

IBM i  
Version 7.2

*Programming  
IBM i globalization*



**Note**

Before using this information and the product it supports, read the information in [“Notices” on page 589](#).

This document may contain references to Licensed Internal Code. Licensed Internal Code is Machine Code and is licensed to you under the terms of the IBM® License Agreement for Machine Code.

© **Copyright International Business Machines Corporation 1998, 2013.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>IBM i globalization.....</b>	<b>1</b>
PDF file for IBM i globalization.....	1
Globalization overview.....	2
Multiple language support.....	2
National language version.....	2
IBM i translations.....	3
National language design in IBM i.....	3
Linguistic and cultural values.....	7
Setting up IBM i with a national language version.....	27
How a language is displayed for IBM i functions.....	27
Installation preparation and national languages.....	29
Checklist: Globalization planning.....	29
Hardware installation and national languages.....	31
Software installation and national languages.....	33
Configuring a national language version.....	33
Scenarios: Setting up IBM i with a national language version.....	42
Developing globalized applications.....	46
Goals and processes.....	47
Designing globalized applications.....	52
Programming considerations in globalized application design.....	90
Delivering globalized applications.....	104
Handling data in globalized applications.....	105
Working with Unicode.....	105
GB18030: The Chinese standard.....	119
Working with CCSIDs.....	119
Working with bidirectional data.....	166
Working with DBCS data.....	168
Working with locales.....	191
Globalization reference information.....	235
National language version feature codes.....	235
Country and region identifiers.....	237
Default system values for national language versions.....	244
System values for other languages with no national language version.....	260
Keyboard reference information.....	262
Code pages.....	277
Character sets.....	279
CCSID reference information.....	308
Locale reference information.....	343
UCS-2 level-1 mapping tables.....	360
Unicode Private Use Area mapping on IBM i.....	403
REXX extension characters.....	406
Default character data conversion that can use substitution.....	432
Globalization checklists.....	435
Related information for IBM i globalization.....	436
<b>Notices.....</b>	<b>589</b>
Programming interface information.....	590
Trademarks.....	591
Terms and conditions.....	591

**Terms and conditions.....593**

---

# IBM i globalization

The IBM i operating system is designed to support the culture and languages of many countries around the world. As companies integrate e-commerce on a global scale into their fundamental business processes, their prospective customers, established customers, and active partners can take advantage of increased revenue and decreased expenses through software globalization.

Globalizing your e-business is no longer a luxury; it is a necessity. As the Internet transcends national and geographical boundaries, the concept of doing business within a single country is quickly giving way to the need to compete in an international marketplace.

Globalized software gives you the following advantages:

- Increased customer satisfaction that can increase sales
- Enhanced customer support communications
- Enhanced global information dissemination
- A better return on Information Technology (IT) investments

This information shows you how to:

- Create an application efficiently and at minimal expense.
- Retrofit existing applications for globalization and create new applications designed for globalization. Designing an application for globalization, however, is usually less expensive than retrofitting an existing application.
- Ensure that the application design does not interfere with the current or planned design of other internationalized applications.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

## **Related information**

[Globalize your On Demand Business Web site](#)

[IBM i Globalization Web site](#)

---

## PDF file for IBM i globalization

You can view and print a PDF file of this information.


To view or download the PDF version of this document, select [IBM i globalization](#) .

### **Saving PDF files**

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF link in your browser.
2. Click the option that saves the PDF locally.
3. Navigate to the directory in which you want to save the PDF.
4. Click **Save**.

### **Downloading Adobe Reader**

You need Adobe Reader installed on your system to view or print these PDFs. You can download a free copy from the [Adobe Web site](http://www.adobe.com/products/acrobat/readstep.html) ([www.adobe.com/products/acrobat/readstep.html](http://www.adobe.com/products/acrobat/readstep.html)) .

## Globalization overview

---

The IBM i operating system supports many languages, thus working as you expect it to work from both a linguistic and a cultural point of view. You can work in the language of your choice. The IBM i operating system also ensures that the data you send to and receive from the system appears in the form and order you expect.

The IBM i operating system uses a common set of program code, regardless of which language you use on the system. For example, the program code on a U.S. English system and the program code on a Spanish system are identical. Different sets of textual data are used, however, for different languages.

*Textual data* is a collective term for menus, displays, lists, prompts, options, online help information, and messages. This means that you see *Help* for the description of a function key for online help information about a U.S. English system, while you see *Ayuda* on a Spanish system. Using the same program code with different sets of textual data allows the IBM i operating system to support more than one language on a single system.

### Multiple language support

The IBM i operating system provides the tools and functions you need to make your applications deliver your business information, such as dates and numbers, in formats that conform to the expectations of users in multiple cultures using multiple languages.

You can enable your system to translate, present, and process data in a global environment.

When you install secondary languages on your system, you can set up your system with user interfaces (that is, textual data) for any of the national language versions (NLVs) provided for the system. To support multiple languages concurrently, you must have adequate storage to install all the necessary secondary languages. You must also install the necessary hardware to support each language.

### National language version

A *national language version (NLV)* is a version of the IBM i operating system that contains a predefined set of language-dependent values, such as date format, time format, and sort sequence, for a particular language.

When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code. If you want to use more than one national language version of a licensed program, you can order additional languages. For example, if you are a German customer, you might need support for both German and French on one system. You can order a national language version for German and a national language version for French.

When you order more than one national language version for a system, you designate one of the versions as the primary language. The primary language you designate is the feature code identified when you ordered the operating system. You designate all other national language versions as secondary languages.

You must order some of the licensed programs for your system with the same language feature code as the primary language of the system. If the language feature code of a licensed program differs from the language feature code of the primary language of the system, the licensed program might not install correctly. Licensed programs with different feature codes as the primary language can be installed as a secondary language.

#### Related concepts

##### National language version feature codes

This table lists the available national language version feature codes on the IBM i operating system. When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code.

##### Setting up IBM i with a national language version

The steps to install and configure a national language version on the IBM i operating system include selecting and installing hardware, installing software, and configuring your environment to run in a globalized setting.

## **IBM i translations**

IBM i licensed programs, or portions of IBM i licensed programs, are translated into languages listed in this topic. Not all portions of IBM i licensed programs are translated into every language.

- Arabic
- Brazilian Portuguese
- Czech
- Danish
- Dutch (this includes Belgian Dutch)
- Finnish
- French (this includes Belgian French, Canadian French and French Multinational Character Set)
- German (this includes German Multinational Character Set)
- Hebrew
- Hungarian
- Icelandic
- Italian (this includes Italian Multinational Character Set)
- Japanese (this includes Japanese Universal)
- Korean
- Norwegian
- Polish
- Portuguese (this includes Portuguese Multinational Character Set)
- Russian
- Simplified Chinese
- Spanish
- Swedish
- Traditional Chinese
- Turkish

## **National language design in IBM i**

The national language design in IBM i defines the functions your application software can use to support national languages.

### **Character representation**

Character representation in the system is controlled by the Character Data Representation Architecture (CDRA).

CDRA identifies characters by encoding scheme identifier (ESid), character set, pairs of character sets and code pages (as needed), and additional coding-related information (as necessary). This identification is established by a system of tags. The tags are handled by the IBM i operating system in a way that ensures character set integrity.

The overall objective of CDRA is to define a method of assigning and preserving the meaning of coded graphic characters through various stages of processing and interchanging.

### **Encoding scheme**

The Character Data Representation Architecture (CDRA) system of tags uses an encoding scheme to specify many rules.

The rules include:

- The coding space (number and allowable value of code points in a code page)
- Rules for sharing the coding space between control and graphic characters
- Rules related to specific options, such as the number bytes required for each character (single-byte, double-byte, or mixed-byte) permitted in that scheme
- Rules related to code extension techniques (if used)

The rules for encoding schemes are followed when code points are assigned to graphic characters in a particular code page. Some common encoding schemes are Extended Binary Coded Decimal Interchange Code (EBCDIC) and American Standard Code for Information Interchange (ASCII).

### **Conversion of character data**

The Character Data Representation Architecture (CDRA) system of tags ensures that you can convert character data in a predictable, repeatable way.

Conversion pertains to converting the code points assigned to one or more characters in one code page to their corresponding code points in another code page. The conversion might cause a single character to map to a sequence of characters, or a sequence of characters to map to a single character. Conversion should not be equated to translating from one language to another.

### **Conversion methods**

The following methods are used for conversion:

- *Round-trip conversion.* The integrity of all character data is maintained from the source coded character set identifier (CCSID) to the target CCSID and back to the source.

When performing a round-trip conversion, you might see incorrect representation of the characters displayed in the target CCSID. The integrity is preserved, however. When the characters are converted back to the source CCSID, they regain their original hexadecimal values and representation.

- *Enforced subset match conversion (substitution).* Characters that exist in both the source and target CCSID have their integrity maintained. Characters in the source CCSID but not in the target CCSID are replaced. Replaced values are also referred to as substitution characters. For EBCDIC encoding, these appear on most display stations as a solid block. For ASCII encoding, these substitution characters appear differently.

This substitution is permanent when converting back to the source CCSID because it is not possible to retrieve the original hexadecimal values.

For a list of CCSID conversions that result in substitution characters, see the Default conversion that might use substitution table.

- *Linguistic conversion.* Also known as best-fit conversion, a partial mapping is done from the source code page to the target code page. The integrity of characters that are in both the target CCSID and the source CCSID are preserved. Characters that are not in the target CCSID are mapped to the most culturally acceptable alternative for that character.

For example, the source CCSID might support an A grave character (À). The target CCSID might not support this character. During the conversion, the most linguistically acceptable character (a Latin capital A) is substituted for the A grave. After the conversion, characters that are not included in the target CCSID are presented to the user as the most linguistically acceptable substitution characters. This substitution is permanent. Any loss of character integrity is permanent.

Through an application programming interface (API), linguistic conversion is available from any supported single-byte CCSID to any other supported single-byte CCSID.



## Related concepts

[Recommendations and guidelines for using CCSIDs](#)

These recommendations are useful when you write globalized applications.

## Related reference

[Default character data conversion that can use substitution](#)

The default CCSID conversions use substitution because the character sets within the CCSIDs are different. The table shows which CCSIDs (From CCSID column) can be substituted by other CCSIDs (To CCSID column).

## Related information

[Character Conversion APIs](#)

## Coded character set identifier values

CDRA defines the range of values for CCSIDs (coded character set identifiers).

The values include:

CCSID value	Purpose or meaning
00000	Use next higher hierarchical CCSID
00001 through 65533	IBM-registered CCSIDs
65534	Refer to lower hierarchical CCSID
65535	No automatic conversion of data between this CCSID and any other CCSID. (This is the default setting of the QCCSID system value.)

CDRA uses a tag field to hold a CCSID value to identify the meaning of coded graphic characters. The tag field might be in a data structure that is logically associated with the data object (explicit tagging), or it might be inherited from the tag field associated with the other objects within the operating system (implicit tagging).

## Related concepts

[Set job attributes \(QSETJOBATR\) system value](#)

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

## Related information

[CCSID values defined on IBM i](#)

## Character data integrity

The Character Data Representation Architecture (CDRA) system of tags uses coded character set identifiers (CCSIDs) to maintain data integrity when character data is passed from system to system or from user to user. CCSIDs assign a value that uniquely identifies the coded graphic character representation used for character data.

## Data integrity is not maintained using CCSID 65535 across countries

The following table shows the meaning of maintaining data integrity. A database file created by a U.S. user contains a dollar sign and is read by a user in the United Kingdom and in Denmark. If the application does not assign CCSID tags that are associated with the data to the file, users see different characters.

Country	Keyboard type	Code page	CCSID	Code point	Character
U.S.	USB	037	65535	X'5B'	\$
U.K.	UKB	285	65535	X'5B'	£
Denmark	DMB	277	65535	X'5B'	Ⓐ

## Data integrity is maintained by using CCSID tags

If the application assigns a CCSID associated with the data to a file, the application can use IBM i CCSID support to maintain the integrity of the data. When the file is created with CCSID 037, the user in the United Kingdom (job CCSID 285) and the user in Denmark (job CCSID 277) see the same character. Database management takes care of the mapping.

Country	Keyboard type	Code page	CCSID	Code point	Character
U.S.	USB	037	00037	X'5B'	\$
U.K.	UKB	285	00285	X'4A'	\$
Denmark	DMB	277	00277	X'67'	\$

CCSID support is particularly important when:

- Multiple national language versions, keyboards, and display stations are installed on the IBM i operating system.
- Multiple systems are sharing data between systems with different national language versions.
- The correct keyboard support for a language is not available when you want to encode data in another language.

### Related concepts

#### CCSID reference information

Coded character set identifier (CCSID) is a 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

### Character processing

Character processing on the IBM i operating system is controlled by specific coding rules and guidelines that ensure consistent processing of character data.

The rules and guidelines cover tasks such as:

- Converting character data to all uppercase or to all lowercase data
- Folding data (substituting printable or displayable characters for those that cannot be printed or displayed on a particular device)
- Processing character data strings
- Classifying characters
- Naming objects
- Determining data, file, and field lengths

### Related concepts

#### Developing globalized applications

Globalized applications are applications that have national language support. National language support allows users to enter, store, process, retrieve, print, and display data in their chosen language. It also allows users to see and enter commands, prompts, messages, and documentation in their chosen language, in formats matching their cultural expectations.

### Character presentation

Character presentation on the IBM i operating system is controlled by coding rules and algorithms that ensure consistent presentation of character data.

These rules and algorithms cover tasks such as:

- Shaping characters
- Truncating characters
- Handling substrings of character data

These rules and algorithms are described in detail in [“Developing globalized applications”](#) on page 46.

### **Globalization hardware support**

Hardware, in this context, means the physical keyboards, displays, printers, and controllers that make up a IBM i product. The extent to which this hardware supports national languages might impose limitations on the degree of support that you can provide with an application.

You must refer to the reference manuals for non-IBM hardware to determine what limitations, if any, are imposed by that hardware.

### **Character data translation**

Translating is changing the meaning of character data from a set of concepts, ideas, and statements in one human language to a culturally similar meaning in another human language.

You can follow the user interface subset of these rules as guidelines to ensure translation goes smoothly. A subset of these rules is provided in [“User interfaces”](#) on page 60.

### **Locales**

A *locale* is an object that can determine how data is processed, printed, and displayed.

Locales are made up of categories that define language, cultural data, and character sets. The locale support is provided to supplement the job value options that the IBM i operating system previously has provided.

Many locales are included with the IBM i operating system. In addition, locale definition source files are provided for locale customization. A locale definition source file contains one or more categories that describe (or make up) a locale.

### **Related concepts**

#### Installing and enabling locales

If you are installing a new release, you can request that library QSYSLOCALE be installed on the system at that time.

#### Working with locales

Locales are used primarily in ILE-based application programs. Additionally, the Retrieve Locale Information (OPM, QLGRVLC; ILE, QlgRetrieveLocaleInformation) API retrieves one or all categories of a locale.

## **Linguistic and cultural values**

Linguistic and cultural conventions include any system values, attributes, or settings that can be altered to suit a country or language.

Examples of linguistic and cultural conventions on the system include date formats and currency symbols.

Some linguistic and cultural conventions might vary by language within a country. For example, language conventions vary in Canada. One set of linguistic conventions apply for French and another set of linguistic conventions apply for English.

### **Where you can change linguistic and cultural values on IBM i**

Settings of cultural and linguistic conventions are supported at different levels in the IBM i operating system.

The system is structured in the following way:

<i>Table 1. System with subsystem A and subsystem B</i>	
<b>System</b>	
<b>Subsystem A</b>	<b>Subsystem B</b>
Job A1 Job A2	Job B1 Job B2

Some linguistic and cultural conventions can be set or changed at the system level, some at the subsystem level, some at the user profile level, and some at job run time. In addition, some cultural and linguistic settings can be set or changed in device descriptions. For example, keyboard types can be changed when creating or changing a display device description.

### **Related tasks**

#### Enabling the secondary language

You must ensure that secondary languages can be used after they have been installed on the system.

### **User profiles**

Individual users can store customized cultural and linguistic values in their user profiles.

These customized values can differ from the system default values and can be used by the IBM i operating system when you set job attributes and object attributes for an individual user. Job attributes can also be used as defaults for setting object attributes that are created or changed under the control of that job.

If you have a single system supporting multiple languages, you should change the user profile to use language and cultural-appropriate values. When you change the character set identifier (CCSID) parameter in the user profile, ensure that the CCSID is set as follows:

- Is set to an SBCS CCSID or to CCSID 65535 for SBCS users
- Is set to a mixed CCSID or to CCSID 65535 for DBCS users
- Is set to a SBCS CCSID for SBCS users on a DBCS system

You can use the Create User Profile (CRTUSRPRF) and the Change User Profile (CHGUSRPRF) commands to customize a user profile.

### **Related reference**

Create User Profile (CRTUSRPRF) command

Change User Profile (CHGUSRPRF) command

### **Subsystems**

A subsystem is a single, predefined operating environment through which the system coordinates the work flow and resource use.

The system can contain several subsystems, all operating independently of each other. Subsystems manage resources. The runtime characteristics of a subsystem are defined in an object called a *subsystem description*.

You can use subsystems to support users in a multilingual environment. You should create a separate subsystem for each set of users with differing needs.

### **Subsystem descriptions for secondary language users**

You can create and use a subsystem description in a multilingual environment. For example, you can create a subsystem for secondary language users (such as QGPL/GERMAN for German language users).

A subsystem description defines how, where, and how much work enters a subsystem, and which resources the subsystem uses to perform the work. An active subsystem takes on the simple name of the subsystem description.

A subsystem description consists of three parts:

- Subsystem attributes
- Workstation entry
- Routing entries

### **Notes:**

1. You can work with existing work entries while the subsystem is active.

2. An IBM-supplied subsystem on a DBCS system is included with a workstation entry to support DBCS display devices.

### **Related concepts**

[Work management](#)

[Packaging and installation process](#)

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

### ***Creating a subsystem description***

IBM-supplied subsystem descriptions have been provided as examples and as backup for user-created subsystem descriptions. Therefore, you should not change the subsystem descriptions in libraries QSYS and QGPL. You should make copies of the subsystem descriptions from these libraries and make changes to the copies.

You can create a subsystem description in two ways. You can either copy and then change an existing subsystem description, or create an entirely new description.

### **To copy an existing subsystem description, follow these steps:**

1. On a command line, type CRTDUPOBJ to create a duplicate object of an existing subsystem description.
2. Change the sign-on display file and the system part of the library list for the secondary language.

### **To create an entirely new subsystem description, follow these steps:**

1. Create a subsystem description (CRTSBSD). Specify a sign-on file from the national language version library and specify the national language version library (QSYSnnnn) as the system-library list entry.
2. Create a job description (CRTJOBDD).
3. Add work entries to the subsystem description.
  - a. ADDWSE (Add work station entry)
  - b. ADDJOBQE ( Add job queue entry)
  - c. ADDCMNE (Add communications entry)
  - d. ADDAJE (Add autostart job entry)
  - e. ADDPJE (Add prestart job entry)
4. CRTCLS (Create a class).
5. ADDRTGE (Add routing entries to the subsystem description).

### **Related reference**

[Create Duplicate Object \(CRTDUPOBJ\) command](#)

[Create Subsystem Description \(CRTSBSD\) command](#)

[Add Work Station Entry \(ADDWSE\) command](#)

[Add Job Queue Entry \(ADDJOBQE\) command](#)

[Add Communications Entry \(ADDCMNE\) command](#)

[Add Autostart Job Entry \(ADDAJE\) command](#)

[Add Prestart Job Entry \(ADDPJE\) command](#)

[Create Class \(CRTCLS\) command](#)

[Add Routing Entry \(ADDRTGE\) command](#)

### **Subsystem attributes**

Subsystem attributes provide the overall characteristics of the subsystem. Attributes include the system-library list entry and a text description of the subsystem description.

For example, you can specify subsystem attributes to support secondary language users:

1. Specify the national language version for the subsystem library entry parameter.

By creating a subsystem for each secondary language on your system, you can ensure that secondary language users have access to textual data in their own language. Within each subsystem, you can arrange the order of libraries in the library list so the textual data for the appropriate secondary language is at the top of the system library list. For example, if you have a primary language of Danish, and a secondary language of German, you can add a library at the top of the system library list in the German subsystem. Jobs running in the German subsystem then use the library at the top of the system part of the library list and a search for German textual data is successful.

If you add a subsystem-library list entry for a national language version library:

- Do not add the library to the QSYSLIBL system value.
  - Be sure that there are no more than 14 libraries in the QSYSLIBL list before adding your additional library entry. (The maximum number of list entries for the system part of the library is 15.)
2. Specify the signon display using the national language version library.
  3. Create or duplicate objects that all users of the secondary national language version need in the national language version library.
  4. Add workstation entries for these workstations that are specifically configured for this national language version.

### **Workstation entry**

A workstation entry, which is an entry in the subsystem description, specifies the workstations from which users can sign on to the subsystem or from which interactive jobs can transfer to the subsystem.

Here are the items that you can specify in a workstation entry. Parameter names are given in parentheses.

- Workstation name or type (WRKSTN or WRKSTNTYPE)
- Job description to be used for jobs started through this workstation entry
- Maximum number of interactive jobs that can be active at the same time through the entry (MAXACT)
- When the work stations are to be allocated, either when the subsystem is started or when an interactive job enters the subsystem through the Transfer Job (TFRJOB) command.

### **Adding, changing, or removing workstation entries**

The following commands allow you to add, change, or remove workstation entries from a subsystem description.

To add a workstation entry to a subsystem description, use the Add Work Station Entry (ADDWSE) command. Here is an example of adding a workstation entry:

```
ADDWSE SBSDB(USERLIB/ABC) WRKSTN(DSP12)
JOBDB(USERLIB/WSE)
```

To specify a different job description for a previously defined workstation entry, use the Change Work Station Entry (CHGWSE) command. Here is an example of changing a workstation entry:

```
CHGWSE SBSDB(USERLIB/ABC) WRKSTN(DSP12)
JOBDB(USERLIB/NEWJD)
```

To remove a workstation entry from a subsystem description, use the Remove Work Station Entry (RMVWSE) command. Here is an example of removing a workstation entry:

```
RMVWSE SBSDB(USERLIB/ABC) WRKSTN(DSP12)
```

## Related reference

[Transfer Job \(TFRJOB\) command](#)

[Add Work Station Entry \(ADDWSE\) command](#)

[Change Work Station Entry \(CHGWSE\) command](#)

[Remove Work Station Entry \(RMVWSE\) command](#)

## Starting a subsystem

After you have created a subsystem that meets your needs, you need to start the subsystem.

To start a subsystem, use the Start Subsystem (STRSBS) command:

```
STRSBS SBSD('library name/subsystem name')
```

For example:

```
STRSBS USERLIB/ABC
```

## Related reference

[Start Subsystem \(STRSBS\) command](#)

## Job attributes

Job attributes are set at the time a job starts.

Some job attributes are set from the user profile. Other job attributes come from system values, from locales, from a Submit Job (SBMJOB) command, a job description, and the Change Job (CHGJOB) command (from which you can change values for attributes while the job is running).

## Related concepts

[Database management](#)

Database management support provides default coded character set identifier (CCSID) values for database files on the system. All database files are assigned a CCSID. At file creation time, the CCSID is either explicitly assigned through DDS, SQL, or IDDU, or implicitly assigned the job default CCSID (DFTCCSID).

## Related reference

[Submit Job \(SBMJOB\) command](#)

[Change Job \(CHGJOB\) command](#)

## Coded character set identifier job attribute

When an interactive job is started on the IBM i operating system, the job CCSID value is taken from the user profile. When a batch job is started, the current job CCSID is used unless a CCSID is specifically entered on the SBJOB command.

For every mixed-byte coded character set CCSID, there is a corresponding SBCS CCSID that is valid. If you specify a mixed-byte coded character set CCSID for an SBCS system, the job CCSID is changed to the corresponding SBCS CCSID.

If a job CCSID is specified as an SBCS CCSID, the job cannot handle DBCS data. If a job CCSID is specified as a mixed CCSID, the job can handle DBCS data. You must use a DBCS-capable display device, though, for the DBCS data in a job to display correctly. You can specify a mixed-byte CCSID for a job only if the DBCS system value (QIGC) value is set to 1 (on). A QIGC value of 1 indicates that a DBCS national language version is installed on the system.

## Job default coded character set identifier

A job attribute, job default CCSID (DFTCCSID), is created for jobs with a CCSID of 65535. The DFTCCSID value is used by a system code when a CCSID other than 65535 is needed.

The DFTCCSID attribute can only be retrieved or displayed. The value of this attribute is determined as follows:

- If the job CCSID is not 65535, the DFTCCSID equals the job CCSID.

- If the job CCSID is 65535, the DFTCCSID value is based on an appropriate value derived from the job language identifier (LANGID).

When the job is running, the system determines the default CCSID for a job using the following logic (you can find the corresponding CCSID for LANGID in default CCSID table):

1. If the job CCSID is set to a value, it uses that value.
2. If the job CCSID is set to \*USRPRF, then the system checks the user profile for the value.
3. If the user profile is set to a value, it uses that value.
4. If the user profile is set to \*SYSVAL, the system checks the system value.
5. If the system value for QCCSID is set to a value, it uses that value.
6. If the system value is set to 65535, the system checks the job's language ID.
7. If the job's LANGID is set to a value, the QTQ\_DEFAULT\_CCSID environment variable is checked for that LANGID value. If the QTQ\_DEFAULT\_CCSID environment variable contains a value for that LANGID, the CCSID specified in the QTQ\_DEFAULT\_CCSID environment variable is used. If the QTQ\_DEFAULT\_CCSID environment variable does not contain a value for the LANGID, the system converts that LANGID to a CCSID.
8. If the job's LANGID is set to \*USRPRF, the system checks the user profile's language ID.
9. If the user profile's LANGID is set to a value, the QTQ\_DEFAULT\_CCSID environment variable is checked for that LANGID value. If the QTQ\_DEFAULT\_CCSID environment variable contains a value for that LANGID, the CCSID specified in the QTQ\_DEFAULT\_CCSID environment variable is used. If the QTQ\_DEFAULT\_CCSID environment variable does not contain a value for the LANGID, the system converts that LANGID to a CCSID.
10. If the user profile's LANGID is set to \*SYSVAL, the QTQ\_DEFAULT\_CCSID environment variable is checked for that LANGID value. If the QTQ\_DEFAULT\_CCSID environment variable contains a value for that LANGID, the CCSID specified in the QTQ\_DEFAULT\_CCSID environment variable is used. If the QTQ\_DEFAULT\_CCSID environment variable does not contain a value for the LANGID, the system converts that LANGID to a CCSID.

### **Related concepts**

#### Database management

Database management support provides default coded character set identifier (CCSID) values for database files on the system. All database files are assigned a CCSID. At file creation time, the CCSID is either explicitly assigned through DDS, SQL, or IDDU, or implicitly assigned the job default CCSID (DFTCCSID).

#### Graphic character conversion tables

Table (\*TBL) objects support non-CCSID conversions from one code page to another. The system-supplied table objects are located in the QUSRSYS library.

#### Language identifiers and associated default CCSIDs

This table shows the language identifiers and the job default CCSID (DFTCCSID) values associated with those identifiers.

### ***Job library list***

The language used for textual data (displays, messages, printed output, and online help information) is controlled by the library list for the job.

Users can place their national language library, before QSYS (the primary language library) and any other national language libraries in their library lists. In this way, users can customize which national language versions of information are presented to them.

### **Related concepts**

#### System library list (QSYSLIBL) system value

The system library list (QSYSLIBL) system value is used as the first part of the library list associated with a job.

#### Packaging and installation process



You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

### **System values**

The system values of the primary language on the system are used as system-wide cultural and linguistic defaults. Therefore, if you change the primary language on the system, each varying system value resets to the default system value of the new primary language.

The following list shows the cultural and linguistic system values. To display or change these values, use the Work with System Value (WRKSYSVAL) command. A subset of language-dependent default system values (QCCSID, QCHRID, QCNTYID, QCURSYM, QDATFMT, QDATSEP, QDECFMT, QKBDTYPE, QLANGID, and QTIMSEP) are shown in Default system values in the Reference section.

### **Related concepts**

[Default system values for national language versions](#)

Jobs and functions on the IBM i operating system use system values as default values.

[System values](#)

[Configuring the primary language](#)

A primary language consists of program code, textual data for each licensed program ordered, and default national language cultural values.

### **Related reference**

[Work with System Value \(WRKSYSVAL\) command](#)

### **Century (QCENTURY) system value**

The century (QCENTURY) system value specifies the century. It is used with the system values QDATE and QYEAR to determine the specific date currently used by the system.

The possible values are:

- 0 (the years from 1928 to 1999)
- 1 (the years from 2000 to 2053)

**Note:** 1900 to 1927 and 2054 to 2099 are not supported years for system time. Applications can, however, support year date ranges from 0001 to 9999.

You can set the value of QCENTURY with the century indicator, or the system sets the value of QCENTURY based on the following two situations:

- At the time of the first IPL, the system sets the initial value of QCENTURY based on the following rules:
  - If QYEAR is equal to or greater than 40, the system assigns a value of 0 to QCENTURY.
  - If QYEAR is less than 40, the system assigns a value of 1 to QCENTURY.
- When QYEAR or the year in QDATE is changed:
  - QCENTURY is set to 0 if QYEAR is 54 to 99
  - QCENTURY is set to 1 if QYEAR is 00 to 27

For example, if you change QYEAR from 95 to 13, the system changes QCENTURY from 0 to 1, indicating a year of 2013. However, if you change QYEAR from 95 to 45, the system will not change QCENTURY, because both 1945 and 2045 are valid dates.

If you change this value, the change takes effect immediately. Changing this value also affects the system value QDATE.

**Note:** The 21st century begins at 0000 hours, 1 January 2001. However, for purposes of common understanding, the 20th/21st century boundary is defined to be between 2400 hours, 31 December 1999 and 0000 hours, 1 January 2000. This allows a discussion of the 21st century to include all dates with a 20xx format inclusive of the year 2000.

### **Related concepts**

[System date \(QDATE\) system value](#)

The system date (QDATE) system value indicates the year, the month, and the day on the system.

Year (QYEAR) system value

The year (QYEAR) system value specifies the last 2 digits of the year on the system.

#### ***Character identifier (QCHRID) system value***

The character identifier (QCHRID) system value specifies the character set and code page CHRID(\*SYSVAL) for the CL commands that create, change, or override display files, display device descriptions, user interface (UIM) menus, panel groups, and printer files.

You can change this value if the system QCCSID system value is set to CCSID 65535. You can also change the QCHRID value if the code page portion of the new QCHRID value is the same as the code page portion of the QCCSID value.

#### **Related concepts**

Object-level coded character set identifier 65535

CCSID 65535 is the default object-level CCSID for message files and message queues.

#### ***Character identifier control (QCHRIDCTL) system value***

The character identifier control (QCHRIDCTL) system value controls the type of CCSID conversion that occurs for display files, printer files, and panel groups.

You must specify the \*CHRIDCTL special value on the CHRID parameter of the create, change, or override command for display files, printer files, and panel groups before this attribute can be used.

Possible values are:

#### **\*DEV D**

The support provided by the \*DEV D special value on the CHRID parameter for display files, printer files, and panel groups.

#### **\*JOBCCSID**

The support provided by the \*JOBCCSID special value on the CHRID parameter for display file, printer files, and panel groups.

#### **Related concepts**

Display files

When a display file object is created, it is tagged with the coded character set identifier (CCSID) of the source file.

#### ***Coded character set identifier (QCCSID) system value***

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

As shipped, the CCSID is set to CCSID 65535. CCSID 65535 means that all character data tagging support on the system is turned off, which is not generally recommended.

If you use Java™ or WebSphere®, or if you plan to transfer data between the IBM i operating system and another client, then this value (or the corresponding value on the user profile) should be set to match the CCSID of your data. If the value is 65 535, then the encoding of the data on the system is unknown.

If you leave this value at 65535, then you cannot get the results you expect when working from a client on the IBM i operating system, or the connection might not work at all.

You can change the coded character set identifier (QCCSID) system value. When you change this value, the default character set and code page system value (QCHRID) is changed to match the character set and code page of the coded character set identifier.

If a job is started with a single byte CCSID, (from either this value or the user profile value) then that job will not support double-byte language (DBCS) users.

#### **Related concepts**

Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

#### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Database file attributes

Database attributes, such as coded character set identifier (CCSID), sort sequence (SRTSEQ), and language identifier (LANGID), are cultural dependent.

#### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

#### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

#### ***Coordinated universal time offset (QUTCFFSET) system value***

The coordinated universal time offset (QUTCFFSET) system value specifies the number of degrees, in hours and minutes, by which your local system differs from the zero meridian.

This value is used by the system when processing alerts that are sent to other systems, as well as by other parts of the system. If systems in a network cross time zones, the QUTCFFSET value is sent in the alert.

This value is 5 characters long. The first character is a plus (+) sign or minus (-) sign. The next 2 characters specify hours ranging from 00 through 24. The last two characters specify minutes ranging from 00 through 59.

For example, you have a network with one system in Brisbane, Queensland, Australia (Eastern Australia standard time zone) and one system in Caracas, Venezuela. You can set QUTCFFSET to +1000 for the Brisbane system and to -0400 for the Caracas system.

The Brisbane system value should be changed each time the daylight saving time begins or ends. Caracas, Venezuela does not observe a daylight saving time, and its system value remains constant.

If you change this value, the change takes effect immediately.

#### ***Country or region identifier (QCNTYID) system value***

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

Setting this system value, along with the QLANGID system value, allows you to choose the correct language dictionary, encoding of data, and advanced linguistics for successful document indexing. There is no validity checking between the QCNTYID system value and the QCCSID system value.

#### **Related concepts**

##### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

##### Coded character set identifier (QCCSID) system value

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

##### Country and region identifiers

This table lists the country and region identifiers.

##### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

##### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

#### Coding globalized applications with high-level languages

Your major goal must be to have only one general set of running code that is common for all language versions and to make your programs table-driven as much as possible.

#### **Currency symbol (QCURSYM) system value**

The currency symbol (QCURSYM) system value verifies the currency symbols specified in the DDS keywords Edit Word (EDTWRD) and Edit Code (EDTCDE).

You can change the currency symbol to correctly reflect the monetary symbol used in your country or location. If you change this system value, the change takes effect immediately.

#### **Related concepts**

##### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

#### **Related reference**

EDTWRD (Edit Word) keyword for display files

EDTCDE (Edit Code) keyword for display files

#### **System date (QDATE) system value**

The system date (QDATE) system value indicates the year, the month, and the day on the system.

This value is made up of the QYEAR, QMONTH, and QDAY system values. The format in which QDATE appears is specified by the QDATFMT system value. You can change the system date. If you change QDATE, the change might affect the system values for QCENTURY, QYEAR, QMONTH, QDAY, and QDAYOFWEEK. Any change you make to QDATE takes effect immediately.

#### **Related concepts**

##### Century (QCENTURY) system value

The century (QCENTURY) system value specifies the century. It is used with the system values QDATE and QYEAR to determine the specific date currently used by the system.

##### Year (QYEAR) system value

The year (QYEAR) system value specifies the last 2 digits of the year on the system.

##### Month of the year (QMONTH) system value

The month of the year (QMONTH) system value indicates the month of the year on the system.

##### Day of the month (QDAY) system value

The day of the month (QDAY) system value indicates the day of the month on the system. This value must be a valid day of the month or of the year if you are using the Julian date format.

##### Day of week (QDAYOFWEEK) system value

The day of week (QDAYOFWEEK) system value specifies the day of the week on the system.

#### **Date format (QDATFMT) system value**

The date format (QDATFMT) system value is used for the default value for the DATFMT job attribute. This system value also determines the format in which a date can be specified on the initial program load (IPL) options prompt.

This system value can be:

- YMD (year, month, day)
- MDY (month, day, year)
- DMY (day, month, year)
- JUL (Julian format, which is year, day of year)

You can change the date format to reflect the format in which months, days, and years are represented in your country or location. If you change this system value, the change takes effect for new jobs that enter the system after you make the change.

### **Related concepts**

#### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

#### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### ***Date separator (QDATSEP) system value***

The date separator (QDATSEP) system value is used as the date separator for the default value of the DATSEP job attribute. It is also used as the date separator you can specify on the initial program load (IPL) options prompt.

You can change the date separator to reflect the character used to separate days, months, and years for your country or location. You can change the date separator to any one of the following values:

- A slash (/) as a date separator
- A hyphen (-) as a date separator
- A period (.) as a date separator
- A comma (,) as a date separator
- A blank ( ) as a date separator

If you change this value, the change takes effect for new jobs that enter the system after you make the change.

### **Related concepts**

#### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

#### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### ***Day of the month (QDAY) system value***

The day of the month (QDAY) system value indicates the day of the month on the system. This value must be a valid day of the month or of the year if you are using the Julian date format.

You can change the day of the month to reflect the current day of the month in your country or location. If you change QDAY, you also change the value for QDATE. A change to this value takes place immediately.

### **Related concepts**

#### System date (QDATE) system value

The system date (QDATE) system value indicates the year, the month, and the day on the system.

### ***Day of week (QDAYOFWEEK) system value***

The day of week (QDAYOFWEEK) system value specifies the day of the week on the system.

This value can be:

- \*SUN (Sunday)
- \*MON (Monday)
- \*TUE (Tuesday)
- \*WED (Wednesday)

- \*THU (Thursday)
- \*FRI (Friday)
- \*SAT (Saturday)

This value cannot be changed. It is set by the system. The value of QDATE determines the value of QDAYOFWEEK.

This value cannot be set correctly if your system is not using the Gregorian calendar.

### **Related concepts**

System date (QDATE) system value

The system date (QDATE) system value indicates the year, the month, and the day on the system.

Leap year adjustment (QLEAPADJ) system value

The Leap year adjustment (QLEAPADJ) system value adjusts the system algorithms for the leap year in different calendar systems.

### **DBCS system indicator (QIGC) system value**

The DBCS system indicator (QIGC) system value specifies whether a double-byte character set (DBCS) national language version (NLV) is installed. This value is set when the primary national language version is installed.

If QIGC is set to 0, no DBCS national language version is installed on the system. When QIGC is set to 0, the coded character set system identifier (QCCSID) must be set to an SBCS coded character set identifier.

If QIGC is set to 1, a DBCS national language version is installed as the primary language on the system. When QIGC is set to 1, the coded character set system identifier (QCCSID) system value should be set to a mixed CCSID (such as 05026) or to CCSID 65535.

Beginning with IBM i V5R3, any NLV can support DBCS. Therefore, QIGC is always set to 1 (or on). If you have applications that check this value, update them to use the job level DBCS indicator. You can use the Retrieve Job Information (QUSRJOBI) API to get the job's IGC value.

You cannot change this value.

### **Related concepts**

Recommendations and guidelines for using CCSIDs

These recommendations are useful when you write globalized applications.

### **Related information**

Retrieve Job Information (QUSRJOBI) API

### **DBCS font name (QIGCCDEFNT) system value**

The DBCS font name (QIGCCDEFNT) system value is used when the system transforms SNA character string (SCS) data with shift in/shift out (SI/SO) characters into a spooled file that is composed of Advanced Function Presentation data stream (AFPDS).

QIGCCDEFNT is a 20-character list of up to 2 values. The first 10 characters contain the font name. The last 10 characters contain the library name. The font name can be only 8 characters. The possible values for the DBCS font name are:

#### **\*NONE**

No font is identified to the system.

#### **Coded font name**

The name of the DBCS font.

The possible values for the library are:

#### **\*LIBL**

The library list is used to locate the font.

#### **\*CURLIB**

The current library is used to locate the font. If no library is specified, library QGPL is used.

**Library name**

The library containing the font.

**Decimal format (QDECFMT) system value**

The decimal format (QDECFMT) system value determines the type of zero suppression and decimal point character used by DDS edit codes 1 through 4 and A through M. It also determines the decimal point character for decimal input fields in the interface.

You can change the decimal format to reflect the way decimals are formatted for your country or location. You can change the decimal format to any one of the following values:

**(blank)**

If you specify a blank, the system uses a period for a decimal point, a comma for a 3-digit grouping character, and zero suppression to the left of the decimal point. For example,

```
One thousand is formatted as 1,000
and
Four one-hundredths is formatted as .04
```

**J**

If you specify a J, the system uses a comma for a decimal point, a period for a 3-digit grouping character, and zero suppression at the second character to the left of the decimal point. For example,

```
One thousand is formatted as 1.000
and
Four one-hundredths is formatted as 0,04
```

**I**

If you specify an I, the system uses a comma for a decimal point, a period for a 3-digit grouping character, and zero suppression to the left of the decimal point. For example,

```
One thousand is formatted as 1.000
and
Four one-hundredths is formatted as ,04
```

A change to this value takes effect immediately.

**Related concepts**Set job attributes (QSETJOBATR) system value

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

Decimal formats

You can change the decimal format with the QDECFMT system value to reflect the way decimals are presented for your country or location.

**Language identifier (QLANGID) system value**

The language identifier (QLANGID) system value specifies the default language identifier for the system.

This value also determines the sort sequence table to be used for sorting character data when the QSRTSEQ system value is set to \*LANGIDSHR or \*LANGIDUNQ.

**Note:** This value is not used to determine the sort sequence table when QSRTSEQ is set either to \*HEX or to a user-specified table.

You can change this system value to reflect the default language identifier for your country or location.

There is no validity checking between the QLANGID system value and the QCCSID system value.



## **Related concepts**

### Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

### Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

### Coded character set identifier (QCCSID) system value

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

### Database file attributes

Database attributes, such as coded character set identifier (CCSID), sort sequence (SRTSEQ), and language identifier (LANGID), are cultural dependent.

### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### Coding globalized applications with high-level languages

Your major goal must be to have only one general set of running code that is common for all language versions and to make your programs table-driven as much as possible.

### Session manager

For all applications that use a session manager, you must ensure that the output data stream has no X'3F' values in it. The IBM i operating system uses X'3F' values to make a screen blank.

### ILE RPG sort sequence

The ILE RPG feature, an option of the IBM Rational Development Studio for i licensed program, provides the possibility for a user to specify a sort sequence table and to use the table in comparison operations that are performed with nonnumeric data.

### DB2 and SQL sort sequence

For Interactive SQL, the SRTSEQ and LANGID parameters can be specified on the STRSQL command. You can change these parameters by using the session services for interactive displays.

### IBM i Access sort sequence

You can specify the sort sequence in IBM i Access functions. When performing queries on the system databases and SQL tables, you can specify the system-supplied or user-supplied sort sequence tables.

### ***Language indicator for keyboard type (QKBDTYPE) system value***

The language indicator for the keyboard type (QKBDTYPE) system value specifies the language character set for the keyboard.

This value is used as the default keyboard type when you create a display device description.

You can change this value to reflect the language of your keyboard.

## **Related concepts**

National language keyboard types and SBCS code pages



This table lists the keyboard types and code pages for each national language supported by the IBM i operating system. The Create Device Display (CRTDEV DSP) command uses the KBDTYPE parameter.

### ***Leap year adjustment (QLEAPADJ) system value***

The Leap year adjustment (QLEAPADJ) system value adjusts the system algorithms for the leap year in different calendar systems.

This system value is set by the operating system. If your system observes the Gregorian calendar, this system value should be zero. You cannot edit this system value.

For more information about the Leap year adjustment (QLEAPADJ) system value, see [Date and time system values: Leap year adjustment](#).

### **Related concepts**

[Day of week \(QDAYOFWEEK\) system value](#)

The day of week (QDAYOFWEEK) system value specifies the day of the week on the system.

[Information in message CPX8416](#)

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### ***Locale (QLOCALE) system value***

The locale (QLOCALE) system value specifies a locale object that can determine how data is processed, printed, and displayed. Locales can define the language used by the system, cultural data of that language, and the type of characters displayed or printed.

The locale path name must be a path name that specifies a locale. A locale is made up of the language, territory, and code set combination used to identify a set of language conventions. The maximum path length allowed for the locale path name on the Change System Value (CHGSYSVAL) command is 1024 bytes.

The allowed values are:

<b>Value</b>	<b>Indication</b>
*NONE:	There is no locale for the QLOCALE system value.
*C:	The C locale is to be used.
*POSIX:	The POSIX locale is to be used.
path-name	The path name of the locale to be used.

### **Related concepts**

[Set job attributes \(QSETJOBATR\) system value](#)

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

### ***Month of the year (QMONTH) system value***

The month of the year (QMONTH) system value indicates the month of the year on the system.

This value must be a number from 1 (January) through 12 (December) if your system date format uses the Gregorian calendar. This value cannot be displayed or changed if your system date format uses the Julian format (year, day of year).

You can change the month to reflect the current month in your country or location. If you change QMONTH, you also change the value for QDATE. A change to this value takes place immediately.

### **Related concepts**

[System date \(QDATE\) system value](#)

The system date (QDATE) system value indicates the year, the month, and the day on the system.

### **Set job attributes (QSETJOBATR) system value**

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

This system value has the following attributes that can be assigned values:

- Coded character set identifier (CCSID)
- Date format (DATFMT)
- Date separator (DATSEP)
- Decimal format (DECFMT)
- Sort sequence (SRTSEQ)
- Time separator (TIMSEP)

The system sets the initial values for these attributes from the locale (QLOCALE) system value.

### **Related concepts**

#### Coded character set identifier values

CDRA defines the range of values for CCSIDs (coded character set identifiers).

#### Date formats

There is no worldwide standard for the presentation of dates. Therefore, the date format should always be stored externally as part of the textual data.

#### Date separators

The date separator for presentation should always be stored externally as part of the textual data.

#### Decimal format (QDECFMT) system value

The decimal format (QDECFMT) system value determines the type of zero suppression and decimal point character used by DDS edit codes 1 through 4 and A through M. It also determines the decimal point character for decimal input fields in the interface.

#### Sort sequences

The IBM i operating system supports sort sequence. By using one of the listed options, you can order your data according to cultural-dependent requirements for specific applications.

#### Time separators

The IBM i operating system allows several valid time separators.

#### Locale (QLOCALE) system value

The locale (QLOCALE) system value specifies a locale object that can determine how data is processed, printed, and displayed. Locales can define the language used by the system, cultural data of that language, and the type of characters displayed or printed.

### **Sort sequence (QSRTSEQ) system value**

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

You can change QSRTSEQ to any one of the following values:

Value	Meaning
*HEX	<p>No sort sequence table is used. The hexadecimal values of the graphic characters are used to determine the sort sequence (a binary sort). This is the only sort sequence available for DBCS data.</p> <p><b>Note:</b> When you specify values other than *HEX for mixed-byte character data, SBCS character data is sorted according to the sort sequence specified. DBCS character data is sorted by hexadecimal values (binary sort).</p>

Value	Meaning
*LANGIDSHR	The sort sequence table can use the same weight for multiple graphic characters. The shared-weight sort table associated with the language specified in the LANGID parameter is used. This sort applies only to SBCS data.
*LANGIDUNQ	The sort sequence table contains uniquely weighted graphic characters. The unique-weight sort table associated with the language specified in the LANGID parameter is used. This sort applies only to SBCS data.
Qualified sort sequence table name	The name and library of the sort sequence table to be used. This value allows you to specify a sort sequence table other than those associated with the language specified in the LANGID parameter. This sort sequence table can be used to sort Unicode and SBCS data.

### Related concepts

#### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Sort sequence tables

A sort sequence table is an object that contains the weight of each single-byte graphic character within a specified coded character set identifier (CCSID). The system-recognized identifier for the sort sequence table object type is \*TBL.

#### Database file attributes

Database attributes, such as coded character set identifier (CCSID), sort sequence (SRTSEQ), and language identifier (LANGID), are cultural dependent.

#### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

#### ILE RPG sort sequence

The ILE RPG feature, an option of the IBM Rational Development Studio for i licensed program, provides the possibility for a user to specify a sort sequence table and to use the table in comparison operations that are performed with nonnumeric data.

#### DB2 and SQL sort sequence

For Interactive SQL, the SRTSEQ and LANGID parameters can be specified on the STRSQL command. You can change these parameters by using the session services for interactive displays.

#### IBM i Access sort sequence

You can specify the sort sequence in IBM i Access functions. When performing queries on the system databases and SQL tables, you can specify the system-supplied or user-supplied sort sequence tables.

#### Sort sequence support in work management

Work management involves the assigning of the SRTSEQ value at the job level, the user profile level, and the system value level.

### **System library list (QSYSLIBL) system value**

The system library list (QSYSLIBL) system value is used as the first part of the library list associated with a job.

The libraries in the system part of the library list of a job are searched before any other libraries in the library list of a job. The list can contain as many as 15 names. You cannot delete or rename a library specified as part of the system library list, because libraries in this library list are locked.

You can change the system library list (QSYSLIBL). If you change QSYSLIBL, the change takes place immediately for new jobs entering the system. The change does not affect running jobs, unless the application in the job accesses the system library list directly.

## Related concepts

### Job library list

The language used for textual data (displays, messages, printed output, and online help information) is controlled by the library list for the job.

### ***Time separator (QTIMSEP) system value***

The time separator (QTIMSEP) system value specifies the character separator for time.

This value is used as the time separator for the default value of the TIMSEP job attribute. This value is also used as the time separator that you can specify on the IPL options prompt.

You can change the time separator to reflect the character used to separate hours and minutes for your country or location. You can change the time separator to any one of the following values:

- A colon (:) as a time separator
- A period (.) as a time separator
- A comma (,) as a time separator
- A blank ( ) as a time separator

If you change this value, the change takes effect for new jobs that enter the system after you make the change.

## Related concepts

### Job attributes

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### ***Year (QYEAR) system value***

The year (QYEAR) system value specifies the last 2 digits of the year on the system.

This value ranges from 0 through 99. The system assigns the first two digits for the year based on the current setting for the QCENTURY system value. If the calculated year falls outside the range of dates supported by the system (1928 to 2053), the QCENTURY system value is changed so that the calculated year is within the supported range.

If you change this system value:

- QCENTURY is set to 0 if QYEAR is 54 to 99
- QCENTURY is set to 1 if QYEAR is 00 to 27

For example, if you change QYEAR from 95 to 13, the system changes QCENTURY from 0 to 1, indicating a year of 2013. However, if you change QYEAR from 95 to 45, the system will not change QCENTURY, because both 1945 and 2045 are valid dates.

If you change this value, the change takes effect immediately. Changing this value also affects the system value QDATE.

## Related concepts

### Century (QCENTURY) system value

The century (QCENTURY) system value specifies the century. It is used with the system values QDATE and QYEAR to determine the specific date currently used by the system.

### System date (QDATE) system value

The system date (QDATE) system value indicates the year, the month, and the day on the system.

### Device descriptions

These control language (CL) command parameters can be used to change cultural and linguistic conventions for some display and printer devices.

**Note:** Some printer device descriptions do not allow you to specify a CHRID.

- Character identifier (CHRID) parameter. You can change the character identifier when you create or change device descriptions for printers and displays. Change the character identifier for a printer or display device using one of the following commands:
  - The Create Device Description (Display) (CRTDEVDSPP)
  - The Change Device Description (Display) (CHGDEVDSPP)
  - The Create Device Description (Printer) (CRTDEVPRT)
  - The Change Device Description (Printer) (CHGDEVPRT)
- Keyboard type (KBDTYPE) parameter. You can set the keyboard language type for a keyboard when you create a device description. Set the keyboard language type using the CHGDEVDSPP command.
- Workstation customization (WSCST) parameter. You can set the workstation customization parameter when creating a device to specify the use of a customized keyboard layout. To set this parameter, the display device must be varied off. You can specify the WSCST parameter when using the CRTDEVDSPP command.
- Language type (LNGTYPE) parameter. When you create an ASCII printer using the CRTDEVPRT command, the LNGTYPE parameter describes the default country or region keyboard language identifier for the printer. When you specify the \*SYSVAL value, the QKBDTYPE system value is used.

### Related concepts

[Packaging and installation process](#)

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

### Related reference

[Create Device Description \(Display\) \(CRTDEVDSPP\) command](#)

[Change Device Description \(Display\) \(CHGDEVDSPP\) command](#)

[Create Device Description \(Printer\) \(CRTDEVPRT\) command](#)

[Change Device Description \(Printer\) \(CHGDEVPRT\) command](#)

### Display and printer files

These keywords and command parameters can be used to change cultural and linguistic values for display files and printer files.

- The Create Display File (CRTDSPF), Change Display File (CHGDSPF), Create Printer File (CRTPRTF), Change Printer File (CHGPRTF), and Override Printer File (OVRPRTF) commands. You can specify a character identifier explicitly:
  - As the QCHRID system value (\*SYSVAL)
  - As a device description or a device default of the output device (\*DEVDD)
  - With the \*JOBCCSID value
  - As using the \*CHRIDCTL system value (\*SYSVAL)
- Character identifier (CHRID) keyword in DDS. Use this field-level keyword to identify fields that should be converted to the character identifier (CHRID) of the device. Use this keyword in conjunction with the CHRID parameter on the CRTDSPF, CHGDSPF, CRTPRTF, CHGPRTF, and OVRPRTF commands. This keyword is ignored, however, when the CHRID parameter of these commands is set to \*JOBCCSID.
- The SRTSEQ parameter and LANGID parameter on the CRTDSPF command. These parameters can be used to specify a sort sequence and a language identifier for a display file.

**Note:** If \*JOBCCSID is not specified for the CHRID parameter of a display file (either directly or indirectly with CHRIDCTL), the CHRID parameter of the display file must be compatible with the job CCSID. Otherwise, unpredictable results might occur when data is displayed or when data is stored in a database file.

### **Related reference**

[Create Display File \(CRTDSPF\) command](#)  
[Change Display File \(CHGDSPF\) command](#)  
[Create Printer File \(CRTPRTF\) command](#)  
[Change Printer File \(CHGPRTF\) command](#)  
[Override with Printer File \(OVRPRTF\) command](#)

### **Database files**

These command parameters and the DDS keywords can be used to change language-dependent values for database files.

You can use the following command parameters:

- The SRTSEQ, LANGID, and CCSID parameters on the Create Physical File (CRTPF) command
- The SRTSEQ, LANGID, and CCSID parameters on the Change Physical File (CHGPF) command
- The parameters on the Copy File (CPYF) command
- The SRTSEQ parameter and LANGID parameter on the Create Logical File (CRTL) command

These parameters can be used to specify a sort sequence and language for a database file.

### **DDS keywords for database files**

You can use the following DDS keywords for database files:

- The CCSID keyword. This keyword can be used to tag character data stored in a database. By default, the CCSID value is taken from the job creating the database file.
- DATFMT, DATSEP, TIMFMT, and TIMSEP keywords in DDS.

The format of the data type Time (T) field is described by DDS with the TIMFMT keyword that can have \*JOB specified for a value. Similarly, the format of the data type Date (L) is described by DDS with the DATFMT keyword that can have \*JOB specified for a value.

Use the TIMSEP and DATSEP keywords to specify date and time separators.

### **Related reference**

[Create Physical File \(CRTPF\) command](#)  
[Change Physical File \(CHGPF\) command](#)  
[Copy File \(CPYF\) command](#)  
[Create Logical File \(CRTL\) command](#)  
[DDS keywords and parameters](#)

### **UIM menus and panel groups**

You can use the CHRID parameter on the Create Menu (CRTMNU) or Create Panel Group (CRTPNLGRP) command to specify a \*JOBCCSID for a menu or a panel group.

The CHRID parameter on the Create Menu (CRTMNU) command for creating menus can be used to specify a \*JOBCCSID value for a menu. Conversion is automatically done between the CHRID parameter of the device and the CCSID value of the menu.

The CHRID parameter on the Create Panel Group (CRTPNLGRP) command for creating panel groups can be used to specify a \*JOBCCSID value for panel groups. Conversion is automatically done between the CHRID of the device and the CCSID of the panel group and the CCSID of the job.

## Related reference

[Create Menu \(CRTMNU\) command](#)

[Create Panel Group \(CRTPNLGRP\) command](#)

## Setting up IBM i with a national language version

---

The steps to install and configure a national language version on the IBM i operating system include selecting and installing hardware, installing software, and configuring your environment to run in a globalized setting.

You can use this information as you install your own systems, and you can apply the principles when you develop applications for customers who are installing their own national language version on IBM i.

The feature code identified when you order an IBM i operating system is the language of your textual data and is called the primary language of the system. Any other language versions that you have ordered are called secondary languages. For secondary languages, the national language version consists of only the textual data for all licensed programs ordered. The program code is not contained in the secondary language version.

The primary language is the language in which the system is serviced and from which all language-dependent or cultural-dependent system values are initialized. In addition, other system objects and functions assume attributes based on the primary language. For example, messages appearing in the history log always appear in the primary language.

### Related concepts

[National language version](#)

A *national language version (NLV)* is a version of the IBM i operating system that contains a predefined set of language-dependent values, such as date format, time format, and sort sequence, for a particular language.

[National language version feature codes](#)

This table lists the available national language version feature codes on the IBM i operating system. When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code.

[Configuring the primary language](#)

A primary language consists of program code, textual data for each licensed program ordered, and default national language cultural values.

[Configuring secondary languages](#)

A secondary language consists of textual data for all licensed programs supported for a national language version.

### Related information

[Installing, upgrading, or deleting IBM i and related software](#)

## How a language is displayed for IBM i functions

If you want information presented in a language other than the primary language of the system, you must first have a secondary language loaded. When a secondary language is loaded, you have three ways to display information in that language.

### Method 1: Placing the language you want at the top of your library list

One way to display information in a secondary language is to change the system part of your library list so the library of the national language you want is positioned before all other libraries in the system library list that contain national language information.



For example, to present the French version of textual data, you can enter the following command to place French information at the top of the library list:

```
CHGSYSLIBL LIB(QSYS2928) OPTION(*ADD)
```

To remove a library from the library list enter:

```
CHGSYSLIBL LIB(QSYS2928) OPTION(*REMOVE)
```

**Note:** The authority included with the CHGSYSLIBL command does not allow all users to run the command. As included, you must have \*ALLOBJ and \*SECADM special authority to use the Change System Library List (CHGSYSLIBL) command.

### Method 2: Creating a subsystem for the language you want

A second way to present information in a different language is to follow these steps:

1. Create a subsystem for the secondary language.
2. Define the subsystem system part of the library list entry with the national language version library for the secondary language.

All jobs running in the subsystem use textual data from the secondary language. All jobs that you submit as batch jobs have the national language version library as the first library on the system part of the library list.

### Method 3: Changing the library list for your job so that the national language version library for the secondary language is the first library on the system part of the library list

A third way to present information in a different language is to change the library list for your job so that the national language version library for the secondary language is the first library on the system part of the library list. All jobs running in the subsystem use textual data from the secondary language. All jobs that you submit as batch jobs have the national language version library as the first library on the system part of the library list.

### How a language of your choice is displayed for licensed programs

Libraries for licensed programs are either added automatically, or must be added by the user, when needed. For example, when a licensed program does not provide a translation for the primary language of the system, you need to choose a secondary language that is supported to interface with that program. After you install the secondary language, the text is found in the appropriate QSYS29xx library. After the secondary language is installed, you can use the one of the three methods that are described to change all system interfaces to the secondary language to use this licensed program. If you want to leave the rest of the interfaces in the default language and enable the secondary language for only this licensed program, you can add the QSYS29xx library to the users part of the library list. These steps enable the licensed program's translation to be found.

An example of French as a secondary language would have the text in library QSYS2928. For more information about the numbers that are used, see [National language version feature codes](#). If you want to add libraries for other licensed programs to your library list, use the CHGLIBL command.

### Related concepts

[Configuring secondary languages](#)



A secondary language consists of textual data for all licensed programs supported for a national language version.

### Related reference

[Change System Library List \(CHGSYSLIBL\) command](#)

[Change Library List \(CHGLIBL\) command](#)

## Installation preparation and national languages

IBM periodically creates program temporary fixes (PTF) to correct existing problems or potential problems within a particular IBM licensed program.

PTFs are designed to fully replace one or more objects in the licensed program. Primary and secondary languages can have language-sensitive online information PTFs.

If the primary language of your system is changed at any time for reasons other than a new release update, the cumulative PTF package of the new primary language should be at the same level as the previous primary national language. PTFs that were associated with the primary language and any secondary language must be applied again. In addition, primary language and secondary language PTFs for the online information need to be ordered by the customer.

For systems running IBM i V6R1, or later, you can set the system service language (the dedicated service tools (DST) language) to a language different from the operating system language. For more information, see [Installing, upgrading, or deleting IBM i and related software](#).

### Related concepts

[Configuring the primary language](#)

A primary language consists of program code, textual data for each licensed program ordered, and default national language cultural values.

[Configuring secondary languages](#)

A secondary language consists of textual data for all licensed programs supported for a national language version.

## Checklist: Globalization planning

When planning to install a multilingual IBM i operating system, start by completing the Globalization planning checklist for globalization and multilingual support. The checklist consists of two parts, which should be completed sequentially.

### Globalization checklist: Part 1

Before you work with a national language, answer the questions in the following table. After you have answered the questions in this table, you can then use [“Globalization checklist: Part 2”](#) on page 30 for planning for multilingual support.

Check off	Question	Response
	What national language version for the primary language are you going to install? (Refer to <a href="#">“Setting up IBM i with a national language version”</a> on page 27.)	
	What program library can it be ordered from? (Refer to <a href="#">“National language version feature codes”</a> on page 235.)	
	Are you going to use a DBCS national language version as a secondary language? (Refer to <a href="#">“Notes on secondary languages when you require English as the primary language”</a> on page 38.)	

Check off	Question	Response
	Are you aware that the latest 5250 PC emulation is necessary to support graphics data format (GDF) type?	
	What national language version for the secondary language are you going to install, if any? (Refer to <a href="#">“Notes on secondary languages when you require English as the primary language”</a> on page 38.)	
	Do you want to change your subsystem to change the language of your initial sign-on display? (Refer to <a href="#">“Notes on secondary languages when you require English as the primary language”</a> on page 38.)	
	What release level of the national language version for the primary language are you ordering? (Refer to <a href="#">“Configuring the primary language”</a> on page 35.)	
	Are the release levels of the national language version for the secondary language the same as the primary language you ordering? (Refer to <a href="#">“Notes on secondary languages when you require English as the primary language”</a> on page 38.)	

### Globalization checklist: Part 2

When you have completed [“Globalization checklist: Part 1”](#) on page 29 of the checklist, answer the additional questions in part 2.

Check off	Question	Response
	What printers support your language from a remote location?	
	What applications support your languages on the local system? (Contact your marketing support representative in your country.)	
	What applications support your languages on the remote system? (Contact your marketing support representative in your country.)	
	Do you want all your database files with the CCSID of the primary language? (Refer to <a href="#">“Database management”</a> on page 124.)	
	Do you want to work with sort sequence tables in your applications? (Refer to <a href="#">“Sort sequence support in programs”</a> on page 161.)	

Check off	Question	Response
	<p>When creating user profiles (user IDs) only certain characters are allowed.</p> <p>You can use any of the following characters in the user profile name:</p> <ul style="list-style-type: none"> <li>• Any letter (A through Z)</li> <li>• Any number (0 through 9)</li> <li>• These special characters: pound (#), dollar (\$), underscore (_), at (@). However, these characters should be avoided for globalized application systems. See <a href="#">“User profile name considerations” on page 33</a> for more information.</li> </ul> <p>See <a href="#">User profiles</a> for more detailed information about user profile considerations.</p>	

## Hardware installation and national languages

When installing or changing a device on your system, you must make sure that the device is configured correctly to reflect the keyboard ID that matches the character set and code page of the job CCSID.

Changing the keyboard configuration of a device results in different behavior, similar to adding a new display or printer to the system.

Panels, menus, and messages used by the installation process do not support right-to-left presentation of data. Therefore, online information for the installation appears left to right, in English, for bidirectional languages (such as Arabic and Hebrew).

### Related concepts

#### Configuring secondary languages

A secondary language consists of textual data for all licensed programs supported for a national language version.

#### Console device

You should make sure that your console device is configured to support the default code page of the primary language you are going to install on the IBM i operating system.

If the console device supports the code page of the new primary language, panels, messages, and online help will display properly after you change the primary language.

You must change the console device to one that supports the code page of the new primary language before doing the IPL that activates the new primary language. Make sure that autoconfig is on before doing this IPL.

#### Scenario: Console configured as a single-byte device

Your system has a primary language of English Uppercase DBCS (feature 2938). You decide to change the primary language to Japanese DBCS (feature 2962).

The existing console device on your system is configured as a single-byte-only English device using a code page of 00037. While a single-byte English device supports the installation of all other single-byte national language versions, it does not support the installation of double-byte national language versions like Japanese or Chinese. You must change the console device to one that supports the Japanese DBCS code page before doing the activation IPL.

If you do not change the console device to one that supports the Japanese DBCS code page, the IPL cannot complete.

## Workstation considerations

In a multilingual environment, different workstations support different languages on the same system.

Any data that is not tagged with CCSIDs should be stored in separate objects, unless the CCSID for each language is the same. Data that is tagged with CCSIDs (such as message files and database files) do not have to be stored in separate objects.

To correctly retrieve, process, and display data that is not tagged with CCSIDs, the application being used needs to be aware of the language differences, and how they relate to the following items:

- Programmable workstations through IBM i Access programs
- Nonprogrammable workstations

**Note:** The 3486, 3487, 3488 model V, and 3489 displays support all languages (except Thai) listed in 3486, 3487, 3488 Model V, and 3489 Keyboard and Display Part Numbers by Language.

- Keyboards
- Telnet or pass-through implications

The characters shown on your workstation depend on the keyboard type defined on your source system. If you pass through to the target system and use a virtual device with a different keyboard type, you might not see the same characters as if you were directly attached to the target system, because the target system uses another language.

## Related concepts

### Keyboard layouts

These keyboard layout samples are provided for your information. The special-character keyboard set is available only with the enhanced keyboard.

## Considerations for changing printers

When changing printers, consider the areas of data interchange, data stream, fonts, and host printer emulation.

- Interchange (z/OS® operating systems sending Advanced Function Presentation (AFP) data for DBCS to IBM i.)

AFP data containing DBCS data can be generated on the IBM i operating system. In addition, the system can receive AFP-generated data from the z/OS system containing DBCS data and print the data on Intelligent Printer Data Stream (IPDS) printers attached to the IBM i platform. The IPDS printers must be configured with \*YES specified for the AFP parameter.

- Data stream

Printers consist of SNA character string (SCS) and IPDS printers.

SNA character string (SCS) is a data stream composed of EBCDIC controls, optionally intermixed with end-user data, which is carried within a request/response unit. Host-attached SCS printers can be configured by the systems engineer or by the customer, using a diskette or selection of keys on the printer. The appropriate printer operator's guide should be used to determine how to configure the SCS printer for the language you are using.

One of the strengths of IPDS is that independent applications can create source data. The source data from independent applications is merged at the printer to create an integrated mixed data page. For example, text data can be produced on an editor, image data can be the output of a scanner stored in a folder, and graphics data be produced by the Business Graphics Utility program. IPDS makes it possible to integrate application output rather than requiring the use of integrated applications.

- Fonts

Font types for IPDS printers can be configured through the use of the Create Device Description (Printer) or Change Device Description (Printer) (CRTDEVPRT or CHGDEVPRT) commands. Fonts can be downloaded from the host or can be saved in printer storage.

For a list of the character identifier (CHRID values) supported by the various printers and languages, see the Printer Device Programming PDF.

- IBM i Access printer to emulate host printer

The IBM i Access programs support multiple languages on a single system. A IBM i Access user (except for host emulation) can use any single language of choice that is installed on the attached IBM i platform. If a IBM i Access user has a host emulation session with five different systems, the user can view a different language on each session. However, the same personal computer ASCII code page must be on all the systems.

See [IBM i Access of your environment](#) for information about installing and configuring attached PC printers.

### **Related concepts**

[Advanced Function Presentation](#)

### **Related reference**

[Create Device Desc \(printer\) \(CRTDEVPRT\) command](#)

[Change Device Desc \(printer\) \(CHGDEVPRT\) command](#)

## **Software installation and national languages**

If your system communicates with systems using different languages, you need to be careful when specifying configuration names that are exchanged with the remote system.

Do not use characters that might not be available on the keyboard used by the remote system; for example, characters such as a dollar sign (\$), pound sign (#), and an at sign (@). For an illustration of the characters that you can use in configuration names, see [“Invariant character set \(and its exceptions\)”](#) on page 298.

You should limit support of configuration names that use characters outside of the invariant character set to those already in use on existing systems.

Configuration names that might be exchanged with remote systems include:

- Network identifiers
- Location names
- Control point names
- Mode description names
- Class-of-service description names
- User IDs (from the directory entry)

For more information about software installation, see the appropriate software product books.

### **Related information**

[Installing, upgrading, or deleting IBM i and related software](#)

## **Configuring a national language version**

You must configure the national language version on your system before the system can meet your business needs in the multilanguage environment.

### **User profile name considerations**

The user profile name identifies the user to the system. This user profile name is also known as the user ID. It is the name that the user types in the *User* prompt on the Signon display.

The user profile name can be a maximum of 10 characters. The characters can be:

- Any letter (A through Z)
- Any number (0 through 9)
- In addition to these characters, three special codepoints are allowed (x'5B', x'7B', x'7C'). For many CCSIDs, including 37, these code points are interpreted as \$, #, and @. For other CCSIDs, however, these code points represent other characters. Although these code points are allowed, you should avoid using them because of the potential misinterpretation when multiple CCSIDs are used on a single

system. For example, a Spanish-speaking person using CCSID 284 might create a user profile with the name ESPA Ñ OL, but an English-speaking person using CCSID 37 might see this name as ESPA#0L.

The user profile name cannot begin with a number.

**Note:** You can create a user profile such that when a user signs on, the user ID is only numerals. To create a profile like this, specify a Q as the first character, such as Q12345. A user can then sign on by entering 12345 or Q12345 for the *User* prompt on the Signon display.

### **Related concepts**

[Packaging and installation process](#)

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

### **Related information**

[User profiles](#)

### **Service tools**

Panels, messages, and online help information for service tools are typically shown in the primary language of the system. Therefore, the workstation from which the system is being serviced must be configured to support the primary language, and the keyboard for the primary language must be attached to that workstation.

Panels, menus, and messages used by the service tools do not support right-to-left presentation of data. Therefore, online information for the service tools appears left to right, in English, for bidirectional languages (such as Arabic and Hebrew).

### **System and user interfaces**

The system interfaces and user interfaces are presented through a workstation or printer.

The workstation controller interprets keystrokes on keyboards according to the mapping determined by the KBDTYPE parameter in the device description. The display presents the data to the user, depending on the code page mapping located in the workstation controller. This code page mapping in the workstation controller is determined by the CHRID parameter in the device description. Each supported keyboard type has a character identifier assigned to it, and the default setting of CHRID in the device description (\*KBDTYPE) refers to that character identifier. Ensure the code page of the emulator is set to match the language of the system. For more information, see the help provided by the emulator.

### **Automatic device configuration**

Automatic configuration defines the local devices and some remote devices to the system.

This means that the devices attached to your system are available for use when the system is running and has a powered-on display. You do not have to use manual configuration to create configuration descriptions for the devices before you can use them. For devices that are able to send configuration information to the workstation controller, the KBDTYPE parameter is set according to the keyboard attached. If the device cannot send KBDTYPE information to the system, the QKBDTYPE keyboard system value is used.

**Note:** If you use manual configuration to set up a device with a different keyboard type than the hardware reports, automatic configuration changes the device description to match the keyboard attached. To avoid this, each time the device is powered on; you can switch automatic configuration off by setting QAUTOCFG system value to 0 (Off).

### **Related information**

[Local Device Configuration PDF](#)

### ***Automatic character set and code page conversion***

The IBM i operating system provides automatic conversion between character set and code pages for all applications that are enabled for national language support.

This automatic conversion can be controlled in the display, menu, or panel source, or through the CHRID parameter on the control language (CL) commands that create these displays. The character set and code page of the device used by the user is determined by the CHRID parameter in the device description. The CHRID value is normally set to \*KBDTYPE.

When the data to be presented is in a character set and code page different from the language of the user, automatic data conversion might occur.

### **Related concepts**

#### Working with CCSIDs

Using the system implementation of Character Data Representation Architecture (CDRA), you can achieve consistent representation, processing, and interchange of coded characters (data) on the IBM i operating system and across IBM Systems. The primary implementation of CDRA on the IBM i operating system is through coded character set identifier (CCSID) support.

#### CCSID reference information

Coded character set identifier (CCSID) is a 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

### ***Printer file conversion***

The printer provides printed output to the user. IBM i printer support does not do any conversion between the different character sets. For the data to be printed, the user must make sure that the proper character set and code page are specified in the printer and the fonts are in the printer.

If the CHRID value of the printer file is set to \*JOBCCSID, the printer joins the CHRID value of the job CCSID to the data to be printed. For externally described printer files, constants within your DDS (data description specification) are converted from the DDS source file CCSID to the character identifier of the job CCSID value.

### **Configuring the primary language**

A primary language consists of program code, textual data for each licensed program ordered, and default national language cultural values.

The primary language is the language in which the system is serviced and from which all language-dependent or cultural-dependent system values are initialized. In addition, other system objects and functions assume attributes based on the primary language. For example, messages appearing in the history log always appear in the primary language.

For each licensed program installed on the system, the national language version for the primary language is in the product library. For example, the IBM i operating system ordered in Spanish is installed in library QSYS as the primary language.

The system provides default system values for each of the primary languages. If some of the defaults do not meet the needs of your users, you can change some language-dependent system values.

### **Selecting and changing the primary language**

Choosing your primary language is important. The IBM i operating system allows you to change your primary language to accommodate your business needs based on the country in which you are operating. Keep in mind, however, that changing the primary language can take several hours or longer to accomplish.

To change a primary language on your system, you can order a different primary language from IBM. If you have a secondary language tape for the language you want as your new primary language, you can change the primary language from that tape. For example, if you have a primary language of U.S. English,



and a secondary language of Canadian French, you can use the Canadian French secondary language tape to change your primary language to Canadian French.

When you change a primary or secondary language, and want to continue receiving software and documentation updates for future releases of licensed programs that you are currently using, contact your IBM representative.

Selecting and changing a primary language affects the following operational characteristics of your system:

- Cultural values of the user.
- Language used to communicate with the system through user interfaces presented through a workstation or printer. See the figure in [“Example: How locales work”](#) on page 212.
- Implied character identifier (CHRID) of the character data stored in objects other than database files, message files, and message queues on the system.

All user-created database files have an implicit CCSID and are tagged with the job default CCSID (DFTCCSID) unless you provide a CCSID at creation time.

- If you change the primary language and the CCSID for the data remains the same, there is no effect on your system. An example is to change the primary language from the German MNCS to the Italian MNCS, of which both use CCSID 00500. The multinational character set refers to character set 00697 and code page 00500.
- If changing the primary language includes changing the CCSID value, the character data in objects other than database files might not be presented properly through the system and user interfaces. The database manager automatically converts character data unless conversion is suppressed by the application that processes the file. Data in objects other than database files are displayed correctly if the CHRID value of the display file, panel group, or menu is \*JOBCCSID.

Because some of the system values are set based on the installed primary language, you should record your current system value settings before you change the primary language of your system. Then, after you change the primary language, you can compare the current system values with the previous system value settings.

When you change the primary language of your system, the CCSID of the text fields in the system-supplied output files might also change. This is because the CCSID is dependent on the installed primary language.

### **Related concepts**

#### Setting up IBM i with a national language version

The steps to install and configure a national language version on the IBM i operating system include selecting and installing hardware, installing software, and configuring your environment to run in a globalized setting.

#### Installation preparation and national languages

IBM periodically creates program temporary fixes (PTF) to correct existing problems or potential problems within a particular IBM licensed program.

#### Default system values for national language versions

Jobs and functions on the IBM i operating system use system values as default values.

#### System values

The system values of the primary language on the system are used as system-wide cultural and linguistic defaults. Therefore, if you change the primary language on the system, each varying system value resets to the default system value of the new primary language.

### **Related information**

#### Changing the primary language of your system or logical partition



## Configuring secondary languages

A secondary language consists of textual data for all licensed programs supported for a national language version.

When you install a secondary language, the textual data for licensed programs installed on your system is copied into the secondary language library. See the chapter called "Installing a Secondary Language" in the Software Installation PDF for instructions on installing secondary languages.

The program code is not included in the secondary language version.

## Secondary language environments

Some multilingual environments have more than one national language version installed. To have a single system support multiple languages, you must have the associated hardware installed. You must also have sufficient disk storage space available to contain all of the system and application textual data for the secondary languages. The amount of disk storage space that is required varies by language and application, but it is typically somewhere in the range of 50 to 300 MB.

The languages currently supported on IBM i as either primary or secondary languages can be found in National Language Version (NLV) feature codes. Listed are the national language versions, their feature codes, and the program libraries from which they are available.

Each of the national language versions available from the program library (primary or secondary) include cultural- and language-dependent system values for that particular language. Date format, date and time separators, code page and character set, and keyboard types are examples. The system values are initially set to the cultural values of the primary language. By setting up a subsystem, however, you can ensure that the cultural values for the secondary languages are set properly for users of the secondary languages.

Applications can use language values that are available in message CPX8416, in file QCPFMSG, accessed using the library list. Message CPX8416 gives the correct values for the primary or secondary language, depending on the library list.

Except for logical partitioning (LPAR), when you use a multilingual environment, the primary language version and any secondary languages must be at the same release level. You must also order and install the correct devices (workstation controllers, display stations, and printers) to support your languages.

## Related concepts

### Setting up IBM i with a national language version

The steps to install and configure a national language version on the IBM i operating system include selecting and installing hardware, installing software, and configuring your environment to run in a globalized setting.

### How a language is displayed for IBM i functions

If you want information presented in a language other than the primary language of the system, you must first have a secondary language loaded. When a secondary language is loaded, you have three ways to display information in that language.

### Installation preparation and national languages

IBM periodically creates program temporary fixes (PTF) to correct existing problems or potential problems within a particular IBM licensed program.

### Hardware installation and national languages

When installing or changing a device on your system, you must make sure that the device is configured correctly to reflect the keyboard ID that matches the character set and code page of the job CCSID.

### National language version feature codes

This table lists the available national language version feature codes on the IBM i operating system. When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code.

### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### ***Notes on secondary languages when you require English as the primary language***

These considerations are important when you require English as the primary language and want to install DBCS secondary languages.

### **If Japanese, Simplified Chinese, Traditional Chinese, or Korean is used as a secondary language and English is required as the primary language**

If Japanese, Simplified Chinese, Traditional Chinese, or Korean is used as a secondary language and English is required as the primary language, use English Uppercase and Lowercase (2984) as a primary language.

### **In other cases**

In cases other than those described in this topic, use English (2924) as a primary language.

### ***Enabling the secondary language***

You must ensure that secondary languages can be used after they have been installed on the system.

To enable the secondary language, follow these steps:

1. Add the secondary language library to the beginning of the user's system part of the library list.

To do this, use one of the following ways:

- Use the Change System Library List (CHGSYSLIBL) command to add the national language library you want to the top of the library list.

The command can be in an initial program specified in the user profile so that the user does not have to enter the command at every sign-on.

The authority included with the CHGSYSLIBL command does not allow all users to run the command. To enable a user to run the CHGSYSLIBL command without granting the user rights to the command, you can write a CL program containing the command. The program is owned by the security officer and adopts the security officer's authority when the program is created. Any user with authority to run the program can use it to change the system part of the library list in the user's job.

- Use a separate subsystem for a secondary language. To do this, follow these steps:
  - a. Create a subsystem description for secondary language users (for example, QGPL/DANISH).
  - b. Specify the secondary language library for the Subsystem library (SYSLIBL) attribute (for example, QSYS2926).
  - c. Specify the sign-on display file from the secondary language library for the Sign-on display file (SGNDSPF) attribute (for example, QSYS2926/QDSIGNON).
  - d. Use the Remove Work Station Entry (RMVWSE) command to remove the appropriate display devices from the interactive subsystem, and then use the Add Work Station Entry (ADDWSE) command to add these devices to the secondary language subsystem.

When you use these commands, no one can be signed on to the devices that you are removing.

- e. If you want to use separate job queues (JOBQ) and output queues (OUTQ) for a secondary language, you can create these queues in the secondary language library (for example, QSYS2926). Attach the job queue to the secondary language subsystem (for example, QGPL/DANISH).
- You might have licensed programs that have secondary language libraries and that are not on the IBM i secondary language tape. You should add those secondary language libraries to the library list before the primary language product libraries. Use the Change System Library List (CHGSYSLIBL) command to add the secondary language libraries to the library list if the product libraries are in the system part of the library list.

2. Specify the keyboard ID for the secondary language in the device description for the display station.
  - a) Turn off your device.
  - b) Use the Change Device Description Display (CHGDEVDSP) command to specify the keyboard ID for the secondary language in the device description.
  - c) Use the Vary Configuration (VRYCFG) command to turn on the device.

3. Change the date format to reflect the date format of your language.

The date format, date separator, and time separator can only be changed using the CHGJOB command for secondary language users. If you use the CHGSYSVAL command to change these values, all primary language users and all secondary language users have this information changed. The following table illustrates this and shows the ways the date and other NLS-related job attributes should be specified for secondary language users.

	CHGJOB	CRTJOB	CHGJOB	CRTUSRPRF	CHGUSRPRF
Date	X	X	X		
Date format	X				
Date separator	X				
Time separator	X				
Character set identifier	X			X	X
Language identifier	X			X	X
Sort sequence	X			X	X
Country or region identifier	X			X	X

**Note:** The following commands are used in this table:

- Change Job (CHGJOB) command
- Create Job Description (CRTJOB) command
- Change Job Description (CHGJOB) command
- Create User Profile (CRTUSRPRF) command
- Change User Profile (CHGUSRPRF) command

4. Change the CCSID value to reflect the CCSID of the secondary language that you want to use.

You can set the CCSID value for all jobs to run under your user profile by using the Change User Profile (CHGUSRPRF) command. This change takes effect for any jobs that enter the system using your profile after you have made the change.

You can set the CCSID value for a batch job to be run using the CCSID parameter on the Submit Job (SBMJOB) command. You can change the CCSID of a job that is running by using the Change Job (CHGJOB) command.

5. Ensure that your data in objects other than database files and message files prints correctly.

To do this, you might want to direct all of your printed output to a print queue that contains printer output only for the character identifier of your language.

- a) Use the Create Output Queue (CRTOUTQ) command to create a printer queue.
- b) Use the OUTQ parameter of the Change Job (CHGJOB) command to change your job output queue.

```
CHGJOB OUTQ(output_queue)
```

**Note:** You can use the Change User Profile (CHGUSRPRF) command instead to make a more permanent change to the OUTQ parameter. Then, each time you sign on to the system, the correct output queue is used.

If the printer supports changing the code page, you can use the \*JOBCCSID value in the printer file.

6. Change other cultural- and language-dependent values to the secondary language you want to use if you do not want to use the system values.

Use the Change Job (CHGJOB) command to change the cultural- and language-dependent values.

### **Related concepts**

#### Working with CCSIDs

Using the system implementation of Character Data Representation Architecture (CDRA), you can achieve consistent representation, processing, and interchange of coded characters (data) on the IBM i operating system and across IBM Systems. The primary implementation of CDRA on the IBM i operating system is through coded character set identifier (CCSID) support.

#### Linguistic and cultural values

Linguistic and cultural conventions include any system values, attributes, or settings that can be altered to suit a country or language.

### **Related reference**

[Change System Library List \(CHGSYSLIBL\) command](#)

[Create Output Queue \(CRTOUTQ\) command](#)

[Change Job \(CHGJOB\) command](#)

### **Multilingual support**

Multilingual support on the IBM i operating system is the support that includes more than one language on one system.

A system that works in multiple languages must be able to handle a variety of cultural and linguistic characteristics such as the following:

- Graphic characters, such as an e accent grave (è)
- Currency symbols, such as the Pound Sterling symbol
- Date formats, such as 24.06.93
- Time formats, such as 23:59
- Sort sequences, such as a, b, c....

The system must also handle differences, such as the direction in which text prints and displays. For example, all text of Latin-based languages, such as French and Spanish, displays from left to right across a display. However, the general direction of Arabic and Hebrew text is from right to left across a display. The system displays text, prints text, and allows data entry left to right for some languages and right to left for other languages.

Printing and displaying text left to right for some languages and right to left for others is not enough, though. Numbers and Latin character phrases that are included in Arabic and Hebrew text display and print from left to right. For example, Hebrew text generally flows from right to left across a display. When Hebrew text includes a street address, the street name flows right to left, but the address number flows left to right. Similarly, if Hebrew text includes a Latin name, such as John Smith, the Latin name flows from left to right. Because this text flows both right to left and left to right (bidirectionally), the system displays and prints text bidirectionally.

### **Multilingual network**

Two or more systems, each using a different primary language, can interchange data. Because data is flowing between systems with different primary languages, the data must have a CCSID assigned. When data has a CCSID assigned, data integrity is maintained. Thus, character data is correctly displayed for the receiving user.

## Installing and enabling locales

If you are installing a new release, you can request that library QSYSLOCALE be installed on the system at that time.

To install library QSYSLOCALE at a later time, type GO LICPGM and press the Enter key. Scroll until you find *Extended NLS Support*. Select option 1 to install *Extended NLS Support*.

Locales can be enabled on the system by using system values or user profiles.

## Enabling locales with system values

Two system values are related to locales:

### QLOCALE

The system value specifying the locale object. The default is \*SYSVAL. Other possible values are:

- \*C  
The C locale is assigned for this user (same result as using \*POSIX)
- \*POSIX  
The POSIX (Portable Operating System Interface for Computer Environments) locale path name is assigned for this user.
- locale path name  
The path name of the locale to be assigned for this user.

### QSETJOBATR

A system value that sets job attributes at job start time. The default is \*SYSVAL. The following values indicate the job attributes that are to be set from the locale object specified by QLOCALE:

- \*CCSID (Coded character set identifier)  
The CCSID associated with a locale when the locale object is created.
- \*DATFMT (Date format)  
The date format is determined from the locale object.
- \*DATSEP (Date separator)  
The date separator is determined from the locale object.
- \*SRTSEQ (Sort sequence)  
The sort sequence is determined from the locale object
- \*TIMSEP (Time separator)  
The time separator is determined from the locale object.
- \*DECfmt (Decimal format)  
The decimal format is determined from the locale object.

## Enabling locales with user profiles

Two parameters on the user profile are related to locales:

### LOCALE

The parameter value specifying the locale object to use for the LANG environment variable. The default is \*NONE. Other possible values are:

- \*SYSVAL  
The system value QLOCALE is used to determine the locale path name to be assigned for this user.
- \*C  
The C locale is assigned for this user (same result as using \*POSIX).
- \*POSIX

The POSIX locale path name is assigned for this user.

- locale path name

The path name of the locale to be assigned for this user.

### **SETJOBATR**

The parameter value that sets job attributes at job start time. The default is \*NONE. If \*SYSVAL is specified, then the attributes are set from the QSETJOBATR value. The same attributes (\*CCSID, \*TIMSEP, \*DATFMT, \*DATSEP, \*DECfmt, \*SRTSEQ) that can be specified on the system value QSETJOBATR can be specified on the SETJOBATR parameter of the user profile.

If you want all users on the system to use locales, setting system values accomplishes this. Alternatively, the user profile is an ideal mechanism if you want to provide locale function to a limited or specific group of users.

### **Related concepts**

#### Locales

A *locale* is an object that can determine how data is processed, printed, and displayed.

#### System-supplied locales and recommended CCSIDs

The system-supplied locale source definition file members are in the optionally installable library QSYSLOCALE in the QLOCALESRC source file. The source file members are encoded in CCSID 37 and are read only.

#### Working with locales

Locales are used primarily in ILE-based application programs. Additionally, the Retrieve Locale Information (OPM, QLGRTVLC; ILE, QlgRetrieveLocaleInformation) API retrieves one or all categories of a locale.

## **Scenarios: Setting up IBM i with a national language version**

These scenarios demonstrate how you can enable multilingual support on the IBM i operating system.

**Note:** For more information about the details described in the scenarios, see the following topics:

- For Unicode database information relating to DDS, see [DDS for physical and logical files](#).
- For Unicode display information with DDS, see [DDS for display files](#).
- For Unicode printing information with DDS, see [DDS for printer files](#).
- For information about using the subsystem description, see [Enabling the secondary language](#).

### **Related concepts**

#### Working with Unicode

*Unicode* is a standard that precisely defines a character set as well as a small number of encodings for it. It enables you to handle text in any language efficiently. It allows a single application to work for a global audience.

### **Scenario: A single system supporting Spanish**

In this scenario, a single system supports Spanish users and applications.

The primary language of the system is Spanish (NLV 2931). Because 2931 is the primary language, the default system settings and IBM i localization preference are set to Spanish.

The user has also created a database file where the fields of interest are defined to contain Unicode, because they plan to use this same database file for both 5250 applications and Java applications.

The following example shows the SQL statement used to create a database containing a Unicode field named PART\_NAME and a non-Unicode field named STOCK\_NUMBER:

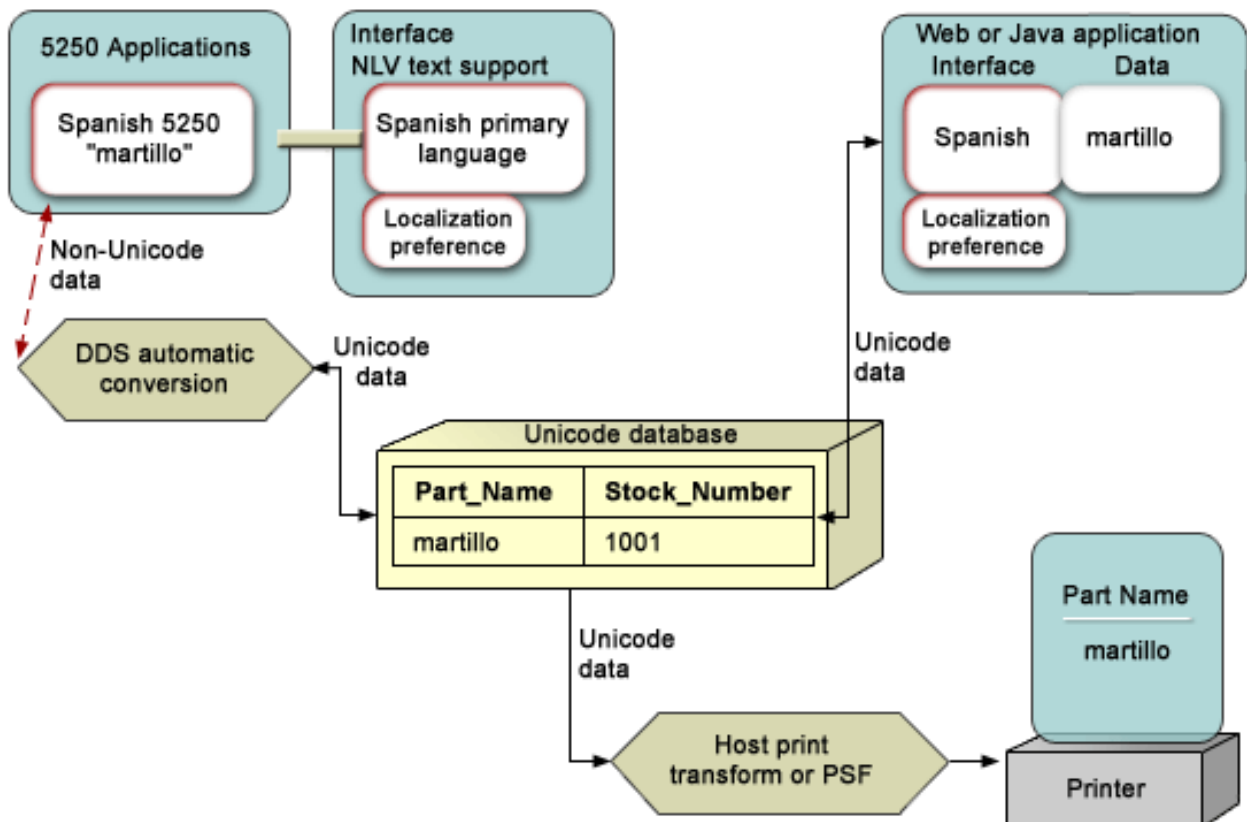
```
CREATE TABLE SAMPLE (PART_NAME GRAPHIC (10) CCSID 1200
NOT NULL WITH DEFAULT, STOCK_NUMBER INT NOT NULL WITH DEFAULT 0)
```

If the user wants to display this data with a web service or Unicode enabled application, then Unicode is the natural encoding for web use, and no conversion is needed. To get the correct localization preference for the Java application, the user sets the Java locale to sp\_SP for Spanish in Spain.

If the user wants to display this data with a 5250 session, then the Unicode field must be converted to the CCSID of the display device. The user only has to set the user profiles's CCSID value to 284 to tell the system that this user is on a Spanish display. This service is provided automatically by the system if requested with the CCSID keyword and the \*CONVERT parameter in DDS.

To print the Unicode data, the user specifies the \*NOCONVERT parameter of the CCSID keyword, and uses the FONTNAME keyword to specify a TrueType font. The unconverted Unicode data can be printed with PSF or with Host Print Transform.

The following figure illustrates this scenario.



**Scenario: A single system supporting Spanish and an existing EBCDIC database**

In this scenario, a single system supports Spanish users and applications and an existing EBCDIC database.

The primary language of the system is Spanish (NLV 2931). Because 2931 is the primary language, the default system settings and IBM i localization preference is set to Spanish.

The user has also created a database file where the fields of interest are defined to contain Unicode, because they plan to use this same database file for both 5250 applications and Java applications. They also have an existing database in which the fields are defined in EBCDIC.

The following example shows the SQL statement used to create the EBCDIC database:

```
CREATE TABLE SAMPLE (PART_NAME CHAR (10) CCSID 284 NOT NULL WITH DEFAULT,
  STOCK_NUMBER INT NOT NULL WITH DEFAULT 0)
```

The following example shows the SQL statement used to create a database containing a Unicode field named PART\_NAME and a non-Unicode field named STOCK\_NUMBER:

```
CREATE TABLE SAMPLE (PART_NAME GRAPHIC (10) CCSID 1200 NOT NULL WITH DEFAULT,  
STOCK_NUMBER INT NOT NULL WITH DEFAULT 0)
```

### **When using the Unicode file**

If the user wants to display this data with a web service or Unicode enabled application, then Unicode is the natural encoding for web use, and no conversion is needed. To get the correct localization preference for the Java application, the user sets the Java locale to sp\_SP for Spanish in Spain.

If the user wants to display this data with a 5250 session, then the Unicode field must be converted to the CCSID of the display device. The user only has to set the user profiles's CCSID value to 284 to tell the system that this user is on a Spanish display. This service is provided automatically by the system if requested with the CCSID keyword and the \*CONVERT parameter in DDS.

To print the Unicode data, the user specifies the \*NOCONVERT parameter of the CCSID keyword and uses the FONTNAME keyword to specify a TrueType font. The unconverted Unicode data can be printed with PSF or with Host Print Transform.

### **When using the EBCDIC file**

If the user wants to display this data with a web service, then the file first must be converted to Unicode. This can be done with the JDBC connector. To get the correct localization preference for the Java application, the user sets the Java locale to sp\_SP for Spanish in Spain.

If the user wants to display this data with a 5250 session, EBCDIC is the natural encoding for the 5250 device and no conversion is needed. To print the EBCDIC data, the user sends the data to the printer. Because EBCDIC is the default encoding for the printer, no conversion is needed.

### **Logical file support**

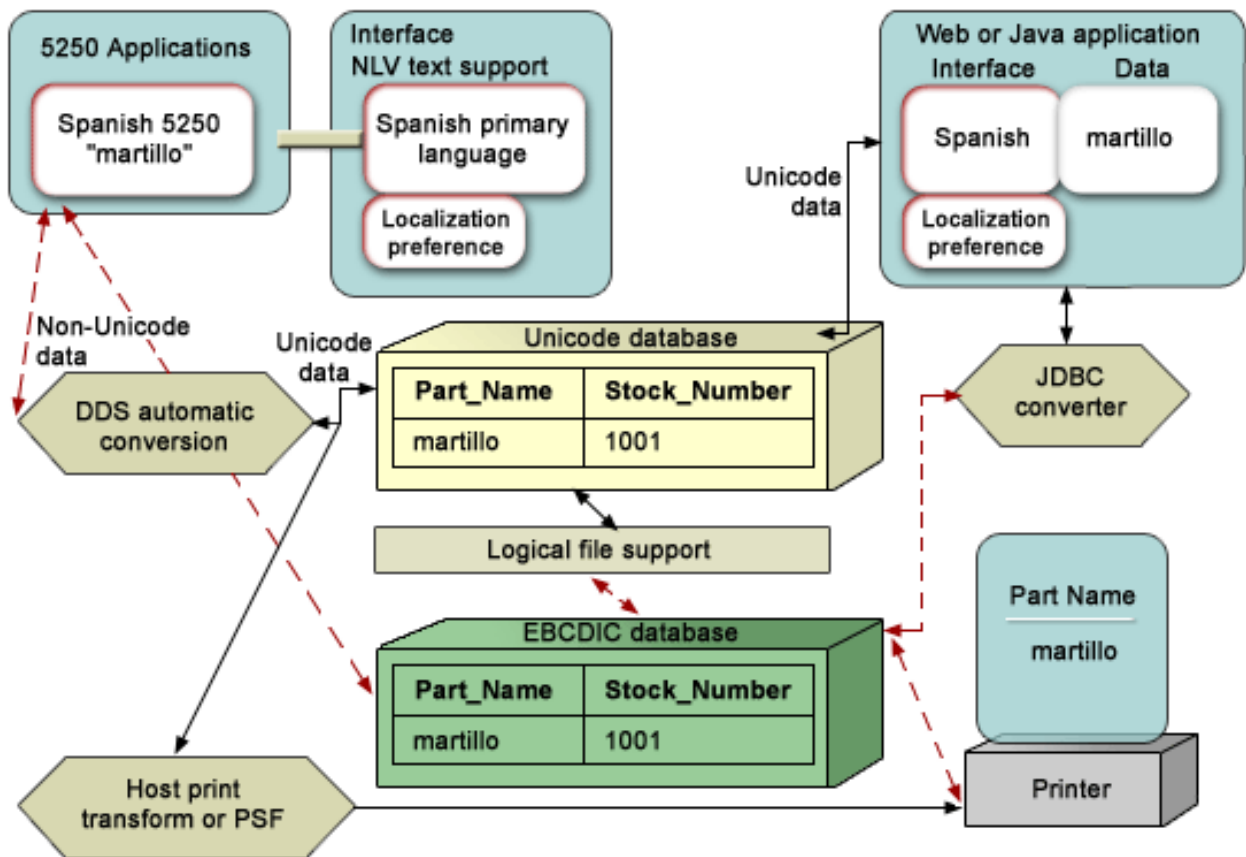
One of the unique features of IBM i is the ability to use the system's logical file support to have either the EBCDIC file appear to the application as a Unicode file, or to have a Unicode file appear to the application as an EBCDIC file. This might be of use if you want to move your database to Unicode, but do not want to update your existing applications.

If the majority of your application's use of the database involves Unicode, you can have the data stored as Unicode, and create a logical view of the file in EBCDIC. You can then have your EBCDIC programs access this logical file and they do not need to be updated to handle Unicode.

If the majority of your application's use of database involves EBCDIC, you can have the data stored as EBCDIC, and create a logical view of the file in Unicode. You can then have your Unicode programs access this logical file and they do not need to be updated to handle EBCDIC. However, because EBCDIC encodes a smaller set of characters than Unicode does, some character loss might occur.

The following figure illustrates this scenario.





### Scenario: A single system supporting English, Japanese, and German

In this scenario, a single system supports English, Japanese, and German users and applications.

The primary language of the system is English (NLV 2924). The system has also been loaded with secondary languages of Japanese (NLV 2962) and German (2929). Because 2924 is the primary language, the default system settings and IBM i localization preference is set to English. Because these three NLVs are installed, each user can work with the system in English, German, or Japanese.

The users see their language of choice and IBM i localization preference from the initial sign-on screen by the use of a subsystem description for each secondary language.

The user has also created a database file in which the fields of interest are defined to contain Unicode. Because Unicode provides a unique number for every character on any platform, in any program, and in any language, one field can contain English, German, and Japanese.

The following example shows the SQL statement used to create a database containing a Unicode field named "PART\_NAME" and a non-Unicode field named "STOCK\_NUMBER":

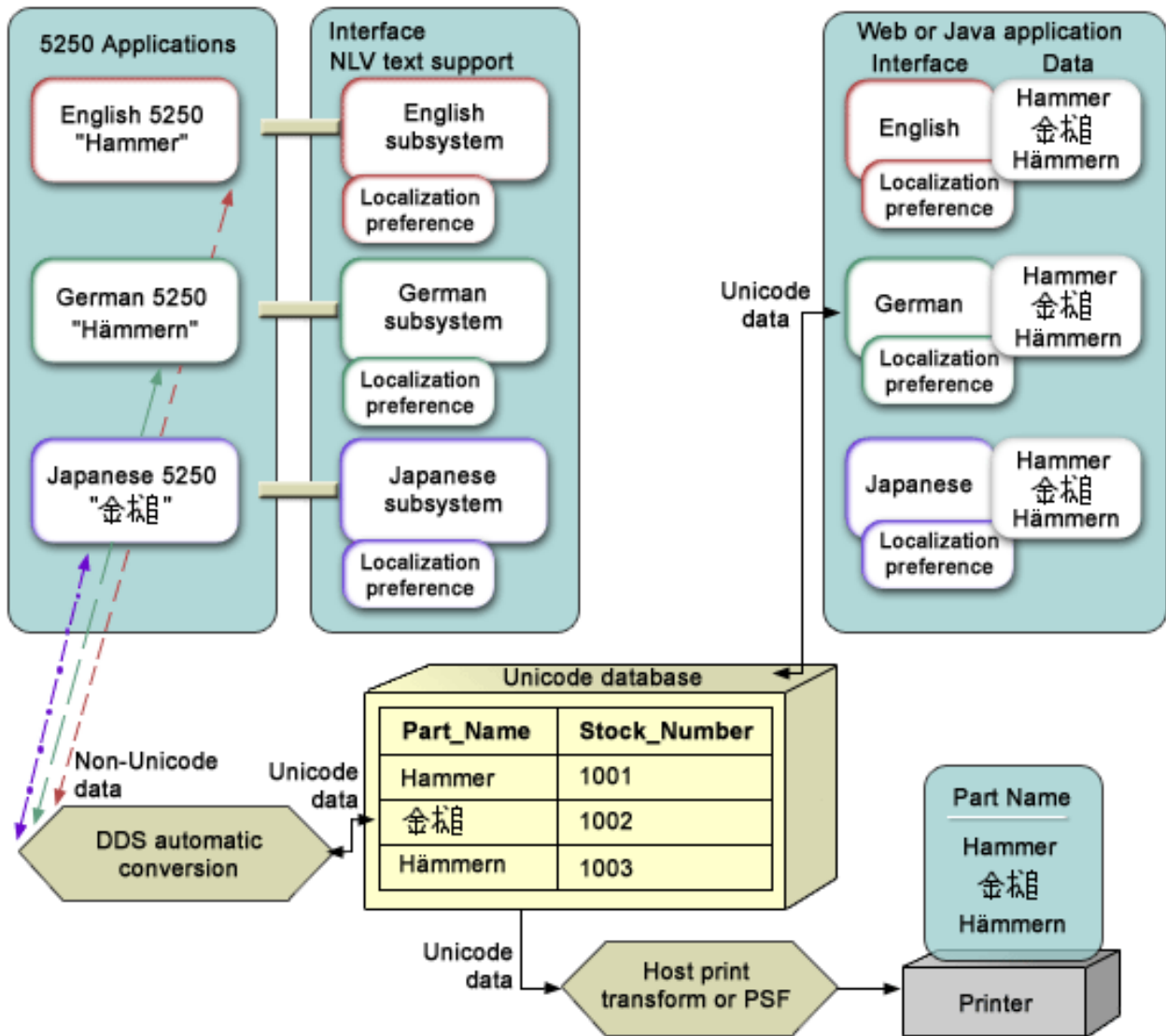
```
CREATE TABLE SAMPLE (PART_NAME GRAPHIC (10) CCSID 1200
NOT NULL WITH DEFAULT, STOCK_NUMBER INT NOT NULL WITH DEFAULT 0)
```

If the user wants to display this data with a web service or Unicode enabled application, then Unicode is the natural encoding for web use and no conversion is needed. To get the correct localization and interface preference for the Java application, the user needs to set the Java locale to the correct value: en\_US for English, Jp\_JA for Japanese, and de\_DE for German.

If the user wants to display this data with a 5250 session, then the Unicode field must be converted to the CCSID of the display device. The user only has to set the user profile's CCSID value to the correct value (37 for English, 1399 for Japanese, and 278 for German) to tell the system what the user's preference is for the display. This service is provided automatically by the system if requested with the CCSID keyword and the \*CONVERT parameter in DDS.

To print the Unicode data, the user specifies the \*NOCONVERT parameter of the CCSID keyword and uses the FONTNAME keyword to specify a TrueType font. The unconverted Unicode data can be printed with PSF or with Host Print Transform.

The following figure illustrates this scenario.



## Developing globalized applications

Globalized applications are applications that have national language support. National language support allows users to enter, store, process, retrieve, print, and display data in their chosen language. It also allows users to see and enter commands, prompts, messages, and documentation in their chosen language, in formats matching their cultural expectations.

Here are some guidelines for designing, developing, and delivering globalized applications:

- Designing functions that are sensitive to national languages
- Supporting various types of hardware
- Translating the textual data in your application
- Making your application available worldwide

Although your reasons might differ, most internationalized applications are created because:

- The market demands globalized software products that have a local feel
- The application is used in a community that represents multiple cultures
- Revenue opportunities are expanded

### **Related concepts**

#### Character processing

Character processing on the IBM i operating system is controlled by specific coding rules and guidelines that ensure consistent processing of character data.

#### Handling data in globalized applications

The IBM i operating system enables you to handle data in a globalized environment. This topic collection describes Unicode and Unicode data, the Chinese standard GB18030, how to use CCSIDs to integrate multiple language environments consistently, and how to use bidirectional data, DBCS data, and locales.

## **Goals and processes**

Before you invest your time and money in the development of globalized applications, you should set up a planning process to consider how to serve your users well.

### **Globalization development goals**

This information assumes certain goals and provides you with recommendations for developing globalized applications.

The recommendations in this topic assume that your basic goals are:

- To create an application efficiently.
- To create an application at minimal expense. You can retrofit existing applications for globalization and create new applications designed for globalization. Designing an application for globalization, however, is typically less expensive than retrofitting an existing application.
- To ensure that the application design does not interfere with the current or planned design of other internationalized applications.
- When creating an application with national language support, you must plan for or put into effect the following tasks:
  - Designing functions that are sensitive to national languages
  - Supporting various types of hardware support
  - Translating the textual data in your application
  - Making your application available worldwide.

### **Globalization development planning processes**

A globalized application should be well planned at every stage in order to save time, effort, and money. You should not have to recompile programs nor repackage data objects. Your product might, however, be required to use a different data object based on the language version you are using. You should have one set of program code and different sets of cultural- and text-dependent code, as needed.

Consider these processes when planning for a globalized application.

### **Market research process**

In the market research process, you must determine for whom you are designing and developing globalized applications. To find the answer, you can ask yourself and your potential customers these types of questions.

### **What are my target markets for today and tomorrow?**

The answer to this question makes a significant difference if you define your market place in different countries or only in the area of your own language, or if you decide to include countries speaking other

languages. For example, if you are coding an application from a Latin-based language, application complexity increases when you decide to include countries using non-Latin languages such as Hebrew, Chinese, or Japanese. The application complexity increases because you need to deal with incompatible characters sets and more complex input methods.

Along with the language problem, there are other areas to consider. You need to understand the culture, habits, ways of doing business, and laws of the target markets. You need to understand the customers' ways of life for you to be accepted as a business partner, to be able to get into the market, and to support them in their countries.

These factors can affect:

- The skills that you need (technical, cultural, language, laws)
- The environments to consider
- Your company structure and support organization
- Your relationship to other companies
- The resources that you need (people, time, and money)

### **Who are the users of my application?**

You must understand the requirements that future users of your application will have. For example, do they want to:

- Work with separate databases for different languages?
- Work with a shared database for all languages?
- Exchange or consolidate data?
- Work with different languages dependent on the user, the company, or the company's customers?
- Use end-user database tools to do their own inquiries on the application database?

All these factors affect the design you choose, the way your application is able to switch from one language environment to another, and how data presentation and conversion take place.

### **How much globalization support is needed?**

After you understand the requirements for your customers and their end users, you can decide what kind of cultural-sensitive information you need to store and maintain, the type of data presentation, which parts you need to translate, and how your application must be able to be integrated in the different environments.

### **What is the cost of the effort?**

To estimate the expected revenue, analyze the places you have chosen as your target market. After you know the requirements, you should be able to determine the effort and costs. This amount allows you to compare the costs against the expected revenue.

### **Which costs more, enabling or retrofitting an application?**

The initial cost of enabling an application for national language support might be higher. But consider that the enabling steps are based more on normal modular and data-driven design techniques, which improve the quality of your application even without NLS enabling. Because a good design helps people to understand and describe the application system, you will receive a certain return on the investment. A good design helps to improve productivity of development and maintenance. You have the additional

effort of designing and implementing the application only once, even for many different language versions. Compared to retrofitting an existing application, it is much less expensive to plan and design it from the very beginning.

### **Development process**

Before you start to develop NLS-enabled applications, you need to consider initial education and the implementation of internationalized applications.

### **Education for developing internationalized applications**

When you intend to develop NLS-enabled applications, you need to consider additional initial education. The following topics are important to learn about:

- General globalization concepts
- Available globalization support on the IBM i operating system
- Available globalization support on other systems and applications with which your application operates
- Isolation of different parts of an application
- Data presentation corresponding to cultural conventions
- Design and coding for textual data parts
- Translation process
- Product and system integration
- Packaging, installation, and setup
- Product support and maintenance

Based on the globalization enabling guidelines, first prepare a prototype application and test the chosen way of implementing the application for your specific environment. Afterward include the globalization enabling guidelines in your general application development processes, guidelines, and standards.

### **Implementing internationalized applications**

When implementing an internationalized application, the most important objective is to produce only one set of running code. You must differentiate consistently between running code and textual data. It is essential that you standardize the chosen approach throughout the whole application. Work with unique and clearly defined naming conventions. To understand and to maintain this information in the application, handle parameters called from a program in a consistent way.

### **Documentation process**

Documentation should provide information for the end users of the application system in their own language. The documentation should also include installation, setup, and customization information for the user, the system operator, and the application system manager.

The user documentation should be textual data that can be easily translated. Whenever possible, combine the online help information and user documentation to reduce the volume of text to translate. Any example displays or print layouts should be produced by the application and included in the documentation.

### **Translation process**

Translating the textual data is a time-consuming process. The textual data should be available to translators early at the development stage, even before the code is stable. When planning for translation, you need to consider translation tools, education, guidelines, instructions, and the glossary as well as physical equipment.

## Physical equipment

Each translator should have equipment compatible with the language being translated. The display stations and keyboards should have all the characters needed to translate, and the printers should be able to print the translated text.

## Translation tools

Provide the translators with tools that increase productivity and that prevent translation of non-textual application data. When purchasing or developing a translation tool, the following features should be included:

- An editor that provides the ability to show displays that can be seen by the user, and the ability to translate the textual data on the system but still protect the parts of the application that are not textual data. The editor should also include functions such as scan and replace, find, copy, move, and delete.
- A dictionary function to provide consistency of words and phrases throughout the product.
- A validation process to check translation errors that might cause the application to malfunction.
- A merge function that provides the ability to merge the translated text into a new version of the original text. This merge function allows for translating only new text, and saves time and effort.
- A print function for validation purposes.

## Translation education

It is important that translators are familiar with the product they are translating and also with the tools they are using. The translation process is not the replacement of one word with another, but the formation of concepts in another language. Knowledge of the product being translated provides more understandable products to the user. Time and resources for educating translators should be planned well in advance.

## Translation guidelines and instructions

Translation guidelines and instructions should be provided to ensure correct translation. For example, to translate an error message properly, it is important to know in what context this message is displayed. A note to translators telling them what error caused the message to be displayed also helps.

## Translation glossary

To ensure accurate translation, use terminology based on definitions in standard, widely available, dictionaries. If your application uses terms not found in standard dictionaries or terms that are used differently from standard definitions, provide a glossary of non-standard terms to the translators. Avoid using abbreviations and acronyms in your application. If you must use abbreviations or acronyms in your application, define them in the glossary. Remember, abbreviations and acronyms that are obvious in your language might not be obvious in another language.

## Testing process

The testing of a globalization-enabled product involves testing the running code, checking the textual data, and integrating the running code and textual data.

1. Test the running code

The running code should be tested in a globalization support environment in order to check all the possible language-dependent combinations. Translators should not test the product functionality.

2. Check the textual data

The textual data should be tested to check correct translation and consistency throughout the product.

### 3. Integrate the running code and textual data

After the textual data and the code have been tested separately, an integration test should be performed to test if the application has taken into account all the globalization-related processing, and that the translation of the textual data has not caused a malfunction in the product.

If your application will also run on a multinational or multilingual system, a separate test that includes more than one set of textual data should be planned.

### **Packaging and installation process**

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

- Store the running code and textual data separately.
- Package the textual data so that customers receive only the textual data in the languages that are ordered. (If the textual data for all languages is sent to all customers, it will waste system resources and lead to maintenance problems.)
- Provide comprehensive installation documents (translated to the language of the person installing the product) to avoid unnecessary operator-related problems and also to avoid the wrong impression right at the beginning that the application is not reliable.

Installation documentation should cover the following topics:

- What is needed to install and run the application, such as hardware and software requirements.
- How to install the application, and how to recover when things go wrong.
- What changes need to be made regarding:
  - Subsystem definitions
  - Device descriptions
  - User profiles
  - System values
  - Library lists
- What are the application limitations?

### **Related concepts**

#### Subsystems

A subsystem is a single, predefined operating environment through which the system coordinates the work flow and resource use.

#### Device descriptions

These control language (CL) command parameters can be used to change cultural and linguistic conventions for some display and printer devices.

#### User profile name considerations

The user profile name identifies the user to the system. This user profile name is also known as the user ID. It is the name that the user types in the *User* prompt on the Signon display.

#### Default system values for national language versions

Jobs and functions on the IBM i operating system use system values as default values.

#### Job library list

The language used for textual data (displays, messages, printed output, and online help information) is controlled by the library list for the job.

#### Delivering globalized applications

As you prepare to deliver your globalized application, you should consider how globalization issues might affect the ways that your customers install and use your application.

### Application maintenance process

Consider these points when planning for the maintenance of a multilingual application.

- The running code must be maintained separately from the textual data. These separate components must be fully synchronized. A redesign in one component might cause a redesign to be made in another.
- Whenever textual data is changed, be sure that it is incorporated in all the languages to which your textual data was translated. In this way, you can ensure a single maintenance level for the complete product.
- Be sure to test the running code for each textual data change that you distribute.

## Designing globalized applications

Your goal in designing international application components is to create components that support national languages independently.

The support of one language should not interfere with the support of another language. The support of one language should not force any reduction in the function of the product for another language.

Your application should be able to support multiple languages simultaneously. For example, support for a double-byte coded character set (DBCS) language should not exclude support for single-byte coded character set (SBCS) languages. When you set up your libraries, consider using multiple textual data libraries, which can be dynamically allocated for testing, packaging, and delivery.

As you develop a globalized application for the IBM i operating system, you must consider these and other unique design considerations that will affect the way you build and code your application.

### Related concepts

Developing applications that process DBCS data

You should design your application programs for processing double-byte data in the same way you design application programs for processing alphanumeric data.

### Checklist: Application design

The checklist provides some guidelines that you can follow when you create an application with national language support.

Complies	Not applicable	Rule
		The existence of a specific character set within a system or its components must not be assumed.
		Converting character case must be definable for each language and code page.
		Folding must be definable for each language and code page. Folding is the process in which characters that can be printed or displayed are substituted for those that cannot be printed or displayed on a particular device.
		The use of a graphic character for software control purposes must not preclude the use of the same character in the text of messages, menus, prompts, input fields, or output fields.
		The set of characters allowed for use in the entry of data must be definable by the system operator, a user, or an application.
		Graphic symbols and icons must be translatable.
		All characters on the active code page must be accessible.



Complies	Not applicable	Rule
		Language-dependent parts of a product must be isolated from non-language-dependent parts for easy modification.
		The design of a product must allow for the national language support of the various components of the product to be independent of each other.
		National language exits must be provided at strategic points.
		Diagnostics must be enabled.
		Logical layouts different from a given physical keyboard layout must be available to the user.
		All user interface text and presentation control information must be isolated from the running code.
		Functions dependent on display field length and display field position, or display field position alone, must not be designed in such a way that they are affected by user-interface text expansion.
		A method must be provided to allow for the identification and tracking of panels and messages during the translation process.
		Variables must be permitted to assume any location and order within a display field.
		Messages and other displayed words or phrases must be complete entities and must not be constructed from individual words or phrases.
		Entry of end-user commands, keywords, or responses must be possible without regard to uppercase or lowercase characters.
		A product with national language-dependent functions must be designed to facilitate the addition of other countries or national languages.
		Lowercase alphabets should not be assumed to be invariant.
		Character sets should be definable by the operator, a user, or an application.
		Special characters, including punctuation marks, should be definable and not program dependent.
		User-interface text modules should be packaged separately from the running code.

### Globalization and localization

National language support enables users to interact with the IBM i operating system in the language of their choice, with results that are culturally acceptable. National language support consists of two parts: globalization and localization.

The IBM i operating system controls the operation of programs and provides services such as controlling resources, scheduling jobs, controlling input and output, and managing data. It is designed to complement and extend the capabilities of the system to provide fully integrated support for interactive and batch applications.

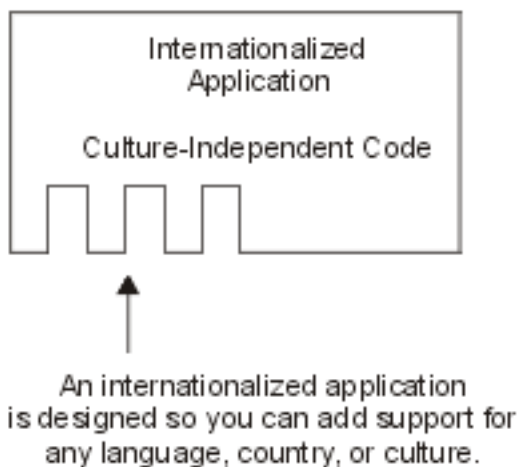
Many functions of the operating system apply directly to interactive data processing. Some of the functions are listed as follows:

- Database support to make up-to-date business data available for rapid retrieval from any workstation

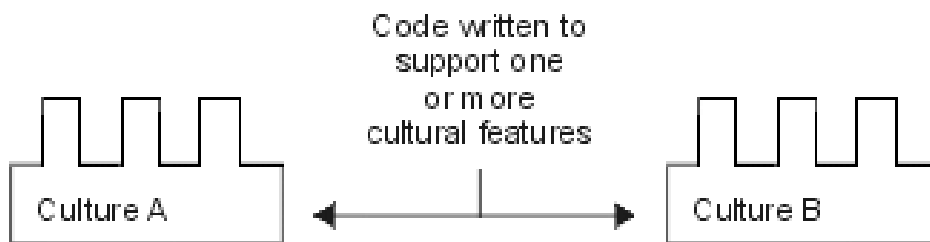
- Work management support to schedule the processing of requests from all work station users
- Application development support that allows online development and testing of new application programs to run at the same time as normal production activities
- System operation support that allows the user responsible for system operations to perform work from the display station using a single control language, complete with prompting and help for all commands
- Help and index search support that allows users to request online information about a wide variety of topics
- Message handling support that allows communication among the system, the user responsible for systems operations, workstation users, and programs running in the system
- Security support to protect data and other system resources from unauthorized access

In addition to these functions, the operating system provides national language support. National language support allows users to interact with the system in the language of their choice, with results that are culturally acceptable. National language support consists of two parts: globalization and localization.

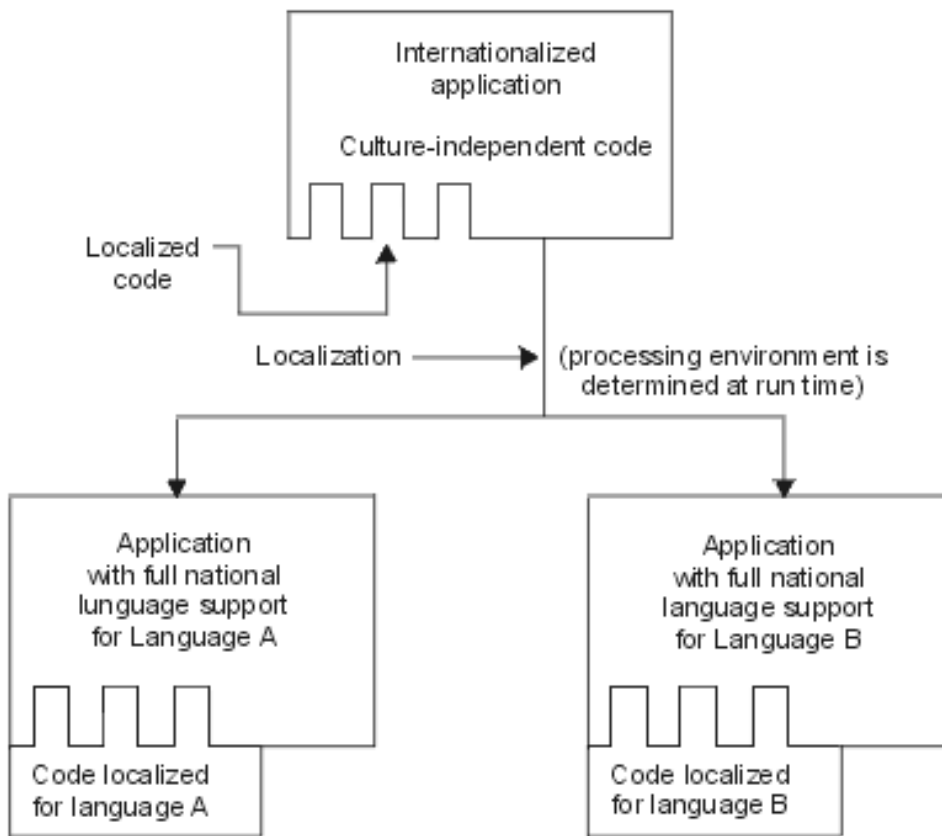
Globalization allows an application to operate in all language environments without any change to the application. This type of design is also known as enabling an application for national language support. A globalized application, shown in the following figure, is culturally neutral.



By contrast, localization allows an application to operate in a specific language, country, or culture. Localization of an application goes a step beyond globalization of the application, as shown in the following figure.



When localized code is integrated with globalized code at run time, the resulting application appears to the user with full national language support. The processing environment defines which localization code is combined with the globalized code at run time, as shown in the following figure.



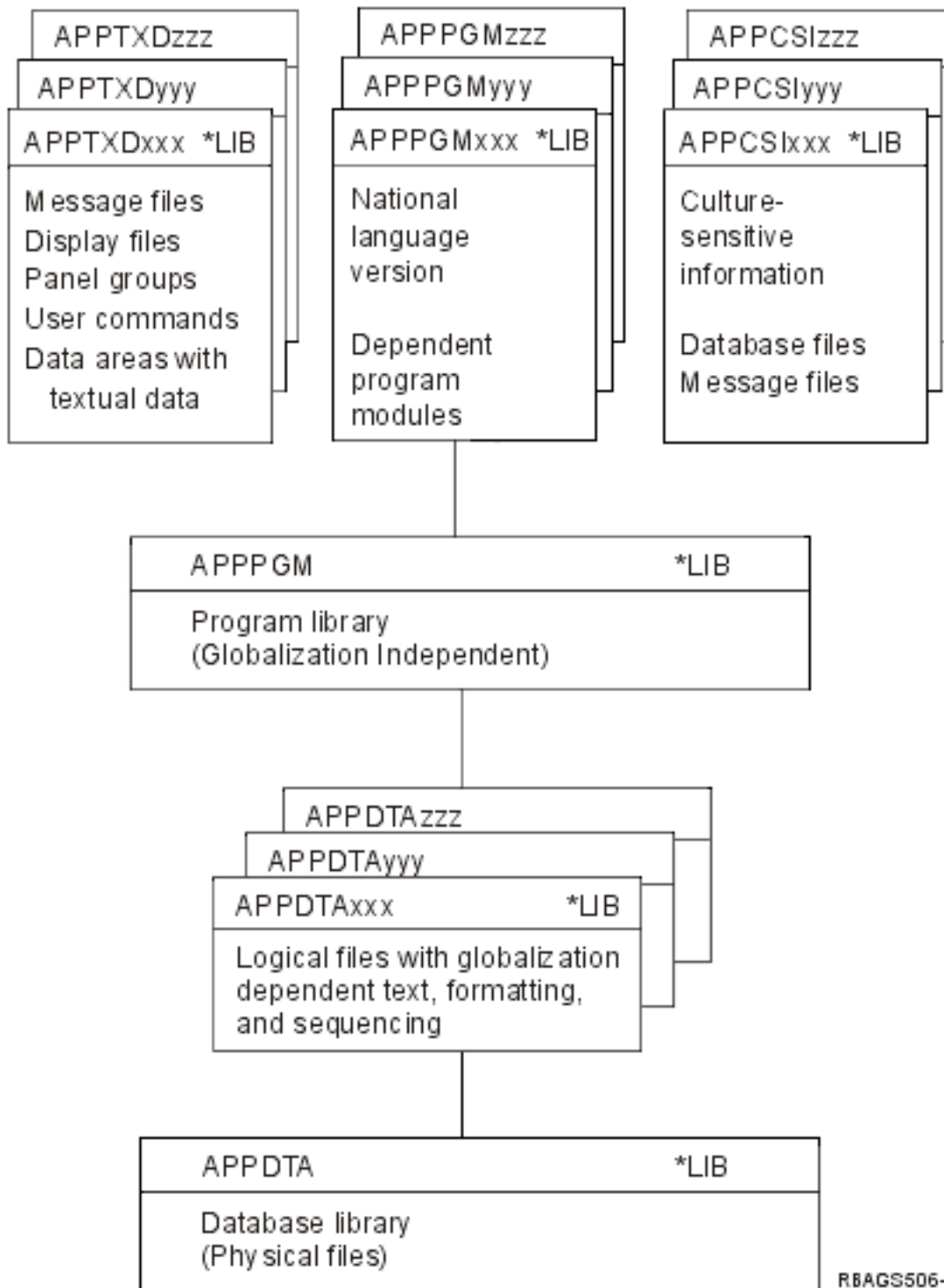
### Application arrangement and architecture

When you design an international application, consider organizing and structuring your application in ways that enable it to be used in an international environment.

In particular, consider the following strategies:

- Separate program modules at appropriate places
- Name application parts appropriately for a multilingual environment
- Refer to specifications whenever possible
- Provide multiple sets of logical files in separate libraries when working with database definitions

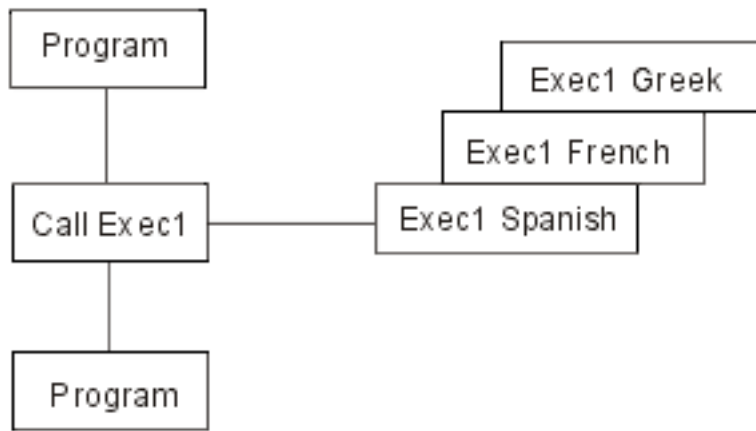
The following figure shows you the recommended way to organize the parts of your application.



### ***Program module separation***

You can separate cultural-dependent parts from your running code and set up cultural-dependent environments. You can do this using system values, user profile attributes, job attributes, and object attributes.

When it is impossible to separate national language and cultural-dependent parts from the running code, you must provide national language exits or calls at all points where functions dependent on national language support are required. The following figure shows a national language exit.



RBAGS504-0

### **Application part names**

When you want to enable your application for different languages and countries, consider the environments of the target systems in your naming conventions.

Use characters that are available, can be displayed, and can be printed in all the target environments. Use only characters of the invariant character set whenever you specify names for:

- Libraries
- Database files
- Device files (display or printer)
- Help panels
- Message files
- User commands
- Programs
- Record formats
- Fields

All other characters either vary their meaning or might not be available on the keyboard.

To create an internationalized application, you need to divide your application objects into related parts that are textual data and nontextual data. Your naming conventions should be able to distinguish between these parts. You should also be able to distinguish between the textual data of different languages. You can do this by separating the objects into different libraries.

### **Scenario: Library naming convention**

Your library naming convention can look like this:

```
AAATTTLL
```

where: **AAA** is the application identification; **TTT** is the type of objects; and **LLL** is the language code.

This naming convention allows you to have all libraries that belong to an application grouped together because you have a unique identifier (AAA) at the beginning.

The second part (TTT) allows you to distinguish between different types of objects:

#### **Textual data**

- Display files
- Printer files
- Message files

- Help panels
- User command
- Cultural values
- Database files with NLS-sensitive information and specifications
- NLS-dependent program modules

**Nontextual data**

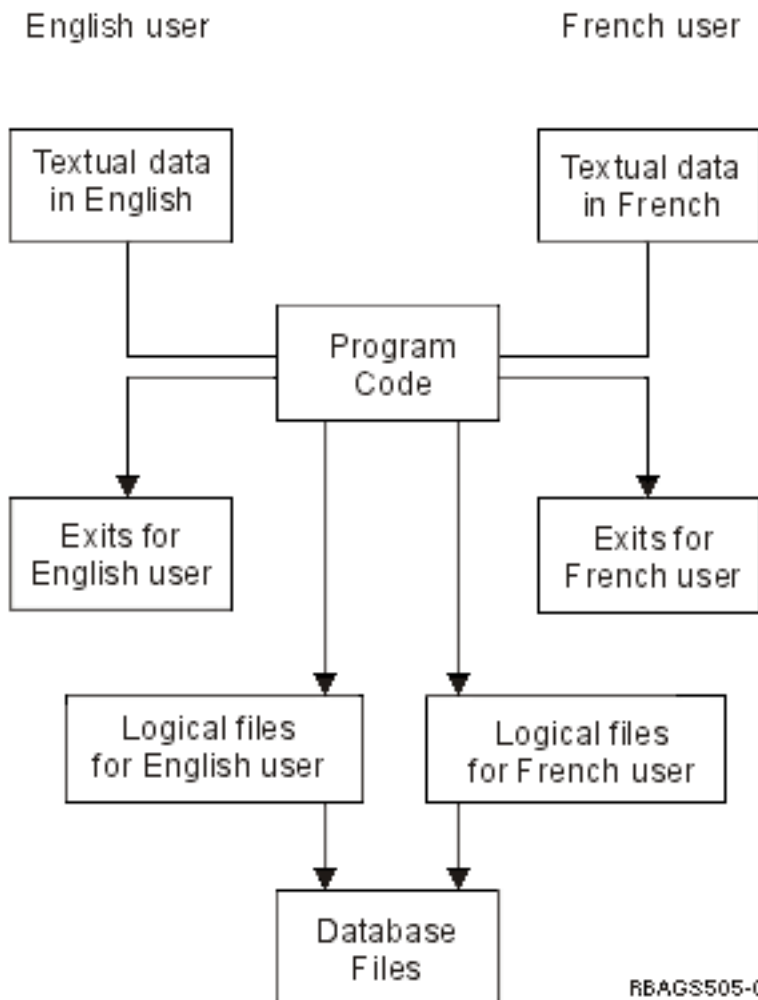
Programs

**Data**

Database files

The third part (LLL) allows you to specify the national language version for all the textual data parts. This allows you to use the same names for objects of the different national language versions within the different libraries. Your program is able to use different objects by just rearranging the library list accordingly when the job is run.

The initial library list can be taken from the job description. You can build a new library list by specifying the library list in the INLLIBL parameter of the Create Job Description (CRTJOBDD) command for a new job description, or of the Change Job Description (CHGJOBDD) command for an existing job description. The following figure shows an example of this.



**Related concepts**

Invariant character set (and its exceptions)

An *invariant character set* is a character set, such as the syntactic character set, whose code point assignments do not change from code page to code page. The table illustrates the invariant character set (character set 00640) on the IBM i operating system.

#### Database definitions

You can define a file to specify certain facts. The specifications are then used in database files.

#### **Specification references**

You should define all your fields first in the field reference file of your application and refer to them wherever you can, in the database specifications, in device file specifications, and in the high-level language programs. This technique helps you to define the field specifications once and use them again.

If you need to distinguish between the same field of different sources, you can rename or qualify them. Whenever you need to change the definition of a specific field, you just need to change the attributes of that field in the field reference file and create the objects again. Then the changes take place automatically in all the different places where the field is used.

For example:

```
|...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
  A                                     REF(field-ref-file-name)
  A field      R record
  A field      R line   pos
or
  A field      R line   pos                REFFLD(ref-field-name)
```

#### **Database definitions**

You can define a file to specify certain facts. The specifications are then used in database files.

Here are some examples of such specifications:

- The object description text of the file
- The explanation text (TEXT keyword) on record formats and field descriptions
- The column headings (COLHDG keyword) on field descriptions
- Date and time formats and separators
- Sort sequence
- Language identifier

The object description text is shown by many database tools, such as Db2® for i SQL, IBM i Access, and data file utility (DFU), on the file selection display.

The column headings are shown by the database tools on the output field definition display. Column headings are also used on screen design aid (SDA) and report layout utility (RLU) as the proposed field-prompting text or heading.

Data management handles date- and time-type fields in the format specified at file-creation time, unless your application or database tool does a conversion to present it according to your request or job demand.

When you want to present all this information according to the language and culture of the user, you need to provide multiple sets of logical files in separate libraries. Along with the translated text, you can specify different date and time formats or different sort sequence and let data management perform the conversion. A similar technique can also be used for numeric-type date fields (unless they are packed), using the substring (SST) function. The user can access the data only through the designated logical views. When you are defining logical files with different sort sequences, avoid using a unique index with a shared-weight table. Although this is possible, a unique index prevents using keys that differ only in characters with the same weight.

The information about Application part names includes a scenario that uses different sets of logical files for different users.

#### **Related concepts**

Application part names

When you want to enable your application for different languages and countries, consider the environments of the target systems in your naming conventions.

### User interfaces

A user interface is the part of a software product that your customer actually sees.

A user interface may include the layout of display screens or printed output, displayed or printed text, commands, online help, and messages. A user interface is also the part of a software product that you must either translate or make cultural changes to for users in other countries or cultures.

The operating system provides specific software functions to help you organize text from your user interface and store that text in a library for easy translation. The operating system also provides you with a user interface manager that provides a consistent user interface. The user interface manager provides comprehensive support for defining and running panels such as displays and online help.

This section provides guidelines that you can follow when designing a user interface for an international application. You should apply these guidelines early in the design process.

### Related concepts

#### Command design

The IBM i operating system allows users to define and create their own commands.

#### Delivering globalized applications

As you prepare to deliver your globalized application, you should consider how globalization issues might affect the ways that your customers install and use your application.

### **Checklist: User interface design**

When creating a user interface with globalized support, you should follow some rules and guidelines.

The rules and guidelines are shown in the following table:

Complies	Not applicable	Rule
		The use of a graphic character for software control purposes must not preclude the use of the same character in the text of messages, menus, prompts, input fields, or output fields.
		Graphic symbols and icons must be translatable.
		Language-dependent parts of a product must be isolated from nonlanguage-dependent parts for easy modification.
		All user interface text and presentation control information must be isolated from the running code.
		Sufficient space must be available for user-interface text expansion caused by translation.
		Functions dependent on display field length and display field position, or display field position alone, must not be designed in such a way that they are affected by user-interface text expansion.
		A method must be provided to allow for the identification and tracking of panels and messages during the translation process.
		Variables must be permitted to assume any location and order within a display field.
		Messages and other displayed words or phrases must be complete entities and must not be constructed from individual words or phrases.
		Entry of end-user commands, keywords, or responses must be possible without regard to uppercase or lowercase characters.



Complies	Not applicable	Rule
		Date and time formats must be selectable.
		Numeric punctuation must be selectable.
		Number rounding and mathematical formats must be selectable.
		Monetary format must be definable.
		The default currency symbol and its abbreviations must be selectable.
		The currency symbol position must be selectable.
		Field sizes for monetary values must be selectable.
		The measurement system must be selectable.
		Lowercase alphabets should not be assumed to be invariant.
		Special characters, including punctuation marks, should be definable and not program dependent.
		User-interface text modules should be packaged separately from the running code.
		User-interface text modules for single-byte coded character set systems should be loaded separately from the running code.
		A consistent convention should be used throughout the product for denoting variables and input fields.
		Words should not be used in place of numbers.
		The terminology in user interface text should be consistent throughout a product.
		Abbreviations should be avoided.
		Slang, jargon, and humor should not be used.
		Trademarks should be identified and explained.
		Ambiguous words should not be used.
		Proper style and sentence structure should be used in user interface text.
		Negative questions should be avoided.

### ***Text translation design***

These general tips help simplify the translation of your textual material.

### **Isolating textual data from running code**

To allow easier translation and to avoid translating the running code, you should separate all textual data from the running code. Only one set of running code is needed, but many translations of the textual data can be done.

### **Providing expansion space**

The space needed to translate text from one language to another varies by language. To ensure that the translated version preserves the concept and keeps usability, allow sufficient presentation space for the textual data expansion. The following table shows recommended expansion space for user interfaces designed using U.S. English.

Number of characters in text	Additional space required
Up to 10	100 to 200%
11 to 20	80 to 100%
21 to 30	60 to 80%
31 to 50	40 to 60%
51 to 70	31 to 40%
Over 70	30%

### Variable placement of an object on the display

Because the position of one display element often is influenced by the position and size of others, some of the elements on the translated version of a display might need to be relocated. The program must continue to respond properly, despite this relocation.

### Flexible order of variables

In order to contain dynamic information, messages typically employ substitution variables. However, each spoken language has its own syntax (order of arrangement of parts of speech). When a message is translated into another language, the position and order of substitution variables might need to change to meet the syntax requirements in the translated language.

### Complete textual data entities

If the final form of the constant text relies on the composition of various parts, it might be untranslatable. This is because the translator might not know which form of the word to use or because there is no combination of parts that work for a different language.

For example, you should define column headings for display screens as complete entities. You should not combine words or parts of words to define column headings. Assume that you are writing an application for scheduling jobs between Monday and Friday. You are creating your application in French. You decide to create column headings for reports and screen displays by combining the first part of the name of the day with the constant DI. Throughout the application, the column and report headings are assembled like this:

First Part of the Name of the Day:	Combine With:	Result:
LUN	DI	LUNDI
MAR	DI	MARDI
MERCRE	DI	MERCREDI
JEU	DI	JEUDI
VENDRE	DI	VENDREDI

When you translate your application from French to German, you cannot combine two parts to create the names of the days: MONTAG, DIENSTAG, MITTWOCH, DONNERSTAG, and FREITAG.

### Treating commands, responses, and keywords like textual data

Commands, responses, and keywords should be translated into the language normally spoken by the user. For example, an English application has been translated into German. If the response is still in English as Yes and No, the German users might feel unfamiliar and uncomfortable in using the program because the responses they are familiar with are Ja and Nein.

### Expressing all text as simply and clearly as possible

- Use simple phrases and sentences and avoid compound phrases. Simple words allow easy translation.
- Make terminology consistent throughout the product.

If consistent terminology is not being adopted throughout the product, translators will waste time trying to determine the appropriate word to be used in translation.

- Include notes to translators in your information for correct word use to prevent any misunderstandings.
- Avoid abbreviations.

Rules for abbreviations vary from language to language. Abbreviations of words can lead to misunderstandings by the translator and by the user.

- Avoid slang, jargon, and humor.

Slang, jargon, and humor are specific for a particular language and cannot be easily translated into another language.

- Avoid negative questions.

Negative questions are often misunderstood by the user. When asking questions, ask them in a positive way.

### ***Textual data code design***

You can use different techniques to specify, store, and use constant text. You can use each technique for specific types of textual data components. Each technique has its advantages and disadvantages.

Application displays, printer file specifications, and user-created commands typically contain a large amount of constant text. Application displays, printer file specifications, and user-created commands also contain input and output fields such as headings, field prompts, instruction lines, and function key descriptions.

### **Related concepts**

#### Constant text strings

When designing your panels, keep in mind that different languages have different space needs for the same expression.

#### Printer file design and translation

Program-described printer files and externally described printer files are two types of printer files. When you design printer files to be translated into a national language version, you should follow some specific guidelines.

#### *Early message binding*

Text can be stored externally from the source code in a separate message file but is bound into the object when it is created.

This technique can be used for:

#### **Display files**

Constants such as titles, instruction lines, option definitions, headings, field prompts, command key descriptions

#### **Printer files**

Constants such as titles, headings, total line descriptions

#### **User commands**

Prompt descriptions on the command definition statements

For device files (display and printer), the message is referred to by the Message Constant (MSGCON) keyword in the DDS source specifications.

For example:

```
A          line pos  MSGCON(length message-ID[*libl/]message-file-name)
                        ^
                        includes expansion space
```

For user commands, the message identifier *xxxxnnn* is specified on the PROMPT keyword instead of a literal. The message file is referred to on the Create Command (CRTCMD) command.

For example:

```
CMD      PROMPT(xxxxxxxxx)
```

The message file name *message-file-name* is in a source file referred to by the following command.

```
CRTCMD CMD(command-name) PGM(library-name/program-name) +  
      PMTFILE([*libl/]message-file-name)
```

Before the object can be created, you must enter the message description into the specified message file. Enter the message description using the Add Message Description (ADDMSGD) command.

For example:

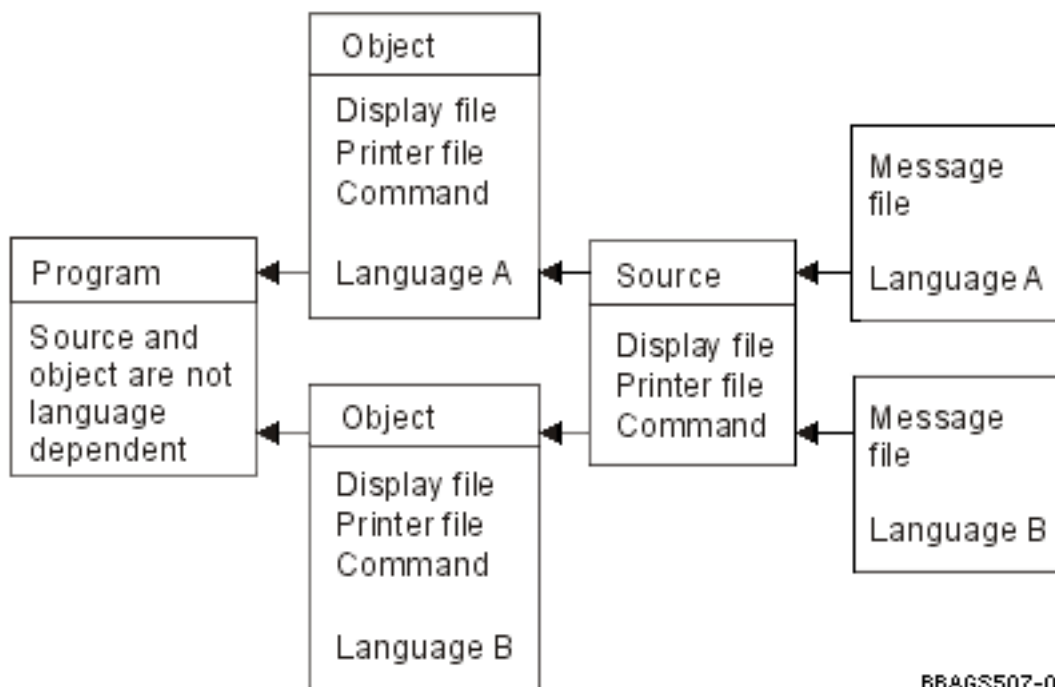
```
ADDMSGD  MSGID(xxxxxxxx) MSGF(library-name/message-file-name) +  
MSG('Text')
```

where *xxxxxxxx* is the message identifier.

This technique allows you to create any number of objects in different languages and to put them into different libraries using the same source code by just assigning another message file at object creation time.

The message file is needed only during the creation of the object. Consider specifying the appropriate length for different languages on the MSGCON keyword. Then make the length information available to the translator.

The following figure shows how early message binding works:



At file creation time, you can choose the appropriate textual data of the language version you want to work with by setting up the library list with the specific library containing the textual data and the program library.

#### **Related concepts**

[MSGCON \(Message Constant\) keyword for display files](#)

[MSGCON \(Message Constant\) keyword in printer files](#)

#### **Related reference**

[Create Command \(CRTCMD\) command](#)

## Add Message Description (ADDMSGD) command

### Late message binding

Text can be stored externally from the DDS source code in a message description and is bound only to the display format at run time.

This technique can be used for:

#### Display files only

Constants such as titles, instruction lines, option definitions, headings, field prompts, command key descriptions (MSGID keyword)

Default values on input fields (MSGID keyword)

Field validation specifications (CHKMSGID keyword)

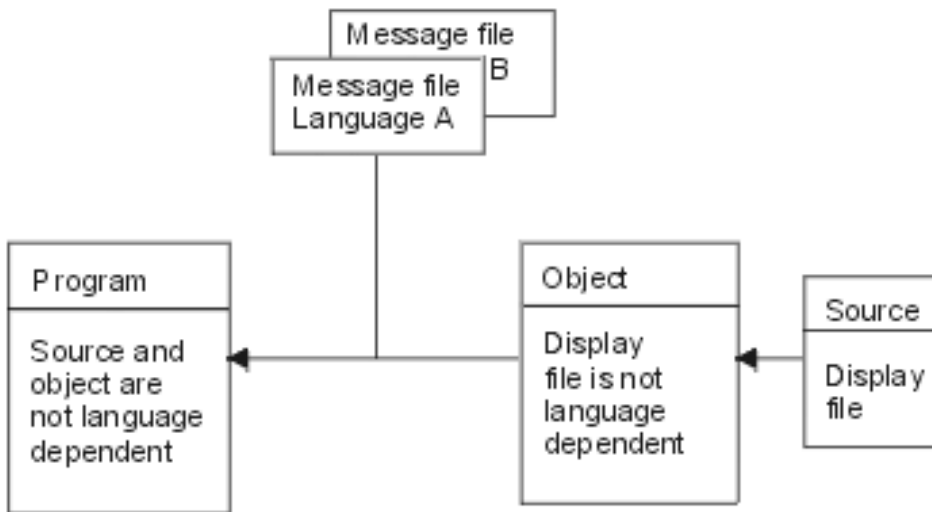
Error messages (ERRMSGID and SFLMSGID keywords)

In the DDS for the display file, the message is specified through the MSGID (Message Identifier) keyword. The message has to be entered into the specified message file using the ADDMSGD (Add Message Description) command.

For example:

```
A   FLD-name  length  line pos  MSGID(message-ID [*libl/]message-filename)
      ^
      includes expansion space
ADDMSGD  MSGID(xxxxxxx) MSGF(library-name/message-file-name) +
MSG('Text')
```

This technique allows you to create any number of message files in different languages and different libraries, with one DDS source code and display file object. During run time, you assign another message file by setting the library list accordingly. The following figure is an example.



**Note:** This technique requires the application to perform all editing based on the cultural convention.

### Direct coding as an unnamed output field

The most common way to define constant text is to specify the text directly in the source code as a literal. While this method is the most common way to define constant text, it is the most difficult to translate. Avoid using this method whenever coding an application, even if the application is not planned for translation.

If you are coding an application that will not be translated, you might want to use this technique for:

#### Display files

Constants such as titles, instruction lines, option definitions, headings, field prompts, command key descriptions

Default values on input fields (DFT keyword)

Error messages (ERRMSG/SFLMSG keyword)

**Printer files**

Constants such as titles, headings, total line descriptions

**User commands**

Prompt descriptions on the command definition statements.

For device files, specify the text as an unnamed field, indicating the starting line and column and the constant text itself.

For example:

```
A          line pos      'Text . . . . . :  
,
```

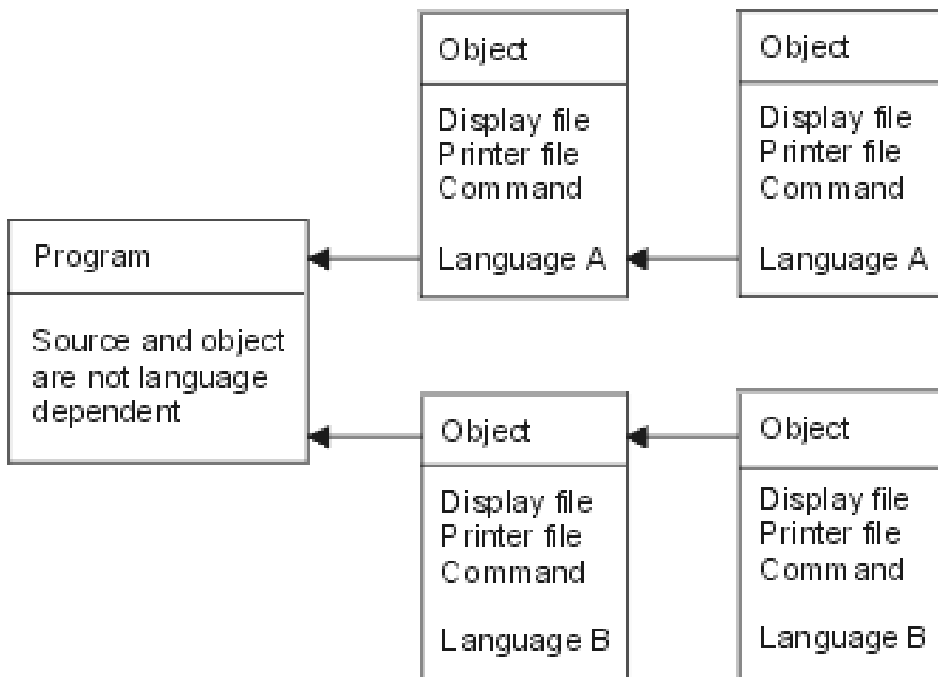
A similar rule applies to user-created commands. Define the text directly on the keywords of your command source statements.

For example:

```
CMD          PROMPT('      Command description      ')
```

When defining the text directly on the keywords, standardize the sizes of the different elements in a large literal, rather than specifying many small single ones as single words. This makes the source code more readable and more flexible for translation.

Consider that the space needed for explanation text can vary from language to language. To have enough room after translation, remember to reserve space initially. The source members need to be translated and the objects need to be created for different languages as shown in the following figure:



Each national language version has one set of programs, but can have multiple sets of source members and data objects. When the application is run, you can choose the appropriate textual data of the language version that you want to work with. This can be done if you set up the system part of the library list with the specific library that contains both the textual data and the program library.

### *Text stored in database files*

Text can be stored externally from the source code in a database file, retrieved by the application program, and then moved to the display or print format at run time.

Instead of coding constants on the DDS, you can specify output fields that can be filled by the program. Consider specifying the appropriate length for different languages on the output fields and making that available to the translator.

This technique can be used for:

#### **Display files**

- All constant text
- Default values on input fields
- Error messages

#### **Printer files**

- All constant text

#### **Programs**

- All constants like compare values, scan characters, and tables.

This technique allows you to create any number of database files in different languages and different libraries, with only one DDS source code and display file object. During run time, you assign the corresponding database file by setting the library list accordingly.

**Note:** This technique requires the application to perform all editing based on the cultural convention.

### ***User interface manager***

The IBM i user interface manager (UIM) is a part of the system that allows you to define panels and dialogs for your application.

UIM provides the following support:

- A tag-based language for describing data and panels.
- A compiler to create panel group objects and menu objects by using the tag-based language.
- A set of application programming interfaces (APIs) to use as panel group objects to display and print panels.

The UIM also provides the following functions:

- Dialog commands for screen management
- Contextual online help
- Pop-up windows
- Menu bars
- Command line for entering CL commands
- Tailoring of the contents of a panel for different users or environments
- Fast paths through menu networks
- Double-byte character set (DBCS) languages
- Bidirectional (BIDI) language support

UIM supports common panel types, such as menus, information displays, list displays, and entry displays. When all display types and interfaces are consistent, users adapt more quickly to new applications.

UIM applications can coexist with and share the requester display device with other open display files that are not under UIM control. However, a UIM panel and a DDS-defined record format cannot appear on the display at the same time. When a UIM panel either replaces a DDS panel or vice versa, the system suspends operations of one file or panel group and restores the display as needed.

### *Online help design*

You can define online help by using panel groups or records. By using panel groups, you can define online help as objects into which user interface manager (UIM) source is entered. By using records, you can define online help as a set of DDS keywords contained in a source file member.

If the user interface manager is used for defining online help, the panel groups are defined either in place of DDS or in the display file. In either case, the encoding of the data to be displayed must be indicated by the CHRID value in the display file or the panel group. A panel group is an object that can be used to contain help information. The operating system uses \*PNLGRP as an identifier for the object type that contains a collection of help information.

### **Guidelines: Online help**

When defining online help information to be translated into national language versions, keep in mind the following considerations about panel groups and records:

- Records do not have word processing available (functions such as spell check and word wrap though system APIs exist to provide spell checking).
- Various IBM i messages and panel groups determine the national language conventions and translations. Not all countries have a national language version available for the operating system. Not all national language versions are completely translated, with many parts still in English. The messages and panel groups that are not translated do not reflect the national language cultural conventions. The command design information includes an example of a translated panel in which part of the panel has remained in English because not all parts of the NLV were translated.
- Allow for translation expansion.

### **Guidelines: DDS online help design**

When multiple languages are installed on one system, the help documents are stored in different folders. The DDS source file needs to be copied, changed, and compiled again for each language on the system.

### **Related concepts**

#### Command design

The IBM i operating system allows users to define and create their own commands.

#### *Index search tags*

Help panel groups can contain index search modules. Index search supplements the help information that is provided for each display.

To use the information in help panel groups for the index search function, you need to add the appropriate UIM tags to your help modules.

Users can access the index search function from any display help that specifies that the index search function is available.

### **The ISCH tag**

The ISCH tag defines the title of a topic in the index and specifies the root words that serve as the link between the topic and the search words (synonyms) entered by the user. The tag appears immediately after the HELP tag to which it refers. There can only be one ISCH tag within a single help module.

For each ISCH tag, there can be several lines of root words, provided that the total number of root words is no more than 50. If more than one line of root words is used, ROOTS= must be repeated at the beginning of the second line and subsequent lines.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```
:PNLGRP.  
:HELP name=entry1.  
:ISCH ROOTS='root1 root2 root3 root4 root5'  
  ROOTS='root6 root7 root8 root9 root10'  
  ROOTS='root11 root12 root13 ... root50'.
```



```
Title of First Topic
```

```
This is the first index search module in this panel group.  
:EHELP.  
:EPNLGRP.
```

The root words on all lines must be enclosed in apostrophes and a period must be placed only at the end of the last line of root words. The topic title follows the period on the ISCH tag and can be placed on the line immediately following the period.

### The ISCHSYN tag

The ISCHSYN tag defines the words (synonyms) that, if entered by a user, match a specific root word. If a word that is entered by a user is a synonym for a root word, then a match is found for each topic whose ISCH tag contains that root.

If you want a word that is used as a root word to be used as a synonym as well, you must include the word as a synonym on the ISCHSYN tag. For example:

```
:ISCHSYN ROOT='ocean'.ocean water sea
```

The synonyms for the ISCHSYN tag must be entered on one line, and at least one ISCHSYN tag must exist for each root word. If more than one line is needed, more ISCHSYN tags can be entered for the same root word.

UIM does not differentiate between synonyms entered in uppercase, lowercase, or mixed case. For this reason, it is not necessary to repeat synonyms to cover all the different cases.

You can use alphabetic or numeric characters for synonyms; however, the following characters (including their hexadecimal equivalents) are not allowed to be used as a synonym or part of a synonym:

- . (period)
- ( (left parenthesis)
- ) (right parenthesis)
- ; (semicolon)
- , (comma)
- ? (question mark)
- : (colon)

The ISCHSYN tags can be placed anywhere in the panel group, but to make maintenance and translation easier, place them all in one area (such as at the beginning of your panel group or in a panel group object that contains only ISCHSYN tags).

### Example: ISCH and ISCHSYN usage

The following example shows some ISCHSYN tags and the ISCH tags that use them:

```
:PNLGRP.  
:ISCHSYN ROOT='ocean'.ocean water sea  
:ISCHSYN ROOT='lake'.lake water pond  
:ISCHSYN ROOT='definition'.definition define description what  
:ISCHSYN ROOT='definition'.summary concept information explanation  
:HELP name='defocean'.  
:ISCH ROOTS='definition ocean'.  
Definition of ocean  
  
An ocean is one of the five large bodies of salt water, which  
together cover nearly three-fourths of the world.  
:EHELP.  
:HELP name='deflake'.  
:ISCH ROOTS='definition lake'.  
Definition of lake  
  
A lake is a body of standing water that is enclosed by land.
```

:EHELP.  
:EPNLGRP.

### *Index search and double-byte character set*

The index search function can be used with either double-byte character set (DBCS) or single-byte character set (SBCS) data. When DBCS data is used, the device from which it is requested must be capable of entering and presenting the data in DBCS.

The object that contains the index search data is marked as containing DBCS data. The system determines if the device is capable of handling the DBCS data.

When the data is being prepared for DBCS format and the index search function is used with that data, consider the following information:

- When the index search data is prepared for a DBCS system, the synonyms entered on the ISCHSYN tag must be in double-byte character mode. That is, the first byte after the tag must be a shift-out character and the last byte of the data must be a shift-in character. The system does not convert data on the ISCHSYN tag to double-byte character data.
- Words must be separated by a single-byte blank. From 1 to 19 double-byte characters can be combined to form a word. Intervening shift-out and shift-in characters are allowed, but are ignored by index search.
- The words that are used to link the ISCH and ISCHSYN tags (the ROOTS attribute of the ISCH tag and the ROOT attribute of ISCHSYN tag) must be identical and should not be entered in DBCS.
- Search words can be entered in either single-byte mode or double-byte mode. Single-byte blanks can be entered to separate the words.

When the search words are shown on the screen, the double-byte character representation (the character that was actually used in the search) is shown. Special processing takes place so that index search is not case sensitive. The search words from the ISCHSYN tag are converted to uppercase using a conversion table for the code page that is specified with the TXTCHRID attribute of the PNLGRP tag. If the search words are DBCS, they are not converted to uppercase. Shift-out and shift-in characters are treated as blanks during parsing; leading and trailing blanks are removed. All SBCS words are converted to uppercase using a conversion table for the code page of the device description.

### **Program message design**

A message can be predefined or immediate.

Consider the following information when you design and code:

- Do not use immediate messages. They are created by the sender or program at the time they are sent and are not stored in a message file. Therefore, they cannot be translated by the translator.
- Use predefined message descriptions that can both:
  - Exist outside of the program that uses them.
  - Be stored in a message file.
- Do not specify the maximum size for a message file. When the message file becomes full, you cannot change the size of the message file. You need to create another message file and add the message description again.

Use the Create Message File (CRTMSGF) command to create a message file to hold the predefined message description. The contents of the predefined message description can be put into a message file by the Add Message Description (ADDMSGD) command.

- Use substitution variables with care. Different languages have different orders for substitution variables. For example, in the English message:

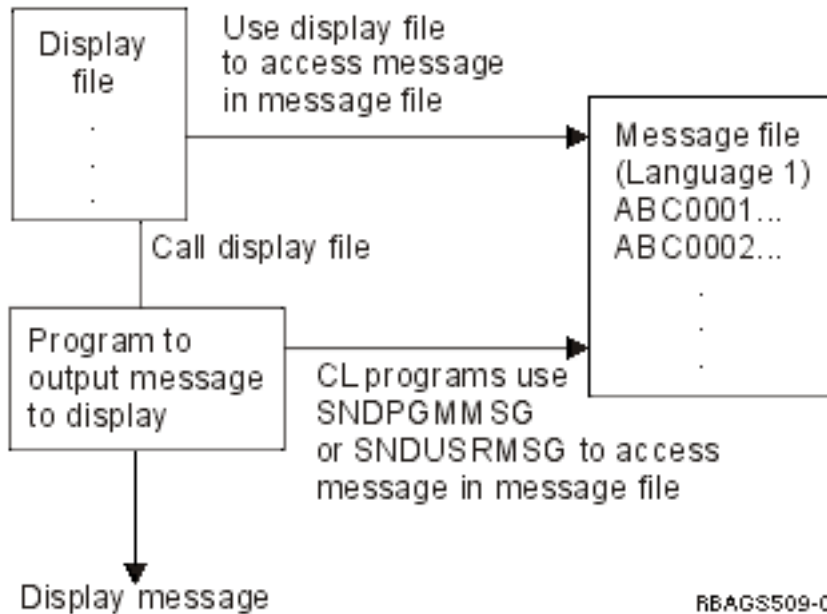
```
File &1 in Library &2 not found.
```

&1; and &2; are the substitution variables. Those substitution variables can appear in different positions for different languages.

- Make your design and coding able to understand a reply code for different languages. For example,

English	Y = Yes
Danish	J = Ja (means Yes)

The following figure shows the creation of different NLV messages from message files.



A program can directly access the message file for program messages, or it can indirectly access the message file through display files for program messages.

### Related concepts

#### CCSID support for messages

You can use CCSID support for handling messages and message catalogs on the IBM i operating system by using commands and application programming interfaces. You can send messages tagged with one CCSID to users with a different CCSID.

### Related reference

- [Create Message File \(CRTMSGF\) command](#)
- [Add Message Description \(ADDMSGD\) command](#)
- [Control language](#)

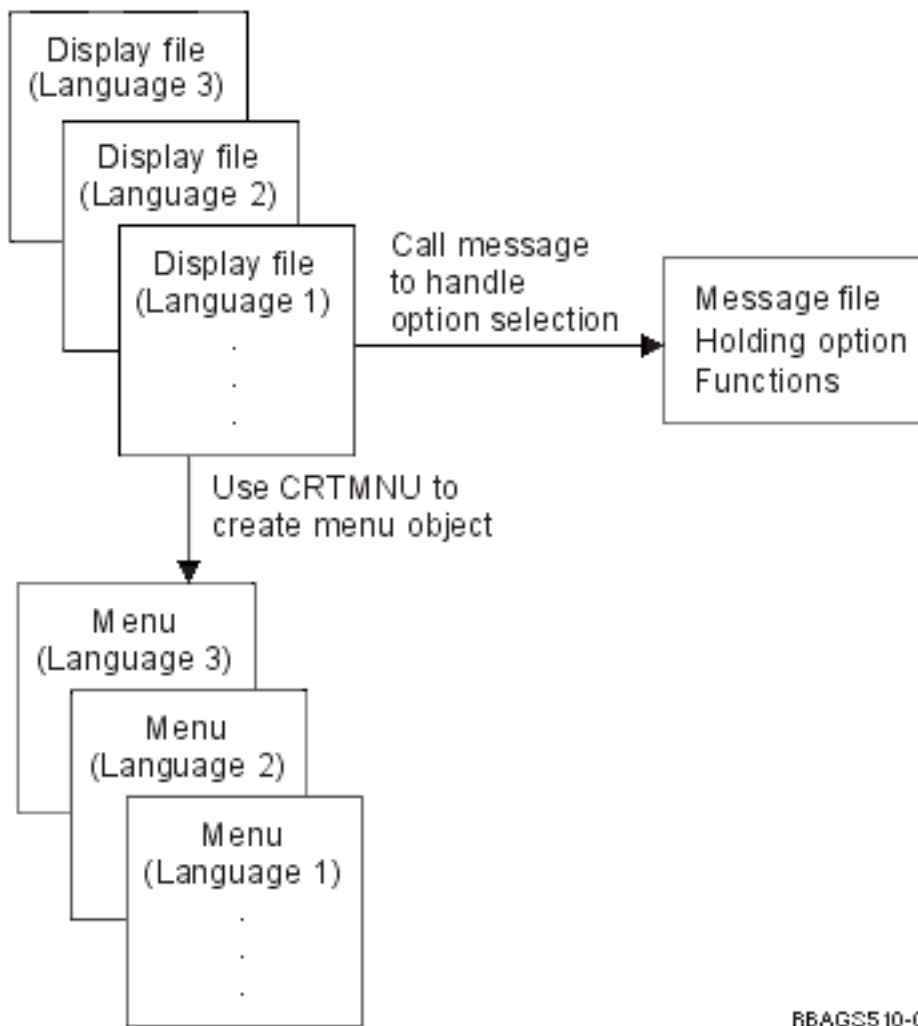
### Menu design

You can define your own menus on the system. The types of user-defined menus include display file menus, UIM (reference) menus, and program menus.

To use an application system, users need to deal with a lot of menus and displays. When an application is being translated from one language to another, a large portion of the literal text to be translated comes from menus.

### Display file menu

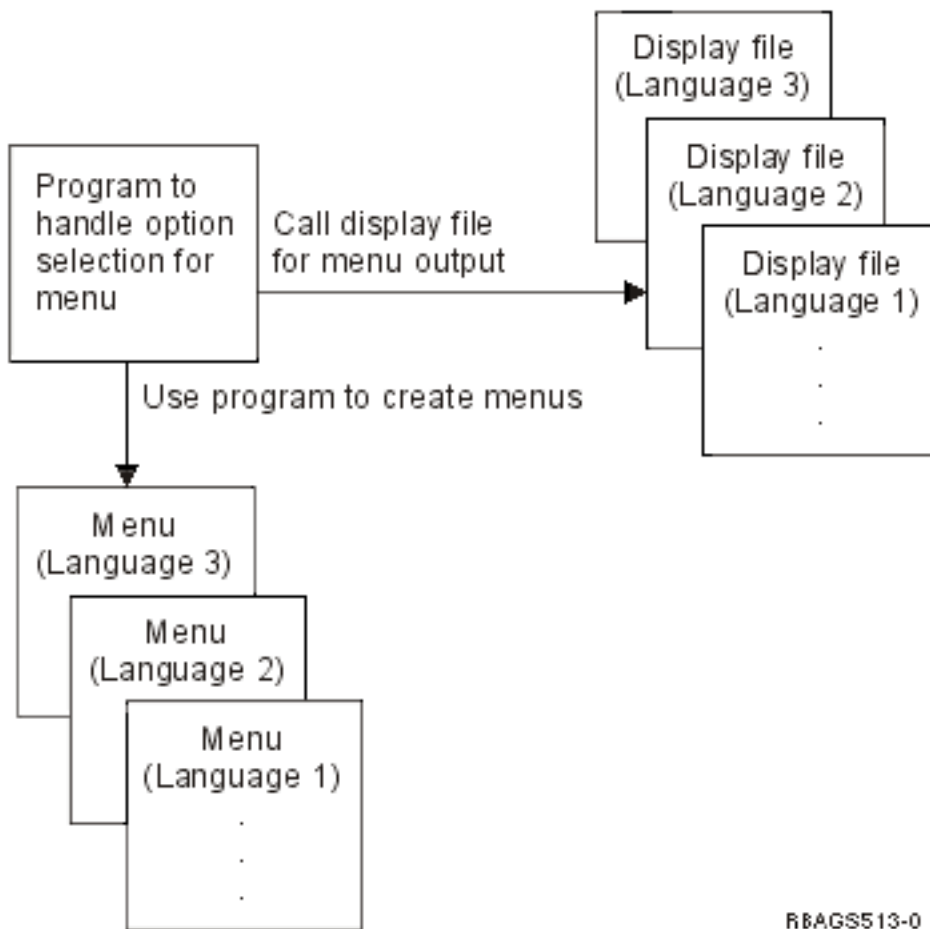
A display file menu uses a display defined by DDS to present a menu format. The menu functions are controlled by a menu object that contains the commands used to run each of the menu options. The following figure shows how display file menus are created for different national language versions.



RBAGS510-0

### Program menu

A program menu uses programs to present the menu format (defined by DDS) and to provide functions necessary to run the menu options. The following figure shows how program menus are created in different national language versions.



RBAGS513-0

### Menu translation

To allow for easy translation into national language versions of your menus:

- Keep the literal text of menus external by holding the constant text as externally defined message descriptions in a message file and by incorporating the text into a menu file when the program is run.
- Be aware of the expansion space needed when a menu is translated from one language to the next. Leave space for translation expansion when you design your menus.
- Be aware of cultural conventions when date, time, or edited fields are displayed on the menu.
- Use numerals 0 through 9, instead of uppercase and lowercase English letters (A through Z), as the option fields for selection. Numeric characters are more standard among different languages.

### Command design

The IBM i operating system allows users to define and create their own commands.

To create a command, you must first define the command through command definition statements. Then use the Create Command (CRTCMD) command to process the command definition statements to create the command definition object.

When you define and create a command, take into consideration the following information:

- Use help panel groups to provide online help information for the command.
- Use message identifiers instead of literal text for the PROMPT keyword on the CL CMD, PARM, ELEM, and QUAL command definition statements.
- Translate the text that is displayed to the right of the prompt line of each parameter on the prompt display. This text is specified by the CHOICE parameter of the PARM command definition statements, so the appearance of the command prompt display maintains its coherency.

- Compile command-prompt text into separate command definition object versions for each national language. Use the Change System Library List (CHGSYSLIBL) command before creating the command to get the national language version prompt text from the correct national language version library.
- The function keys of the command prompt display are provided by the operating system. If the NLV of the operating system is different from the NLV of the command, two different languages appear on the command prompt display. For example, when translating an English display into German, both the English and German appear on the command prompt display.

The Control language information includes additional information about creating and defining commands.

### **Related concepts**

#### Online help design

You can define online help by using panel groups or records. By using panel groups, you can define online help as objects into which user interface manager (UIM) source is entered. By using records, you can define online help as a set of DDS keywords contained in a source file member.

#### User interfaces

A user interface is the part of a software product that your customer actually sees.

### **Related reference**

Create Command (CRTCMD) command

Change System Library List (CHGSYSLIBL) command

Control language

### ***Cultural-dependent design***

Different countries might have different standards, which you must consider when developing an NLS-enabled application. This culturally sensitive information must be placed outside the program the same way as the textual data is handled.

Many languages have characters (such as common-usage vowels essential to the correct spelling of a word) outside of the A-Z alphabet that must be considered for collating purposes.

Through system values, the system supplies linguistic support, cultural support, and the ordering of data.

### **Related concepts**

Default system values for national language versions

Jobs and functions on the IBM i operating system use system values as default values.

Field editing specifications

For the edit specification of your numeric, date, and time fields, you must consider the different cultural conventions of the users.

#### *Database file attributes*

Database attributes, such as coded character set identifier (CCSID), sort sequence (SRTSEQ), and language identifier (LANGID), are cultural dependent.

The CCSID attribute applies only to physical files. The SRTSEQ and LANGID attributes can be used with both physical files and logical files. A logical file can have a CCSID value only when it has taken the CCSID from the physical file. The database attributes are stored with the data. They are static in the sense that they cannot be dynamically altered by the process of accessing the data.

### **Related concepts**

Coded character set identifier (QCCSID) system value

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

### *Job attributes*

Some job attributes are cultural dependent. Through cultural-dependent attributes, the system provides linguistic support, cultural support, and the ordering of data.

- Coded character set identifier (CCSID)
- Sort sequence (SRTSEQ)
- Language identifier (LANGID)
- Country or region identifier (CNTRYID)
- Date format (DATFMT)
- Date separator (DATSEP)
- Decimal format (DECFMT)
- Time separator (TIMSEP)

The default values for CCSID, SRTSEQ, LANGID, and CNTRYID attributes are set from the user profile when the job starts. The values for CCSID, DATFMT, DATSEP, DECFMT, SRTSEQ, and TIMESEP can be set from the LOCALE and SETJOBATR attributes associated with the user profile. When you use the Change Job (CHGJOB) command, you can override the values specified for any of the listed job attributes.

### **Related concepts**

#### Coded character set identifier (QCCSID) system value

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

#### Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

#### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

#### Date format (QDATFMT) system value

The date format (QDATFMT) system value is used for the default value for the DATFMT job attribute. This system value also determines the format in which a date can be specified on the initial program load (IPL) options prompt.

#### Date separator (QDATSEP) system value

The date separator (QDATSEP) system value is used as the date separator for the default value of the DATSEP job attribute. It is also used as the date separator you can specify on the initial program load (IPL) options prompt.

#### Decimal format (QDECFMT) system value

The decimal format (QDECFMT) system value determines the type of zero suppression and decimal point character used by DDS edit codes 1 through 4 and A through M. It also determines the decimal point character for decimal input fields in the interface.

#### Time separator (QTIMSEP) system value

The time separator (QTIMSEP) system value specifies the character separator for time.

### **Related reference**

Change Job (CHGJOB) command

### Program attributes

The SRTSEQ and LANGID parameters can be specified as program attributes belonging to a \*PGM object type.

The LANGID parameter is used together with the SRTSEQ parameter only when the SRTSEQ value is set to \*LANGIDUNQ or \*LANGIDSHR. Otherwise, the LANGID parameter is not used.

If a program explicitly refers to a sort sequence or a language identifier, then those attributes stored in the program object take effect. The \*JOB RUN value for these parameters is used to refer to the attributes of the job running the program. \*JOB RUN makes it possible to use a single set of programs processing data according to different sort sequences. The \*JOB RUN value affects only the processing of data, however, not the retrieval sequence of data. The retrieval sequence is determined by the database attributes. To retrieve data in a sort sequence different than what is defined in the database, use logical files that are built separately.

### Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

The system message contains these values:

- Code page and character set
- Currency symbol
- Date format
- Date separator
- Decimal format
- Leap year adjustment
- Coded character set identifier
- Time separator
- Language identifier
- Country or region identifier

Cultural-dependent fields in the panel or display should not contain hard-coded values. These fields should be defined with the maximum length permitted for the field on the display.

If your application is to support users in languages other than the primary language, the callable routines should use the CPX8416 message values. A callable routine uses the cultural values for the primary language to determine the contents of the field (for example, date format) and places these values on the display. NLS system values maintained in message CPX8416 determine the format of the cultural values appearing in the cultural-dependent fields.

Your application can use the details from the system message.

The following table shows the layout for message CPX8416. This example shows the values in the text column using the English uppercase and lowercase NLV (feature 2924).

	Field	Start	Length	Justify
<b>Description</b> value	QCHRID 697 37	0001 0012	10 21	L L
<b>Description</b> value	QCURSYM \$	0034 0045	10 01	L L
<b>Description</b> value	QDATFMT MDY	0047 0058	10 03	L L



	Field	Start	Length	Justify
<b>Description</b> value	QDATSEP /	0062 0073	10 01	L L
<b>Description</b> value	QDECFMT	0075 0086	10 01	L L
<b>Description</b> value	QLEAPADJ 0	0088 0099	10 01	L L
<b>Description</b> value	QCCSID 37	0101 0112	10 05	L L
<b>Description</b> value	QTIMSEP :	0118 0129	10 01	L L
<b>Description</b> value	QLANGID ENU	0131 0142	10 03	L L
<b>Description</b> value	QCNTYID US	0146 0157	10 02	L L
<b>Description</b> value	QIGCCDEFNT *NONE	0160 0171	10 21	L L

### Related concepts

#### Configuring secondary languages

A secondary language consists of textual data for all licensed programs supported for a national language version.

#### Currency symbol (QCURSYM) system value

The currency symbol (QCURSYM) system value verifies the currency symbols specified in the DDS keywords Edit Word (EDTWRD) and Edit Code (EDTCDE).

#### Date format (QDATFMT) system value

The date format (QDATFMT) system value is used for the default value for the DATFMT job attribute. This system value also determines the format in which a date can be specified on the initial program load (IPL) options prompt.

#### Date separator (QDATSEP) system value

The date separator (QDATSEP) system value is used as the date separator for the default value of the DATSEP job attribute. It is also used as the date separator you can specify on the initial program load (IPL) options prompt.

#### Decimal format (QDECFMT) system value

The decimal format (QDECFMT) system value determines the type of zero suppression and decimal point character used by DDS edit codes 1 through 4 and A through M. It also determines the decimal point character for decimal input fields in the interface.

#### Leap year adjustment (QLEAPADJ) system value

The Leap year adjustment (QLEAPADJ) system value adjusts the system algorithms for the leap year in different calendar systems.

#### Coded character set identifier (QCCSID) system value

The coded character set identifier (QCCSID) system value specifies the CCSID for the IBM i operating system.

#### Time separator (QTIMSEP) system value

The time separator (QTIMSEP) system value specifies the character separator for time.

#### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

#### English Uppercase and Lowercase (Feature 2924)

The table shows the default system values for the English Uppercase and Lowercase (Feature 2924) national language version.

#### *Date formats*

There is no worldwide standard for the presentation of dates. Therefore, the date format should always be stored externally as part of the textual data.

The valid date formats on the operating system are:

- \*MDY (Month, day, year)
- \*DMY (Day, month, year)
- \*YMD (Year, Month, Day)
- \*JUL (yy/ddd)
- \*ISO (YYYY-MM-DD)
- \*USA (MM/DD/YYYY)
- \*EUR (DD.MM.YYYY)
- \*JIS (YYYY-MM-DD)

**Note:** Some operating system functions do not support all of the previous date formats.

In database files, dates can be stored as:

- Normal numeric data fields
- SAA date data-types

When you store dates as numeric data, your application needs to specify the format in which it is stored and presented.

When you store dates as data type DATE (L), you can specify the format with the DDS keyword DATFMT on the database file. The date is shown in this predefined format as character data, including the date separators.

If date sorting and other processing is needed, the date should be stored in \*ISO format (YYYY-MM-DD) and converted to another format during the input and output operations. Write a high-level language routine to convert dates.

#### **Related concepts**

##### Set job attributes (QSETJOBATR) system value

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

#### **Related information**

##### DATFMT (Date Format) keyword for physical and logical files

#### *Date separators*

The date separator for presentation should always be stored externally as part of the textual data.

The following list shows valid date separators:

- / (slash)

- - (dash)
- . (period)
- , (comma)
- (blank)

When you use decimal fields for dates, not only must your application specify the format, but it also must handle the date separators during the input operation and presentation.

When you use date-type fields, the date separators are always included in the date. To change the date separator, you can write a high-level language routine to convert dates.

### Related concepts

#### Set job attributes (QSETJOBATR) system value

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

#### Edit date presentation

You need to handle the presentation of dates in display and printer files differently, depending on how they are stored.

- As a normal decimal data field

Your application program has responsibility for the way the date is entered, stored, and presented. The application must check to see that the date is entered in the right format, remove any date separators, convert the date to another format when necessary, and edit it on the display file or printer file.

The DDS keyword DATE is used as an output-only field. DATE uses the job attributes DATE, DATFMT, and DATSEP. You can edit DATE using the edit code keyword, EDTCDE, for 6- and 8-digit date fields.

Editing with EDTCDE includes the following changes to the appearance of displayed fields, depending on which edit code is specified:

- Leading zeros are suppressed.
- Zero values can be displayed as zero or blanks.
- The field can be further edited using a user-defined edit code.

For all other types of fields using the EDTCDE Y keyword, the program has to specify the format, and the system uses the date separator of the job that created the device file. The date separator is integrated in the object, and you are not able to change it dynamically at run time.

- As an SAA data type DATE (L) field

The DDS date format (DATFMT) keyword allows you to specify different date formats, including default date separators, at the database field level. For the \*MDY, \*DMY, \*YMD, and \*JUL parameters, the default date separator can be changed with the date separator (DATSEP) keyword. The \*ISO, \*USA, \*EUR, and \*JIS values have a fixed separator, and the DATSEP keyword is not allowed with these values. The DATFMT and DATSEP keywords allow you to specify the format and editing characters for storing date fields. The date is shown as a character string, including the separators.

Any format conversion between the date input and the format the database asks for can be done by:

- Application program routines
- Field mapping through logical files that define different date formats and separators

For example, you can provide a date conversion that is dependent on the actual job attributes by using the following CL program:

```
PGM      PARM(&fromfmt &fromfld &tofld );
DCL      VAR(&fromfmt); TYPE(*CHAR) LEN(4)
DCL      VAR(&fromfld); TYPE(*CHAR) LEN(10)
DCL      VAR(&tofld); TYPE(*CHAR) LEN(10)
CVTDAT   DATE(&fromfld); TOVAR(&tofld);
FROMFMT(&fromfmt); TOFMT(*JOB) TOSEP(*JOB)
ENDPGM
```

Your application program has to pass the format of the date you want to convert and the date itself to the CL program. The CL program assumes that the job attributes represent the way the user expects to see date fields edited. It retrieves these values and does the conversion, conforming to these values, and passes back the date in that way. The \*ISO, \*USA, \*EUR, and \*JIS values have a fixed separator that cannot be changed. If the TOFMT parameter contains one of these values, the TOSEP value is ignored.

### **Related information**

[DATE \(Date\) keyword for display files](#)

[DATFMT \(Date Format\) keyword for display files](#)

[DATSEP \(Date Separator\) keyword for display files](#)

#### *Time formats*

The IBM i operating system supports several time formats.

- \*HMS (hh:mm:ss)
- \*ISO (hh.mm.ss)
- \*USA (hh:mm AM or hh:mm PM)
- \*EUR (hh.mm.ss)
- \*JIS (hh:mm:ss)

The system value QTIME has one format (hhmmss). The time separator value is determined by the QTIMSEP system value.

The time format for presentation should always be stored externally as part of the textual data.

In database files, times can be stored as:

- Normal numeric data fields
- SAA time data-types

When you store the time as numeric data, your application needs to specify the format in which it is stored and presented.

When you store the time as data type TIME (T), you can specify the format with the DDS keyword TIMFMT on the database file. The time is sorted in this predefined format as character data, including the time separators. To convert time fields from one format to another, write a CL program or high-level language routine to do the conversion.

#### *Time separators*

The IBM i operating system allows several valid time separators.

- : (colon)
- . (period)
- (blank)
- , (comma)

The time separator for presentation should always be stored externally as part of the textual data.

When you use decimal-data fields for time fields, your application needs to specify the format and time separators on the input and presentation operations.

When you use time-type fields, the time separators are always included in the time field. To change the time separators, write a CL program or high-level language routine to do the conversion.

### **Related concepts**

[Set job attributes \(QSETJOBATR\) system value](#)

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

### *Edit time presentation*

You need to handle the presentation of times in display files and printer files differently, depending on the way they are stored.

- As a decimal data field

Your application program has responsibility for the way the value is entered, stored, and presented. The program must check for the correct format, eliminate the time separators, convert the time to another format when necessary, and edit it on the display file or printer file.

The editing can be done by specifying the edit word (EDTWRD) for the field. The TIME keyword is an output-only field. Both the edit word and TIME keyword use the information available at creation time. The time separators are integrated in the device file object.

Both ways force you to have different copies of the source and objects for different editing requirements.

- As an SAA data type TIME (T) field

The operating system allows you to specify different time formats and time separators on the database file level. The TIME keywords allow you to specify the format and editing characters for storing time fields. The time type field is shown as a character string that includes the separators.

As an SAA data type, you can specify such time fields as normal character fields on the display file or printer file. On an input operation, your program has to check entered values for the correct format and separators and move them over to the database field. On an output operation, you just move the character string from the database file field to the device file field, including the separators. Any format conversion between the input and output format and the format that the database asks for can be done by either of the following two ways:

- Application program routines
- Field mapping through logical files that define different time format and separators

### *Decimal formats*

You can change the decimal format with the QDECFMT system value to reflect the way decimals are presented for your country or location.

#### **Related concepts**

##### Decimal format (QDECFMT) system value

The decimal format (QDECFMT) system value determines the type of zero suppression and decimal point character used by DDS edit codes 1 through 4 and A through M. It also determines the decimal point character for decimal input fields in the interface.

### *Sort sequences*

The IBM i operating system supports sort sequence. By using one of the listed options, you can order your data according to cultural-dependent requirements for specific applications.

- Hexadecimal sorting (sort sequence tables not used). This is the default.
- A user-supplied or system-supplied shared-weight sort sequence table or unique-weight sort sequence table, determined by the SRTSEQ parameter.

The following example shows how to use one DDS source file to create database files with different sort sequences. The following steps can be performed:

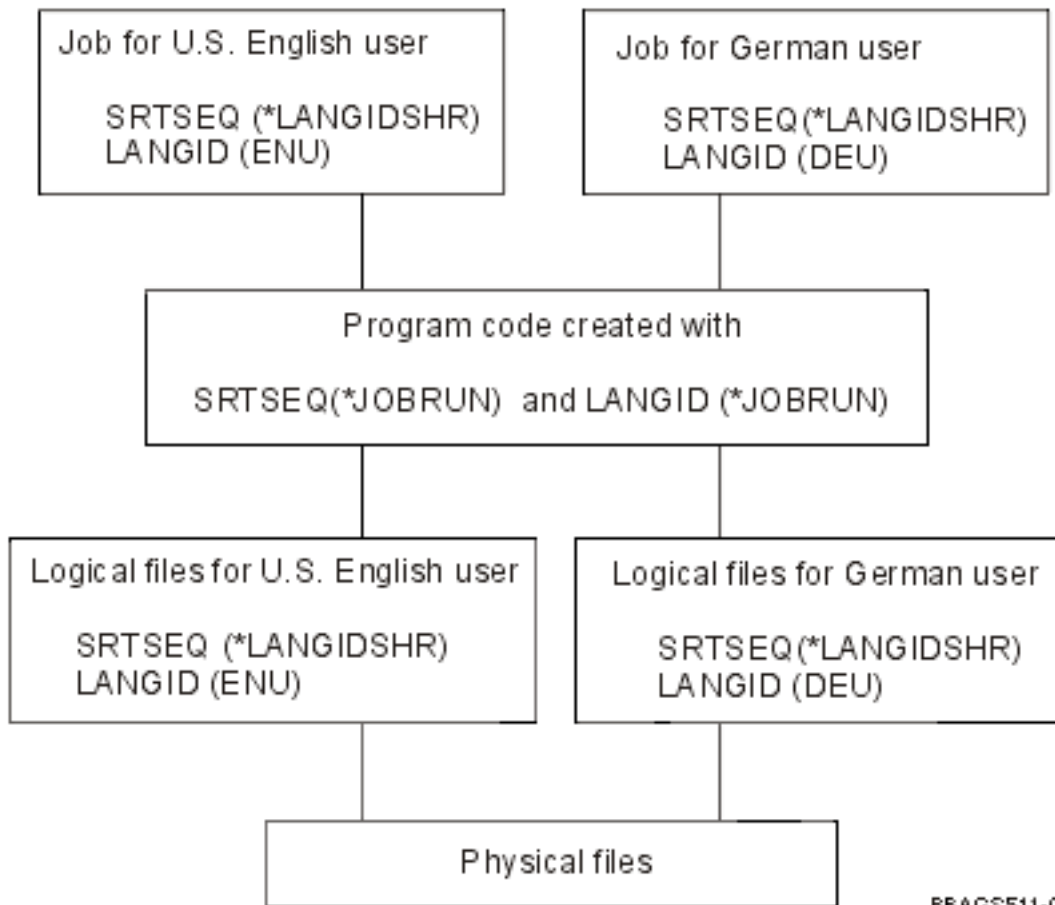
```
CRTxF      FILE(*CURLIB/NAME)
           SRTSEQ(*JOB)
           LANGID(*JOB)
```

You can then change the job attributes to create files with different sort sequences.

The CL program and high-level language programs can be created by specifying either early binding or late binding of a sort sequence. With early binding of a sort sequence, the sort sequence table to be used

is determined at compile time. With late binding of a sort sequence, the sort sequence table to be used is determined at run time.

Late binding makes it possible to use one set of programs in different national language environments. The following figure illustrates using different sort sequences for different jobs with one set of physical files and program code. The sort sequence table defined for the job and used by the program should be the same as (or compatible with) the sort sequence table assigned to the logical files accessed through the library list.



### Designing for running with different sort sequences:

If your program is expected to run with different sort sequences, consider the following conditions:

- Presenting the data in different order.
- Processing different records.
  - Specifying selection criteria such as less than or greater than can result in selecting different records. The selection criteria equal to can result in selecting a different number of records when the shared-weight sort sequence table is used.
- Processing of a conditional branch may be different.

**Note:** System lists (such as the output from the WRKOBJ command) are not affected by sort sequence support.

You can use the DDS file-level keyword alternate sequence (ALTSEQ) to specify the sequencing table and the library in which it is contained. The system-supplied sort sequence tables with shared and unique weight can be used for defining the alternative collating sequence.

The alternative collating sequence table is inserted into the file at compile time and is not needed at run time. You can have different files containing different collating sequences using one set of DDS.

**Note:** The alternative collating sequence defined in your database files must also be defined in your application programs; otherwise, you might get unexpected results.

The DDS ALTSEQ keyword provides limited support for sequencing. It has no effect on select/omit logic. The ALTSEQ keyword can only be used with the SRTSEQ(\*SRC) parameter on the Create Physical File (CRTPF) and Create Logical File (CRTLFL) commands.

### **Related concepts**

[Set job attributes \(QSETJOBATR\) system value](#)

The set job attributes (QSETJOBATR) system value sets job attributes at job startup time.

[ALTSEQ \(Alternate Collating Sequence\) keyword](#)

[Character sorting](#)

Traditionally, information is displayed in sorted order to enable users to easily find the items they are looking for. However, users of different languages might have very different expectations of what a sorted list should look like.

### **Related reference**

[Create Physical File \(CRTPF\) command](#)

[Create Logical File \(CRTLFL\) command](#)

### **Display file design**

Application panels typically consist of major elements, such as constant text strings, input and output fields, and cursor positioning specifications.

**Note:** You can handle these either as a program-described or an externally described file using DDS. The information found in this topic is based on the externally described technique using DDS.

*Constant text strings*

When designing your panels, keep in mind that different languages have different space needs for the same expression.

Do not place many fields on the same line, except for a list panel that has column headings instead of field prompts. Do not overload the panels with information. Choose one of the techniques described under Textual data code design to make your panels.

### **Related concepts**

[Textual data code design](#)

You can use different techniques to specify, store, and use constant text. You can use each technique for specific types of textual data components. Each technique has its advantages and disadvantages.

*Input and output fields*

You must define fields according to the needs of the different languages, countries, cultures, currencies, and laws that you want to address with your application.

For example, assume that you want to store the British pound and the Japanese yen in the same field as the United States dollar. You must set the field size to accommodate the higher number of digits needed for the British pound.

*Field editing specifications*

For the edit specification of your numeric, date, and time fields, you must consider the different cultural conventions of the users.

Do not code the format and editing instructions in your application program in a way that requires program modification when another convention is needed.

### **Related concepts**

[Cultural-dependent design](#)

Different countries might have different standards, which you must consider when developing an NLS-enabled application. This culturally sensitive information must be placed outside the program the same way as the textual data is handled.

#### *Cursor positioning specifications*

Do not specify cursor positioning values to fixed locations on the screen, because different languages have different space requirements.

When you work with different display files, you can adjust them with the translation process. When you need to work with field-independent cursor locations, store the positional information outside of your code and retrieve the variable values for the keyword within your program.

For example:

```
A    record-name    CSRLOC(field-name-1 field-name-2)
```

Cursor positioning on the field level is more useful in an NLS environment. For normal records, this is done by specifying the DSPATR(PC) keyword on a specific field. For subfiles, the cursor can be positioned using SFLRCDNBR(CURSORS) keyword on a special positioning field. In addition, the subfile record number must be stored in that field before the format is written.

For example:

```
A    field-name    4S 0B line pos SFLRCDNBR(CURSORS)
```

**Note:** The name of the record and field where the cursor is positioned, the subfile relative record number, and subfile fold/truncate indicator can be returned to your application program. This function is provided by hidden fields on the DDS keywords RTNCSRLOC, SFLCSRNRN, and SFLMODE.

#### **Related information**

[RTNCSRLOC \(Return Cursor Location\) keyword for display files](#)

[SFLCSRNRN \(Subfile Cursor Relative Record Number\) keyword for display files](#)

[SFLMODE \(Subfile Mode\) keyword for display files](#)

#### *Input field default values*

You can use these methods to put default values into the input fields of your display. Users can override the default values with their own data.

- Getting information from program

Never hard code the values as a literal if they are language or cultural-dependent values. Use values you can get from the system-provided information, such as system or job date, or get the values from a data object, such as a database file or data area from outside of the program.

- Using DDS keywords DFT (Default) or DFTVAL (Default Value)

Specify the default input value directly on the DDS after the keyword. The DDS keyword DFT is for input-only (I) fields. For output-only (O) or input-output (B) fields, use the keyword DFTVAL.

For example:

```
A    field-name    length type I   line pos  DFT('default  ')
OI
A    field-name    length type O/B line pos  DFTVAL('default value ')
```

- Using DDS keyword MSGID (Message Identification)

Using the Message Identification (MSGID) keyword allows you to retrieve the content of a specified message description when the program is run and to put that value as a default in your display file field. The field must be input-output capable (B) for you to use this technique.

For example:

```
A    field-name    length type B line pos  MSGID(message-id [*libl/message-file])
```



This allows you to use different message files for each national language version by setting the library list accordingly when the program is run.

### Related information

[DFT \(Default\) keyword for display files](#)

[DFTVAL \(Default Value\) keyword for display files](#)

[MSGID \(Message Identifier\) keyword for display files](#)

#### *Field validation specifications*

Some DDS keywords provide validation checks on input-capable fields on your display.

- RANGE (Range checking)
- VALUES (Values checking)
- CMP and COMP (Comparison)
- CHECK (Check validity, keyboard control and cursor control)

Using the DDS keywords with any hard-coded values that are language, country, or cultural-dependent makes duplication and modification of the DDS and the application program necessary.

### Example: Validation checks

An example of field validation checks on input-capable fields on your display using the DDS keywords VALUES, COMP, and CHECK follows:

```
A    field-name  length type usage line pos  VALUES('Y' 'N')
or
A    field-name  length type usage line pos  COMP(EQ 'US$')
or
A    field-name  length type usage line pos  CHECK(M10 or M11)
(Modulus checking)
or
A    field-name  length type usage line pos  CHECK(RL)
(Right-to-left support)
```

Validation checks are provided according to the sort sequence defined for the display file at creation time. You can use the same DDS source file to create objects for different languages. For example, the following command creates a display object tagged with the Latin 1 sort sequence table:

```
CRTDSPF FILE(name) SRTSEQ(*LANGIDSHR) LANGID(DEU)
```

The following specification:

```
A    field-name  length type usage line pos  COMP(EQ 'a')
```

accepts all lowercase, uppercase, and accented characters, as defined by the shared-weight in the Latin 1 sort sequence.

In addition, note that all the checks specified using those DDS keywords are done by the data management function of the operating system. Any error message caused by wrong input or handling by the user appears in the language of the operating system. This can be the primary language or a secondary language, depending how the library list of the job is set up.

You can override this when you use the additional DDS keyword CHKMSGID (Check Message Identifier). This keyword allows you to specify your own customized messages and message file to be used by the checking routines of the operating system.

For example:

```
A    field-name  length type usage RANGE(1 999)
A                                CHKMSGID(USR1234 [*lib1/]APPMSGF [&MSGFLD1])
A    MSGFLD1    length type  P    TEXT('Message data field')

and

ADDMSGD  MSGID(USR1234) MSGF(APPTXDENU/APPMSGF)
```

```

MSG('Value &1; is out of range 1 to 999')
and
ADDMSGD MSGID(USR1234) MSGF(APPTXDDEU/APPMSGF)

```

```
MSG('Wert &1; ist ausserhalb des gültigen Bereichs 1 bis 999')
```

To use different message files of different library names, do not specify a fixed library name. You can use a message file for different languages by setting the library list when you run the program.

### Related information

[RANGE \(Range\) keyword for display files](#)

[VALUES \(Values\) keyword for display files](#)

[CMP \(Comparison\) keyword for display files](#)

[CHECK \(Check\) keyword for display files](#)

[CHKMSGID \(Check Message Identifier\) keyword for display files](#)

### Error messages

You can provide error messages in a display file by specifying text as constant, or by using predefined messages.

- Specifying text as constant on ERRMSG or SFLMSG keywords

Specify the text directly as a constant on the DDS keyword. When you want to have more than one language, you must duplicate the DDS source code and translate constants within the DDS specifications. You can then create a separate display file object for each language.

- Using predefined messages on ERRMSGID or SFLMSGID keyword

When using predefined messages instead of constants, you need to have multiple display files.

Instead of using different display files, exchange only the used message file by setting the library according to the language that you want to use.

For example:

```

A   field-name   length type usage  EDTCDE(x)
A 61
A   MSGFLD2     length type  P   TEXT('Message data field')

and

ADDMSGD  MSGID(USR3456) MSGF(APPTXDENU/APPMSGF)
         MSG('Delivery date &1; is earlier than production end date &2')

and

ADDMSGD  MSGID(USR3456) MSGF(APPTXDDEU/APPMSGF)
         MSG('Lieferdatum &1; ist . . .')
.
.
.

```

### Related information

[ERRMSG \(Error Message\) and ERRMSGID \(Error Message Identifier\) keywords for display files](#)

[SFLMSG \(Subfile Message\) and SFLMSGID \(Subfile Message Identifier\) keywords for display files](#)

### Printer file design and translation

Program-described printer files and externally described printer files are two types of printer files. When you design printer files to be translated into a national language version, you should follow some specific guidelines.

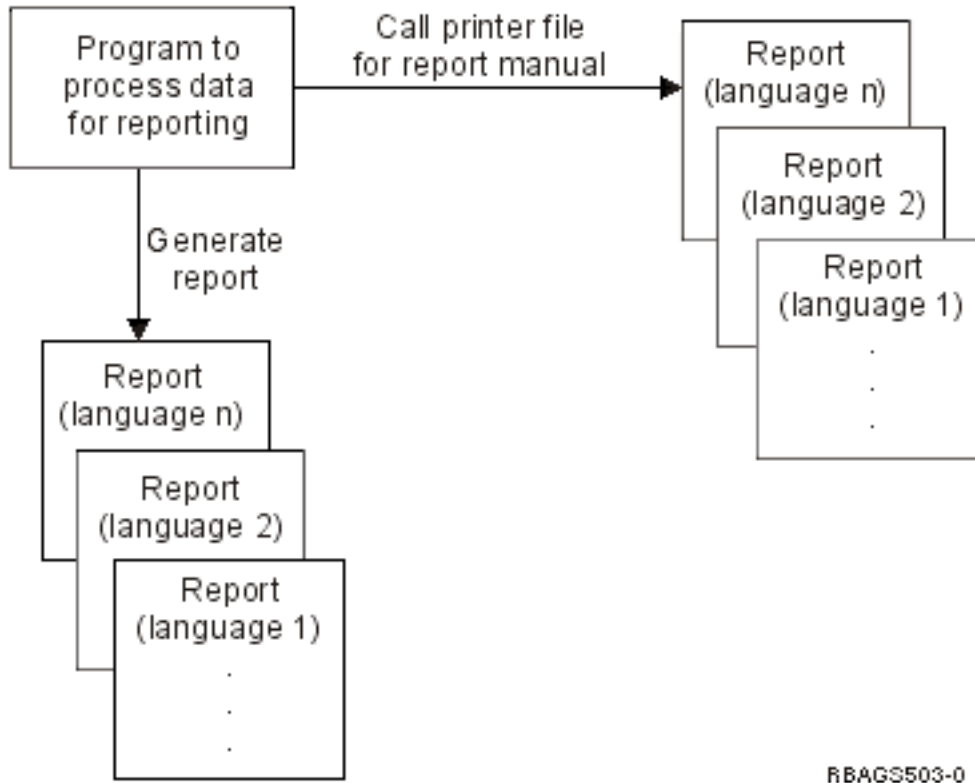
- Program-described printer files

Program-described files rely on the high-level language program to define records and fields to be printed.

- Externally described printer files

Externally described printer files use DDS rather than the high-level language to define records and fields to be printed.

The following figure shows how externally described printer files are used in creating reports for a different national language version.



### Printer file translation

When you design printer files to be translated into a national language version, consider these guidelines:

- Use externally described printer files to define records and fields to be printed. Avoid using program-described printer files. Program-described printer files are described inside the high-level language program. Translators trying to translate text imbedded within the program can mistakenly translate literals that are within your program.
- Print data in one national graphic character set on devices that support the corresponding character sets and code pages. Not all printers support all CHRID parameters.
- Use the MSGCON keyword to access the constant text described in the message file. A printer file does not have the MSGID keyword. However, the techniques of direct coding as unnamed output field (literal) and storing text in a database file can be used to specify the constant text in a printer file.
- Take culture conventions into consideration when bar codes are being described in the printer file. Different countries have different standards for bar codes.
- When entering data, consider these parameters on the Create Printer File (CRTPTF) command.
  - PAGESIZE (page size)  
Different countries have different page-size standards.
  - OVRFLW (overflow line number)  
The overflow line number must be less than or equal to the page length.
  - CHRID (character set and code page)

If the CHRID parameter of the printer file is set to \*DEVLD, the printer uses the character identifier that was set on the control panel or specified in the device description.

If the CHRID parameter of the printer file is set to a specific value, this value determines the code page and character set used to print the data. For externally described printer files, the CHRID parameter is used only for fields that also have the CHRID DDS keyword specified. For all other fields, the code page and character set used is the same as if \*DEVN was specified.

If the CHRID parameter of the printer file is set to \*JOBCCSID, constant text from an externally described printer file is converted to the CCSID of the job. The printer data stream is tagged with the CHRID taken from the job CCSID, using this CHRID value to print the data. When using the \*JOBCCSID value on the CHRID parameter, the CHRID DDS keyword is ignored.

**Note:** All code pages and character sets cannot be handled by all printers.

### **Related concepts**

[Textual data code design](#)

You can use different techniques to specify, store, and use constant text. You can use each technique for specific types of textual data components. Each technique has its advantages and disadvantages.

### **Related reference**

[Create Printer File \(CRTPRTF\) command](#)

### **Source file design**

Database source files are implicitly assigned the CCSID of the job when they are created, unless they have been explicitly assigned a CCSID value through the CCSID parameter on the Create Physical File (CRTPF) or Create Source Physical File (CRTSRCPF) command.

If the job CCSID is 65535, the job default CCSID (DFTCCSID) is used as the implicitly assigned CCSID. The job default CCSID is determined by the system language identifier value and the job DBCS-capable indicator.

### **Related reference**

[Create Physical File \(CRTPF\) command](#)

[Create Source Physical File \(CRTSRCPF\) command](#)

### **Character data representation architecture design**

To enable your application for a multilingual environment, avoid coding CCSID values directly in your DDS for physical files. When database sharing takes place, you need to define your files with the CCSID of the primary language or use Unicode.

- Avoid coding CCSID values directly in your DDS for physical files. When creating different physical files for different languages, change the CCSID for your job (using the CHGJOB command). Only one set of DDS source code needs to be maintained.

Conversions between all CCSIDs might not make sense in all cases. For example, if you access a Greek database with a CCSID of 00875 from a German display station with a job CCSID of 00273, you see garbled data on your display.

Countries outside the Latin-1 character set use character sets that include non-Latin characters. No meaningful conversion is possible between the non-Latin code points and the Latin code points. Arabic, Greek, Hebrew, and Turkish are SBCS languages with non-Latin characters.

- When database sharing takes place, define your files with the CCSID of the primary language being used. Make sure that all users have the CCSID of the language that they use defined in their user profile.

### **Related concepts**

[Working with CCSIDs](#)

Using the system implementation of Character Data Representation Architecture (CDRA), you can achieve consistent representation, processing, and interchange of coded characters (data) on the IBM i operating system and across IBM Systems. The primary implementation of CDRA on the IBM i operating system is through coded character set identifier (CCSID) support.

[Working with Unicode](#)

Unicode is a standard that precisely defines a character set as well as a small number of encodings for it. It enables you to handle text in any language efficiently. It allows a single application to work for a global audience.

### Related reference

[Change Job \(CHGJOB\) command](#)

#### *Use of the Send Network File command*

When you use the Send Network File (SNDNETF) command, the data (if the command is sending a member only) is assumed to be in the CCSID of the job that is running the command. Therefore, no conversion takes place.

When the data is received, you should store the member in a file with the same CCSID as the originating file.

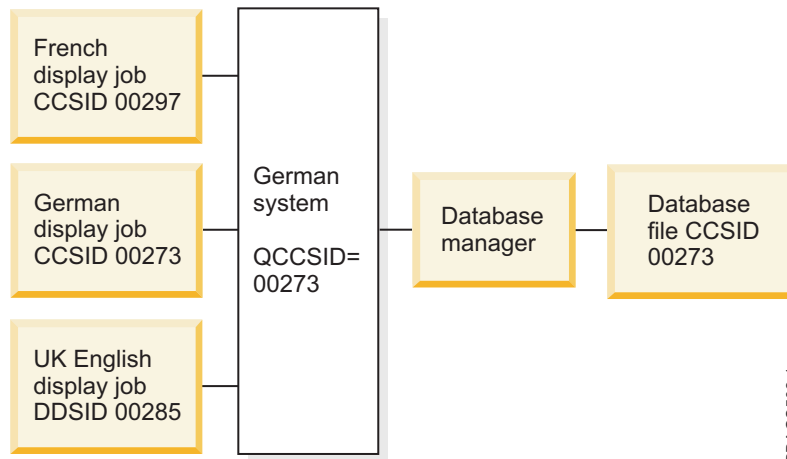
If the receiver does not know the CCSID of the incoming file member, it can be received into a file with a CCSID of 65535, which indicates that no conversion takes place.

### Related reference

[Send Network File \(SNDNETF\) command](#)

#### *Scenario: Multilingual single system*

This scenario shows a multilingual single system with German as the primary language and English and French as secondary languages. All users enter data into the same database file.



On this multilingual system, all users are entering character data into a single file with CCSID 00273 (German), and character data entered from the English and French display stations is being mapped into the German file.

To preserve correct mapping, fields defined as character fields should be actual character fields. If the fields contain application development values (for example, control characters or fields that are not used as real character fields), the fields either should be specified as hexadecimal fields or assigned a CCSID value of 65535.

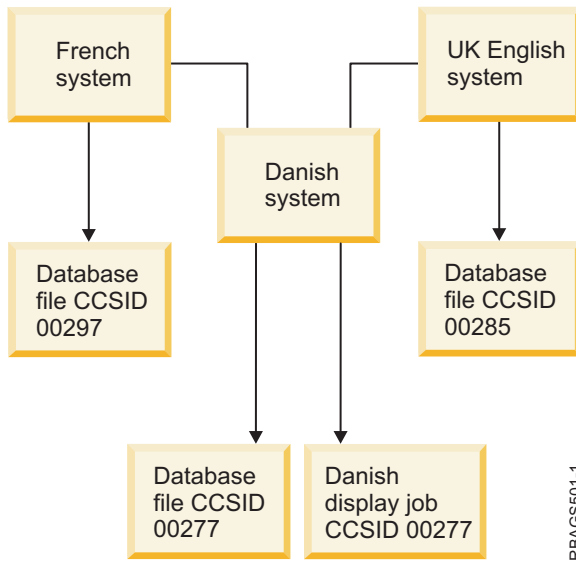
Using CCSIDs, characters that cannot be converted between different code pages are replaced with a substitution code. If you are using a user-defined data stream (UDDS) to format and lay out your display (instead of using DDS), you might get substitution codes returned after the system reads and inserts that data in your user-defined data stream. Substitution codes might cause unpredictable results on the display.

#### *Scenario: Multilingual network*

This scenario shows a multilingual network with three systems located in three different countries, each with a different language.

In this example, the application on the Danish system is using distributed relational database. All national characters (regardless of the language that the data is stored in) are displayed correctly at the Danish display. When the CCSID of the language is used by the database, the integrity of the database is

preserved. The conversion of data between the different code pages is completely automatic and part of the database management.



### **Handling languages with no NLV support**

If you need to support a language that does not have a supported national language version, follow these steps.

1. Study the available national language versions. Find out which national language version most closely resembles your language in character representation.
2. Install the most appropriate national language version as your primary language.
3. Modify the system values to meet your cultural needs. For example, set date and time formats to meet those of the culture that you are supporting.
4. Configure your workstations and printers to match your primary language. Then, handle discrepancies between support for the installed NLV and your own language.

**Note:** The workstation customization functions can support only those capabilities built into your hardware. You cannot support functions through workstation customization that your hardware is unable to support.

5. Use the Create Table (CRTTBL) command to create a sort sequence table based on the existing table that most nearly matches the appropriate sorting sequence for your language.
6. If your language is a DBCS language, create your own characters (UDC) to represent missing characters in the code page associated with the NLV you installed. UDC is an acronym for a user-defined character that is created through the character generator utility (CGU). CGU is an extension of the code page with special user-defined ideographic characters, symbols, or logos.

### **Related concepts**

#### System values for other languages with no national language version

Some of the system values are associated with languages and countries that do not have a national language version. You should set these values immediately after initially installing the IBM i operating system.

### **Related reference**

Create Table (CRTTBL) command

## **Programming considerations in globalized application design**

As you develop your globalized applications, the national language version environment often requires that you pay additional attention to how you prepare and compile your code.

## **Coding globalized applications with high-level languages**

Your major goal must be to have only one general set of running code that is common for all language versions and to make your programs table-driven as much as possible.

You should do as follows:

- Base validity checks on database accesses and message files rather than on hard-coded literals or tables.
- Base calculations on variable factors retrieved from a file rather than coding them inline.
- Place cultural-dependent functions into separate modules of the application and call them when you cannot code them flexibly.

Do not use hard-coded values unless they are fully language and cultural independent on comparison, scan, replace, or call operations. In addition, do not use uppercase or lowercase-sensitive values. For example, never hard code Yes and No (Y or N) responses in your program, because these values are different for every language, and should be part of the textual data.

For literals and constants in source code, use characters only from the invariant character set. If input data is checked for validity in the program, make sure that the characters checked belong to the invariant character set; otherwise you might get a situation where the user is requested to enter a character that is not even on his keyboard. For example, the left brace ( { ) and right brace ( } ) do not appear on Arabic keyboards.

Do not use compile-time arrays to hold messages or any other language or cultural-sensitive data.

For better performance, when you need to call external NLS-dependent modules, call them by a fixed name as a literal (but based on the library list) rather than by a variable field containing the program name. This allows your application to call the modules of different libraries based on the associated library list.

To allow users to work with an application in the language and habits of their culture, specify the editing values (for example, date, time, and date separators) as dependent on the language and country or region. You can then retrieve them according to the information in the user profile. The parameters are LANGID (language identifier) and CNTRYID (country or region identifier). You need to retrieve the cultural-sensitive information only once at program initiation. You can do this by an initial CL program or by the high-level language program and prepare them as:

- Parameters on the call operation
- Parameters on the local data area (LDA)
- Program load tables

Using an initial program allows you to set the user's job attributes to present a consistent application.

### **Related concepts**

#### Invariant character set (and its exceptions)

An *invariant character set* is a character set, such as the syntactic character set, whose code point assignments do not change from code page to code page. The table illustrates the invariant character set (character set 00640) on the IBM i operating system.

#### Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

### **Language compilers CCSID**

Some language compilers expect syntactical operators and the naming convention for the source code to be in CCSID 00037.

You can refer to the documentation for the language compiler you use.

For these compilers, incorrect mapping occurs if the source is compiled with a CCSID other than 00037 or 65535. You must ensure that these compilers receive any variant characters used in language syntax in CCSID 00037.

### ILE language compilers

When an ILE C, ILE RPG, or ILE COBOL program is compiled, source from database source files is converted to the CCSID of the primary source file.

Compilers for these languages can handle syntactical operators in most CCSIDs. These compilers can also handle naming conventions for the source code in most CCSIDs.

### Non-ILE language compilers

When a non-ILE CL, non-ILE RPG, or non-ILE COBOL program is compiled, source from database source files is converted to the CCSID of the job.

If you do not want your names, constants, or literals converted to the CCSID of the job, you can change your job CCSID to 65535. Your constants, literals and names then remain intact.

**Note:** REXX/400 procedures and the literal data coded within them are not converted to the job CCSID.

### Example 1

The following example shows a sample non-ILE RPG program. This example shows English source on a system in the United States.

```
* RPG Source      (Source file created using CCSID 00037 but tagged
*                with CCSID 65535)
FFILE1 IF E              DISK              80
C                READ FILE1
C* Test char
C*
C                FLD1      IFEQ '$'
C                ...
C* Move char
C*
C                MOVE FLD1      FLD$
C                ...
C*
C                SETON              LR
```

### Example 2

In Finland, the program in the first example does not compile because the field name FLD\$ contains a variant character (the dollar sign). The variant character represents a different code point in a code page other than 00037. This figure shows the same sample non-ILE RPG program as English (U.S.) source on a system in Finland (CCSID 278).

```
* RPG Source      (Source file created with CCSID 00037, but tagged
*                with 65535)
FFILE1 IF E              DISK              80
C                READ FILE1
C* Test char
C*
```

```
C FLD1 IFEQ ' ¤ '
```

```
C
C* Move char      ...
C*
```



```
C MOVE FLD1 FLDá
```

```
C  
C*      ...  
C      SETON                                LR
```

### Example 3

You can correct this error by changing the file CCSID to 00037 and setting the job CCSID to 00278 (for Finland). The following example shows the changed file as seen English source in Finland.

```
* RPG Source (Source file created using CCSID 00037 and tagged  
* with CCSID 00037)  
FFILE1 IF E DISK 80  
C      READ FILE1  
C* Test char  
C*  
C      FLD1 IFEQ '$'  
C      ...  
C* Move char  
C*  
C      MOVE FLD1 FLD$  
C      ...  
C*  
C      SETON                                LR
```

### Session manager

For all applications that use a session manager, you must ensure that the output data stream has no X'3F' values in it. The IBM i operating system uses X'3F' values to make a screen blank.

### General sort sequence

The sort sequence used by a program might influence the program logic. The following figure shows an example of this.

Using the Latin 1 shared-weight sort sequence, character test 3 is equivalent to character test 4 (not all characters are shown). When using hexadecimal or unique sorting, they are completely different. The following example shows an RPG program using different sort sequences.

```
* RPG Source (Program created with Latin 1 sort sequence)  
*  
C* Test char 3  
C*  
C      FLD1 IFEQ 'a'  
C      ...  
C* Test char 4  
C*  
C      FLD1 IFEQ 'a'  
C      FLD1 OREQ 'A'
```

```
C FLD1 OREQ 'á'
```

```
C FLD1 OREQ 'Ä'
```

```
C  
C*      ...  
C      SETON                                LR
```

If you compile the program with \*JOB RUN specified for the SRTSEQ parameter and \*JOB RUN specified for the LANGID parameter, the sort sequence table used at run time is not known at compile time.

The IBM DB2® Query Manager and SQL Development Kit for i licensed program and ILE C have additional special considerations.

### Related concepts

[Language identifier \(QLANGID\) system value](#)

The language identifier (QLANGID) system value specifies the default language identifier for the system.

### ***ILE C considerations***

When you compile programs with ILE C, consider this information.

- You can compile a source file in any EBCDIC code page except code page 00290.
- If the CCSID of the primary source file is 65535, code page 00037 is assumed.
- All secondary source files are converted to the CCSID of the primary source file.

**Note:** While most secondary source files are converted to the CCSID of the primary source file, some conversions are not supported. Contact your IBM service representative if you require support for an unsupported CCSID conversion.

- If the CCSID of the secondary source files is 65535, no conversion takes place.
- Any modules are created in the code page of the primary source file. A module is an operating system object that can be a collection of one or more procedures and one or more definitions for external or internal variables. A module is compiled from source code.
- When binding modules of different CCSIDs, no conversion takes place and unpredictable results might occur.
- You can use the trigraph support for the C characters that are not available on all keyboards. Trigraph support generally uses invariant characters to represent variant characters. For example, the left bracket (l) is represented by ??{.

The ILE C runtime library functions that parse strings containing variant characters use the variant character code point value associated with the CCSID of the job.

### ***ILE RPG sort sequence***

The ILE RPG feature, an option of the IBM Rational Development Studio for i licensed program, provides the possibility for a user to specify a sort sequence table and to use the table in comparison operations that are performed with nonnumeric data.

For each of the supported languages, two tables (a shared-weight table and a unique-weight table) are included with the system. With sort sequence support, you can create sort sequence tables based on the existing ones.

The control specifications provide the ILE RPG compiler with information about your program and your system. The sort sequence used in ILE RPG programs is controlled by all of the following items:

- The control specifications.
- The SRTSEQ (sort sequence table) parameter on the Create RPG Module and the Create Bound RPG Program commands.
- The LANGID (language identifier) parameter on the Create RPG Module and the Create Bound RPG Program commands.

The alternative collating sequence field (ALTSEQ) in the control specifications allows the following values:

#### **blank**

No alternative collating sequence is used in the RPG program. The normal collating sequence is used in the RPG program. The compile options SRTSEQ and LANGID are ignored.

#### **\*NONE**

No alternative collating sequence is used in the RPG program. The normal collating sequence is used in the RPG program. The compile options SRTSEQ and LANGID are ignored.

#### **\*SRC**

The alternative collating sequence is used in the RPG program, according to the tables entered at the end of the RPG program. The alternative collating sequence table is loaded at compile time, and ordering, sorting, comparing, and match field processing is done according to that table.

The SORTA and LOOKUP operation codes do not use specified alternative collating sequence tables.

The SRTSEQ and LANGID parameters on the Create RPG Module and Create Bound RPG Program commands are ignored.

## **\*EXT**

The alternative collating sequence is specified outside of the RPG program. RPG compiler imports an external sort sequence table, based on the SRTSEQ and LANGID parameters on the Create RPG Module and the Create Bound RPG Program commands.

The SORTA and LOOKUP function with the arrays and tables at compile time and processing time take effect only when you specify D in the control specifications.

The sort sequence table to be used by the program can be determined at compile time or when the job is run. If the SRTSEQ parameter of the Create RPG Module and Create Bound RPG Program commands:

- Is set to \*HEX, no sort sequence table is used.
- Specifies a table name, then that table is stored with the program object to be used when the job is run. For system-supplied default sort sequence tables for the supported languages, refer to Sort sequence tables.
- Is set to \*LANGIDSHR or \*LANGIDUNQ, the shared-weight or unique-weight table for the language determined by the LANGID parameter is stored with the program object. For a list of valid language identifiers, refer to Language and country and region identifiers.
- Is set to \*JOB, the SRTSEQ parameter of the compile time job is used to determine the sort sequence. The table is stored with the program object.
- Is set to \*JOB RUN, the attributes of the job running the compiled program determine the sort sequence to be used. If the SRTSEQ attribute of the job refers to the LANGID, the LANGID stored with the program object is used. If the LANGID stored with the program is also \*JOB RUN, the LANGID of the runtime job is used.

## **Notes:**

1. If the table to be stored with the program object at compile time does not exist, a table defining hexadecimal sort sequence and tagged with a CCSID value of 65535 is used.
2. If the sort sequence table and the CCSID of the job running the program differ, the table is converted to the CCSID of the job.

## **SORTA and LOOKUP operation codes**

The implementation of compare operation codes, match field and control field processing with the sort sequence tables is the same for the alternative collating sequence and for the sort sequence support. Compare operation codes are ANDxx, COMP, CABxx, CASxx, DOUxx, DOWxx, IFxx, ORxx, and WHxx. Additional functions provided with the SORTA and LOOKUP operation codes follow:

### **SORTA**

The data in the array is sorted according to the sort sequence table.

### **LOOKUP**

To provide proper table searching, the sort sequence table is used with the search arguments in the arrays and tables.

The search argument and either the table or array element are compared using the sort sequence table.

The array and table data are checked using the sort sequence table, whenever ascending or descending sequence is specified. If the SRTSEQ and LANGID parameter values resolve to retrieve the sort sequence table again at run time, then the array and table elements are loaded without a sequence check at the compile time. The sequence checks are performed at run time, according to the sort sequence table.

## **Related concepts**

[Sort sequence \(QSRTSEQ\) system value](#)

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

[Language identifier \(QLANGID\) system value](#)

The language identifier (QLANGID) system value specifies the default language identifier for the system.

#### Sort sequence tables

A sort sequence table is an object that contains the weight of each single-byte graphic character within a specified coded character set identifier (CCSID). The system-recognized identifier for the sort sequence table object type is \*TBL.

#### Country and region identifiers

This table lists the country and region identifiers.

#### **ILE COBOL sort sequence**

The ILE COBOL feature, an option of the IBM Rational Development Studio for i licensed program, uses the sort sequence support in several ways.

- Create COBOL Module command
- Create Bound COBOL Program command
- PROCESS clause
- ALPHABET clause

The ILE COBOL licensed program uses sort sequence tables that are system supplied or user supplied.

#### **Creating COBOL module and creating bound COBOL program commands**

These CL commands have two compiler options relating to sort sequence support: the SRTSEQ parameter and LANGID parameter. The SRTSEQ parameter allows the user to specify any of the system-supplied or user-supplied sort sequence tables residing in a specified library. You can specify whether the sort sequence table should be taken at compile time or run time. Also, you can choose between the shared-weight and unique-weight tables.

With the LANGID parameter, you can specify one of the system-defined language identifiers, or leave that parameter to be defined at the run time.

The meanings of the SRTSEQ and LANGID parameters on the Create COBOL Module and Create Bound COBOL Program commands are the same as on the Create RPG Module and Create Bound RPG Program commands as described in [“ILE RPG sort sequence” on page 94](#).

#### **PROCESS statement**

Sort sequence support options can be supplied in the PROCESS statement. The syntax for that command is like that for the Create COBOL Module and Create Bound COBOL program commands. The only exception to this is that the values for the parameters in the PROCESS statement are entered without an asterisk (\*) for the predefined values. Any options specified in the PROCESS statement override the corresponding options on the Create COBOL Module and Create COBOL program commands.

#### **ALPHABET clause**

The alphabet-name in the ALPHABET clause of the SPECIAL-NAMES paragraph can use the NLSSORT option. Use the SRTSEQ and LANGID parameters of the compiler for alternative collating sequence options. Otherwise, it means the same as the NATIVE option.

The following COBOL lines are affected by the NLSSORT option:

- PROGRAM COLLATING SEQUENCE phrase of OBJECT-COMPUTER paragraph

When evaluating the result of nonnumeric comparisons, the alphabet name has to be referenced in this phrase to enable the program to use the specified sort sequence options. This option also applies to the nonnumeric sort or merge operation. Otherwise, the hexadecimal collating sequence is used.

- ALPHABET CLAUSE in the SPECIAL-NAMES paragraph

This clause should specify the NLSSORT option.

- COLLATING SEQUENCE in the MERGE (or SORT) statement

This phrase is used to specify the collating sequence to be used for nonnumeric comparisons for the KEY data name in the MERGE or SORT operation. If omitted, the PROGRAM COLLATING SEQUENCE clause in the OBJECT-COMPUTER paragraph defines the collating sequence to be used. If neither is specified, hexadecimal collating sequence is used.

- Nonnumeric relation names and condition names

The selected sort sequence table affects the result of certain statements, using nonnumeric relation names and condition names: EVALUATE, IF, PERFORM...UNTIL, SEARCH and START. The truth values of the nonnumeric comparisons depend on the corresponding weights of the characters in the selected sort sequence table. For example, if you specify unique-weight table (LANGIDUNQ) for French (Latin 1), the following statement is true for the single value of the variable ITEM-1,e.

```
IF ITEM-1 = "e"
```

If you specify a shared-weight table (LANGIDSHR) for French (Latin 1), the same statement is true for several values of the variable ITEM-1. All have the same shared weight of 77:

lowercase e (e), uppercase e (E),

lowercase e acute (É), uppercase e acute (É),

lowercase e grave (È), uppercase e grave (È),

lowercase e caret (Ê), uppercase e caret (Ê),

lowercase e umlaut (Ë), uppercase e umlaut (Ë)

### **DB2 and SQL sort sequence**

For Interactive SQL, the SRTSEQ and LANGID parameters can be specified on the STRSQL command. You can change these parameters by using the session services for interactive displays.

Sort sequence tables are used for all string comparisons. String comparisons are performed in the following SQL statements:

- ORDER BY clause
- WHERE clause
- GROUP clause
- HAVING clause
- UNION and UNION ALL clauses
- DISTINCT clause
- BETWEEN predicate
- IN predicate
- LIKE predicate
- MIN and MAX scalar functions
- MIN and MAX column functions

In addition, any indexes or views that are created using the CREATE INDEX or the CREATE VIEW statements are created with the specified sort sequence table.

### **IBM DB2 Query Manager and SQL Development Kit for i**

The IBM DB2 Query Manager and SQL Development Kit for i licensed program does not assume a particular CCSID when precompiling source. Any variant characters in the language syntax, such as the not (¬) symbol, are assumed to be encoded in the CCSID of the source file.

For example, if the source file has a CCSID of 00037, the `␣` not symbol is correctly interpreted to be at code point X'5F'. If the source file has a CCSID of 00500, however, the `␣` not symbol is correctly interpreted to be at code point X'BA'.

A literal is stored in the CCSID of the source file.

The IBM DB2 Query Manager and SQL Development Kit for i licensed program calls the appropriate language compiler to create an SQL program; therefore, you must consider the general guidelines for high-level languages.

### **Related concepts**

Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

### ***IBM i Access sort sequence***

You can specify the sort sequence in IBM i Access functions. When performing queries on the system databases and SQL tables, you can specify the system-supplied or user-supplied sort sequence tables.

### **Remote SQL support**

You can specify the way the selected data has to be sorted when performing the query. For that purpose, sort fields must be specified in the ORDER BY clause. The following clauses also use the specified sort sequence:

- WHERE clause
- GROUP BY clause
- HAVING clause
- JOIN BY clause
- UNION clause
- DISTINCT clause
- IN predicate
- LIKE predicate
- BETWEEN predicate
- RANGE predicate
- MAX function
- MIN function

The actual sort sequence table is retrieved from the job attributes of the user. The SRTSEQ and LANGID parameters can be affected through changing the user profile or changing the job attributes.

### **Data transfer support**

When transferring data from the system to the workstation, you can specify the sort sequence to be applied on selected data. The sort sequence table is also used in the following string comparison operations:

- WHERE clause
- GROUP BY clause
- HAVING clause

- JOIN BY clause
- IN predicate
- LIKE predicate
- BETWEEN predicate
- MAX function
- MIN function

You can specify in the OPTION statement the following parameters related to sort sequence:

- SRTSEQ (sort sequence table)
  - \*JOB
  - \*HEX
  - \*LANGIDSHR
  - \*LANGIDUNQ
  - \*LIBL/sort-seq-table-name
  - \*CURLIB/sort-seq-table-name
  - library-name/sort-seq-table-name
- LANGID (language identifier)
  - \*JOB
  - language-identifier

You can choose the appropriate sort sequence through options on IBM i Access displays.

### Related concepts

Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

Language identifier (QLANGID) system value

The language identifier (QLANGID) system value specifies the default language identifier for the system.

### Coding globalized applications that use bidirectional data

When you are developing NLV-enabled applications, you should consider some specific restrictions on bidirectional languages.

- Bidirectional language display layout

The presentation of data should have a right-to-left orientation. Literals should appear on the right side of the fields that they describe. The following examples illustrate a U.S. English display with a left-to-right orientation and the same display in a right-to-left orientation.

#### Left-to-right layout of a U.S. English display

```

Display employee record (DSPEMPRCD)
Type choices, press enter.

Employee code . . . . . _____ Code, *ALL
Field name . . . . . _____ Name, *ALL
File name . . . . . _____ Name
Library name . . . . . _____ Name, *LIBL
Output to . . . . . _____ *CONS, *PRINT
  
```

#### Right-to-left layout of a U.S. English display

(DSPEMPRCD) drocer eeyolpme yalpsiD  
 .retne sserp ,seciohc epyT

```
*ALL ,edoC _____ . . . . . edoc eeyolpmE
*ALL ,emaN _____ . . . . . eman dieiF
      ,emaN _____ . . . . . eman eliF
*LIBL ,emaN _____ . . . . . eman yrarbiL
*CONS ,*PRINT _____ . . . . . ot tuptuO
```

- Long fields in bidirectional languages

Avoid defining input fields that span more than one line. When the field is displayed or printed as one entity, the result for bidirectional languages is not what the user intended.

- Variable positioning in bidirectional languages

Your application must allow for variables to be in any order. For example, consider the following message in English:

```
File &1 in library &2 not found
```

When translated to another language, the message might look like this:

```
dnuof ton &2 yrarbil ni &1 eliF
```

In this case, variable 2 is positioned before variable 1.

- CHECK(RL) and CHECK(RB) keywords with bidirectional languages

These options are valid only for display stations capable of right-to-left movement, and have the following restrictions:

- Option indicators are not valid with cursor control codes.
- CHECK(RZ) and CHECK(RB) are not valid with these keywords.
- A field that spans more than one line gives a warning message.
- The check digit for modulus checking is the farthest-right byte in the field.
- CHECK(RL) applies to character fields only.

- Online information for bidirectional languages

The special bidirectional tags have a restriction. When combining online help information from several panel groups that do not have the same value for the BIDI tag, the user must use the hot key sequence to read the opposite orientation of the online help information.

- CCSIDs for bidirectional languages

As bidirectional languages have special character sets that are unique, no exchange of data into other languages is feasible. You might need to use data mapping between EBCDIC and ASCII data streams, however. For example, you need data mapping between EBCDIC and ASCII data streams if you are using Distributed Relational Database Architecture™ (DRDA).

When exchanging data in a language that uses Latin characters and when special characters that are not part of the invariant character set are needed, use CCSID 00424 for Hebrew and CCSID 00420 for Arabic for data mapping to take place.

### Related concepts

[CCSID reference information](#)



Coded character set identifier (CCSID) is a 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

#### Working with bidirectional data

Arabic and Hebrew languages use an alphabet written and read from right to left. Numerics and Latin text imbedded in the right-to-left text are written and read from left to right. Therefore, these languages are called bidirectional languages.

#### **Related reference**

Checklist: Bidirectional support guidelines

When you create an application with bidirectional support, follow the guidelines in this table.

#### **Using message catalogs**

The IBM i operating system can use message catalogs to store messages. Messages in a message catalog are grouped as sets. Each message has a unique number within a set.

You can create a message catalog as a stream file, a source file member, or a user space object type from one or more source files.

Because you can store message catalogs as stream files, you can use directories to isolate messages for specific products or national language versions.

#### **Creating or updating a message catalog with the GENCAT and MRGMSGCLG commands**

You can use both the Generate Message Catalog (GENCAT) command or the Merge Message Catalog (MRGMSGCLG) command to create or update a message catalog. When a message catalog exists, continued use of these commands updates a catalog by comparing the original messages to the messages in the source. New message text replaces specific messages without changing the other messages within the set. With these commands, you can add or delete messages from an existing set of messages. You can also delete sets of messages from an existing message catalog.

#### **Related concepts**

CCSID support for messages

You can use CCSID support for handling messages and message catalogs on the IBM i operating system by using commands and application programming interfaces. You can send messages tagged with one CCSID to users with a different CCSID.

#### **Related reference**

Generate Message Catalog (GENCAT) command

Merge Message Catalog (MRGMSGCLG) command

#### **Source for message catalogs**

The source for a message catalog is either a source physical file, a stream file, or multiple files. The source contains fields to define set numbers, message numbers, message text, or to specify sets to delete.

The following information provides additional information and examples relating to message catalogs.

#### **Message catalog source format**

A message catalog contains five fields of message text source lines. A single blank character separates each of the five fields. Any other blank characters are considered as part of the subsequent field data. See “Special characters and escape sequences” on page 103 for additional information.

**Note:** Enter the key fields exactly as in the following list, using the dollar sign (\$) and lowercase characters. Definitions for maximum and minimum values are stored in QSYSINC/QSYS/LIMITS.

- **\$ comment**

A line that begins with \$ that is followed by one or more blank characters is treated as a comment line. A comment line should be placed directly beneath the message to which it refers. Place comments for an entire set directly below the \$set directive in the source file.

- **\$quote C**

This line specifies an optional quote character *C* that is used to surround message text. This character enables trailing spaces or null (empty) messages to be visible in a message source line. By default, or if an empty \$quote directive is supplied, no quoting of message text is recognized.

- **\$set n comment**

This line specifies the set identifier of the messages to follow until the next \$set or end-of-file appears. The *N* denotes the set identifier that is defined by a number between 1 and NL\_SETMAX. Place set identifiers in ascending order within a single source file. They do not need to be contiguous. A character string that follows a set identifier is treated as a comment and ignored.

- **\$delsetncomment**

This line deletes message set *n* from an existing message catalog. The *n* specifies the set number. Data that follows the set number is treated as a comment. The \$set and \$delset identifiers can both be in the message catalog source or the field tags.

- **m message text**

The *m* specifies the message identifier that is defined by a number between 1 and NL\_SETMAX. The message text is stored in the message catalog with message identifier *m* with the set identifier that is specified in the last \$set directive. If the message text is empty and a blank character field separator is present, it stores an empty string in the message catalog. Existing messages get deleted from the catalog if the message line does not have a field separator or MESSAGE TEXT and a NEWLINE or carriage return follows the message line. Message identifiers must be in ascending order, noncontiguous, and within a single set. The length of the MESSAGE TEXT must be in the range of 0 to NL\_TEXTMAX.

**Note:** Empty lines in a message text source file will be ignored.

## Messages programming format

MESSAGES should follow these recommendations:

- The last line of all messages should end with \n.
- The second and remaining lines of a message should begin with \t, indicating a tab.
- All lines of messages that continue to the next line should end with \n\, indicating that the message continues to the next line.
- The quotation mark at the end or beginning of a line should be omitted. The quotation mark delineates the beginning and end of a complete message.

## Using multiple source files

You can specify multiple source files for the source file parameter. The messages that are contained in all of the files must follow the same rules for sets and messages as defined in a single source file. For example, the first source file contains messages in sets 1 through 3. The next source file must begin with set 3 and have a message number greater than the last message number in the first source file. If not, it must contain sets that begin with a number higher than the highest number (set 3) in the previous source file.

## Replacing messages

Messages in an existing message catalog can be replaced by specifying a source file that contains the same set number and message number as the message text you want to change. All other messages in the source file remain the same. To update a value for the \$QUOTE in a catalog, use the same \$QUOTE character in subsequent source files.

## Example source for a message catalog

Here is a sample format for the source that is used to create a message catalog. A quotation mark delineates each message. The message text that is stored in the message catalog has had the extraneous blank characters removed. This example describes three sets of messages. Set 2 is deleted while sets 1 and 3 remain stored in the message catalog.

```
$ Messages for my new product
$quote "

$set 1

1 "Error occurred.\n"
$ The next message is continued on the next line.
2 "This is a very long message \n\
\t that requires another line to display. \n"
3 "Specify a value greater than %d.\n"
4 "File %c cannot be used at this time.\n"

$set 2
1 "Error %d occurred. \n"
2 "Flag not set.\n"
3 "Number of arguments must be %d.\n"

$set 4
1 "Before using this command, you must \
set the correct values in the %c box.\n"
2 "You have not properly NLS enabled this function.\n"
10 "Messages should end with a %c.\n"

$delset 2
```

**Note:** Message 2 in set 1 will be displayed in two lines. Message 1 in set 4 will display as a one line message.

Here is an example for using the MRGMSGCLG command to create a message catalog.

```
MRGMSGCLG CLGFILE('/MYPRODUCT/MESSAGES?US')
SRCFILE('QSYS.LIB/MYLIB.LIB/MYSOURCE.FILE/US.MBR')
CLGCCSID(*SRCCSID) SRCCSID(*SRCFILE)
TEXT('Message catalog for USA')
```

This example creates a message catalog into the stream file US in directory /MYPRODUCT/MESSAGES using the source from MYLIB library in file MYSOURCE and member US. The CCSID of the data in the message catalog is the same as the CCSID tag of the source file.

## Special characters and escape sequences

Text strings can contain special characters and escape sequences as defined in the following table.

Description of special characters	Sequence
\	\\
backspace	\b
carriage return	\r
form feed	\f

Description of special characters	Sequence
horizontal tab	\t
NEWLINE	\n
octal bit pattern	\ddd <b>Note:</b> The escape sequence \ddd consists of a backslash followed by up to three octal digits that specify the value of the required character. If the character following the backslash is not an octal digit, the backslash and data following are included as part of the text.

**Opening, extracting, and closing message catalogs**

After you have created a message catalog, you can use these functions: CATOPEN(), CATGETS(), and CATCLOSE().

**CATOPEN()**

Opens a message catalog

**CATGETS()**

Extracts a message from a message catalog, given a set identifier and a message identifier

**CATCLOSE()**

Closes the message catalog

The C function CATOPEN opens the message catalog. If no slash (/) characters are found in the name, the NLSPATH environment variable and the LC\_MESSAGES category are used to find the specified message catalog. If the name contains one or more slash (/) characters, the name is interpreted as a path name of the catalog to open.

A default path is used if there is no NLSPATH environment variable or a message catalog cannot be found in the NLSPATH path specified. If the value of oflag is NO\_CAT\_LOCALE the environment variable setting of LC\_MESSAGES may affect the default path. If the value of oflag is zero the LANG environment variable may affect it also.

**Related information**

[WebSphere Development Studio: ILE C/C++ Language Reference PDF](#)

**Delivering globalized applications**

As you prepare to deliver your globalized application, you should consider how globalization issues might affect the ways that your customers install and use your application.

**Hardware support for multilingual systems**

Hardware, in this context, means the physical keyboards, displays, printers, and controllers that make up a IBM i platform. The extent to which this hardware supports national languages might impose limitations on the degree of support that you can provide with an application. You must refer to the reference manuals for non-IBM hardware to determine what limitations, if any, are imposed by that hardware.

**Character data translation**

Translating is changing the meaning of character data from a set of concepts, ideas, and statements in one human language to a culturally similar meaning in another human language. You can follow some basic rules to ensure translation goes smoothly. A subset of these rules is provided in the User interfaces topic.

## Delivering your globalized application to customers

Delivering your application to customers includes the processes of packaging, servicing, supporting, and educating users about your application. You must consider various tasks when following these processes in different countries and cultures throughout the world.

### Related concepts

#### User interfaces

A user interface is the part of a software product that your customer actually sees.

#### Packaging and installation process

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

## Handling data in globalized applications

---

The IBM i operating system enables you to handle data in a globalized environment. This topic collection describes Unicode and Unicode data, the Chinese standard GB18030, how to use CCSIDs to integrate multiple language environments consistently, and how to use bidirectional data, DBCS data, and locales.

One of the most critical challenges you might encounter as you work with globalized systems and applications is the effective interaction with data. The operating system provides a wide range of options that you can use to ensure that data is viewed and processed seamlessly across national languages.

### Related concepts

#### Developing globalized applications

Globalized applications are applications that have national language support. National language support allows users to enter, store, process, retrieve, print, and display data in their chosen language. It also allows users to see and enter commands, prompts, messages, and documentation in their chosen language, in formats matching their cultural expectations.

## Working with Unicode

*Unicode* is a standard that precisely defines a character set as well as a small number of encodings for it. It enables you to handle text in any language efficiently. It allows a single application to work for a global audience.

Before Unicode, the encoding systems that existed did not cover all the necessary numbers, characters, and symbols in use. Different encoding systems might assign the same number to different characters. If you used the wrong encoding system, your output might not have been what you expected to see.


Unicode provides a unique number for every character, regardless of platform, language, or program. Using Unicode, you can develop a software product that works with various platforms, languages, and countries. Unicode also allows data to be transported through many different systems. Modern systems provide Internationalization solutions based on Unicode.

Unicode was developed as a single-coded character set that contains support for the common languages around the world. The first version of Unicode used 16-bit numbers, which allowed for encoding 65 536 characters without complicated multibyte schemes. With the inclusion of more characters, and following implementation needs of many different platforms, Unicode was extended to allow more than one million characters. In addition, other encoding schemes were added, such as UTF-8, UTF-16, and UTF-32. This introduced more complexity into the Unicode standard, but far less than managing a large number of different encodings.

The original Unicode repertoire covered all major languages commonly used in computing. Unicode continues to grow and to include more scripts.

The design of Unicode differs in several ways from traditional character sets and encoding schemes:

- Its repertoire enables users to include text efficiently in almost all languages within a single document.

- It can be encoded in a byte-based way with one or more bytes per character, but the default encoding scheme uses 16-bit units that allow much simpler processing for all common characters.
- Many characters, such as letters with accents and umlauts, can be combined from the base character and accent or umlaut modifiers. This combining reduces the number of different characters that need to be encoded separately. *Precomposed* variants for characters that existed in common character sets at the time were included for compatibility. For example, Latin small letter A used with a combining tilde results in .

Characters and their usage are well-defined and described. Traditional character sets typically provide only the name or a picture of a character and its number and byte encoding; Unicode has a comprehensive database of properties available. It also defines a number of processes and algorithms for dealing with many aspects of text processing to make it more interoperable.

The early inclusion of all characters of commonly used character sets makes Unicode a useful mechanism for converting between traditional character sets, and makes it feasible to process non-Unicode text by first converting the text into Unicode, processing the text, and then converting it back to the original encoding without loss of data.

### **Related concepts**

Scenarios: [Setting up IBM i with a national language version](#)

These scenarios demonstrate how you can enable multilingual support on the IBM i operating system.

[Character data representation architecture design](#)

To enable your application for a multilingual environment, avoid coding CCSID values directly in your DDS for physical files. When database sharing takes place, you need to define your files with the CCSID of the primary language or use Unicode.

### **Related information**

[Unicode Home Page](#)

### **Why use Unicode**

Unicode has many advantageous functions.

The operating system provides multilingual support. Unicode provides the means to store and retrieve data in the user's national language of choice in a single file, and therefore provides for one database file to support all text needs, regardless of the language of the input device. For example, the same parts file can have Greek, Russian, and English descriptions and names in it.

### **Related concepts**

[GB18030: The Chinese standard](#)

GB 18030-2000 is a Chinese standard that specifies an extended code page for use in the Chinese market.

### **Different encodings of Unicode**

The Unicode standard has several main ways in which a Unicode value can be encoded. They are UTF-8, UTF-16, and UTF-32. Unicode Transformation Format (UTF) is the algorithmic mapping from every Unicode value to a unique byte sequence.

#### **UTF-8**

UTF-8 converts Unicode data through a mathematical algorithm so that UTF-8 uses 8 data bits to encode the data, keeps all ASCII codes from 00 to 7F encoded as themselves, and contains nulls only when they are the intended characters.

For example, the string "ABC" in Unicode is "004100420043"x. However, in UTF-8 it is "414243".

Because UTF-8 allows Unicode data to flow over an 8-bit network without the network needing to know that it is Unicode, UTF-8 is used to store Unicode on several UNIX platforms and is used as the default encoding for most new internet standards.

UTF-8 is used mainly as a direct replacement for older MBCS encodings, which all use 8-bit code units, but it takes some more code to process it. It is a good encoding if 90% of your data is English, because all English letters use only one byte.

The IBM i operating system supports UTF-8 encoding with CCSID 1208. Beginning with IBM i V5R3, CCSID 1208 is supported in database.

### **UTF-16**

UTF-16 is an encoding of Unicode in which each character is composed of either one or two 16-bit elements.

Unicode was originally designed as a pure 16-bit encoding, aimed at representing all modern scripts. Over time, and especially after the addition of over 14 500 composite characters for compatibility with established sets, it became clear that 16 bits were not sufficient for most users. Out of this arose UTF-16.

UTF-16 allows access to about 60 000 characters as single Unicode 16-bit units. It can access an additional 1 000 000 characters by a mechanism known as surrogate pairs.

Two ranges of Unicode code values are reserved for the high (first) and low (second) values of these pairs. Highs are from 0xD800 to 0xDBFF, and lows from 0xDC00 to 0xDFFF. Because the most common characters have already been encoded in the first 64 000 values, the characters requiring surrogate pairs are relatively rare.

UTF-16 is extremely well designed as the best compromise between handling and space, and all commonly used characters can be stored with one code unit per code point. This is the default encoding for Unicode.

The IBM i operating system supports UTF-16 encoding with CCSID 1200 (and CCSID 13488). Beginning with IBM i V5R3, CCSID 1200 is supported in database. CCSID 13488 has been supported in database for several releases.

### **Related concepts**

#### UCS-2 and its relationship to Unicode (UTF-16)

The UCS-2 standard, an early version of Unicode, is limited to 65 535 characters. However, the data processing industry needs over 94 000 characters; the UCS-2 standard has been superseded by the Unicode UTF-16 standard.

### **Related information**

[Unicode Home Page](#)

### **UTF-32**

UTF-32 is an encoding of Unicode in which each character is composed of 4 bytes.

The IBM i operating system does not support UTF-32 encoding with a CCSID value.

Unicode was originally designed as a pure 16-bit encoding, aimed at representing all modern scripts. Over time, and especially after the addition of over 14 500 composite characters for compatibility with established sets, it became clear that 16 bits were not sufficient for many users. Out of this arose UTF-32.

UTF-32 allows characters to be encoded as 4 bytes at any code point from 00000000 to 0010FFFF. For example, the string *ABC* in UTF-32 is encoded as `x"00000004100000004200000043"`.

### **Related information**

[Unicode Home Page](#)

### **UCS-2 and its relationship to Unicode (UTF-16)**

The UCS-2 standard, an early version of Unicode, is limited to 65 535 characters. However, the data processing industry needs over 94 000 characters; the UCS-2 standard has been superseded by the Unicode UTF-16 standard.

The IBM i operating system supports CCSID 13488, defined as UCS-2, and CCSID 1200, defined as UTF-16. The system treats both CCSID 13488 and CCSID 1200 as UTF-16 encodings.

Using either scheme, you will have the same results for almost all system operations. However, certain SQL functions that operate on a character boundary defined by the SQL standard can produce different results. For instance, the SQL functions of CHARACTER, LENGTH, POSITION, and SUBSTRING distinguish UTF-16 and UCS-2, and therefore you get different results. See the SQL reference for more information about these functions.

## **UCS, UCS-2 (Universal Multiple-Octet Coded Character Set)**

The ISO 10646 standard is a character code designed to encode text for storage in computer files. The design of the ISO 10646 standard is based on today's prevalent character code, ASCII (and ISO 8859-1, an extended version of the ASCII code). But ISO 10646 goes beyond ASCII's ability to encode only the Latin alphabet. The ISO 10646 encoding provides the capability to encode all of the characters used for written languages throughout the world.

### **Related concepts**

#### UTF-16

UTF-16 is an encoding of Unicode in which each character is composed of either one or two 16-bit elements.

### **Related information**

#### SQL reference

### **How Unicode relates to prior standards such as ASCII and EBCDIC**

The Unicode standard is advantageous to other standards. It can reduce the complexity of handling character data in globalized applications.

### **Evolving standards based on limited platforms**

The representation of character data in modern computer systems can be fairly complicated, depending on the needs of your globalized application. One of the reasons for this complexity is that the methods for handling this data have evolved from early methods that served less complicated environments and hardware platforms.

In fact, many early decisions about how to encode characters on a system were guided by the functional requirements of specific devices, such as the early Telex (TTY) terminals and punch card technologies. For example, the Delete character (with an ASCII value of x'7F') was required in order to punch out all of the holes in a column of a punch card to signify that the column should be ignored. The storage capacities of these early computing systems placed additional limitations on system and application designers.

The character encoding schemes that have grown out of these early systems were built on this historical foundation:

- The ASCII (American Standard Code for Information Interchange) character set uses 7-bit units, with a trivial encoding designed for 7-bit bytes. It is the most important character set in use today, despite its limitation to very few characters, because its design is the foundation for most modern character sets. ASCII provides only 128 numeric values, and 33 of those are reserved for special functions.
- The EBCDIC (Extended Binary-Coded Decimal Interchange Code) character set and a number of associated character sets, designed by IBM for its mainframes, uses 8-bit bytes. It was developed at a similar time as ASCII, and shares the same set of base characters and has other similar properties. Unlike ASCII, the Latin letters are not combined in two blocks for upper- and lower-case. Instead, the letters are arranged so that their hexadecimal values have second digits of 1 through 9 (another punch card-friendly design).

### **Historical simplicity creates modern complexity**

The physical and functional limitations of the early character sets gave way to rapidly expanding hardware and functional capabilities. Character representation on computing systems became less dependent on hardware; instead, software designers used the existing encoding schemes to accommodate the needs of an increasingly global community of computer users.

### **Character sets for many characters**

The most common encodings (character encoding schemes) use a single byte per character, and they are often called single-byte character sets (SBCS). They are all limited to 256 characters. Because of this, none of them can even cover all of the accented letters for the Western European languages. Consequently, many different such encodings were created over time to fulfill the needs of different user



communities. The most widely used SBCS encoding today, after ASCII, is ISO-8859-1. It is an 8-bit superset of ASCII and provides most of the characters necessary for Western Europe.

However, East Asian writing systems needed a way to store over 10 000 characters, and so double-byte character sets (DBCS) were developed to provide enough space for the thousands of ideographic characters in East Asian writing systems. Here, the encoding is still byte-based, but each two bytes together represent a single character.

Even in East Asia, text contains letters from small alphabets like Latin or Katakana. These are represented more efficiently with single bytes. Multi-byte character sets (MBCS) provide for this by using a variable number of bytes per character, which distinguishes them from the DBCS encodings. MBCSs are often compatible with ASCII; that is, the Latin letters are represented in such encodings with the same bytes that ASCII uses. Some less often used characters may be encoded using three or even four bytes.

An important feature of MBCSs is that they have byte value ranges that are dedicated for lead bytes and trail bytes. Special ranges for lead bytes, the first bytes in multibyte sequences, make it possible to decide how many bytes belong together to encode a single character. Traditional MBCS encodings are designed so that it is easy to go forwards through a stream of bytes and read characters. However, it is often complicated and very dependent on the properties of the encoding to go backwards in text: going backwards, it is often hard to find out which variable number of bytes represents a single character, and sometimes it is necessary to go forward from the beginning of the text to do this.

Examples of commonly used MBCS encodings are Shift-JIS and EUC-JP (for Japanese), with up to 2 or 3 bytes per character.

### **Stateful encodings**

Some encodings are stateful; they have bytes or byte sequences that switch the meanings of the following bytes. Simple encodings, like mixed-byte EBCDIC, use Shift-In and Shift-Out control characters (bytes) to switch between two states. Sometimes, the bytes after a Shift-In are interpreted as a certain SBCS encoding, and the bytes after a Shift-Out as a certain DBCS encoding. This is very different from an MBCS encoding where the bytes for each character indicate the length of the byte sequence.

The most common stateful encoding is ISO 2022 and its language-specific variations. It uses Escape sequences (byte sequences starting with an ASCII Escape character, byte value 27) to switch between many different embedded encodings. It can also *announce* encodings that are to be used with special shifting characters in the embedded byte stream. Language-specific variants like ISO-2022-JP limit the set of embeddable encodings and specify only a small set of acceptable Escape sequences for them.

Such encodings are very powerful for data exchange but hard to use in an application. Their flexibility allows you to embed many other encodings, but direct use in programs and conversions to and from other encodings are complicated. For direct use, a program has to keep track not only of the current position in the text, but also of the state--which embeddable encoding is currently active--or must be able to determine the state for a position from considerable context. For conversions to other encodings, converting software might need to have mappings for many embeddable encodings, and for conversions from other encodings, special code must figure out which embeddable encoding to choose for each character.

### **Why Unicode?**

Hundreds of encodings have been developed, each for small groups of languages and special purposes. As a result, the interpretation of text, input, sorting, display, and storage depends on the knowledge of all the different types of character sets and their encodings. Programs are written to either handle one single encoding at a time and switch between them, or to convert between external and internal encodings.

Part of the problem is that there is no single, authoritative source of precise definitions of many of the encodings and their names. Transferring of text from one machine to another one often causes some loss of information. Also, if a program has the code and the data to perform conversion between a significant subset of traditional encodings, then it carries several megabytes of data around.

Unicode provides a single character set that covers the languages of the world, and a small number of machine-friendly encoding forms and schemes to fit the needs of existing applications and protocols. It is

designed for best interoperability with both ASCII and ISO-8859-1, the most widely used character sets, to make it easier for Unicode to be used in applications and protocols.

Unicode is in use today, and it is the preferred character set for the Internet, especially for HTML and XML. It is slowly being adopted for use in e-mail, too. Its most attractive property is that it covers all the characters of the world (with exceptions, which will be added in the future). Unicode makes it possible to access and manipulate characters by unique numbers (that is, their Unicode code points) and use older encodings only for input and output, if at all.

### **International Components for Unicode**

The International Components for Unicode (ICU) is a C library that provides a full-featured, industrial strength, Unicode support.

The library provides:

- Calendar support
- Character set conversions
- Collation (language-sensitive)
- Date and time formatting
- Locales (many)
- Message catalogs (resources)
- Message formatting
- Normalization
- Number and currency formatting
- Time zones
- Transliteration
- Word, line, and sentence breaks

ICU is a collaborative, open-source development project jointly managed by a group of companies and individual volunteers throughout the world, using the Internet and the Web to communicate, plan, and develop the software and documentation.

ICU is open source. For more information about ICU license, see the International Components for Unicode Web site.

### **Related concepts**

[International Components for Unicode](#)

### **Mapping of data**

The IBM i operating system uses the EBCDIC encoding scheme. However, not all clients attached to the system use an EBCDIC encoding scheme to store, retrieve, and process data. Therefore, some clients use Unicode as an exchange mechanism that is safe across all platforms.

Some clients might use ASCII, PC DATA, or other encoding schemes. They can use Unicode to prevent the loss of data due to incomplete conversion between encoding schemes and code pages.

### ***Example: Displaying data without Unicode***

This example highlights two users on the same system, one English and the other Greek. The English user has the display device CCSID set to 37, and the Greek user has the display device CCSID set to 875. Both users query, update, and replace data in the DATABASE1 database.

DATABASE1 is tagged with CCSID 37.

Problems with data integrity develop because users are operating with CCSIDs that have varied character support. That is, not all characters in CCSID 37 are available in CCSID 875 and vice versa.

Assume that the following names are to be entered by the English-speaking user (display device supports a CCSID of 37):

- Ά alson
- Gifford

When these entries are stored, the data integrity remains intact. That is, an Ά is stored as an Ά. This is because the display device CCSID and the database CCSID are both 37.

Assume the following names are also input into DATABASE1 by the Greek-speaking user (display device CCSID of 875):

- Μ π έ ν
- Ω ρ ι μα

DATABASE1 now consists of the following logical entries:

- Ά alson
- Gifford
- Μ π έ ν
- Ω ρ ι μα

The Greek characters that make up the name are stored as those characters only if the same character exists within CCSID 37. If the character does not exist, the system converts the characters using a predetermined algorithm to a code point from code page 37. The algorithm converts Ω to Ά.

The following list shows the code point used to store the first character of each name in DATABASE1. (Using only the first character makes the example easier by eliminating long strings of code points which are shown if the code point is presented for each character in the name.)

**Name**

CCSID 37 Stored Code Point (Hexadecimal)

Ά alson  
67...

Gifford  
C7...

Μ π έ ν  
53...

Ω ρ ι μα  
67...

The next step in this example is to show how data can be incorrectly selected due to the character conversion when it was stored in the database.

Assume that the Greek user wants to find all names beginning with Ω. The following SQL statement can provide two names: Ω ρ ι μα and Ά alson

Select from DATABASE1 where name LIKE ' Ω %'

The search yielded an unexpected name (Ά alson). This is because the first character in Ά alson is stored with the same code point as the first character in Ω ρ ι μα.

**Related concepts**

[Example: Displaying data with Unicode](#)

This example highlights two users on the same system, one English and the other Greek. The English user has the display device CCSID set to 37, and the Greek user has the display device CCSID set to 875. Both users query, update, and replace data in the DATABASE1 database.

**Example: Displaying data with Unicode**

This example highlights two users on the same system, one English and the other Greek. The English user has the display device CCSID set to 37, and the Greek user has the display device CCSID set to 875. Both users query, update, and replace data in the DATABASE1 database.

DATABASE1 is tagged with CCSID 13488.

This example, using unicode as the CCSID of DATABASE1, shows how data integrity is maintained both in storing and retrieving data.

As in the example of Displaying data without Unicode information, one user is English using CCSID 37 and the other user is Greek using CCSID 875.

DATABASE1 is used as in the previous example. However DATABASE1 is now defined with CCSID 13488. (13488 is a unicode CCSID.)

- Å alson
- Gifford
- Μ π έ ν
- Ω ρ ι μ α

The key difference in using unicode as the CCSID of DATABASE1 is that data integrity is maintained for each user who inputs data to the database. That is each character, regardless of the CCSID of the inputting device, is stored with a unique code point. (Remember that in this example the CCSID of DATABASE1 is 13488.)

**Name**

CCSID 13488 Stored Code Point (Hexadecimal)

Å alson  
00C5 ...

Gifford  
0047 ...

Μ π έ  
03A9 ...

Ω ρ ι μ α  
039C ...

Assume that the Greek user wants to find all names beginning with Ω. The following SQL statement can provide one name, Ω Ω ρ ι μ α, as compared to two in the previous example:

```
Select from DATABASE1 where Substr(name,1,1) = ' Ω '
```

The reason for this is that each character stored in a unicode tagged database has a unique code point. This contrasts to the example of Displaying data without Unicode information that had the first character in Å alson stored with the same code point as the first character in Ω ρ ι μ α

**Related concepts**

[Example: Displaying data without Unicode](#)

This example highlights two users on the same system, one English and the other Greek. The English user has the display device CCSID set to 37, and the Greek user has the display device CCSID set to 875. Both users query, update, and replace data in the DATABASE1 database.

### Unicode on IBM i

The IBM i operating system provides support for Unicode.

Unicode cannot be specified as a value for:

- The system CCSID
- A user profile CCSID
- A job CCSID

The IBM i operating system provides external support for Unicode in the following parts of the system:

- Database files and functions
- Db2 for i
- DDS
- Display file and panel groups
- ILE high-level languages such as RPG
- Message handling and message catalogs
- Query files and tools
- SQL tables
- Unicode variables in UIM

Several other IBM i functions use Unicode internally so that character data integrity is maintained for users across multilingual platforms.

**Note:** These topics do not give detailed information about application development as it relates to the implementation of Unicode. Rather, they provide highlights of IBM i support for Unicode. Where possible, these topics provide reference to a book that provides detailed information for unicode implementation. You need to have the information about the Unicode standard available and understand the information.

### Related information

[Unicode Home Page](#)

### *Database files and functions*

When you create Unicode database applications, you need to consider the implications for creating physical files, creating logical files, and for database input/output.

### Creating physical files

Unicode graphic fields can be created in physical files. This is done by specifying a G data type and a Unicode CCSID for the CCSID keyword.

The following example shows the DDS for a physical file containing four fields, and the command for creating the file:

```
A          R  FMT1
A          EMPNO          6A
A          NAME           30G          CCSID(1200)
A          DESCR1         500G          CCSID(1200)  VARLEN
A          DESCR2         500A

CRTPF FILE(UNICODEPF) SRCFILE(CLR/QDDSSRC)
```

In the example:

- The first field, EMPNO, is a character field of length 6. The CCSID of the EMPNO field is the SBCS CCSID of the job. The decision was made to use a character field because the EMPNO field contains only numerics and Unicode support is not needed.
- The NAME and DESCR1 fields are both Unicode fields. Both of these fields may need to contain data from more than one EBCDIC code page so the decision was made to make these fields Unicode graphic.
- The DESCR2 field is the SBCS CCSID of the job. This field is used as illustration of mapping to a logical field in [Creating logical files](#).

You can specify the default (DFT) keyword for Unicode graphic fields. The default value can be specified as SBCS, bracketed-DBCS, or bracketed-DBCS-graphic character strings. If you do not specify the DFT keyword, the default value for fixed-length Unicode fields is the Unicode blank (hexadecimal 0020). For varying-length Unicode fields, the default is the empty string.

### Creating logical files

You can use logical files to map Unicode data to and from character, DBCS-open, or DBCS-graphic. This allows Unicode graphic data to be manipulated in a character based form.

The following example shows the DDS for a logical file containing 4 character fields. The Unicode graphic data is converted to character data when reading from the logical file, and character data is converted to Unicode graphic data when writing to the file.

```

R FMT1                PFILE(UNICODEPF1)
A                    EMPNO
A                    NAME          A          CCSID(37)
A                    DESCR1        A          CCSID(37)
A                    DESCR2        G          CCSID(1200)

```

### Database input/output

Whenever reading or writing data from or to a field tagged with a Unicode CCSID to the job physical files, the data is passed as Unicode data without any conversions occurring. Regardless of the job CCSID, data is passed as Unicode data. When writing data to a logical file, the *from* CCSID is the job CCSID; however, if the job CCSID is 65535, the *from* CCSID is the CCSID of the field in the logical file.

Here are some scenarios from the previous physical and logical files. For the scenarios, the job CCSID is 297.

**Scenario 1.** When reading the data from the physical file:

- EMPNO is converted from its CCSID to 297.
- NAME is not converted but is left as Unicode data.
- DESCR1 is not converted but is left as Unicode data.
- DESCR2 is converted from its CCSID to 297.

**Scenario 2.** When writing the data to the physical file:

- EMPNO is converted from 297 to its CCSID.
- NAME is not converted but is left as Unicode data.
- DESCR1 is not converted but is left as Unicode data.
- DESCR2 is converted from 297 to its CCSID.

**Scenario 3.** When reading the data from the logical file:

- EMPNO is converted from its CCSID to 297.
- NAME is converted from Unicode data to character data with a CCSID of 297.
- DESCR1 is converted from Unicode data to character data with a CCSID of 297.

- DESCR2 is converted from character data to Unicode data and not converted to the job CCSID.

**Scenario 4.** When writing the data to the logical file:

- EMPNO is converted from 297 to its CCSID.
- NAME is converted from 297 to Unicode data.
- DESCR1 is converted from 297 to Unicode data.
- DESCR2 is converted from Unicode to its CCSID in the physical file.

**Scenario 5.** If the job was 65535, the conversions for the previous fields are:

- EMPNO is not converted.
- NAME is converted from 37 to Unicode data.
- DESCR1 is converted from 37 to Unicode data.
- DESCR2 is converted from Unicode to its CCSID in the physical file.

### **Related concepts**

Object-level coded character set identifier 65535

CCSID 65535 is the default object-level CCSID for message files and message queues.

### ***Db2 for i***

When using Db2 for i applications, you need to be aware of some restrictions of Unicode and some commands.

- Implicit conversion when comparing Unicode fields with character/IGC/graphic fields as well as with literals and host variables can occur.
- Physical and logical files with Unicode fields cannot have their CCSIDs changed with the Change Physical File (CHGPF) command.
- A Unicode CCSID is not allowed on the CHGPF command.
- The Copy File (CPYF) and Copy From Query File (CPYFRMQRYP) commands with FMTOPT(\*MAP) specified is not allowed when copying from or to a Unicode graphic field unless:
  - The corresponding field is a Unicode or DBCS-graphic field.
  - The corresponding field is a character, DBCS-open, DBCS-either, or DBCS-only field with a CCSID other than 65535.
- The Copy File (CPYF) command supports copying of SBCS character, DBCS-open, DBCS-only, DBCS-either, and DBCS-graphic fields to and from Unicode graphic fields. There is limited support for Unicode on the FROMKEY, TOKEY, INCCHAR, and INCREL parameters.

### **Related concepts**

Object-level coded character set identifier 65535

CCSID 65535 is the default object-level CCSID for message files and message queues.

### **Related reference**

Change Physical File (CHGPF) command

Copy File (CPYF) command

Copy From Query File (CPYFRMQRYP) command

### ***SQL tables***

SQL supports tables that contain Unicode graphic columns by specifying a Unicode CCSID for the GRAPHIC and VARGRAPHIC data types.

The following SQL example creates the table U\_TABLE. U\_TABLE contains one character column called EMPNO, and two Unicode graphic columns. NAME is a fixed-length Unicode graphic column and DESCRIPTION is a variable-length Unicode graphic column. The decision was made to use a character field because the EMPNO field only contains numerics and Unicode support is not needed. The NAME and

DESCRIPTION fields are both Unicode fields. Both of these fields may contain data from more than one EBCDIC code page.

```
CREATE TABLE U_TABLE (EMPNO CHAR(6) NOT NULL,  
NAME GRAPHIC(30) CCSID 1200,  
DESCRIPTION VARGRAPHIC(500) CCSID 1200)
```

### Inserting data

SBCS character, mixed character, and DBCS graphic data can be inserted into Unicode graphic columns using the SQL INSERT statement. Db2 for i SQL converts the data to Unicode graphic data. In SQL programs, the DECLARE VARIABLE statement can be used to attach a Unicode CCSID to graphic host variables.

The following SQL example converts character data to Unicode graphic data for the NAME and DESCRIPTION columns and inserts the row into the U\_TABLE.

```
INSERT INTO U_TABLE VALUES('000001','John Doe','Engineer')
```

### Selecting Unicode data

Implicit conversion of Unicode graphic data is supported on a FETCH or select INTO and CALL.

In the following example, the EMPNO column is returned in empno\_hv as character data. The NAME column is returned in name\_hv as Unicode graphic data because name\_hv is a Unicode variable. It is not converted to character, mixed character, or DBCS graphic.

```
...  
char empno_hv[7];  
wchar_t name_hv[31];  
EXEC SQL DECLARE :name_hv VARIABLE CCSID 13488;  
...  
EXEC SQL SELECT EMPNO, NAME  
INTO :empno_hv, :name_hv  
FROM U_TABLE;  
...
```

To return Unicode graphic data as EBCDIC data, the prior example can be changed to return the Unicode data as character data, EMPNO and NAME are returned in the job CCSID.

```
...  
char empno_hv[7];  
char name_hv[31];  
...  
EXEC SQL SELECT EMPNO, NAME  
INTO :empno_hv, :name_hv  
FROM U_TABLE;  
...
```

When doing selection, implicit conversions are done when comparing Unicode graphic data and character or DBCS graphic data.

The following example converts the character string 'John Doe' to Unicode graphic and then selects the rows where the NAME column is 'John Doe'.

```
EXEC SQL DECLARE C1 CURSOR FOR  
SELECT *  
FROM U_TABLE  
WHERE NAME = 'John Doe';
```

The SQL Reference information includes additional information about using SQL with Unicode graphic data.



## Related concepts

[SQL reference](#)

### **Query files and tools**

The Open Query File (OPNQRYF) command can retrieve or perform the selection of Unicode data. IBM Query for i, DB2 Query Manager, and IBM i DB2 Query Management all have Unicode support.

### **Open query file (OPNQRYF) command considerations**

The Open Query File (OPNQRYF) command, shown as follows, can retrieve or perform selection of Unicode data. Using the MAPFLD parameter, data can be mapped to or from Unicode.

```
OPNQRYF FILE(U_TABLE)
QRYSLT(' NAME=MAPNAME ')
MAPFLD((MAPNAME 'John Doe' *GRAPHIC *N *N 1200))
```

### **Interactive query tools considerations**

IBM Query for i, DB2 Query Manager, and IBM i DB2 Query Management all have Unicode support. Unicode data can be displayed or printed on a report by implicitly converting to either character or mixed art.

### **Related reference**

[Open Query File \(OPNQRYF\) command](#)

### **Related information**

[Query Manager Use PDF](#)

[Query Management Programming PDF](#)

### **Data description specifications**

In data description specifications (DDS), you use the CCSID file-, record-, or field-level keyword to specify that a G-type field supports Unicode data instead of DBCS-graphical data.

### **Related concepts**

[CCSID \(Coded Character Set Identifier\) keyword for physical and logical files](#)

### **Display files and panel groups**

The 5250 data stream has an option to support Unicode. Some 5250 emulators like IBM i Access Client Solutions or IBM Rational Host On-Demand support the sending of Unicode to the device. Other 5250 emulators, like IBM Personal Communication do not support Unicode in the data stream.

It is suggested that all Unicode capable fields be initialized in the output buffer before writing the fields to the screen. Unpredictable results can occur if default initialization is allowed to take place.

### **Unicode not supported**

If Unicode data is not supported, conversions between the Unicode data and EBCDIC are necessary during input and output operations. On output, the Unicode data is converted to the CCSID of the device. On input, the data is converted from the device CCSID to the Unicode CCSID.

Because the device CCSID, which is determined from the device configuration, determines what the Unicode data is converted to, the converted data can appear differently on different devices since some devices may not support some scripts.

**Note:** Display devices do not have a 'CCSID' parameter, rather, they use the Character identifier (CHRID) parameter to determine the device CCSID. This means the CHRID configured must map to a supported CCSID and not all CCSIDs defined on the system are supported.

**Using the \*LEN or \*MIN option on the CCSID keyword in DDS.**

If you are not dealing with Japanese, Chinese or Korean data it may be possible to use \*MIN with the CCSID keyword since this will show a smaller field on most screens.

For example if we define a field to be 10

```
A          field1          10G  B  7  32CCSID(13488)
```

The Latin characters of "abcdefghij" will be displayed as:

```
abcdefghij_____
```

and there will be 10 empty screen positions after the text. Using \*MIN, like CCSID(13488, \*MIN) will result in only showing the 10 screen positions as in:

```
abcdefghij
```

(See the note on \*MIN below for Unicode supported devices)

If you are dealing with Japanese, Chinese or Korean data it may be necessary to use the length keyword in DDS so as to provide more space in order to hold shift-in and shift-out characters used to show those characters on the EBCDIC device.

### Unicode is supported

For the case where Unicode is sent to the device, one potential issue to be aware of is the screen space needed for a Unicode field. Different characters in different scripts can display as different widths in the same Unicode field. Some take 1 screen space whereas others can take 2. This means a user can type more data into the field than will be returned to the application.

For example if we define a field to be 10

```
A          field1          10G  B  7  32CCSID(13488)
```

The Latin characters of "abcdefghij" will be displayed as:

```
abcdefghij_____
```

and there will be 10 empty screen positions after the text. This may look like the user could add more characters since the field shows as 20 screen positions, 10 of which are blank, however since each character takes 2 bytes in storage you can only enter 10. Any character entered after 10 will be discarded when the screen is returned to the program

The same field showing 10 ideographic characters will need 20 screen positions since each ideographic character needs 2 screen spaces to be displayed. The ideographic characters of "赵钱孙李周吴郑王冯陈" will be displayed as:

```
赵钱孙李周吴郑王冯陈
```

and there will be no empty screen positions after the text.

**Note:** Using \*MIN with the CCSID keyword can cause problems since \*MIN will restrict the data sent to the device or subfile to the smallest size, in our example 10 screen positions. As show above the ideographic text needs 20 screen positions so the use if \*MIN is not recommended for Unicode capable devices.

### Related concepts

[Unicode considerations for database files](#)

### **Unicode variables in user interface manager**

This example shows how to define a Unicode variable in user interface manager (UIM).

```
1      :class name=example basetype='graphic 6 13488' width=10,  
2  
3      :class name=example2 basetype='graphic 10 13488' width=20.  
4
```

Line 1 defines a class for variables that will contain 6 Unicode characters and is to be displayed in a field that is 10 bytes long.

Line 3 defines a class for variables that will contain 10 Unicode characters and is to be displayed in a field that is 20 bytes long.

#### **Related reference**

[DDS concepts](#)

#### **Related information**

[Application Display Programming PDF](#)

## **GB18030: The Chinese standard**

GB 18030-2000 is a Chinese standard that specifies an extended code page for use in the Chinese market.

The IBM i operating system supports this encoding with CCSID 1392. Generally, you should use Unicode instead of CCSID 1392 for complete national language support. CCSID 1392 is provided if you need to handle or interchange GB18030 encoded data.

### **A brief history of major GB code pages**

A common base code page standard for Chinese is GB 2312-1980. It encodes more than 6000 frequently used Chinese ideographs. With the growing importance of Unicode and the parallel standard ISO 10646 (which was adopted by China as GB 13000), an extension of GB 2312-1980 was created. This extension was called GBK; it encoded all 20 902 unified ideographs that are assigned in Unicode 2.1. GBK is not a formal standard, but is a widely implemented specification.

Unicode 3.0 added more than 6000 ideographs, and version 3.1 added about 42 000 additional ideographs.

GB 18030 was created as an update of GBK for Unicode 3.0 with an extension that covers all of Unicode. It has the following general features:

- GB 18030 character assignments are compatible with the GB 2312-1980 standard and the GBK specification.
- The mapping table between GB 18030 and Unicode is compatible with the one between GB 2312-1980 and Unicode. With some exceptions (with the one between GBK and Unicode), most of the changes compared to the GBK mapping table are due to updates for Unicode 3.0.
- GB 18030 specifies a mapping table that covers all Unicode code points. It is functionally similar to a UTF (Unicode Transformation Format) while maintaining compatibility of GB-encoded text with GBK and GB 2312-1980.

#### **Related concepts**

[Why use Unicode](#)

Unicode has many advantageous functions.

#### **Related information**

[Unicode Home Page](#)

## **Working with CCSIDs**

Using the system implementation of Character Data Representation Architecture (CDRA), you can achieve consistent representation, processing, and interchange of coded characters (data) on the IBM i operating

system and across IBM Systems. The primary implementation of CDRA on the IBM i operating system is through coded character set identifier (CCSID) support.

### **Related concepts**

#### Automatic character set and code page conversion

The IBM i operating system provides automatic conversion between character set and code pages for all applications that are enabled for national language support.

#### Character data representation architecture design

To enable your application for a multilingual environment, avoid coding CCSID values directly in your DDS for physical files. When database sharing takes place, you need to define your files with the CCSID of the primary language or use Unicode.

#### Sort sequence tables

A sort sequence table is an object that contains the weight of each single-byte graphic character within a specified coded character set identifier (CCSID). The system-recognized identifier for the sort sequence table object type is \*TBL.

### **Related tasks**

#### Enabling the secondary language

You must ensure that secondary languages can be used after they have been installed on the system.

### **Recommendations and guidelines for using CCSIDs**

These recommendations are useful when you write globalized applications.

- Because the system is included with a default CCSID of 65535, character data conversions do not normally occur in applications. You should look over the CCSID information in this topic, however, because the system might need to participate in a multilingual environment, a network, or exchanging data at a later time.
- Applications implementing their own mapping scheme should use CCSID 65535, where a CCSID assignment is necessary. For example, depending on what an application does, it might need to use CCSID 65535 for the files, or it might need to use CCSID 65535 for the jobs. Because other applications may require CCSIDs other than 65535, consider changing such applications by replacing the mapping scheme with CCSID support.
- Correctly define fields based on their usage. If fields contain application-dependent values (for example, control characters or fields that are not used as real character fields), define the fields as hexadecimal data or character fields with CCSID 65535.
- Avoid using characters that are not in the invariant character set for names and literals in programs.

Follow these guidelines when using CCSIDs:

- Use CCSIDs in multilingual applications to maintain character integrity in database files, displays, and printed data.
- You can find a suggested CCSID for a language in Language identifiers and associated default CCSIDs.
- If the QIGC system value is set on, set QCCSID as a mixed CCSID or 65535.
- If you use DBCS support, set the job CCSID to a mixed CCSID. If you do not, set the job CCSID to a single-byte CCSID.
- Ensure that the QCHRID code page is compatible with the character set and code page of the QCCSID value, unless the QCCSID value is 65535. If the QCCSID value is changed to a value that is incompatible with the current QCHRID value, the QCHRID value is changed to a compatible value by the system.
- If you use a user-defined data stream (UDDS), remove any X'3F' values inserted by CCSID conversions. Otherwise, your data can cause the system to blank out a screen. Some CCSID conversions use a X'3F' value for a substitution character.
- If you are using any interactive jobs, such as Application Development ToolSet/400, ensure that the code page of the job CCSID matches the code page of the keyboard type. If these CCSID values do not match, or the job CCSID is 65535, unpredictable results might occur.
- Be aware that the \*JOBCCSID support is not used by any system-supplied displays or panel groups, although CHRIDCTL support is used.

- Be aware of character data that has been defined or specified as control information. For new database files, fields that contain control information should be defined as hexadecimal data type or use CCSID 65535 instead of another CCSID.
- Because of workstation hardware restrictions, you might not see all of the characters on displays other than 3486, 3487, 3488, or Personal System/2 (PS/2) displays when CCSID conversion occurs. However, the character data is retained in the system.
- Be aware that when a CCSID conversion is performed, substitution characters might cause a loss of data. The situation occurs if enforced subset match conversion is performed.

### Related concepts

#### Object-level coded character set identifier 65535

CCSID 65535 is the default object-level CCSID for message files and message queues.

#### Language identifiers and associated default CCSIDs

This table shows the language identifiers and the job default CCSID (DFTCCSID) values associated with those identifiers.

#### DBCS system indicator (QIGC) system value

The DBCS system indicator (QIGC) system value specifies whether a double-byte character set (DBCS) national language version (NLV) is installed. This value is set when the primary national language version is installed.

#### National language keyboard types and SBCS code pages

This table lists the keyboard types and code pages for each national language supported by the IBM i operating system. The Create Device Display (CRTDEV DSP) command uses the KBDTYPE parameter.

#### Conversion of character data

The Character Data Representation Architecture (CDRA) system of tags ensures that you can convert character data in a predictable, repeatable way.

### IBM i function support for CCSIDs

The system provides CCSID support in the functions as shown in the table.

Function	Description of support
CL commands	Some control language (CL) commands have internal functions that support CCSID conversions. For more information about CL commands that support CCSID conversions, see the CL Reference topic.
Copy	Coded character set identifier (CCSID) support is built into the copy function. The Copy File (CPYF) and Copy from Query File (CPYFRMQRYP) commands support CCSIDs. To use the CPYF command to change a physical file, see Changing the CCSID of a physical file. The Copy Source File (CPYSRCF) command supports CCSID conversion.
Database management	Database management support provides default coded character set identifier (CCSID) values for database files on the system.

Function	Description of support
DDM	<p>Coded character set identifier (CCSID) support is built into distributed data management (DDM). DDM provides support to pass CCSID tags in homogeneous environments. DDM passes a CCSID parameter when sending files. With DDM, you can also specify a CCSID when creating files on a remote system. DDM only converts data to the job CCSID of the source system when:</p> <ul style="list-style-type: none"> <li>• The IBM i operating system is running on both the source and target systems.</li> <li>• The source and target systems are at an operating system level of Version 2 Release 1.1 or later.</li> </ul> <p>Program-described files are always created with a CCSID of 65535 if they are created:</p> <ul style="list-style-type: none"> <li>• On a target system running a release level from OS/400® Version 2 Release 1.1 through OS/400 Version 2 Release 3</li> <li>• From a source system that is not running the IBM i operating system</li> <li>• From a source system that is a system at a release level before OS/400 Version 2 Release 1.1</li> </ul> <p>You can use the Submit Remote Command (SBMRMTCMD) command on a source system to change the file CCSID (externally described files only) by specifying the Change Physical File (CHGPF) command and the CCSID parameter.</p>
DDS	<p>Coded character set identifier (CCSID) support is built into data description specifications (DDS). DDS supports file-level and field-level CCSID keywords for all character fields in physical files. DDS also supports file-level and field-level keywords for all DBCS fields in physical files.</p>
Distributed relational database	<p>Coded character set identifier (CCSID) support is built into distributed relational database. Distributed relational database passes the CCSID of an application requester (AR) job to an application server (AS) job and vice versa during connect processing. Distributed relational database also performs a conversion of error information and text-describing fields according to the job CCSID.</p> <p>Distributed relational database uses CCSID information to determine how to build data exchanged between application requester jobs and application server jobs. It also uses CCSID information to describe data exchanged between application requester jobs and application server jobs (for example, a format description).</p>

Function	Description of support
IDDU	Coded character set identifier (CCSID) support is built into interactive data definition utility (IDDU). Interactive data definition utility provides support to specify a CCSID for a character field or a DBCS field.
Open Query File (OPNQRYF)	<p>Coded character set identifier (CCSID) support is built into IBM i query. You can use the Open Query File (OPNQRYF) command to specify a CCSID on the MAPFLD parameter. The MAPFLD parameter specifies the definition of query fields that are either mapped to, or derived from, other fields.</p> <p>IBM i query supports CCSID conversion on CHAR, OPEN, EITHER, and UCS-2 graphic field operators for join, record selection, group-by, and minimum or maximum values functions. CCSID conversion is performed whenever fields do not have the same CCSID value. After the query is opened, database management support converts data read or written to the database files as described in the Database Management topic.</p> <p>IBM i query does not support CCSID conversion if at least one of the fields is assigned a CCSID of 65535.</p>
Query management	<p>Coded character set identifier (CCSID) support is built into query management. Query management assigns a CCSID to queries and forms. Query management:</p> <ul style="list-style-type: none"> <li>• Converts queries to the job CCSID.</li> <li>• Presents data to the display device using the job CCSID.</li> <li>• Assigns a CCSID to the files it creates.</li> </ul>
SNA	Coded character set identifier (CCSID) support is built into SNA Distributed Services (SNADS). SNADS supports CCSIDs by any user ID, system name, or destination queue name. However, other SNADS services such as SNDNETF do not provide CCSID conversion.
Work management	Work management support provides the function to assign or change coded character set identifier (CCSID) values at three different levels. See the Work management topic for details.
Workstations	The workstation management function provides support for display files, printer files, and panel groups. See Workstation function management for details.

Function	Description of support
Message management	<p>Coded character set identifier (CCSID) support is built into the system's message support. Use CCSID support for handling messages and message catalogs on IBM i:</p> <ul style="list-style-type: none"> <li>• You can send messages tagged with one CCSID to users with a different CCSID.</li> <li>• You can use CCSID support to handle messages by using commands and application programming interfaces.</li> </ul>

### Related concepts

[Changing the CCSID of a physical file](#)

### Related reference

[Control language \(CL\)](#)

[Copy File \(CPYF\) command](#)

[Copy From Query File \(CPYFRMQRYF\) command](#)

[Copy Source File \(CPYSRCF\) command](#)

[Submit Remote Command \(SBMRMTCMD\) command](#)

[Change Physical File \(CHGPF\) command](#)

[Open Query File \(OPNQRYF\) command](#)

### Database management

Database management support provides default coded character set identifier (CCSID) values for database files on the system. All database files are assigned a CCSID. At file creation time, the CCSID is either explicitly assigned through DDS, SQL, or IDDU, or implicitly assigned the job default CCSID (DFTCCSID).

### Database files support for CCSIDs

IBM system files and licensed program database files are created with the CCSID of choice for each of the national language versions. Only the customer files are automatically assigned the CCSID of the job creating the file. You can use the Display File Description (DSPFD) command to view the CCSID of a file.

Program-described files are assigned CCSID 65535. If a CCSID is not explicitly specified on the CRTPF or CRTSRCPF command, database source files default to the job default CCSID at file creation.

If a database logical file is defined over several physical files, it is assigned a CCSID at the field level and assumes the CCSID value of the physical file. Logical files cannot be explicitly assigned a CCSID value.

### Database fields and support for CCSIDs

Except for numeric database fields, database fields are supported by CCSIDs. You can use the Display File Field Description (DSPFFD) command to view the CCSID of the fields in a file.

Hexadecimal fields are assigned CCSID 65535.

An implicit CCSID value is assigned to the following fields if a CCSID was not explicitly assigned through DDS, SQL, or IDDU at file creation:

- Physical-file character
- DBCS-open
- DBCS-only
- DBCS-either



- Graphic

The implicitly assigned CCSID is the job default CCSID, or a CCSID associated with the job default CCSID.

- A character field is assigned the single-byte character set (SBCS) CCSID that is associated with the job default CCSID.
- A DBCS-open, DBCS-only, and DBCS-either field is assigned the mixed byte CCSID.
- A Graphic field is assigned the double-byte character set (DBCS) CCSID that is associated with the job default CCSID.

For example, if the job default CCSID is 5026 (which is a CCSID that identifies mixed data), an SBCS character field is assigned the SBCS CCSID associated with 5026. Thus, the CCSID for that field is 290. If there is no CCSID of the required character set type then a CCSID of 65535 is used.

Database logical-file fields are assigned a CCSID value based on their data type and the data type of the underlying physical file field.

### **Database management and conversion support for CCSIDs**

Database management support converts non-graphic character data read from, or written to, database files using the file CCSID and the job CCSID.

- If data is being read from a database file and the CCSID of the file is the same as the job CCSID, no conversion is done.
- If data is being read from a database file and the CCSID of the file and the job CCSID are different, the data is converted to the CCSID of the job.
- If data is being written to a database file and the CCSID of the file is the same as the job CCSID, no conversion is done.
- If data is being written to a database file and the CCSID of the file and the job CCSID are different, the data is converted to match the CCSID of the file.

No conversion is performed if either the CCSID of the job or the CCSID of the database file is equal to 65535.

### **Related concepts**

Job attributes

Job attributes are set at the time a job starts.

Language identifiers and associated default CCSIDs

This table shows the language identifiers and the job default CCSID (DFTCCSID) values associated with those identifiers.

### **Related tasks**

Job default coded character set identifier

A job attribute, job default CCSID (DFTCCSID), is created for jobs with a CCSID of 65535. The DFTCCSID value is used by a system code when a CCSID other than 65535 is needed.

### **Related reference**

Display File Description (DSPFD) command

Display File Field Description (DSPFFD) command

### **Work management**

The job's coded character set identifier (CCSID) value is assigned or changed at any one of these levels: job level, user profile level, or system level.

All jobs run with a CCSID value established at one of these levels:

- *Job level.* A CCSID is assigned to a job.

- *User profile level.* A CCSID is specified in a user profile and the value is assigned to all jobs run under that user profile. The CCSID can be set or changed with the Create User Profile (CRTUSRPRF) and Change User Profile (CHGUSRPRF) commands.
- *System level.* The system value QCCSID is the default CCSID for all jobs running on the system. QCCSID can be set or changed with the CHGSYSVAL and WRKSYSVAL commands.

Work management support initializes the job CCSID for an interactive job to the CCSID on the user profile when the job starts. If \*SYSVAL is specified for the CCSID on the user profile, work management support gets the CCSID from the system value (QCCSID). For batch jobs, the CCSID of the current job is used as the default CCSID for the submitted job.

You can change the CCSID of a job by using the Change Job (CHGJOB) command. Make a note of the current job CCSID. You can use it later to reset the job CCSID to its original value, if necessary. The new CCSID value is reflected in the job immediately. The job DFTCCSID cannot be changed. To retrieve the CCSID or DFTCCSID for a job, use the Retrieve Job Attributes (RTVJOB) command or the Retrieve Job Information QUSRJOBI application programming interface (API). Interactively, use the Work with Job (WRKJOB) command and select the Display Job Definition Attributes option on the Work with Job display.

### **Related reference**

[Create User Profile \(CRTUSRPRF\) command](#)

[Change User Profile \(CHGUSRPRF\) command](#)

[Change System Value \(CHGSYSVAL\) command](#)

[Work with System Value \(WRKSYSVAL\) command](#)

[Retrieve Job Attributes \(RTVJOB\) command](#)

[Retrieve Job Information \(QUSRJOBI\) API](#)

[Work with Job \(WRKJOB\) command](#)

### **Workstation function management**

Workstation function management involves working with display files, printer files, as well as panel group objects and user interface manager (UIM) menus.

**Note:** All source files on the system are tagged with a coded character set identifier (CCSID).

#### *Display files*

When a display file object is created, it is tagged with the coded character set identifier (CCSID) of the source file.

At compile time:

- All character data is read from the primary source file without any character conversion being performed.
- User message text (identified by the MSGCON keyword in DDS) remains the same because it is assumed to be in the same CCSID as the primary source file.

At run time, the constant data is converted based on the CHRID parameter value used to create the display file object. This conversion is optional and can occur only when the CHRID is set to \*JOBCCSID or indirectly with CHRIDCTL. This conversion is from the display file CCSID to the character identifier (CHRID) of the device. The field-level keyword NOCCSID (no coded character set identifier) allows the user to specify fields within the DDS that are never to be converted.

**Note:** To use data management support of CCSIDs, you must change source physical files tagged with CCSID 65535 to a CCSID value that is associated with the data.

### **CHRID parameter on the Create Display File command**

The CHRID parameter on the Create Display File (CRTDSPF) command affects the conversion that occurs for the display file.

If the \*JOBCCSID value is specified on the CHRID parameter of the CRTDSPF command:

- Input characters are converted from the device character identifier (CHRID) to the job CCSID.
- Character data is sent to output-capable fields and converted from the job CCSID to the device CHRID.
- Constant text from the display file is converted from the CCSID of the display file to the CHRID of the device.
- All message files are tagged with a CCSID. Message text is converted from the CCSID of the message file to the CHRID of the device. When message files are tagged with a CCSID of 65535 (the system default), it is assumed that the contents of the message files are already in the CHRID of the device. To ensure that appropriate conversions occur, you can enable CCSID support for messages.
- Message replacement data is converted from the CCSID of the job, or from the CCSID of the display file, to the CHRID of the device.
- All status messages that are tagged with a CCSID other than 65535 are converted to the CHRID of the device.
- Message text for messages on a message line or in a message subfile (identified by the ERRMSG, ERRMSGID, SFLMSG, and SFLMSGID keywords in DDS) is converted from the message file CCSID to the device CHRID.

If a specific value is specified for the CHRID parameter on the CRTDSPF command, conversion is done between the CHRID specified on the CRTDSPF command and the CHRID of the device. This conversion affects only fields defined with the CHRID DDS keyword.

If the \*DEVD value is specified on the CHRID parameter of the CRTDSPF command, no conversion is performed. This is the default setting.

### **Migration of display files with CCSID 65535**

All source files in Version 3 of the IBM i licensed program have an implicit CCSID value of 65535. To have appropriate CCSID support, display files must be recompiled with a source file that has a CCSID value other than 65535 if either of the following conditions are true:

- The display file was originally compiled from a source file with a CCSID value of 65535.
- The display file was originally compiled before Version 2 Release 3 Modification 0 of the OS/400 licensed program.

By recompiling, the display file object is tagged and all necessary conversions take place when needed.

No conversions take place if the source files are explicitly tagged CCSID 65535.

### **Related concepts**

Character identifier control (QCHRIDCTL) system value

The character identifier control (QCHRIDCTL) system value controls the type of CCSID conversion that occurs for display files, printer files, and panel groups.

Changing the CCSID of a physical file

You can use the Change Physical File (CHGPF) command to change the coded character set identifier (CCSID) of a physical file. However, the physical file cannot be changed under certain conditions.

CCSID support for messages

You can use CCSID support for handling messages and message catalogs on the IBM i operating system by using commands and application programming interfaces. You can send messages tagged with one CCSID to users with a different CCSID.

### **Related reference**

Create Display File (CRTDSPF) command

### **Related information**

MSGCON (Message Constant) keyword for display files

ERRMSG (Error Message) and ERRMSGID (Error Message Identifier) keywords for display files

SFLMSG (Subfile Message) and SFLMSGID (Subfile Message Identifier) keywords for display files

### *Printer files*

When a printer file object is created, it is tagged with the coded character set identifier (CCSID) of the source file. Processing of the source files for printer files is the same as for display files. At compile time, all character data is read from the primary source file without any character conversion being performed.

When printing to the device, if the \*JOBCCSID value is specified on the CHRID parameter of the CRTPRTF command:

- Constant text from an externally described printer file is converted from the CCSID of the printer file to the CCSID of the job.
- Character data sent to output fields is assumed to be already converted to the job CCSID.

If the printer data stream is tagged with the character identifier (CHRID) derived from the CCSID of the job, the CHRID value is used by the printer to interpret the data. The CHRID value is ignored for printers not supporting this function.

If a specific value is set for the CHRID parameter on the CRTPRTF command:

- For externally described printer files, fields that specify the CHRID DDS keyword use the CHRID value specified on the printer file. The remainder of the file is printed as if \*DEV D was specified for the CHRID parameter on the CRTPRTF command.
- For program-described printer files, the printer data stream uses the CHRID value specified on the printer file.

If the \*DEV D parameter is specified on the CHRID parameter of the CRTPRTF command, no conversion is performed.

The CHRID information is determined by either the printer hardware or by the device description. If the CHRID information is obtained from the device description, it is then sent to the printer.

### **Related reference**

[Create Printer File \(CRTPRTF\) command](#)

### *User interface manager menus and panel groups*

Like display files and printer files, panel group objects and user interface manager (UIM) menus are tagged with the CCSID of the primary source file. The contents of embedded source members are converted to this CCSID.

When the panel group or UIM menu is created with \*JOBCCSID specified for the CHRID parameter, conversion is performed at run time. Conversion is performed between the CCSIDs of the panel group or menu, the job, and the CHRID of the display or printer.

### **CCSID conversions of user interface manager menu and panel groups**

The following CCSID conversions occur for displays of panel groups and UIM menus:

- Text in the panel group is converted from the panel group CCSID to the device CHRID.
- Text in the UIM menu is converted from the UIM menu CCSID to the CHRID of the device.
- Variables from the user job are converted from the job CCSID to the device CHRID.
- Variables from the job are converted from the CHRID of the device to the job CCSID.
- Online help information imported from a different panel group is converted from the imported panel group CCSID to the device CHRID.

### **CCSID conversions when printing UIM menus and panel groups**

CCSID conversions for printed UIM menus and panel groups are shown in the following table. In this table, xxx and yyy are explicitly assigned CCSID values. For example, a printer file CHRID is explicitly assigned a value of 00697 00037. The panel group is set to \*JOBCCSID. The panel group constant text is converted from the panel group primary source file tagged with CCSID 00500 to the printer file CHRID 00697 00037.

Printer file CHRID is	And the panel group or menu CCSID is xxx	or *JOBCCSID	or *DEVD
yyy	No conversion occurs for panel group constant text.	Panel group constant text is converted from panel group primary source file CCSID to yyy.	No conversion occurs for panel group constant text.
	Variables with CHRID=PNLGRP on class tag are converted from xxx to yyy.	Variables with CHRID=PNLGRP on class tag are converted from job CCSID to yyy.	No conversion occurs for variables with CHRID=PNLGRP on class tag.
	No conversion occurs for variables without CHRID=PNLGRP on class tag.	Variables without CHRID=PNLGRP on class tag are converted from job CCSID to YYY.	No conversion occurs for variables without CHRID=PNLGRP on class tag.
*JOBCCSID	No conversion occurs for panel group constant text.	Panel group constant text is converted from panel group primary source file CCSID to job CCSID.	Panel group constant text is converted from panel group primary source file CCSID to job CCSID.
	Variables with CHRID=PNLGRP on class tag are converted from XXX to job CCSID.	No conversion occurs for variables with CHRID=PNLGRP on class tag.	No conversion occurs for variables with CHRID=PNLGRP on class tag.
	No conversion occurs for variables without CHRID=PNLGRP on class tag.	No conversion occurs for variables without CHRID=PNLGRP on class tag.	No conversion occurs for variables without CHRID=PNLGRP on class tag.
*DEVD	No conversion occurs for panel group constant text.	Panel group constant text is converted from panel group primary source file CCSID to job CCSID. This conversion occurs because variables are in the job CCSID and the device CHRID is unknown.	No conversion occurs for panel group constant text.
	No conversion occurs for variables with CHRID=PNLGRP on class tag.	No conversion occurs for variables with CHRID=PNLGRP on class tag.	No conversion occurs for variables with CHRID=PNLGRP on class tag.
	No conversion occurs for variables without CHRID=PNLGRP on class tag.	No conversion occurs for variables without CHRID=PNLGRP on class tag.	No conversion occurs for variables without CHRID=PNLGRP on class tag.

### ***CCSID support for messages***

You can use CCSID support for handling messages and message catalogs on the IBM i operating system by using commands and application programming interfaces. You can send messages tagged with one CCSID to users with a different CCSID.

**Note:** You do not need a multinational character set (MNCS) when using CCSIDs for handling messages.

For example, if you do not set CCSID support on, the following message, encoded in CCSID 00037:

```
Joe, I need to see you right away!
```

appears to a user with CCSID 00500 as

```
Joe, I need to see you right away]
```

Instead of seeing an exclamation mark (!), Joe sees a right square bracket (]). If you set CCSID support on, the text in a message encoded in CCSID 00037 is converted to CCSID 00500. Both the person sending the message and the person receiving the message see identical text.

CCSID support helps preserve data integrity in messages. As you read through this information, you will see other advantages to using CCSID support for messages.

### **Message-handling commands for CCSIDs**

The following message handling commands support CCSIDs:

- CRTMSGF (Create Message File)
- CRTMSGQ (Create Message Queue)
- CHGMSGQ (Change Message Queue)
- ADDRPLYE (Add Reply List Entry)
- CHGRPLYE (Change Reply List Entry)
- CHGMSGD (Change Message Description)
- RTVMSG (Retrieve Message)
- RCVMSG (Receive Message)
- SNDBRKMSG (Send Break Message)
- SNDMSG (Send Message)
- SNDPGMMSG (Send Program Message)
- SNDRPY (Send Reply)
- SNDUSRMSG (Send user Message)

### **Related concepts**

[Program message design](#)

A message can be predefined or immediate.

[Display files](#)

When a display file object is created, it is tagged with the coded character set identifier (CCSID) of the source file.

[Using message catalogs](#)

The IBM i operating system can use message catalogs to store messages. Messages in a message catalog are grouped as sets. Each message has a unique number within a set.

### **Related reference**

[Create Message File \(CRTMSGF\) command](#)

[Create Message Queue \(CRTMSGQ\) command](#)

[Change Message Queue \(CHGMSGQ\) command](#)

[Add Reply List Entry \(ADDRPLYE\) command](#)

[Change Reply List Entry \(CHGRPLYE\) command](#)

[Change Message Description \(CHGMSGD\) command](#)

[Retrieve Message \(RTVMSG\) command](#)

[Receive Message \(RCVMSG\) command](#)

[Send Break Message \(SNDBRKMSG\) command](#)

[Send Message \(SNDMSG\) command](#)

Send Program Message (SNDPGMMSG) command

Send Reply (SNDRPY) command

Send User Message (SNDUSRMSG) command

*Handling messages with a specific object-level CCSID*

These listed objects support CCSIDs. Each of them has an object-level CCSID.

- Message files
- Message queues
- Job message queues
- System reply lists
- History log

The object-level CCSID is the CCSID in which all the messages in that object are encoded.

*Object-level coded character set identifier 65535*

CCSID 65535 is the default object-level CCSID for message files and message queues.

If an object has a CCSID of 65535, no conversions occur when adding messages to that object or when receiving messages from that object. Use CCSID 65535 if you do not want CCSID processing to occur.

CCSID 65535 is also known as \*HEX.

### **Object-level coded character set identifier 65534**

CCSID 65534 is the default object-level CCSID for job message queues, system reply lists, and the history log. If the CCSID of an object is 65534, each message in the object has its own CCSID. No conversion occurs when a message is added to the object. When a message is received, it is converted based on the CCSID stored with the message.

CCSID 65534 is also known as \*MSG or \*MSGD.

CCSID 65534 is the preferred setting for object-level CCSIDs. An object-level CCSID of 65534 requires fewer CCSID conversions. Fewer CCSID conversions of text result in better performance and improved data integrity.

### **Related concepts**

Character identifier (QCHRID) system value

The character identifier (QCHRID) system value specifies the character set and code page CHRID(\*SYSVAL) for the CL commands that create, change, or override display files, display device descriptions, user interface (UIM) menus, panel groups, and printer files.

Database files and functions

When you create Unicode database applications, you need to consider the implications for creating physical files, creating logical files, and for database input/output.

Db2 for i

When using Db2 for i applications, you need to be aware of some restrictions of Unicode and some commands.

Recommendations and guidelines for using CCSIDs

These recommendations are useful when you write globalized applications.

*Using a specific object-level CCSID for handling messages*

If the CCSID of an object is any value other than 65535 or 65534, all messages in that object are considered encoded in that CCSID. The object-level CCSID overrides the CCSID stored with the messages.

Use this type of object-level CCSID if both of the following conditions are true:

- You expect the object to be sent messages or have message descriptions added in a CCSID different from the CCSID in which you will receive the messages or retrieve the message descriptions.
- You intend to receive the same message or retrieve the same message description many times.

If these conditions are true, set the object-level CCSID to the CCSID in which you will receive or retrieve the messages. When the system uses this type of object-level CCSID, the message text or data is converted at the time the message is sent or is added to the object. No conversion occurs when the message is received or retrieved because the text and data are already in the CCSID requested on the receive operation or retrieve operation.

Do not change system-supplied message files to use this type of object-level CCSID. Each system-supplied message description is tagged separately. No one object-level CCSID value can represent all of the message descriptions in the message file. Changing the object-level CCSID of a system-supplied message file to anything other than CCSID 65535 or CCSID 65534 might cause unpredictable results.

#### *Message-level support*

When a message is sent to a message queue, you must communicate the CCSID of the replacement data or the immediate message text to the IBM i operating system. Use the CCSID parameter on any of the send message commands or APIs to communicate this CCSID to the system.

The default CCSID setting in the send message commands and APIs indicate that the replacement data or immediate message text is in the CCSID of the job that is running the command or API. You can override the job default CCSID value by specifying a different CCSID value.

If the replacement data or immediate message text supplied is not in the CCSID specified, incorrect conversion results may occur. See [Can I correct the CCSID of a message?](#) if this occurs.

### **Determining the CCSID of a message file**

To determine the CCSID of a message file, follow these steps:

1. Type

```
WRKMSGD MSGF(MYLIB/MYMSGF)
```

where MYLIB is the library in which the message file is stored and MYMSGF is the name of the message file.

2. Press F22 (Display list details).

**Note:** You can also use the Retrieve Message File Attributes (QMHRMFAT) application programming interface (API) to determine the CCSID of a message file.

For job message queues, system reply lists, and the history log, the object-level CCSID is always 65534. You cannot change nor display object-level CCSIDs for job message queues, system reply lists, and the history log.

#### **Related concepts**

[Can I correct the CCSID of a message?](#)

You cannot correct the message-level CCSID of a message. You can change the message queue CCSID to match the message-level CCSID. You can also delete the message and send it again with the correct message-level CCSID.

#### *Message-level CCSID with a message queue CCSID of 65535 or 65534*

When a message is sent to the message queue and the CCSID of the message queue is 65535 or 65534, no conversion occurs on the message. The message-level CCSID is set to the CCSID specified.

For example, message queue MYMSGQ has a CCSID of 65534. You enter the following Send Message command:

```
SNDMSG MSG('MSG #1') CCSID(37) TOMSGQ(MYLIB/MYMSGQ)
```

The immediate message text, MSG #1, is not converted when added to the message queue. The message is tagged with CCSID 00037.



### *Message-level CCSID with a specific message queue CCSID*

When a message is sent to the message queue and the CCSID of the message queue is something other than 65535 or 65534, the replacement data or immediate message text is converted to the CCSID of the message queue. The message is then tagged with the CCSID of the message queue.

For example, message queue MYMSGQ has a CCSID of 00277. The replacement data for TST0002 is defined as \*CCHAR data. You enter the following Send Program Message command:

```
SNDPGMMSG MSGDTA(X'0006D4E2C7407BF2') MSGID(TST0002) MSGF(MYMSGF)
CCSID(37) TOMSGQ(MYLIB/MYMSGQ)
```

The replacement data is converted from CCSID 00037 to CCSID 00277 before it is sent to the message queue. X'0006' is the length required for variable-length fields. X'D4E2C7407BF2' is MSG #2 on code page 00037. The number sign (#), X'7B' on code page 00037, is converted to a number sign, X'4A' on code page 00277. All other code points do not change during the conversion because they are the same on both code page 00037 and code page 00277.

When the replacement data or immediate message text of a message is 65535 and it is sent to a message queue with a CCSID other than 65535 or 65534, no conversion occurs. However, the message is tagged with the CCSID of the message queue. Therefore, messages can be tagged with an incorrect CCSID when you send them to a message queue with a CCSID that overrides the message-level CCSID.

For example, message queue MYMSGQ has a CCSID of 00277. You enter the following Send Message command:

```
SNDMSG MSG('MSG #2') TOMSGQ(MYLIB/MYMSGQ) CCSID(*HEX)
```

The immediate message text MSG #2 is not converted before it is sent to the message queue. Although the immediate message text is not converted to CCSID 00277, it is displayed using CCSID 00277. Unless you entered the Send Message command from a device configured to support code page 00277, you lost the integrity of the immediate message text.

### *Message-level CCSID when a message queue CCSID conversion error occurs*

If a conversion error occurs while sending a message to a message queue, the message is still sent to the message queue. However, the immediate text or data of the message is not converted.

A diagnostic message is sent and the message is tagged with the message-level CCSID specified on the send command or API, not with the CCSID of the message queue.

You can recover the replacement data or immediate message text with the proper CCSID setting. First, set the message queue CCSID to 65534. Then use the Receive Message command or API to return the correct message-level CCSID.

### *Message-level CCSID when a message is a stored message*

If a message is a stored message, the message-level CCSID applies only to \*CCHAR replacement data. The CCSID of the first- and second-level text of the message is retrieved from the message file.

Replies to stored messages are never converted from one CCSID to another. Only replies to immediate messages are affected by CCSID processing.

### *Message description-level support*

When a message description is added to a message file, the CCSID of the message text must be communicated to the IBM i operating system.

You can use the CCSID parameter on the Add Message Description (ADDMSGD) or the Change Message Description (CHGMSGD) command to communicate this CCSID to the operating system.

The default settings of these commands indicate that the message text is in the CCSID of the job that is running the command. You can change this value by specifying a different CCSID value. You can also change this value by indicating that no CCSID processing should occur. You indicate that no CCSID processing should occur on the message text by specifying a CCSID value of 65535 (\*HEX).

If you set CCSID processing on, system-supplied display files and printer files that display or print message descriptions convert the CCSID of the message file to the CCSID of the job before displaying them or printing them. To print and display the messages correctly, your job CCSID setting must be the same as the code page portion of your device CHRID setting.

All message descriptions that existed in a message file that was created before V3R1 are tagged with CCSID 65535 on the first use or handling of that message description.

If the text of a message is not in the CCSID specified, incorrect conversion results might occur. See [Can I correct the CCSID of a message description?](#) if this occurs.

### **Related concepts**

[Can I correct the CCSID of a message?](#)

You cannot correct the message-level CCSID of a message. You can change the message queue CCSID to match the message-level CCSID. You can also delete the message and send it again with the correct message-level CCSID.

### **Related reference**

[Add Message Description \(ADDMSGD\) command](#)

[Change Message Description \(CHGMSGD\) command](#)

*Message file with a CCSID of 65535 or 65534*

If the CCSID of the message file is 65535 or 65534, no conversion occurs on the message description when it is added to the file. The message description CCSID is set to the CCSID specified on the ADDMSGD or CHGMSGD command.

For example, a message file MYMSGF has a CCSID of 65534. The job that is running is in CCSID 00037. You enter an ADDMSGD command, as follows:

```
ADDMSGD MSG('MSG #1') MSGID(TST0001) MSGF(MYMSGF)
```

The message text, MSG #1, is not converted when added to the message file. The message text is tagged 00037 because the CCSID parameter was not coded on the ADDMSGD command and the default CCSID parameter is \*JOB.

### **Related reference**

[Add Message Description \(ADDMSGD\) command](#)

[Change Message Description \(CHGMSGD\) command](#)

*Message file with a specific CCSID*

If the CCSID of the message file is something other than 65535 or 65534, the first- and second-level text of the message description is converted from the CCSID specified to the CCSID of the message file. It is then tagged with the CCSID of the message file.

For example, message file MYMSGF has a CCSID of 00277. The job that is running is in CCSID 00037. You enter the following command:

```
ADDMSGD MSG('MSG #2') MSGID(TST0002) MSGF(MYMSGF) CCSID(37)
```

Message 'MSG #2' is converted from CCSID 00037 to CCSID 00277 before it is added to the message file. The number sign (#), X'7B' on code page 00037, is converted to the number sign (#), X'4A', on code page 00277. No other code points change during the conversion because they are the same on both code page 00037 and code page 00277.

When the text of a message description is specified as 65535 and it is added to a message file, no conversion occurs. If the CCSID of the message file is not 65535 or 65534, the message text is tagged with the CCSID of the message file.

When the message file CCSID is not 65535 or 65534, the message file CCSID overrides message description CCSIDs. Keep this rule in mind when adding and changing message descriptions to a message file with a CCSID other than 65535 or 65534. Otherwise, a message description can be marked incorrectly.

For example, message file MYMSGF has a CCSID of 00277. You enter the following command:

```
ADDMSGD MSG('MSG #2') MSGID(TST0002) MSGF(MYMSGF) CCSID(*HEX)
```

Message text 'MSG #2' is not converted before it is added to the message file. Because the CCSID of the message file is 00277, the message text is tagged with CCSID 00277.

If the command was run in a job CCSID where the number sign (#) occupies a code point different than the code point for the number signon code page 00277, the message is displayed incorrectly.

A conversion error may occur while adding or changing a message description in a message file. If a conversion error occurs, the message description is still either added to or changed in the message file. The text of the message description, however, is not converted. A diagnostic message is sent and the message description is tagged with the CCSID specified, not with the CCSID of the message file.

When a conversion error occurs, you can recover the correct CCSID tagging for the message description by setting the message file CCSID to 65534. Then you can retrieve the correct CCSID for the message description using the Retrieve Message (RTVMSG) command or the Retrieve Message (QMHRTVM) API.

The CCSID of a message description applies only to first- and second-level message text.

### **Related reference**

[Retrieve Message \(RTVMSG\) command](#)

[Retrieve Message \(QMHRTVM\) API](#)

#### *Changing the CCSID of a message description*

When you take the option to change a message description from the Work with Message Descriptions display, all current values for the selected message description are retrieved and placed on the prompt display.

The first- and second-level text are converted from the CCSID of the message file to the CCSID of the job before they are put on the prompt display.

\*JOB is displayed for the CCSID keyword and has two different meanings depending on what you do on the prompt display. If you change any part of the first- or second-level text, \*JOB means that the text is converted from the CCSID of the job to the CCSID of the message file when you press the Enter key. If the text is unchanged, \*JOB works like \*SAME, and none of the following texts are changed:

- The first-level message text
- The second-level message text
- The CCSID of the message description

Both the first- and second-level text of a message description must be in the same CCSID. If you change the CCSID of one level, the system automatically converts the other level to match.

### **Example: Changing a message description**

The CCSID of message file MYMSGF is 65534. The CCSID of the job that is running WRKMSGD is 00277. The CCSID of the message description is 00037.

Select option 2 to change a message description. The text of the message description is converted from CCSID 00037 to 00277 before being placed on the prompt display.

If only the first-level text is changed, the 00277-tagged text is stored in the message file. The CCSID of the message description is changed to 00277. The 00277-tagged second-level text is also stored in the message file to keep both the first- and second-level text in the same CCSID.

### *Message queues*

If you set CCSID processing on, system-supplied display files and printer files that display or print messages convert the CCSID of the message queue to the CCSID of the job before displaying or printing the messages.

To print and display the messages correctly, your job CCSID setting must be the same as the code page portion of your device CHRID setting.

All messages that existed on a message queue that was created in a release before V3R1 of the operating system are assigned CCSID 65535 on the first use of that message.

### **Determining the CCSID of a message queue**

To determine the CCSID of a message queue, follow these steps:

#### 1. Type

```
DSPMSG MSGQ(MYLIB/MYMSGQ) ASTLVL(*BASIC)
```

where MYLIB is the library in which the message queue is stored and MYMSGQ is the name of the message queue.

#### 2. Press F22 (Display list details).

**Note:** You can also use the Retrieve Message Queue Attributes (QMHRMQAT) application programming interface (API) to determine the CCSID of a message queue.

For job message queues, system reply lists, and the history log, the object-level CCSID is always 65534. You cannot change nor display object-level CCSIDs for job message queues, system reply lists, and the history log.

### **Related concepts**

#### Job message queues

The CCSID for all job message queues is 65534. You cannot change or display this value. A job message queue CCSID of 65534 requires fewer CCSID conversions. Fewer CCSID conversions of text result in better performance and improved data integrity.

#### System reply list

The system reply list has a CCSID of 65534. You cannot change or display this value.

### **Related tasks**

#### History log

The history log is a database file that is tagged with CCSID 65535. You cannot change the CCSID of the history log. No conversions occur when you do database retrievals from the history file.

### **Related reference**

Retrieve Nonprogram Message Queue Attributes (QMHRMQAT) API

#### *Job message queues*

The CCSID for all job message queues is 65534. You cannot change or display this value. A job message queue CCSID of 65534 requires fewer CCSID conversions. Fewer CCSID conversions of text result in better performance and improved data integrity.

The CCSID of each message in the job log is used for CCSID processing. No conversion occurs when a message is sent to the job log.

**Note:** Request messages are always tagged with a CCSID of 65535 and are never converted.

If you set CCSID processing on, system-supplied display files and printer files that display or print job logs convert the CCSID of the messages to the CCSID of the job before displaying or printing the messages. To print and display the messages correctly, your job CCSID setting must be the same as the code page portion of your device CHRID setting. Status messages that appear on line 24 of a display are converted to the CCSID of the device before they are shown.

## Related concepts

### Message queues

If you set CCSID processing on, system-supplied display files and printer files that display or print messages convert the CCSID of the message queue to the CCSID of the job before displaying or printing the messages.

## Related tasks

### History log

The history log is a database file that is tagged with CCSID 65535. You cannot change the CCSID of the history log. No conversions occur when you do database retrievals from the history file.

### *System reply list*

The system reply list has a CCSID of 65534. You cannot change or display this value.

The only part of the system reply list that is affected by CCSID processing is the Compare data field. If the Compare data field references replacement data that is defined as \*CCHAR, the data being compared must be in a common CCSID before the comparison is done.

Any reply list entry that has compare data is tagged with the CCSID supplied on the ADDRPLYE or CHGRPLYE commands. When the system reply list is used, the replacement data is converted to the CCSID of the compare data before the comparison is made and before the message is sent to the message queue. This ensures that the data is in a common CCSID before the comparison is done.

## Example: System reply list and converted-character compare data

Enter the following Add Reply List Entry command:

```
ADDRPLYE SEQNBR(101) MSGID(TST0010) CMPDTA(X'00017B') RPY(*DFT) +  
CCSID(37)
```

X'7B' is the number sign (#) on code page 00037. X'0001' is the length required for variable-length fields. The compare data is not converted when added to the system reply list. It is tagged with CCSID 00037. Message TST0010 has one replacement data field that is defined as \*CCHAR with (\*VARY 2) for its length. Message queue MYMSGQ has a CCSID of 00278.

Send message TST0010 in a job that has the system reply list turned on using the following Send Program Message command:

```
SNDPGMMSG MSGID(TST0010) MSGF(MYLIB/MYMSGF) MSGTYPE(*INQ) +  
TOMSGQ(MYLIB/MYMSGQ) MSGDTA(X'00014A') CCSID(277)
```

The replacement data is converted from CCSID 00277 to CCSID 00037 and then compared with the compare data. The conversion results in replacement data X'00017B'. A match is found and the default reply is sent when this message is added to the message queue.

When the message is added to the message queue, the replacement data is converted from CCSID 00277 to CCSID 00278. The message queue CCSID does not matter when trying to match the compare data. The replacement data is converted to X'000163' when it is sent to the message queue and tagged 00278. X'63' is the code point for the number sign (#) in code page 00278.

## Related concepts

### Message queues

If you set CCSID processing on, system-supplied display files and printer files that display or print messages convert the CCSID of the message queue to the CCSID of the job before displaying or printing the messages.

## Related reference

Add Reply List Entry (ADDRPLYE) command

Change Reply List Entry (CHGRPLYE) command

### *History log*

The history log is a database file that is tagged with CCSID 65535. You cannot change the CCSID of the history log. No conversions occur when you do database retrievals from the history file.

You can use CCSID processing when working with the history log. The CCSID of the replacement data or immediate message text is added to the history log record. If the history log record is for a stored message, CCSID processing occurs only for the \*CCHAR replacement data in that record.

You can retrieve a message from the history log and convert it into a specific CCSID by following these steps:

1. Obtain the input variables &MSGFL, &MSGF, &MSGID, &MSGDTA, and &MDTACCSID, from the history log record. (See the CL Programming PDF for the layout of the history log record.)
2. Enter the following Retrieve Message command:

```
RTVMSG MSGF(&MSGFL/&MSGF); MSGID(&MSGID); MSGDTA(&MSGDTA); +  
MDTACCSID(&MDTACCSID); MSG(&MSG);
```

If you set CCSID processing on, system-supplied display files and printer files that display or print history log records convert the CCSID of the messages to the CCSID of the job before displaying or printing the messages. To print and display the messages correctly, your job CCSID setting must be the same as the code page portion of your device CHRID setting.

### **Related concepts**

#### Message queues

If you set CCSID processing on, system-supplied display files and printer files that display or print messages convert the CCSID of the message queue to the CCSID of the job before displaying or printing the messages.

#### Job message queues

The CCSID for all job message queues is 65534. You cannot change or display this value. A job message queue CCSID of 65534 requires fewer CCSID conversions. Fewer CCSID conversions of text result in better performance and improved data integrity.

### **Related information**

#### CL Programming

#### *Setting up CCSID support for message handling*

The default setting of the CCSID for creating message queues and message files is 65535. Most message files delivered with the operating system have a CCSID of 65535.

Most message descriptions in system-supplied message files are tagged with a CCSID that corresponds to the national language version with which they are included.

Some message descriptions are not assigned a CCSID that corresponds to the national language version. These message descriptions are tagged 65535 and are not converted when used.

Messages sent to a message queue that has a CCSID of 65535 are not converted when placed on the queue. Message descriptions added to a message file that has a CCSID of 65535 are not converted when placed in the file. These messages and message descriptions are tagged with a CCSID associated with their text or data. By tagging them with a CCSID associated with their text or data, they are given the correct CCSID if the object-level CCSID is changed to 65534.

You can set CCSID support on for handling a specific message queue. For example, to set CCSID handling on for message queue MYMSGQ in library MYLIB, type:

```
CHGMSGQ MSGQ(MYLIB/MYMSGQ) CCSID(65534)
```

The Change Message Queue (CHGMSGQ) command also allows you to turn on CCSID support for more than one message queue at a time.

You can set CCSID support on for handling a specific message file. For example, to set CCSID handling on for message file MYMSGF in library MYLIB, type:

```
CHGMSGF MSGF(MYLIB/MYMSGF) CCSID(65534)
```

The Change Message File (CHGMSGF) command also allows you to turn on CCSID support for more than one message file at a time.

### **Related reference**

[Change Message Queue \(CHGMSGQ\) command](#)

[Change Message File \(CHGMSGF\) command](#)

#### *CCSID support for message catalogs*

The Message catalog CCSID (CLGCCSID) parameter allows you to specify the CCSID for storing data in a message catalog.

The Source file CCSID (SRCCCSID) parameter allows you to specify the CCSID of a source file. Data from the source is converted to the CCSID of the message catalog if the CCSIDs for both are not the same. This is also the default action. The source can be in any CCSID that supports conversion to any other CCSID.

The CCSID of the original message catalog is used to update the message catalog. It can be single or mixed and in extended binary-coded decimal interchange code (EBCDIC), American National Standard Code for Information Interchange (ASCII), or UCS-2. If the catalog is a QSYS source file member that does not exist, the CCSID of the existing file is used. The value that is specified on the CLGCCSID parameter is used if the CCSID of the file is 65535.

#### *Converted character replacement data type field*

A replacement data type field supports CCSID processing. This replacement data type field is called a convertible character field (\*CCHAR). A \*CCHAR replacement data type field is a variable-length field, and therefore the field might increase or decrease in length when it is converted.

### **Example: Adding a message description with CCSID support**

The following example shows how to add the message description TST0006 to message file MYMSGF. The message description has 2 replacement data type fields. One field is a character field length 10. The other field is a convertible character field with varying length. Use the ADDMSGD command as follows:

```
ADDMSGD MSG('This is *CHAR &1; This is *CCHAR &2;') MSGID(TST0006) +  
MSGF(MYLIB/MYMSGF) FMT((*CHAR 10) (*CCHAR *VARY 2))
```

### **Related reference**

[Add Message Description \(ADDMSGD\) command](#)

#### *Retrieving messages*

The Retrieve Message (RTVMSG) command and retrieve message (QMHRTVM) application programming interface (API) have a CCSID-to-convert-to parameter. This parameter determines which CCSID the first- and second-level text is converted to before the text is returned to the user.

The Retrieve Message command and the Retrieve Message API also have a replacement data CCSID parameter. This parameter communicates the CCSID of the replacement data to the system. The replacement data CCSID applies only to the parts of the replacement data that correspond to \*CCHAR type data. No other replacement data is converted.

The Retrieve Message command and Retrieve Message API convert the first- and second-level text from the CCSID of the message file to the CCSID on the CCSID-to convert-to parameter. Any replacement data that is \*CCHAR data is converted from the replacement data CCSID to the CCSID-to-convert-to CCSID before being substituted into the correct replacement variables. The default for both parameters is \*JOB, which means that the CCSID of the job is used.

## Retrieve Message command CCSID return fields

Three CCSID return fields are supported by the Retrieve Message (RTVMSG) command:

- TXTCCSID
- TXTCCSTA
- MDTACCSTA

### Example 1: Retrieving a message with CCSID support

Message file MYMSGF has a CCSID of 65534. The CCSID of the message description is used to determine the CCSID from which to convert the message text. The CCSID of the message description (TST0003) is 00037. The first-level text is:

```
'MSG #3 is &1;'
```

&1 is defined as a \*CCHAR variable field with a length of (\*VARY 2). Enter the following RTVMSG (Retrieve Message) command:

```
RTVMSG MSGF(MYMSGF) MSGID(TST0003) MSG(&MSG); CCSID(277) +  
MDTACCSID(277) MSGDATA(X'0002D6D2')
```

In the message data, the first 2 bytes are a length field with a value of 2. All \*VARY fields begin with a length. The next 2 characters are the actual \*CCHAR data with a value of X'D6D2'. X'D6D2' represents the characters O and K on code page 00277.

The first-level text is converted from CCSID 00037 to CCSID 00277. The replacement data is not converted before it is substituted for &1; because the replacement data CCSID matches the CCSID-to-convert-to parameter. As a result, the text returned in the variable &MSG is:

```
'MSG #3 is OK.'
```

The code point for the number sign (#) is the only change that occurred in the conversion. The number sign was converted from code point X'7B' in code page 00037 to code point X'4A' in code page 00277. All other code points in the text of the message matched in code page 00037 and code page 00277.

**Note:** If the CCSID of a message file is 65535, no conversion occurs, even though the message description CCSID is 00037. The CCSID of the message file always takes precedence over the message description CCSID.

### Example 2: Using return fields and converted character data

Message description TST0005 has the following first-level text:

```
This is *CHAR &1; This is *CCHAR &2;
```

The message description is defined in message file MYMSGF, which has a CCSID of 65535. &1; is defined as a \*CHAR field of length 1. &2; is defined as a \*CCHAR field (\*VARY 2) in length. The CCSID of the message description does not matter because the CCSID of the message file is not 65534. You enter the following RTVMSG command:

```
RTVMSG MSGF(MYMSGF) MSGID(TST0005) MSG(&MSG); CCSID(260) +  
MDTACCSID(37) MSGDATA(X'5A00015A') TXTCCSID(&TXTCCSID);
```

**Note:** X'5A' is the exclamation point (!) on code page 00037.

These are the returned values from the RTVMSG command:

- &MSG = 'This is \*CHAR. This is \*CCHAR !.'



The EBCDIC value of the \*CHAR character is X'5A'. X'5A' appears as an acute accent ( ´ ) on code page 00260. The \*CHAR data did not convert because only \*CCHAR data supports CCSID processing. The '&1' stayed at X'5A', while '&2' converted to X'4F'. X'4F' is the exclamation point on code page 00260.

- &TXTCCSID = 65535

The TXTCCSID variable is set to 65535 because no conversion occurred. When no conversion occurs, the CCSID (if it is not 65534) of the message file is returned.

### Related reference

[Retrieve Message \(RTVMSG\) command](#)

[Retrieve Message \(QMHRVTM\) API](#)

*CCSID of the text returned (TXTCCSID) return field*

TXTCCSID is the CCSID of the text returned.

If a conversion occurs and is successful, this value is always equal to the CCSID-to-convert-to value. If a conversion occurs and is not successful, this is the CCSID of the message file unless the CCSID of the message file is 65534. If the CCSID of the message file is 65534, the CCSID of the message description is returned.

For example, message file MYMSGF has a CCSID of 65534. Your program needs to know the CCSID of message description TST0004. Specify the RTVMSG command as follows:

```
RTVMSG MSGF(MYMSGF) MSGID(TST0004) CCSID(*HEX)
TXTCCSID(&TXTCCSID);
```

The CCSID of the message description is returned in the variable &TXTCCSID because you specified \*HEX for the CCSID-to-convert-to parameter. \*HEX means no conversion is to occur. If no conversion occurs and the message file CCSID is 65534, the message description CCSID is returned.

You can also obtain the message description CCSID from the Work with Message Descriptions (WRKMSGD) display.

1. On the WRKMSGD display, select option 5 to display details.
2. From the Select Message Details to Display menu, select option 5 to display message attributes.
3. Page forward to the CCSID value.

The message description CCSID is shown if the CCSID of the message file is 65534. If the CCSID of the message file is not 65534, the CCSID of the message file is shown.

### Related reference

[Work with Message Descriptions \(WRKMSGD\) command](#)

*CCSID conversion status indicator (TXTCCSTA) return field*

TXTCCSTA is the text CCSID conversion status indicator. Return codes help you determine what happened when the system converted your message text to the CCSID-to-convert-to parameter.

Positive return code numbers indicate that your conversion was successful. A successful return code does not always indicate that a conversion occurred. Negative return code numbers indicate that a conversion error occurred.

The following list shows the available return codes:

**0**

No conversion was necessary. The CCSID of the text matched the CCSID that you wanted the text converted to.

**1**

No conversion occurred. Either the text was 65535 or the CCSID that you wanted the text converted to was 65535.

**2**

No conversion occurred. You did not ask for any text to be returned.

**3**

The text was converted to the CCSID specified. The conversion operation used the linguistic conversion tables.

**4**

A conversion error occurred when the conversion operation used the linguistic conversion tables. The conversion operation then used a default conversion table. The default conversion completed without error.

**-1**

An error occurred on both the linguistic and default conversions. The text was not converted.

### **Related concepts**

Replacement data CCSID conversion status indicator (MDTACCSTA) return field

MDTACCSTA is the replacement data CCSID conversion status indicator. Return codes help you determine what happened when the system converted your replacement data to the CCSID-to-convert-to parameter.

*Replacement data CCSID conversion status indicator (MDTACCSTA) return field*

MDTACCSTA is the replacement data CCSID conversion status indicator. Return codes help you determine what happened when the system converted your replacement data to the CCSID-to-convert-to parameter.

Positive return code numbers indicate that your conversion was successful. A successful return code does not always indicate that a conversion occurred. Negative return code numbers indicate that a conversion error occurred. These return codes are similar to the TXTCCSTA return codes. The return codes apply to the conversion that takes place on any \*CCHAR replacement data being converted from the replacement data CCSID to the CCSID-to-convert-to value.

The following list shows the available return codes:

**0**

No conversion was necessary. The CCSID of the replacement data matched the CCSID that you wanted the text converted to.

**1**

No conversion occurred. Either the replacement data was 65535 or the CCSID that you wanted the replacement data converted to was 65535.

**2**

No conversion occurred. Either you did not ask for any replacement data to be returned or no \*CCHAR replacement data fields were defined for the message description being retrieved.

**3**

The replacement data was converted to the CCSID specified. The conversion operation used the linguistic conversion tables.

**4**

A conversion error occurred when the conversion operation used the linguistic conversion tables. The conversion operation then used a default conversion table. The default conversion completed without error.

**-1**

An error occurred on both the linguistic and default conversions. The replacement data was not converted.

### **Related concepts**

CCSID conversion status indicator (TXTCCSTA) return field

TXTCSTA is the text CCSID conversion status indicator. Return codes help you determine what happened when the system converted your message text to the CCSID-to-convert-to parameter.

### *Receiving messages*

The Receive Message (RCVMSG) command, the Receive Nonprogram Message (QMHRVCM) API, and the Receive Program Message (QMHRVPM) API have a CCSID-to-convert-to parameter. This parameter determines which CCSID the text or data is converted to before it is returned to the user.

The Receive Message command and APIs convert the text or data from the CCSID of the message queue or message file to the CCSID supplied on the CCSID-to-convert-to parameter. When replacement data is returned, only the \*CCHAR data is converted from the CCSID of the message queue to the CCSID-to-convert-to value.

If the CCSID of the message file or message queue is 65534, the text or data is converted from the CCSID of the message description or message to the CCSID supplied on the CCSID-to-convert-to parameter.

The default for the CCSID-to-convert-to parameter is \*JOB, which means that the CCSID of the job performing the receive operation is used.

### **Receive Message command CCSID return fields**

Two CCSID return fields are supported by the Receive Message (RCVMSG) command:

- TXTCCSID
- DTACCSID

### **Receive Message API CCSID return fields**

The Receive Message (QMHRVCM) API and the Receive Program Message (QMHRVPM) API support the return fields defined in TXTCCSID return field for receive message command and DTACCSID return field for receive message command. The Receive Message API and the Receive Program Message API also support two additional return fields.

### **Example 1: Using the CCSID return fields**

Message description TST0005 has the following first-level text:

```
This is &CHAR &1; This is *CCHAR &2;
```

'&1' is defined as a \*CHAR field of length 1. '&2' is defined as a \*CCHAR field (\*VARY 2) in length.

Message file MYMSGF has a CCSID of 65534. TST0005 is defined in message file MYMSGF. The message description CCSID is 65535. The CCSID of message queue MYMSGQ is 65534.

You enter the following Send Program Message command:

```
SNDPGMMSG MSGF(MYMSGF) MSGID(TST0005) CCSID(37) TOMSGQ(MYLIB/MYMSGQ) +  
MSGDTA(X'7B00017B')
```

The message is not converted when it is sent to message queue MYMSGQ because the message queue CCSID is 65534. The message is tagged with CCSID 00037.

You enter the following Receive Message command to receive the message just sent:

```
RCVMSG MSGQ(MYLIB/MYMSGQ) MSG(&MSG); DTACCSID(&DTACCSID); +  
CCSID(277) MSGDTA(&MSGDTA); TXTCCSID(&TXTCCSID);
```

**Note:** X'7B' is the number sign (#) on code page 00037.

Because the message description is tagged 65535, no conversion is performed when retrieving the message text of TST0005. The replacement data is tagged 00037. The \*CCHAR part of the message data is converted from CCSID 00037 to CCSID 00277 before being inserted for &2; \*CHAR data is never converted.

The following table shows the returned values after the Receive Message command runs:

Value	Description
&MSG =	<p>This is &amp;CHAR <b>Æ</b>. This is *CCHAR #.</p> <p>The *CHAR data was not converted when substituted for &amp;1; The *CHAR data remains X'7B'. X'7B' is the code point on code page 00277 for A ligature (<b>Æ</b>).</p> <p>The *CCHAR data was converted to X'4A' before it was substituted for &amp;2; X'4A' is the code point on code page 00277 for the number sign (#).</p>
&TXTCCSID = 65535	The &TXTCCSID variable was set to 65535 because no conversion occurred. When no conversion occurs, the CCSID of the message description is returned if the CCSID of the message file is 65534.
&DTACCSID = 00277	The &DTACCSID variable was set to 00277 because a conversion occurred.

### Example 2: Receiving a message with CCSID support

Message file MYMSGF has a CCSID of 00037. Message queue MYMSGQ has a CCSID of 65534. The message being received has a message-level CCSID of 00277. CCSID 65534 uses the message-level CCSID when determining the CCSID the replacement data is to be converted from.

The message being received is a stored message. The stored message has \*CCHAR replacement data. The CCSID of the job is 00278. You enter the following Receive Message command:

```
RCVMSG MSGQ(MYMSGQ) MSG(&MSG); MSGDTA(&MSGDTA);
```

The first-level text of the stored message that you receive is converted from CCSID 00037 to CCSID 00278. The replacement data of the message that you receive is converted from CCSID 00277 to CCSID 00278. Then the replacement data is substituted into the first-level text and returned in &MSG.

Both the first-level text and the replacement data of the message that you received are converted to the CCSID of the job because the CCSID of the job is the default for the CCSID-to-convert-to parameter.

Two different conversions must occur because only the replacement data is stored in the message queue for stored messages. The text of a stored message must be retrieved from the message file. If the message contained other replacement data type fields that were not defined as \*CCHAR, the non-\*CCHAR data is not converted before being returned.

**Note:** If the CCSID of the message queue is 00278, no conversion occurs on the replacement data before the message is returned, even though the message CCSID is 00277. Remember that the message queue CCSID takes precedence over the message-level CCSID.

### Related reference

[Receive Message \(RCVMSG\) command](#)

[Receive Nonprogram Message \(QMHRVPM\) API](#)

[Receive Program Message \(QMHRVPM\) API](#)

#### *CCSID of the message text returned (TXTCCSID) return field*

TXTCCSID is the CCSID of the message text returned. If a conversion occurs and the conversion is successful, this value is always the same as the CCSID-to-convert-to value.

For immediate text, if the conversion is not successful, TXTCCSID is the CCSID of the message queue, unless the message queue is 65534. If the message queue is 65534, TXTCCSID is the message-level CCSID of the immediate text.

For a stored message, if the conversion is not successful, TXTCCSID is the CCSID of the message file that contains the stored message, unless the message file is 65534. If the CCSID of the message file is 65534, TXTCCSID is the CCSID of the message description for the stored message.

#### *CCSID of the replacement data returned (DTACCSID) return field*

DTACCSID is the CCSID of the replacement data returned. DTACCSID applies only to those parts of the replacement data defined as \*CCHAR.

If the message being received is an immediate message, a value of 0 is returned. If a conversion occurs and the conversion is successful, this value is the same as the CCSID-to-convert-to value.

If the conversion is not successful, the DTACCSID returned is the CCSID of the message queue, unless the CCSID of the message queue is 65534. If the CCSID of the message queue is 65534, the DTACCSID returned is the CCSID of the message.

For example, a stored message TST0004 from message file MYMSGF is sent to message queue YOURMSGQ with replacement data. TST0004 is defined with \*CCHAR replacement data. Message file MYMSGF is 65534. Message queue YOURMSGQ has a CCSID of 00037.

Your program needs to know the CCSID of the message description and the replacement data sent to message queue YOURMSGQ. You enter the following Receive Message command:

```
RCVMSG MSGQ(YOURMSGQ) CCSID(*HEX) TXTCCSID(&TXTCCSID); DTACCSID(&DTACCSID);
```

The message description CCSID is returned in the variable &TXTCCSID. The message description CCSID is returned because you specified \*HEX for the CCSID-to-convert-to parameter. \*HEX means that no conversion is to occur. If no conversion occurs and the message file CCSID tag is 65534, the CCSID of the message description is returned.

The CCSID of message queue YOURMSGQ (00037) is returned in the variable &DTACCSID. The message queue CCSID is returned because it is not 65534.

You can also obtain the message-level CCSID using the Display Messages (DSPMSG) display.

1. From the Display Messages display, press Help to display the Additional Message Information display.
2. Press F9 (Display Message Details).

This displays the message-level CCSID when the CCSID of the message queue that this message is on is 65534. Otherwise, the CCSID of the message queue is displayed.

#### *Common questions about CCSID support for handling messages*

Here are some common questions asked about CCSID support for handling messages.

##### *When is the job default CCSID used for handling messages?*

A job default CCSID is always a CCSID with an encoding scheme of 1100 (single-byte EBCDIC) or 1301 (mixed-byte EBCDIC).

The job default CCSID is used whenever both of the following conditions are true:

- A conversion occurs from a CCSID with an encoding scheme other than 1100 or 1301 to a job CCSID.
- The job CCSID is 65535.

For example, ASCII data must be converted to a CCSID associated with the data when asked to convert to the CCSID of a job. The job default CCSID is used because it is never CCSID 65535.

*How can I determine if a message description is defined with \*CCHAR?*

You can use the Work with Message Description (WRKMSGD) command to determine if a message description is defined with \*CCHAR data. You can also use the Retrieve Message (QMHRVTM) API to return the replacement data format fields.

**Related reference**

[Work With Message Description \(WRKMSGD\) command](#)

[Application programming interfaces \(APIs\)](#)

*Can the length of \*CCHAR replacement data change?*

The length of \*CCHAR replacement data can change. This is why \*CCHAR replacement data is required to be a variable-length field.

The length of the field grows when converting from an SBCS CCSID to the UCS-2 Level-1 CCSID. The length of the field shrinks when converting from the UCS-2 Level-1 CCSID to an SBCS CCSID.

For example, you define message description TST0011 as 'Printer &1; has error &2;' in message file MYMSGF that has a CCSID of 65535. '&1' is defined as \*CCHAR data (\*VARY 2) in length. This is the name of the printer. &2; is defined as a \*CHAR data with a length of 1. This is an error code. Enter the following Send Program Message command to send this message to message queue MYMSGQ:

```
SNDPGMMSG MSGID(TST0011) MSGF(MYLIB/MYMSGF) TOMSGQ(MYLIB/MYMSGQ) +  
MSGDTA(X'000400500030F1') CCSID(61952)
```

X'0004' is the length of the variable \*CCHAR data. X'00500030' represents the characters P0 in CCSID 61952. If message queue MYMSGQ has a CCSID of 00037, the replacement data is converted to X'0002D7F0F1' before it is sent to the message queue. If message queue MYMSGQ has a CCSID of 65535, the data is not converted when it is sent to the message queue.

Your application programs cannot hard-code the position of the return code in this example. When message queue MYMSGQ has a CCSID of 00037, the return code is 5 bytes into the message text. When message queue MYMSGQ has a CCSID of 65535, the return code is 7 bytes into the message text.

*Can I correct the CCSID of a message queue?*

You might have a message queue that has a CCSID that does not match the CCSID of the messages on it. This typically results from sending messages with a message-level CCSID of 65535 to a message queue with a CCSID that is not 65534 or 65535.

If all of the messages on a message queue have the same message-level CCSID and you know the message-level CCSID, you can enter the following command:

```
CHGMSGQ MSGQ(MYMSGQ) CCSID(nnnnn)
```

In this example, MYMSGQ is the name of the message queue and nnnnn is the message-level CCSID.

If you do not know the CCSID of all the messages on the queue or if the messages on the queue have different CCSIDs, the message queue should have a CCSID of 65535 or 65534. You can change the message queue CCSID to 65535. Or, you can follow these steps:

1. Delete all of the messages.
2. Change the CCSID of the message queue to 65534.
3. Send all of the messages again.

*Can I correct the CCSID of a message file?*

You might have a message file that has a CCSID that does not match the CCSID of the message descriptions in it. This typically results from adding message descriptions with a message-level CCSID of 65535 to a message file with a CCSID that is not 65534 or 65535.

If all of the message descriptions in a message file have the same message-level CCSID, and you know the message-level CCSID, you can enter the following command:

```
CHGMSGF MSGF(MYMSGF) CCSID(nnnnn)
```

In this example, MYMSGF is the name of the message file and nnnnn is the message-level CCSID.

If you do not know the CCSID of all the message descriptions in the file or if the message descriptions in the file have different CCSIDs, the message file should have a CCSID of 65535 or 65534. You can handle this situation in either of the following ways:

- Change the CCSID of the message file to 65535.
- Follow these steps:
  1. Change the CCSID of the message file to 65534.
  2. Change the message-level CCSID of each message description to the correct value.

### **Related concepts**

Can I correct the CCSID of a message description?

You can use the Change Message Description (CHGMSGD) command to change the CCSID of a message description. If you do not change the first- or second-level text at the same time that you change the message description CCSID, the text remains unchanged. Only the CCSID changes.

*Can I correct the CCSID of a message?*

You cannot correct the message-level CCSID of a message. You can change the message queue CCSID to match the message-level CCSID. You can also delete the message and send it again with the correct message-level CCSID.

### **Related concepts**

Message-level support

When a message is sent to a message queue, you must communicate the CCSID of the replacement data or the immediate message text to the IBM i operating system. Use the CCSID parameter on any of the send message commands or APIs to communicate this CCSID to the system.

Message description-level support

When a message description is added to a message file, the CCSID of the message text must be communicated to the IBM i operating system.

*Can I correct the CCSID of a message description?*

You can use the Change Message Description (CHGMSGD) command to change the CCSID of a message description. If you do not change the first- or second-level text at the same time that you change the message description CCSID, the text remains unchanged. Only the CCSID changes.

For example, you can enter the following Change Message Description command to correct the CCSID of a message description without changing any of the first- or second-level message text:

```
CHGMSGD MSGF(MYLIB/MYMSGQ) MSGID(TST0001) CCSID(37)
```

### **Related tasks**

Can I correct the CCSID of a message file?

You might have a message file that has a CCSID that does not match the CCSID of the message descriptions in it. This typically results from adding message descriptions with a message-level CCSID of 65535 to a message file with a CCSID that is not 65534 or 65535.

### **Related reference**

Change Message Description (CHGMSGD) command

### **Changing the CCSID of a physical file**

You can use the Change Physical File (CHGPF) command to change the coded character set identifier (CCSID) of a physical file. However, the physical file cannot be changed under certain conditions.

A physical file cannot be changed if one or more of the following conditions exist when working with a logical file defined over a physical file:

- The logical file has a sort sequence table associated with the CCSID of the physical file and the CCSID you want to change to is incompatible. That is, conversion between the original CCSID and the CCSID you want to change to is not allowed because all the characters of the original CCSID are not in the new CCSID.

- The logical file has a sort sequence table associated with the CCSID of the physical file and the CCSID you want to change to is incompatible. Additionally, the logical file has fields defined with CCSIDs that are not compatible to the new CCSID you want to change the physical file to. Again, conversion between the original CCSID and the CCSID you want to change to is not allowed because all the characters of the original CCSID of the logical file or the fields with specific CCSIDs are not in the new CCSID.
- A select/omit or join logical file, or both that performs select/omits or joins between physical file fields that have different CCSIDs.
- A join logical file with a sort sequence table such that the CCSID of the logical file's secondary access path is different than the CCSID to which the physical file is being changed.

### **Related concepts**

#### Display files

When a display file object is created, it is tagged with the coded character set identifier (CCSID) of the source file.

### **Related reference**

#### Change Physical File (CHGPF) command

### **Character sorting**

Traditionally, information is displayed in sorted order to enable users to easily find the items they are looking for. However, users of different languages might have very different expectations of what a sorted list should look like.

Not only does the alphabetical order vary from one language to another, but it also can vary from document to document within the same language. For example, phonebook ordering might be different than dictionary ordering. String comparison is one of the basic functions most applications require, and yet implementations often do not match local conventions.

For example, here are some of the ways languages vary in the ordering of strings:

- The letters A through Z can be sorted in a different order than in English. For example, in Lithuanian, the letter y is sorted between the letters i and k.
- Accented letters can be treated as distinct letters. For example, Å in Danish is treated as a separate letter that sorts just after the letter Z.
- Unaccented letters that are considered distinct in one language can be indistinct in another. For example, the letters v and w are two different letters according to English. However, v and w are considered variant forms of the same letter in Swedish.

IBM i lets you customize the sequence in which characters are sorted. You can use any of the following methods to do this:

- IBM i sort sequence support
- ICU-based sort support
- Locale sort support. If your application uses locales, you can use the sorting support provided by the LC\_COLLATE locale category.

### **Related concepts**

#### Sort sequences

The IBM i operating system supports sort sequence. By using one of the listed options, you can order your data according to cultural-dependent requirements for specific applications.

#### LC\_COLLATE category



The LC\_COLLATE category defines character or string collation information. Within LC\_COLLATE you can specify a sort sequence to use using the *cpysyscol* keyword. The *cpysyscol* keyword value is used in place of the LC\_COLLATE category definitions.

### **IBM i sort sequence support**

IBM i sort sequence support is the sort support that the system has provided for multiple releases. Using this support, users can create their own defined sort tables. However, the sort is limited to a single weight for each individual character.

IBM i sort sequence also provides full support only for single-byte character set (SBCS) and CCSID 13488. These sort sequences are supported over most of the system.

#### *Sort sequence tables*

A sort sequence table is an object that contains the weight of each single-byte graphic character within a specified coded character set identifier (CCSID). The system-recognized identifier for the sort sequence table object type is \*TBL.

Depending on your requirements, you can define a table to have either a unique weight for each graphic character or shared weights for some graphic characters. If you define a table that contains unique weights for each character within the character set, your table is known as a unique-weight table. If you define a table that contains some graphic characters that share the same weight, your table is known as a shared-weight table. For example, if you want to sort the graphic character capital letter A and the graphic character small letter a together, you can define a shared-weight table. If you want to sort these graphic characters separately, you can define a unique-weight table.

A set of sort sequence tables is included with the system. This set of tables defines both unique-weight and shared-weight sort sequences for all single-byte character set (SBCS) languages.

### **Sort sequence table implementation notes**

Sort sequence support does not take into consideration the following information:

- Special cases of single characters that should be handled as multiple characters (such as the German characters sharp).
- Sequences of characters that should be treated as a single character (such as the Danish aa, Hungarian ly, Serbian lj, Spanish ll).
- Nonalphanumeric characters that should be ignored because they are embedded in alphanumeric strings (such as the hyphen in co-op).
- Prefixes that should be ignored (such as *Van der* in the name *Van der Pool*).
- Program-described files.
- DBCS code pages.

If a sort sequence table has a weight other than hexadecimal 40 assigned to the blank character, unpredictable results can occur when strings of unequal lengths are compared.

### **Sort sequence tables included with the system**

You can use the Work with Tables (WRKTBL) command to view the contents of the sort sequence tables that are included with the IBM i operating system. The tables are located in the QSYS library.

When looking at these tables, consider the following information:

- Several tables included with the system represent a single sort sequence, each encoded with a different coded character set identifier (CCSID) value. Not all of the characters in a given sort sequence exist in every CCSID in which the sort sequence is encoded.
- Use the language identifier (LANGID) parameter and the sort sequence (SRTSEQ) parameter to access the unique-weight tables (\*LANGIDUNQ) or the shared-weight tables (\*LANGIDSHR).

- When using the sort sequence, the relative weights shown in these tables differ from the actual weights in the sort sequence table on the system. The relative weights shown in these tables are examples only.
- The relative unique weight of a character is shown by the order of the characters in the sort sequence table. The relative unique weight is determined by assigning a weight of 1 to the first character in the sort sequence table and incrementing by 1 for each of the following characters until the end of the table is reached.
- GCGID is the graphic character global identifier.

For example, the Arabic sort sequence table shows the relative sort sequence weights for characters that are sorted using the Arabic sort sequence table.

### How to build sort sequence tables

To create a user-defined sort sequence table, copy an existing sort sequence table using the Create Table (CRTTBL) command, and then modify the copy of the table. Table functions allow you to perform the following tasks:

- Use a definition stored in a source member.
- Create a table based on another sort sequence table using an interactive interface.

You can create a sort sequence table (MYTEST) from a copy of an existing table using the following CRTTBL command:

```
CRTTBL TBL(MYTEST) SRCFILE(*PROMPT) TBLTYPE(*SRTSEQ)
BASESRTSEQ(QSYS/QLA10025S) CCSID(037)
```

This command displays a sort sequence table that you can modify. Your table is created from a function key on this display. Your resulting table has a coded character set identifier (CCSID) value of 00037. The table is named MYTEST and is stored in the current library.

The following table shows one way in which the resulting characters may be shown on the first display of the MYTEST sort sequence table. The actual panel shows characters instead of text descriptions. For example, the character shown for sequence 0100 is a question mark (?), and the character shown for sequence 0070 is a colon (:).

**Note:** The characters that you actually see on the first display of the MYTEST sort sequence table might vary, depending on the device that you use.

Sequence	Character
0010	Equal sign
0020	Overline
0030	(SHY)
0040	Hyphen
0050	Comma
0060	Semi-colon
0070	Colon
0080	Exclamation mark
0090	Inverted exclamation mark
0100	Question mark
0110	Inverted question mark
0120	Slash
0130	Period

Sequence	Character
0140	Acute accent mark
0150	Grave accent mark
0160	Caret
0170	Right square bracket
0180	Tilde
0190	Small multiply dot
0200	Comma

You can make changes to the tables to move characters in each code page to the preferred position for the national language sort sequence table. The ordering is done by increments of 10. Therefore, the first value is 10, then 20, and so on. If some characters have a shared weight, these groups of characters have the same sequenced weight.

### Related concepts

[Sort sequence \(QSRTSEQ\) system value](#)

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

[ILE RPG sort sequence](#)

The ILE RPG feature, an option of the IBM Rational Development Studio for i licensed program, provides the possibility for a user to specify a sort sequence table and to use the table in comparison operations that are performed with nonnumeric data.

[Working with CCSIDs](#)

Using the system implementation of Character Data Representation Architecture (CDRA), you can achieve consistent representation, processing, and interchange of coded characters (data) on the IBM i operating system and across IBM Systems. The primary implementation of CDRA on the IBM i operating system is through coded character set identifier (CCSID) support.

[Code pages](#)

Several IBM code pages match the International Standard ISO/IEC 8859. The IBM i operating system supports parts of the ISO Standard with equivalent IBM code pages.

### Related reference

[Create Table \(CRTTBL\) command](#)

*Sort sequence for Arabic*

The Arabic sort sequence table is used with the Arabic language.

GCGID	Character	Shared weight	Unique weight
SP010000	٠	1	1
SP090000	—	2	2
SP320000	٠١٢٣٤٥٦٧٨٩	3	3
SP100000	-	4	4
SP080000	+	5	5
SP080007	٠	6	6

GCGID	Character	Shared weight	Unique weight
SP140000	;	7	7
SP140007	;	8	8
SP130000	:	9	9
SP020000	!	10	10
SP150000	?	11	11
SP150007	?	12	12
SP120000	/	13	13
SP110000	.	14	14
SD130000	,	15	15
SD190000	~	16	16
SP050000	,	17	17
SP040000	"	18	18
SP060000	(	19	19
SP070000	)	20	20
SM060000	[	21	21
SM080000	]	22	22
SM110000	{	23	23
SM140000	}	24	24
SM050000	@	25	25
SC010000	¤	26	26
SC040000	€	27	27
SC030000	\$	28	28

GCGID	Character	Shared weight	Unique weight
SM040007	*	29	29
SM070000	\	30	30
SM030000	&	31	31
SM010000	#	32	32
SM020007	%	33	33
SA010000	+	34	34
SA060000	÷	35	35
SA070000	x	36	36
SA030000	<	37	37
SA040000	=	38	38
SA050000	>	39	39
SM660000	└	40	40
SM130000		41	41
SM650000	!	42	42
SP300000	(NSP)	43	43
SP310000	(NSP)	43	44
SM870000	‘	43	45
ND100000	0	44	46
ND100001	.	44	47
ND010000	1	45	48
ND010001	ı	45	49
ND020000	2	46	50
ND020001	ı	46	51
ND030000	3	47	52
ND030001	ı	47	53

<b>GCGID</b>	<b>Character</b>	<b>Shared weight</b>	<b>Unique weight</b>
ND040000	4	48	54
ND040001	ᵢ	48	55
ND050000	5	49	56
ND050001	ᵒ	49	57
ND060000	6	50	58
ND060001	ᶒ	50	59
ND070000	7	51	60
ND070001	γ	51	61
ND080000	8	52	62
ND080001	Λ	52	63
ND090000	9	53	64
ND090001	ᶓ	53	65
LA010000	a	54	66
LA020000	A	54	67
LB010000	b	55	68
LB020000	B	55	69
LC010000	c	56	70
LC020000	C	56	71
LD010000	d	57	72
LD020000	D	57	73
LE010000	e	58	74
LE020000	E	58	75
LF010000	f	59	76
LF020000	F	59	77
LG010000	g	60	78

<b>GCGID</b>	<b>Character</b>	<b>Shared weight</b>	<b>Unique weight</b>
LG020000	<b>G</b>	60	79
LH010000	<b>h</b>	61	80
LH020000	<b>H</b>	61	81
LI010000	<b>i</b>	62	82
LI020000	<b>I</b>	62	83
LJ010000	<b>j</b>	63	84
LJ020000	<b>J</b>	63	85
LK010000	<b>k</b>	64	86
LK020000	<b>K</b>	64	87
LL010000	<b>l</b>	65	88
LL020000	<b>L</b>	65	89
LM010000	<b>m</b>	66	90
LM020000	<b>M</b>	66	91
LN010000	<b>n</b>	67	92
LN020000	<b>N</b>	67	93
LO010000	<b>o</b>	68	94
LO020000	<b>O</b>	68	95
LP010000	<b>p</b>	69	96
LP020000	<b>P</b>	69	97
LQ010000	<b>q</b>	70	98
LQ020000	<b>Q</b>	70	99
LR010000	<b>r</b>	71	100
LR020000	<b>R</b>	71	101
LS010000	<b>s</b>	72	102
LS020000	<b>S</b>	72	103

<b>GCGID</b>	<b>Character</b>	<b>Shared weight</b>	<b>Unique weight</b>
LT010000	t	73	104
LT020000	T	73	105
LU010000	u	74	106
LU020000	U	74	107
LV010000	v	75	108
LV020000	V	75	109
LW010000	w	76	110
LW020000	W	76	111
LX010000	x	77	112
LX020000	X	77	113
LY010000	y	78	114
LY020000	Y	78	115
LZ010000	z	79	116
LZ020000	Z	79	117
AX300000	*	80	118
AA210000	†	80	119
AA210001	‡	80	120
AA210002	£	80	121
AA210006	§	80	122
AA310000	†	80	123
AA310001	‡	80	124
AA310002	£	80	125
AA310006	§	80	126
AW310000	§	80	127
AA310401	¶	80	128



<b>GCGID</b>	<b>Character</b>	<b>Shared weight</b>	<b>Unique weight</b>
AA310406	آ	80	129
AY310001	ع	80	130
AY310000	أ	80	131
AA010000	ا	81	132
AA010001	آ	81	133
AA010002	أ	81	134
AA010006	آ	81	135
AB010000	ب	82	136
AB010003	ب	82	137
AT020000	ت	83	138
AT010000	ت	83	139
AT010003	ت	83	140
AT470000	ت	84	141
AT470003	ت	84	142
AG230000	ج	85	143
AG230003	ج	85	144
AH450000	ح	86	145
AH450003	ح	86	146
AH470000	ح	87	147
AH470003	ح	87	148
AD010000	د	88	149
AD470000	د	89	150
AR010000	ر	90	151
AZ010000	ز	91	152
AS010006	س	92	153

GCGID	Character	Shared weight	Unique weight
AS010000	ا	92	154
AS010003	آ	92	155
AS230006	ش	93	156
AS230000	س	93	157
AS230003	ص	93	158
AS450006	ض	94	159
AS450000	ط	94	160
AS450003	ظ	94	161
AD450006	ق	95	162
AD450000	ك	95	163
AD450003	گ	95	164
AT450000	ت	96	165
AZ450000	ث	97	166
AC470000	ج	98	167
AC470002	ح	98	168
AC470003	خ	98	169
AC470004	د	98	170
AG310000	ذ	99	171
AG310002	ر	99	172
AG310003	ز	99	173
AG310004	س	99	174
AF010000	ف	100	175
AF010003	ب	100	176
AQ010000	ق	101	177
AQ010003	ن	101	178

<b>GCGID</b>	<b>Character</b>	<b>Shared weight</b>	<b>Unique weight</b>
AK010000	ك	102	179
AK010003	ڪ	102	180
AL010000	ل	103	181
AL010003	ل	103	182
AL220000	ف	104	183
AL220003	ف	104	184
AL320000	ي	104	185
AL320003	ي	104	186
AL020000	ا	104	187
AL020003	ا	104	188
AM010000	م	105	189
AM010003	م	105	190
AN010000	ن	106	191
AN010003	ن	106	192
AH010000	ه	107	193
AH010003	ه	107	194
AH010007	ه	107	195
AH010004	ه	107	196
AW010000	و	108	197
AA020000	ع	109	198
AA020001	ع	109	199
AA020002	ع	109	200
AY010000	ي	109	201
AY010001	ي	109	202
AY010002	ي	109	203

GCGID	Character	Shared weight	Unique weight
AY010003	⎵	109	204
AA070000	⎵	110	205
AU070000	⎵	111	206
AI070000	⎵	112	207
AA050000	⎵	113	208
AA050004	⎵	113	209
AU050000	⎵	114	210
AU050004	⎵	114	211
AI050000	⎵	115	212
AI050004	⎵	115	213
AE050000	⎵	116	214
AE050004	⎵	116	215
AX100000	⎵	117	216
AX100004	⎵	117	217
SM860000	⎵	118	218

#### *Sort sequence support*

The sort sequence support is provided in these IBM i functions.

- A user interface for creating new tables based on system-supplied sort sequence tables
- The Work with Tables (WRKTBL) command for creating and displaying tables
- The Create Table (CRTTBL) command for creating tables
- CL, ILE RPG IV, and ILE COBOL for compilers.
- Program support
- Work management support
- Database management support
- Other system components support

#### **Related reference**

[Work with Tables \(WRKTBL\) command](#)

[Create Table \(CRTTBL\) command](#)

### *Sort sequence support in programs*

You can assign a sort sequence to a program that is used for ordering and comparing data by specifying the sort sequence to be used at compilation time.

Specify the sort sequence to be used with the sort sequence (SRTSEQ) parameter and language identifier (LANGID) parameters of the create program commands. Valid SRTSEQ parameter values are:

- SRTSEQ(\*HEX) means that no sort sequence should be used (hexadecimal sorting).
- SRTSEQ(\*LANGIDUNQ) or SRTSEQ(\*LANGIDSHR) means that the unique- or shared-weight sort sequence, determined by the LANGID parameter, should be used.
- A name for the system-supplied or user-supplied sort sequence name can be specified explicitly on the SRTSEQ parameter. If you explicitly specify a sort sequence name, the LANGID parameter is ignored.
- SRTSEQ(\*JOB) or LANGID(\*JOB) means that the sort sequence to be used is determined by the value associated with the job when the program is created.
- SRTSEQ(\*JOB RUN) or LANGID(\*JOB RUN) means that the sort sequence to be used is determined by the values from the job when the program is run.

The first three options assign the sort sequence to the program object at creation time. This sequence is always used when the program is run. Using the \*JOB RUN value on the SRTSEQ or LANGID parameters, however, provides the possibility for dynamically assigning sort sequence to the program.

### *Sort sequence support in work management*

Work management involves the assigning of the SRTSEQ value at the job level, the user profile level, and the system value level.

**Sort sequence support at the job level:** A sort sequence (SRTSEQ) value is assigned to a job. It is valid on the Submit Job (SBMJOB), Batch Job (BCHJOB), and the Change Job (CHGJOB) commands. If a program is created with SRTSEQ(\*JOB), the sort sequence is set from the job sort sequence. If a program is created with SRTSEQ(\*JOB RUN), the sort sequence is set from the job sort sequence at run time.

**Sort sequence support at the user profile level:** The user profile assigns a SRTSEQ value to a user and, by default, to all jobs running under this user profile. The user profile SRTSEQ value defaults to the sort sequence system value (QSRTSEQ).

**Sort sequence support at the system value level:** The QSRTSEQ system value defines a sort sequence that can be referred to by other objects. The QSRTSEQ system value should be set according to the requirements of the primary language used on the system.

### **Related concepts**

#### Sort sequence (QSRTSEQ) system value

The sort sequence (QSRTSEQ) system value, along with the QLANGID system value, determines the sort sequence table to be used for sorting character data.

### *Sort sequence support in database management*

Database management supports the SRTSEQ and LANGID parameters on the Create Physical File (CRTPF) and Create Logical File (CRTLF) commands.

The LANGID and SRTSEQ parameters determine a sort sequence table. The sort sequence table is captured at file creation time and is stored as an attribute of the file. The SRTSEQ job attribute has no effect on the processing of an existing database file. The sort sequence table associated with the file is used for key sequencing, select logic fields and omit logic fields, and for join field functions.

The ALTSEQ keyword in DDS can also be used to specify a sort sequence table. The ALTSEQ keyword applies only to the key fields, not to the select logic fields and the omit logic fields. If the SRTSEQ parameter is specified on the CRTPF command or the CRTLF commands and the ALTSEQ keyword in the DDS source file specify a sort sequence table, an error message is sent and the file is not created.

The default SRTSEQ parameter on CRTPF and CRTLF commands is \*SRC, which indicates that the sort sequence table on the ALTSEQ keyword should be used. If ALTSEQ is not used in DDS, the SRTSEQ attribute of the job determines the file attributes when creating or changing the file.

## How sort sequences are specified for database management

Sort sequence tables can be specified in the following areas:

- IBM Query for i support

External sort sequence tables (including those included with the system) and user-defined tables can be specified.

- IBM DB2 Query Manager and SQL Development Kit for i

The Create Structured Query Language xxx (CRTSQLxxx) commands and the Start Structured Query Language (STRSQL) command support the SRTSEQ and LANGID parameters.

A sort sequence table can be specified when a query object is being defined with the Work with Queries display. The sort sequence (SRTSEQ) value and language identifier (LANGID) value are specified on the Specify Sort Sequence display.

- Db2 for i Query Management

The Create Query Management Query (CRTQMORY) command supports the SRTSEQ and LANGID parameters.

### Related concepts

[Database programming](#)

### Related reference

[Create Physical File \(CRTPF\) command](#)

[Create Logical File \(CRTLF\) command](#)

[Start SQL Interactive Session \(STRSQL\) command](#)

[Create Query Management Query \(CRTQMORY\) command](#)

### Related information

[Arranging key fields using the SRTSEQ parameter](#)

*Sort sequence support in other system components*

Sort sequence support is found in these components of the system.

- CRTCLPGM (Create Control Language Program) command

The LANGID and SRTSEQ parameters are supported.

- DSPPGM (Display Program) command

The LANGID and SRTSEQ values that were specified when the program was created are displayed.

- CRTDSPF (Create Display File) command

The LANGID and SRTSEQ parameters are supported. The values of the RANGE, VALUES, and COMP keywords are validated when the display file is compiled.

- High-level languages

Using ILE COBOL and ILE RPG IV languages, you can specify SRTSEQ and LANGID values directly on the Create Bound Program (CRTBNDXXX) commands. Original Program Model RPG and COBOL compilers use the Create Program (CRTXXXPGM) commands. With ILE C, you can also specify SRTSEQ and LANGID values when you create a locale. You can then associate the locale with a program.

- IBM i Access

The transfer function allows a sort sequence table to be specified when you perform queries on database files and SQL tables.

### Related reference

[Display Program \(DSPPGM\) command](#)

[Create Display File \(CRTDSPF\) command](#)

### Related information

[Create CL Program \(CRTCLPGM\) command](#)

## System i Access

### Sort sequence scenarios

This table shows characters you can sort using a binary, a shared-weight, and a unique-weight sort sequence for the Danish code page 00277.

Character name	Character illustration	Code point in code page 277	Shared sort weight	Unique sort weight
AE ligature	Æ	X'7B'	96	183
O slash	Ø	X'7C'	97	187
A overcircle	À	X'5B'	98	191
Latin capital N	Ń	X'D5'	83	132
Latin capital Z	Ż	X'E9'	95	181
O umlaut	Ö	X'EC'	97	189
Latin capital A	Ą	X'C1'	70	77

Using the information in the previous table, the characters are sorted in ascending order as shown in the following table.

Position in ascending order	Binary sort	Shared weight sort	Unique weight sort
First	A overcircle	Latin capital A	Latin capital A
Second	AE ligature	Latin capital N	Latin capital N
Third	O slash	Latin capital Z	Latin capital Z
Fourth	Latin capital A	AE ligature	AE ligature
Fifth	Latin capital N	O umlaut	O slash
Sixth	Latin capital Z	O slash	O umlaut
Seventh	O umlaut	A overcircle	A overcircle

The following table shows an example of a shared-weight sort sequence, a unique weight sort sequence, and the binary sort sequence for English code page 00037.

Binary sort sequence	Shared-weight sort sequence using LANGID(ENU) and SRTSEQ(*LANGIDSHR)	Unique-weight sort sequence using LANGID(ENU) and SRTSEQ(*LANGIDUNQ)
Jones, Mary	JOHNSON, JOHN	JOHNSON, JOHN
JOHNSON, JOHN	JONES, MARTIN	Jones, Mary
JONES, MARTIN	Jones, Mary	JONES, MARTIN
Smith, Ron	SMITH, ROBERT	Smith, Ron
SMITH, ROBERT	Smith, Ron	SMITH, ROBERT

### Sort sequence types

The IBM i operating system provides a set of shared-weight and unique-weight sort sequence tables for SBCS languages.

A shared-weight sequence is a sort sequence in which some graphic characters may have the same weight as some other characters in the sequence. Those with the same weight sort together as though they were the same character. For example, the letters *a* and *A* might both have the same value 24. This ensures that words such as *able* and *Able* are kept together in a list. In a simple sort table, *a* and *A* might share the value 24, and *b* and *B* might share the value 25 and so on.

A unique-weight sequence is a sort sequence in which each graphic character has a weight different from the weight of every other graphic character in the sequence.

### ICU-based sort support

International Components for Unicode (ICU) based sort support is based on the ICU collation services, which provide a multiple-weight (tertiary level) sort support.

ICU-based sort supports the sorting of data in most CCSIDs supported by the operating system. It also provides a multiple-weight sort based on strings. ICU-based sort support does not allow users to generate additional sort tables.

### ICU-based sort sequence types

An ICU locale sequence is a sort sequence in which a tertiary level sort is used in determining a sort key based on the text string.

In tertiary level sorting, upper and lower case differences in characters are distinguished (for example, "ao" < "Ao" < "a ð"). In addition, a variant of a letter differs from the base form on the tertiary level (such as "a" and " ã").

### ICU locales and sort tables provided by the system

The ICU locale data covers 82 different languages, further divided into 197 regions and variants. For each language, data such as days of the week, months, and their abbreviations are defined. The sort sequence tables for ICU that are defined on the system are based on these locales and regions, as listed in the table in this topic.

The system provides a table (\*TBL) object for each of these locales for you to use when you specify an ICU-based sort. For example, the table object QSYS/ES\_MX directs the database to use the 2.6 version of the ICU sort for Spanish in Mexico. The table object QSYS/I34ES\_MX directs the database to use the 3.4 version of the ICU sort for Spanish in Mexico. ICU provides over 100 locale choices that you can use. See the following example table for a list of the Spanish choices. For the full list of locales that you can use for sort choices, see the [IBM Globalization - ICU Web page](#).

For systems running IBM i V6R1, or later, the system ICU sort support has been updated to ICU 3.4. For systems running IBM i V5R4, or earlier, this support is based on ICU 2.6. The ICU 3.4 tables follow the naming of I34xx\_yy. You can see these choices by looking at the table objects on the operating system. Use the following command to see the ICU 3.4 tables: WRKTBL TBL (QSYS/I34\*)

If your applications use ICU sort support, you need to update them to use the 3.4 version of the support for better performance.

Example locale name	Language or variant	Region
ES	Spanish	
ES_AR	Spanish	Argentina
ES_BO	Spanish	Bolivia
ES_CL	Spanish	Chile
ES_CO	Spanish	Colombia
ES_CR	Spanish	Costa Rica



Example locale name	Language or variant	Region
ES_DO	Spanish	Dominican Republic
ES_EC	Spanish	Ecuador
ES_ES	Spanish	Spain
ES_ES_PREE	Spanish Pre-Euro	Spain
ES_GT	Spanish	Guatemala
ES_HN	Spanish	Honduras
ES_MX	Spanish	Mexico
ES_NI	Spanish	Nicaragua
ES_PA	Spanish	Panama
ES_PE	Spanish	Peru
ES_PR	Spanish	Puerto Rico
ES_PY	Spanish	Paraguay
ES_SV	Spanish	El Salvador
ES_US	Spanish	United States
ES_UY	Spanish	Uruguay
ES_VE	Spanish	Venezuela
ES__TRADIT	Spanish Traditional	

#### Sort sequence scenarios

This table shows an example of the results you might obtain, given the same input data but different ICU locales specified on the sort request. The two locales used are EN\_US and FR\_FR.

en_us ICU locale sort sequence using SRTSEQ(EN_US)	fr_fr ICU locale sort sequence using SRTSEQ(FR_FR)
cote	cote
cot é	c ô te
c ô te	cot é
c ô t é	c ô t é

#### Related concepts

[ICU sort sequence](#)

### ICU-based sort sequence support

IBM i support for ICU sort sequence is provided in IBM i functions in the Work management and Database management topics.

#### **Related concepts**

[Work management](#)

#### **Related information**

[Database](#)

## Working with bidirectional data

Arabic and Hebrew languages use an alphabet written and read from right to left. Numerics and Latin text imbedded in the right-to-left text are written and read from left to right. Therefore, these languages are called bidirectional languages.

Because bidirectional languages are written and read from right to left, you should avoid using the terms left and right. For example, *right margin* in Hebrew or Arabic documents is the beginning of the line and not the end. Use the words *start* and *end* in place of the words *right* and *left*.

Hebrew and Arabic have no case-sensitive characters. To avoid the incorrect presentation of characters, no case-sensitive checking or substitution should be performed. In addition, the Arabic language does not use abbreviations, therefore, you should use only complete words.

#### **Related concepts**

[Coding globalized applications that use bidirectional data](#)

When you are developing NLV-enabled applications, you should consider some specific restrictions on bidirectional languages.

#### **Bidirectional application support**

Workstations, display files, and the user interface manager (UIM) provide support for bidirectional applications.

#### **Workstation support**

Workstations that have the ability to display Arabic and Hebrew character sets also have the ability of right-to-left cursor movement. Right-to-left cursor movement on input fields can be achieved in one of the following ways:

- Pressing a special function key available on Hebrew and Arabic keyboards called the reverse key. This is a toggle function that moves the cursor to the other side of the field, allows for cursor movement in the opposite direction, and also changes the language layer from Latin to Hebrew or Arabic and back again.
- Using the DDS cursor control codes for display files. When the CHECK keyword is used with a cursor-controlled code, it specifies that the cursor is to move from right to left. The following parameters are valid cursor control codes:
  - CHECK (RL): Moves the cursor from right to left in specified nonnumeric input fields or in all nonnumeric input fields on the display.
  - CHECK (RLTB): Moves the cursor from right to left between fields.

When using these parameters, consider the following information:

- Modulus check digit verification is supported, but the check digit is the byte to the extreme right of the field.
- A field for which right-to-left cursor movement is specified can occupy more than one line on the display. However, the cursor still moves from the top of the display to the bottom.
- You cannot use right-to-left cursor movement with user-defined data streams.

**Note:** If no cursor positioning is specified in the display file or by the program, the cursor is placed in the input-capable field to the extreme left of the top line.

## Display file support

The system does not check to make sure that all display files that open to the display station are capable of right-to-left cursor movement. Therefore, it is the responsibility of application programmers to ensure that the appropriate display files are used.

## User interface manager support

The user interface manager gives the following bidirectional support for creating online information and panels:

- BIDI= NONE | RTL | LTR

This attribute controls the directional orientation of the panels in the panel group.

RTL indicates that the panel in the panel group is bidirectional and should be displayed with a right-to-left orientation.

LTR indicates that the panel in the panel group is bidirectional and should be displayed with a left-to-right orientation.

- :RT and :ERT

Reverse-direction-text tags indicate that the enclosed text has an orientation that is opposite to the orientation of the panel group.

For a list of UIM tags, see the Application Display Programming PDF.

## Related reference

[DDS concepts](#)

## Related information

[Application Display Programming PDF](#)

## Checklist: Bidirectional support guidelines

When you create an application with bidirectional support, follow the guidelines in this table.

Complies	Not applicable	Rule
		Software design must allow for bidirectional data to be passed to applications in the same order that a speaker of the language can spell it out.
		The product design must allow for the implementation of the correct handling of bidirectional keyboard and presentation functions.
		Designing of a function that implies logical movement of cursor or characters must permit mirroring of that function.
		Keys or operations labeled with directional icons or symbols must perform according to the icon or symbol.
		Keyboard nomenclature for mirrored functions must be independent of the direction of data or text entry.
		Display functions must not assume a left-to-right orientation.
		Field attributes must contain room for directional information.
		Indicator location must be reserved for the current direction of the cursor (direction of input).
		The design must allow for independent handling of graphic and text orientation.

Complies	Not applicable	Rule
		Provision must be made to allow shape determination to be performed.
		The deshaping must be definable.
		Provision must be made to allow the selection of the appropriate presentation shape for the numerals.
		Characters must be allowed to touch each other on printers and displays.
		Indicator locations should be reserved for screen and field orientation, current level of nesting, status of push (nesting mechanism), and status of symmetric swapping.
		The design should provide for a method to indicate to the user the nesting structure of a string.
		A system-wide method of deshaping Arabic characters or character strings should be provided.
		An indicator location should be provided for the status of shape determination.
		A method should be provided so that proportional spacing can be provided.
		A method should be provided to allow alignment of the baseline of Arabic and Latin characters (including Hindi and Arabic shapes for numerals).

### Related concepts

[Coding globalized applications that use bidirectional data](#)

When you are developing NLV-enabled applications, you should consider some specific restrictions on bidirectional languages.

### Working with DBCS data

A DBCS file is a file that contains double-byte data or a file that is used to process double-byte data, for example, Japanese, Chinese, or Korean. Other files are called alphanumeric files. You can view DBCS files on display, printer, tape, diskette, and ICF devices.

A more modern method to support DBCS data is to use Unicode instead of DBCS fields. (IBM suggests that you use Unicode to develop new applications.)

You use data description specifications (DDS) to describe DBCS-capable device files.

You should indicate that a file is DBCS in one or more of the following situations:

- The file receives input, or displays or prints output, which has double-byte characters.
- The file contains double-byte literals.
- The file has double-byte literals in the DDS that are used in the file at processing time (such as constant fields and error messages).
- The DDS of the file includes DBCS keywords.
- The file stores double-byte data (database files).

### DBCS strings in a mixed data stream

Typically, both single-byte characters and double-byte characters are used in a DBCS environment. For example, an accounting firm in Japan uses both English and Japanese for the spreadsheet. If both English and Japanese are being encoded as mixed SBCS and DBCS, the product must be able to understand a mixed character set that contains both single-byte coded characters and double-byte coded characters.

In IBM systems that use EBCDIC, a DBCS string is bracketed in a mixed data stream by a shift-out (SO) control character and a shift-in (SI) control character.

The following example shows the coding for a mixed string:

```
sss (SO) D1D2D (SI) ssss
```

The following example shows the coding for a mixed hexadecimal string:

```
818283 0E 41424143 0F 818283
```

### Supported code ranges

The IBM i operating system supports Japanese, Korean, Simplified Chinese, and Traditional Chinese character-set code ranges.

Using the IBM i Access Family of products, the systems also provide support for these non-IBM personal computer DBCS code pages:

- Republic of Korea National Standard graphic character set (KS)
- Taiwan Industry Standard graphic character set (Big5)
- The People's Republic of China National Standard graphic character set (GB)

### Related concepts

[DDS concepts](#)

### Related information

[System i Access](#)

### Checklist: DBCS application design

When you create an application with double-byte coded character set (DBCS) support, follow these guidelines.

A complete list of these guidelines, as well as a full description of each guideline, is included in *Volume 1 Designing Enabled Products, Rules and Guidelines* (SE09-8001). For your convenience, a subset of these guidelines is provided in the following table.

Complies	Not applicable	Rule
		Double-byte coded character set code points in the graphic character range must be used only for graphic characters and must not be used for control purposes.
		Single-byte meaning must not be drawn from either byte of double-byte coded data.
		Double-byte coded character set character generators must be capable of producing user-accessible graphic characters.
		The ability to switch between single-byte coded character set and double-byte coded character set and the coexistence of single-byte coded character set and double-byte coded character set in the same session must be possible.
		User-interface text modules for double-byte coded character set systems must be loaded separately from the running code.

### Developing applications that process DBCS data

You should design your application programs for processing double-byte data in the same way you design application programs for processing alphanumeric data.

Here are some additional considerations:

- Make sure that the double-byte data is always processed in a double-byte unit and does not split a double-byte character.
- Identify double-byte data used in the database files.
- Design display and printer formats that can be used with double-byte data.
- If needed, provide DBCS conversion as a means of entering double-byte data for interactive applications. Use the DDS keyword for DBCS conversion (IGCCNV) to specify DBCS conversion in display files. Because DBCS workstations provide a variety of double-byte data entry methods, you are not required to use the IBM i DBCS conversion function to enter double-byte data.
- Create double-byte messages to be used by the program.
- Specify extended character processing so that the system prints and displays all double-byte data.
- Determine whether additional double-byte characters need to be defined. User-defined characters can be defined and maintained using the character generator utility (CGU). Information about CGU can be found in the *ADTS/400: Character Generator Utility*, book SC09-1769-00.

### **Related concepts**

#### Designing globalized applications

Your goal in designing international application components is to create components that support national languages independently.

#### ***Use of double-byte data***

You can use double-byte data in several ways.

- As data in files:
  - Data in database files.
  - Data entered in input-capable and data displayed in output-capable fields of display files.
  - Data printed in output-capable fields in printer files.
  - Data used as literals in display files and printer files.
- As the text of messages.
- As the text of object descriptions.
- As literals and constants, and as data to be processed by high-level language programs.

Double-byte data can be displayed only at DBCS displays and printed only on DBCS printers. Double-byte data can be written onto diskette, tape, disk, and optical storage.

#### **Where you cannot use double-byte data:**

You cannot use double-byte data in the following ways:

- As IBM i object names.
- As command names or variable names in control language (CL) and other high-level languages.
- As displayed or printed output on alphanumeric workstations.

#### **Double-byte character size:**

When displayed or printed, double-byte characters typically are twice as wide as single-byte characters.

Consider the width of double-byte characters when you calculate the length of a double-byte data field because field lengths are typically identified as the number of single-byte character positions used. The DDS concepts information includes more information about calculating the length of fields containing double-byte data.

#### **Related reference**

DDS concepts

#### **Related information**

Application Display Programming PDF

### ***DBCS coding considerations***

If the application will be used in a DBCS environment, ensure that it is DBCS-enabled. Here are some suggestions to consider when you develop the general product design.

- Reserve more expansion space for DBCS textual data translation than you reserve for SBCS textual data translation. (It is possible, however, that the number of bytes used may be reduced when a SBCS sentence is being translated into DBCS.)
- Ensure programs can understand shift-out and shift-in delimiters. Otherwise, EBCDIC mixed-byte character strings cannot be handled.
- Do not enable short responses for DBCS. For short responses, it is difficult to shift in and out of DBCS. The yes and no are examples of short responses.
- Remember to use the graphic data type G where appropriate.
- Remember that the 5494 remote controller supports the graphic data type.
- Be careful when converting mixed data between DBCS-host code and DBCS-PC code, because the transition may change the data length. Losing and gaining SO and SI character pairs can upset field-length calculations.
- Make sure the double-byte data is always processed in a double-byte unit. Do not split a double-byte character.
- Design the display as well as the print format to avoid the problem of truncation of a double-byte character into two single-byte units.

### ***Creating physical files***

When you create a physical file, display file, and printer file for a DBCS environment, consider the IGCDTA parameter in the these commands.

- Create Physical File (CRTPF) command

If DBCS fields are described in DDS, the system treats the file as a DBCS file. Otherwise, specify \*YES for the parameter of the CRTPF command so that the file can contain double-byte character set data. However, the system ignores the IGCDTA parameter value when a value for the RCDLEN parameter is not specified.

- Create Display File (CRTDSPF) and Create Printer File (CRTPRTF) commands

Specify \*YES for the parameter when using the CRTDSPF or CRTPRTF commands to create the externally described files. Then DBCS attributes, in addition to those defined in the DDS, can be specified.

### **Related reference**

[Create Physical File \(CRTPF\) command](#)

[Create Display File \(CRTDSPF\) command](#)

[Create Printer File \(CRTPRTF\) command](#)

### ***Target physical files***

When you specify OUTPUT(\*OUTFILE) with the Copy Spooled File (CPYSPLF), Display Spooled File (DSPSPLF), or Work with Spooled Files (WRKSPLF) command under the DBCS version of the IBM i operating system, the target physical file must be DBCS-enabled.

**Note:** The primary language of the system must support the double-byte character set to allow DBCS-enabled applications.

Use the QIGC system value to check if a DBCS version of the system is installed. Because it is set by the system, it cannot be changed. This system value can be referred to in an application program. QIGC can be:

- 0 (DBCS version is not installed)
- 1 (DBCS version is installed)

A DBCS system allows for concurrent use of SBCS and DBCS data. When the QIGC system value is 1, you should not assume all jobs are DBCS.

## Related reference

[Copy Spooled File \(CPYSPLF\) command](#)

[Display Spooled File \(DSPSPLF\) command](#)

[Work with Spooled Files \(WRKSPLF\) command](#)

### *Using DBCS CCSIDs*

You should use DBCS CCSIDs for DBCS languages. When you design an application to be used in the DBCS environment, consider this information.

- If the QIGC system value is set on, system value QCCSID must have the value of a mixed CCSID.
- If the DBCS and SBCS language users are sharing the same system, they may want to store their data in different databases. It is possible to create DBCS-capable and SBCS-capable physical files in the same system. The CCSID parameter on the CRTPF command or the CCSID keyword on the physical file DDS definition can be used to specify the CCSID value that the data is stored in.
- If a CCSID was not explicitly assigned through DDS at file creation time, the database physical file character J (DBCS-only), E (DBCS-either), O (DBCS-open) or G (DBCS-graphic) fields are implicitly assigned a CCSID value.

## Related reference

[Create Physical File \(CRTPF\) command](#)

### *Using DDS keywords*

With DDS keywords, you can specify alternative ways to enter data through display files, change input- and output-capable alphanumeric data fields to DBCS data fields, or specify the special features of the DBCS printer output.

- CHRSIZ (Character Size)  
This printer file keyword can expand the printer characters to twice the normal size (width and height). This keyword is valid only for IPDS printers and for printer files with a device type of \*IPDS or \*AFPDS specified.
- CONCAT (Concatenate)  
This keyword can be used only on logical files. This keyword does not support concatenation of a character field and a data type O field.
- DFLIN (Define Line)  
The printer file keyword draws horizontal and vertical lines.
- IGCALTTYP (DBCS Alternative Data Type)  
This display and printer keyword is used to change input- and output-capable character fields to DBCS fields with data type O.
- IGCANKCNV (Alphanumeric-to-DBCS Conversion)  
This printer file keyword converts alphanumeric SBCS characters to equivalent DBCS characters. Printed SBCS alphanumeric characters have the same appearance as printed DBCS characters. The printed DBCS characters, however, are twice as wide as the equivalent SBCS alphanumeric characters.
- IGCCDEFNT (DBCS Coded Font)  
This printer file keyword specifies the DBCS coded font for printing a named or constant field (or fields).
- IGCCNV (DBCS Conversion)  
This is a display file keyword that enables DBCS conversion.
- IGCCHRTT (DBCS Character Rotation).  
This printer file keyword rotates each DBCS character 90 degrees counterclockwise before printing. By rotating characters, the system prints them in reading sequence. This keyword should be used only for printer files to be printed with 5553 printers or IPDS AFP(\*YES) printers.



## Related reference

[DDS concepts](#)

## Related information

[Application Display Programming PDF](#)

### *DBCS file data types*

This table summarizes the changes of the data type of a field in a physical file when the data type is being referred to in a logical file.


Physical file data types	Logical file data types
J	J, O, E, H, G
O	O, H
E	E, O, H
A	A, O, E, H
H	J, O, E, A, H
G	G, O, J, E

### *The Katakana code page (00290)*

The Katakana code page (code page 00290) for Japanese supports English and single-byte Katakana (phonetics) characters.

The lowercase English characters are located at code points different from other code pages, and some older hardware might not be able to display English uppercase, lowercase, and Katakana characters concurrently. Therefore, you need to take special considerations if the application is going to support this code page:

- Avoid using the lowercase alphabet for syntactic characters.
- Avoid using the SBCS lowercase alphabet with Japanese DBCS messages.

For more information about code page 290, see the [IBM Coded Character Sets and Related Resources: Globalizing your e-business Web site](#) .

### *Unicode support and IBM DBCS displays*

The IBM i operating system supports Unicode. IBM DBCS-capable display stations, however, do not support Unicode data.

If you are designing an application to handle unicode data for display on an IBM DBCS-capable display, you must convert the data to a mixed-byte CCSID before sending the data to the display station.

### **Processing double-byte characters**

Because of the large number of double-byte characters, the IBM i operating system needs more information to identify each double-byte character than is needed to identify each alphanumeric character.

There are two types of double-byte characters: basic and extended. These characters are typically processed by the device on which the characters are displayed or printed.

### **Basic double-byte characters**

Basic characters are frequently used double-byte characters that reside in the hardware of a DBCS-capable device. The number of double-byte characters stored in the device varies with the language supported and the storage size of the device. A DBCS-capable device can display or print basic characters without using the extended character processing function of the operating system.

## Double-byte extended characters

When processing extended characters, the device requires the assistance of the system. The system must tell the device what the character looks like before the device can display or print the character. Extended characters are stored in a DBCS font table, not in the DBCS-capable device. When displaying or printing extended characters, the device receives them from the DBCS font table under control of the operating system.

Extended character processing is a function of the operating system that is required to make characters stored in a DBCS font table available to a DBCS-capable device.

To request extended character processing, specify the double-byte extended character parameter, IGCEXNCHR(\*YES), on the file creation command when you create a display (CRTDSPF) or create a printer file (CRTPRPF) command that processes double-byte data. Because IGCEXNCHR(\*YES) is the default value, the system automatically processes extended characters unless you instruct it otherwise. You can change this file attribute by using the change file (CHGDSPF) or (CHGPRPF) command. You can override the file attribute with the override display file (OVRDSPF) or override printerfile (OVRPRTF) command. For example, to override the display file DBCSDSPF so that extended characters are processed, enter:

```
OVRDSPF DSPF(DBCSDSPF) IGCEXNCHR(*YES)
```

### Notes:

1. The system ignores the IGCEXNCHR parameter when processing alphanumeric files.
2. When you use the Japanese 5583 Printer to print extended characters, you must use the Kanji print function of the Advanced DBCS Printer Support licensed program.

## What happens when extended characters are not processed

When extended characters are not processed, the following events happen:

- Basic double-byte characters are displayed and printed.
- On displays, the system displays the undefined character where it would otherwise display extended characters.
- On printed output, the system prints the undefined character where it would otherwise print extended characters.
- The extended characters, though not displayed or printed, are stored correctly in the system.

### Related reference

[Create Display File \(CRTDSPF\) command](#)

[Create Printer File \(CRTPRPF\) command](#)

[Change Display File \(CHGDSPF\) command](#)

[Change Printer File \(CHGPRPF\) command](#)

[Override with Display File \(OVRDSPF\) command](#)

[Override with Printer File \(OVRPRTF\) command](#)

### Display support

The IBM i operating system inserts shift-control characters into DBCS-only fields automatically. The use of DBCS input fields affects the total number of input fields allowed on a display. Alphanumeric displays cannot display double-byte data correctly.

## Inserting shift-control characters

The system inserts shift-control characters into DBCS-only fields automatically.

To insert shift-control characters into open fields or either fields, follow these steps:

1. Position the cursor in the field in which you want to insert double-byte data.
2. Press the Insert Shift Control Character key (according to your DBCS display user's guide).

The system inserts a pair of shift-control characters at the same time. The system leaves the cursor under the shift-in character and puts the keyboard in insert mode. Insert double-byte characters between the shift-control characters.

To find out if a field already has the shift-control characters, press the Display Shift Control Character key.

DBCS-graphic fields store double-byte characters without requiring the use of shift control characters. Shift control characters should not be inserted in graphic fields.

### **Number of displayed extended characters**

The system can display up to 512 different extended characters on a Japanese display at one time. Additional extended characters are displayed as undefined characters. However, the additional extended characters are stored correctly in the system.

### **Number of input fields on a display**

The use of DBCS input fields affects the total number of input fields allowed on a display. For a local 5250 display, you can specify as many as 256 input fields. However, each three instances of a DBCS field reduces the maximum number of fields by one. For example, if there are nine DBCS fields on a display, then the maximum is  $256 - (9/3) = 253$  input fields.

### **Effects of displaying double-byte data at alphanumeric workstations**

Alphanumeric displays cannot display double-byte data correctly. If you try to display double-byte data at an alphanumeric display, the following happens:

- The system sends an inquiry message to that display, asking whether you want to continue using the program with double-byte data or to cancel it.
- If you continue using the program, the system ignores the shift-control characters and interprets the double-byte characters as though they were single-byte characters. The displayed double-byte data does not make sense.

### ***Making printer files DBCS capable***

The system uses printer files to process data that will be printed or displayed. When the data involved contains double-byte characters, the printer file that is used to process the data must be capable of processing double-byte data.

In many cases, printer files are used by the system to produce data that will eventually be printed or displayed. In these cases, the data is first placed into a spooled file using one of the IBM-supplied printer files. The data is then taken from the spooled file and is displayed or printed based on the user's request.

When the data involved contains double-byte characters, the printer file that is used to place the data into the spooled file must be capable of processing double-byte data. A printer file is capable of processing double-byte data when \*YES is specified on the IGCDTA parameter for the file. In most cases, the system recognizes the occurrence of double-byte data and takes appropriate measures to ensure the printer file that is used is capable of processing double-byte data.

In some cases, however, the system cannot recognize the occurrence of double-byte data and might attempt to use a printer file that is not capable of processing double-byte data. If this occurs, the output at the display or printer may not be readable. This can happen when object descriptions containing double-byte characters are to be displayed or printed on an alphanumeric device.

To ensure that you receive correct results when you display or print double-byte characters, some recommendations should be followed. Action is required on your part if you have a single-byte national language installed as a secondary language. Printer files that are received as part of the DBCS version of a product are always capable of processing DBCS data.

You should complete the following recommended actions after the product or feature has been installed:

1. If all printers and display devices attached to your system are DBCS-capable, you can enable all printer files for double-byte data. For IBM-supplied printer files that are received as part of a single-byte secondary language feature, you can enable all printer files by issuing the following command:

```
CHGPRTF FILE(*ALL/*ALL) IGCDTA(*YES)
```

After this command has been completed, all printer files in all libraries will be enabled for double-byte data. The change will be permanent.

2. If all printer and display devices that are attached to your system are not DBCS-capable, you should not enable all IBM-supplied printer files.

Instead, use the library search capabilities of the system to control which printer files will be used for any particular job. When the potential exists that double-byte data will be encountered, the library list for the job should be such that the printer files that are DBCS-enabled will be found first in the library list. Conversely, if only single-byte data is expected to be encountered, the library list should be set up so the printer files that are not enabled for DBCS will be found first. In this way, the printer file capabilities will match the type of data that will be processed. The decision as to what type of printer file to use is made on the basis of what type of data will be processed. The device that will be used to actually display or print the data may also influence this decision.

In some cases it may be desirable to make the printer file only temporarily DBCS-capable instead of making a permanent change. For a specific job, you can make this temporary change by using the OVRPRTF command.

To temporarily enable a specific printer file, you can use the following command:

```
OVRPRTF FILE(filename) IGCDTA(*YES)
```

Where *filename* is the name of the printer file you want to enable.

### Related reference

[Override with Printer File \(OVRPRTF\) command](#)

### Copying spooled and nonspooled DBCS files

When you copy spooled files to a database file, the database file must have been created with the IGCDTA(\*YES) value specified. You can use the Copy File (CPYF) command to copy double-byte data from one file to another.

An extra column is reserved for the shift-out character, if the database file contains double-byte data.

### Copying spooled files

Copy spooled files that have double-byte data by using the Copy Spooled File (CPYSPLF) command. However, the database file to which the file is being copied must have been created with the IGCDTA(\*YES) value specified.

When copying spooled files to a database file that contains double-byte data, an extra column is reserved for the shift-out character. This shift-out character is placed between the control information for the record and the user data. The following table shows the shift-out character column number, based on the value specified for the Control character (CTLCHAR) keyword:

CTLCHAR value	Column for shift-out character
*NONE	1
*FCFC	2
*PRTCTL	5
*S36FMT	10

## Copying nonspooled DBCS files

You can use the Copy File (CPYF) command to copy double-byte data from one file to another.

When you copy data from a double-byte database file to an alphanumeric database file, specify one of the following parameters on the CPYF command:

- If both files are source files or if both files are database files, you can specify either the FMTOPT(\*MAP) parameter or the FMTOPT(\*NOCHK) parameter.
- If one file is a source file and the other file is a database file, specify the FMT(\*CVTSRC) parameter.

When you copy DBCS files to alphanumeric files, the system sends you an informational message describing the difference in file types.

Either the FMTOPT(\*MAP) or FMTOPT(\*NOCHK) option of the copy file function must be specified for copies from a physical or logical file to a physical file when there are fields with the same name in the from-file and to-file, but the data type for fields is as shown in the following table:

From-file field data type	To-file field data type
A (character)	J (DBCS-only)
O (DBCS-open)	J (DBCS-only)
O (DBCS-open)	E (DBCS-either)
E (DBCS-either)	J (DBCS-only)
J (DBCS-only)	G (DBCS-graphic)
O (DBCS-open)	G (DBCS-graphic)
E (DBCS-either)	G (DBCS-graphic)
G (DBCS-graphic)	J (DBCS-only)
G (DBCS-graphic)	O (DBCS-open)
G (DBCS-graphic)	E (DBCS-either)

When you use FMTOPT(\*MAP) on the CPYF command to copy data to a DBCS-only field or DBCS-graphic field, the corresponding field in the from-file must not be:

- Less than a 2-byte character field
- An odd-byte-length character field
- An odd-byte-length DBCS-open field

If you attempt to copy with one of these specified in the from-field, an error message is sent.

When you copy double-byte data from one database file to another with the FMTOPT(\*MAP) parameter specified, double-byte data is copied correctly. The system performs correct padding and truncation of double-byte data to ensure data integrity.

When using the CPYF command with FMTOPT(\*MAP) to copy a DBCS-open field to a graphic field, a conversion error occurs if the DBCS-open field contains any SBCS data (including blanks).

### Related reference

[Copy Spooled File \(CPYSPLF\) command](#)

[Copy File \(CPYF\) command](#)

### Changing alphanumeric programs to DBCS programs

If an alphanumeric application program uses externally described files, you can change that application program to a DBCS application program by changing the externally described files.

To convert an application program, follow these steps:

1. Create a duplicate copy of the source statements for the alphanumeric file that you want to change.

2. Change alphanumeric constants and literals to double-byte constants and literals.
3. Change fields in the file to the open (O) data type or specify the Alternative Data Type (IGCALTTYP) DDS keyword so that you can enter both double-byte and alphanumeric data in these fields.  
You might want to change the length of the fields as the double-byte data takes more space.
4. Store the converted file in a separate library. Give the file the same name as its alphanumeric version.
5. When you want to use the changed file in a job, change the library list, using the Change Library List (CHGLIBL) command, for the job in which the file is used.  
The library in which the DBCS display file is stored is then checked before the library in which the alphanumeric version of the file is stored.

### **Related concepts**

[IGCALTTYP \(Alternative Data Type\) keyword](#)

### **Related reference**

[Change Library List \(CHGLIBL\) command](#)

### **Entering DBCS text in CL commands**

You can use double-byte character data anywhere in a CL command that descriptive text can be used.

Enter double-byte character text as follows:

1. Begin the double-byte character text with a single quotation mark (').
2. Enter a shift-out character.
3. Enter the double-byte character text.
4. Enter a shift-in character.
5. End the double-byte character text with a single quotation mark (').

For example, to enter the double-byte character literal ABC, enter the following characters, where SO represents the shift-out character and SI represents the shift-in character:

```
'SOABCSI'
```

Limit the length of a double-byte character text description of an object to 14 double-byte characters, plus the shift control characters, to make sure that the description is properly displayed and printed.

### **DBCS conversion**

You can enter an alphanumeric entry or DBCS code and convert the entry or code to its related DBCS word. DBCS conversion is intended for Japanese character sets and its use is limited for application to other double-byte character sets.

When you use DBCS displays to enter double-byte data, you can use the various data entry methods supported on the display, or you can choose to use the system DBCS conversion support.

Specifically, you can convert the following characters:

- A string of alphanumeric characters to a DBCS word
- English alphanumeric characters to double-byte alphanumeric characters
- Alphanumeric Katakana to double-byte Hiragana and Katakana letters
- A DBCS code to its corresponding double-byte character
- A DBCS number to its corresponding double-byte character

### *Conversion dictionaries*

The DBCS conversion dictionary is a collection of alphanumeric entries and their related DBCS words. The system refers to the dictionary when performing DBCS conversion.

All DBCS conversion dictionaries have an object type of \*IGCDCT. A system-supplied and a user-created dictionary are used with DBCS conversion.

## User-created dictionary

A user-created dictionary contains any alphanumeric entries and related DBCS words that you choose to include. You might create a user dictionary to contain words unique to your business or words that you use regularly but that are not included in the system-supplied dictionary.

You can create one or more DBCS conversion dictionaries with any name and store them in any library. When performing DBCS conversion, however, the system only refers to the first user dictionary named QUSRIGCDCT in the user's library list, no matter how many dictionaries you have or what they are named. Make sure that the library list is properly specified so that the system checks the correct dictionary.

During DBCS conversion, the system checks QUSRIGCDCT before checking QSYSIGCDCT.

## DBCS conversion dictionary commands

You can use the following commands to perform object management functions with the DBCS conversion dictionary. Specify the OBJTYPE(\*IGCDCT) parameter when entering these commands:

- CHGOBJOWN: Change the owner of a DBCS conversion dictionary
- CHKOBJ: Check a DBCS conversion dictionary
- CRTDUPOBJ: Create a duplicate object of the dictionary
- DMPOBJ: Dump a DBCS conversion dictionary
- DMPSYSOBJ: Dump the system-supplied dictionary
- DSPOBJAUT: Display a user's authority to the dictionary
- GRTOBJAUT: Grant authority to use the dictionary
- MOVOBJ: Move the dictionary to another library
- RNMOBJ: Rename the dictionary
- RSTOBJ: Restore the dictionary
- RVKOBJAUT: Revoke authority to use the dictionary
- SAVOBJ: Save the dictionary
- SAVCHGOBJ: Save a changed dictionary

The system saves or restores DBCS conversion dictionaries when you use these commands:

- RSTLIB: Restore a library in which the dictionary is stored
- SAVLIB: Save a library in which the dictionary is stored
- SAVSYS: Save QSYSIGCDCT, the system DBCS conversion dictionary, when saving the system

You can use the following commands to create, edit, display, and delete a dictionary:

- CRTIGCDCT: Create DBCS Conversion Dictionary
- EDTIGCDCT: Edit DBCS Conversion Dictionary
- DSPIGCDCT: Display DBCS Conversion Dictionary
- DLTIGCDCT: Delete DBCS Conversion Dictionary

## Related reference

[Change Object Owner \(CHGOBJOWN\) command](#)

[Check Object \(CHKOBJ\) command](#)

[Create Duplicate Object \(CRTDUPOBJ\) command](#)

[Dump Object \(DMPOBJ\) command](#)

[Dump System Object \(DMPSYSOBJ\) command](#)

[Display Object Authority \(DSPOBJAUT\) command](#)

[Grant Object Authority \(GRTOBJAUT\) command](#)

[Move Object \(MOVOBJ\) command](#)

[Rename Object \(RNMOBJ\) command](#)

[Restore Object \(RSTOBJ\) command](#)  
[Revoke Object Authority \(RVKOBJAUT\) command](#)  
[Save Object \(SAVOBJ\) command](#)  
[Save Changed Objects \(SAVCHGOBJ\) command](#)  
[Restore Library \(RSTLIB\) command](#)  
[Save Library \(SAVLIB\) command](#)  
[Save System \(SAVSYS\) command](#)  
[Create DBCS Conversion Dict \(CRTIGCDCT\) command](#)  
[Edit DBCS Conversion Dict \(EDTIGCDCT\) command](#)  
[Display DBCS Conversion Dict \(DSPIGCDCT\) command](#)  
[Delete DBCS Conversion Dict \(DLTIGCDCT\) command](#)

*Working with DBCS conversion dictionaries*

You can create, edit, display, print, and delete conversion dictionaries.

### **Creating a DBCS conversion dictionary**

To create a DBCS conversion dictionary, follow these steps:

1. Use the Create DBCS Conversion Dictionary (CRTIGCDCT) command.
2. Name the dictionary, QUSRIGCDCT, so it can be used during DBCS conversion. The system uses the dictionary if it is the first user-created dictionary found when searching a user's library list.

You might call the dictionary by another name while it is being created to prevent application programs from using it for conversion. Later, change the dictionary name using the Rename Object (RNMOBJ) command.

3. Use the EDTIGCDCT command to put entries and related words into the dictionary after creating it.

### **Editing a DBCS conversion dictionary**

Use the Edit DBCS conversion dictionary (EDTIGCDCT) command to edit the DBCS conversion dictionary. Use editing to add user-defined characters to the dictionary, so that users can enter characters using DBCS conversion, and rearrange terms in a DBCS conversion dictionary to suit individual needs.

The display needed for use while editing the DBCS conversion dictionary depends on the value that you entered for the ENTRY parameter on the EDTIGCDCT command:

- If you specified a specific string with the ENTRY parameter or if you want to display double-byte characters, you must use a DBCS display.
- If you did not specify a specific string with the ENTRY parameter, or if you do not want to display double-byte characters, use either a DBCS display, or a 24-row by 80-column alphanumeric display.

You can perform the following editing operations on a user-created DBCS conversion dictionary:

- Add entries to the dictionary (including adding the first entries to the dictionary after it is created). The dictionary can contain as many as 99 999 entries.
- Delete entries from the dictionary.
- Change entries in the dictionary, such as replacing the DBCS words related to an alphanumeric entry.
- Move the DBCS words related to an alphanumeric entry to rearrange the order in which they appear during DBCS conversion.

The only editing function that you can perform with QSYSIGCDCT, the system-supplied dictionary, is to move DBCS words related to an alphanumeric entry. Move words in order to rearrange the order in which they appear during DBCS conversion.



## Displaying and printing the DBCS conversion dictionary

Use the Display DBCS Conversion Dictionary (DSPIGCDCT) command to display and print the DBCS conversion dictionary. You can display or print the entire dictionary or just a certain part of it, depending on the value you specify for the ENTRY parameter. For example, to print the entry ABC from the dictionary QUSRIGCDCT and its related words, enter the following command:

```
DSPIGCDCT IGCDCT(DBCSLIB/QUSRIGCDCT) +  
ENTRY(ABC) OUTPUT(*PRINT)
```

To display all of the entries from the system-supplied dictionary QSYSIGCDCT and their related words, enter the following command:

```
DSPIGCDCT IGCDCT(QSYS/QSYSIGCDCT)
```

## Deleting a DBCS conversion dictionary

Use the Delete DBCS Conversion Dictionary (DLTIGCDCT) command to delete a DBCS conversion dictionary from the system. In order to delete the dictionary, you must have object existence authority to the dictionary and object operational authorities to the library in which the dictionary is stored.

When you delete a dictionary, make sure that you specify the correct library name. It is possible that many users have their own dictionaries, each named QUSRIGCDCT, stored in their libraries. If you do not specify any library name, the system deletes the first DBCS conversion dictionary in your library list.

### Related reference

[Create DBCS Conversion Dictionary \(CRTIGCDCT\) command](#)

[Edit DBCS Conversion Dictionary \(EDTIGCDCT\) command](#)

[Display DBCS Conversion Dictionary \(DSPIGCDCT\) command](#)

[Delete DBCS Conversion Dictionary \(DLTIGCDCT\) command](#)

### *Japanese DBCS conversion*

DBCS conversion is intended for Japanese character sets and its use is limited for application to other double-byte character sets.

When you use DBCS displays to enter double-byte data, you can use the various data entry methods supported on the display, or you may choose to use the IBM i DBCS conversion support. DBCS conversion lets you enter an alphanumeric entry or DBCS code and convert the entry or code to its related DBCS word.

Specifically, DBCS conversion lets you convert the following characters:

- A string of alphanumeric characters to a DBCS word
- English alphanumeric characters to double-byte alphanumeric characters
- Alphanumeric Katakana to double-byte Hiragana and Katakana letters
- A DBCS code to its corresponding double-byte character
- A DBCS number to its corresponding double-byte character

## Japanese system-supplied dictionary

The QSYSIGCDCT is the system-supplied dictionary that is stored in the library, QSYS. It is a collection of entries with a Japanese pronunciation, expressed in alphanumeric characters, and the DBCS words related to those entries. The system checks this dictionary second when performing DBCS conversion.

QSYSIGCDCT contains these entries:

- Personal names
  - Family names

- First names
- Organization names
  - Private enterprises registered in the security market
  - Public corporations
  - Typical organizations in the central and local governments
  - Most universities and colleges
- Addresses
  - Public administration units within the prefectures
  - Towns and streets in 11 major cities
- Business terms, such as department names and position titles commonly used in enterprises
- Individual double-byte characters, including basic double-byte characters, as defined by IBM

You cannot add or delete entries from this dictionary. However, you may rearrange the related DBCS words so that the words used most frequently are displayed first during DBCS conversion.

### ***SQL and DBCS***

The basic symbols of keywords and operators in the SQL language are single-byte characters that are part of all character sets supported by the IBM relational database products. Characters of the language are classified as letters, digits, or special characters.

### **SQL host identifiers and double-byte characters**

A host-identifier is a name declared in the host program. The rules for forming a host-identifier are the rules of the host language, except that DBCS characters cannot be used.

### **SQL character subtypes and double-byte characters**

Each character string is further defined as follows:

- **Bit data:**

Data that is not associated with a coded character set and is never converted. The CCSID for bit data is 65535.

- **SBCS data:**

Data in which every character is represented by a single byte. Each SBCS data character string has an associated CCSID. If necessary, an SBCS data character string is converted before it is used in an operation with a character string that has a different CCSID.

- **Mixed data:**

Data that contains a mixture of characters from a single-byte character set (SBCS) and a double-byte character set (DBCS). Each mixed data character string has an associated CCSID. If necessary, a mixed data character string is converted before an operation with a character string that has a different CCSID. If mixed data contains a DBCS character, it cannot be converted to SBCS data.

The database manager does not recognize subclasses of double-byte characters, and it does not assign any specific meaning to particular double-byte codes. However, if you choose to use mixed data, then two single-byte EBCDIC codes are given special meanings:

- X'0E', the shift-out character, is used to mark the beginning of a sequence of double-byte codes.
- X'0F', the shift-in character, is used to mark the end of a sequence of double-byte codes.

In order for the database manager to recognize double-byte characters in a mixed data character string, the following condition must be met:

- Within the string, the double-byte characters must be enclosed between paired shift-out and shift-in characters.

The pairing is detected as the string is read from left to right. The code X'0E' is recognized as a shift out character if X'0F' occurs later; otherwise, it is invalid. The first X'0F' following the X'0E' that is on a double-byte boundary is the paired shift-in character. Any X'0F' that is not on a double-byte boundary is not recognized.

There must be an even number of bytes between the paired characters, and each pair of bytes is considered to be a double-byte character. There can be more than one set of paired shift-out and shift-in characters in the string.

The length of a mixed data character string is its total number of bytes, counting two bytes for each double-byte character and one byte for each shift-out or shift-in character.

When the job CCSID indicates that DBCS is allowed, CREATE TABLE will create character columns as DBCS-Open fields, unless FOR BIT DATA, FOR SBCS DATA, or an SBCS CCSID is specified. The SQL user will see these as character fields, but the system database support will see them as DBCS-Open fields.

#### *SQL graphic strings*

A graphic string is a sequence of double-byte characters that do not include shift-out or shift-in characters.

The length of the string is the number of its characters. Like character strings, graphic strings can be empty.

Every graphic string has a CCSID that identifies a double-byte coded character set. If necessary, a graphic string is converted before it is used in an operation with a graphic string that has a different CCSID.

#### **SQL fixed-length and double-byte characters**

All values of a fixed-length graphic-string column have the same length, which is determined by the length attribute of the column. The length attribute must be between 1 through 16383 inclusive.

#### **SQL graphic-string constants**

A graphic-string constant is a varying-length graphic string. The length of the specified string cannot be greater than 16370.

In the normal form, the SQL delimiters and the G or the N are SBCS characters. The SBCS apostrophe (') is the EBCDIC apostrophe, X'7D'.

In the PL/I form, the apostrophes and the G are DBCS characters. Two consecutive DBCS string delimiters are used to represent one string delimiter within the string. Notice that this PL/I form is only valid for static statements embedded in PL/I programs.

A hexadecimal graphic constant is also supported. The form of the hexadecimal graphic constant is:

```
GX' ssss '
```

In the constant, **ssss** represents a string from 0 to 32766 hexadecimal digits. The number of characters between the string delimiters must be an even multiple of 4. Each group of 4 digits represents a single graphic character. The hexadecimal for shift-in and shift-out (X'0E' and X'0F') are not included in the string.

The CCSID assigned to constants is the DBCS CCSID associated with the CCSID of the source unless the source is encoded in a foreign encoding scheme (such as ASCII). In this case, the CCSID assigned to the constant is the DBCS CCSID associated with the default CCSID of the application server when the SQL statement containing the constant is prepared. If there is no DBCS CCSID associated with the CCSID of the source, the CCSID is 65535.

#### *SQL assignments and comparisons*

The basic operations of SQL are assignment and comparison. Assignment operations are performed during the running of CALL, INSERT, UPDATE, FETCH, and SELECT INTO statements. Comparison

operations are performed during the running of statements that include predicates and other language elements such as MAX, MIN, DISTINCT, GROUP BY, and ORDER BY.

The basic rule for both operations is that the data type of the operands involved must be compatible. The compatibility rule also applies to UNION, concatenation, and the VALUE, COALESCE, MIN, and MAX scalar functions.

### SQL string assignments and double-byte characters

The basic rule for string assignments is that the length of a string assigned to a column must not be greater than the length attribute of the column. (Trailing blanks are normally included in the length of the string. For string assignment, however, trailing blanks are not included in the length of the string.)

If the string contains mixed data, the assignment rules may require truncation within a sequence of double-byte codes. To prevent the loss of the shift-in character that ends the double-byte sequence, additional characters might be truncated from the end of the string, and a shift-in character added. In the truncated result, there is always an even number of bytes between each shift-out character and its matching shift-in character.

Character, DBCS-only, DBCS-open, and DBCS-either are not compatible with graphic types for assignment.

#### SQL conversion rules

When two strings are compared, one of the strings is first converted, if necessary, to the coded character set of the other string.

Character conversion is necessary only if all of the following conditions are true:

- The CCSIDs of the two strings are different.
- Neither CCSID is 65535.
- The string selected for conversion is neither null nor empty.
- The CCSID conversion selection table indicates that conversion is necessary.

If one string has an SBCS CCSID and the other is the same type of operand and has a mixed data CCSID, the SBCS data character string is converted. Otherwise, the string selected for conversion depends on the type of each operand. The following table shows which operand is selected for conversion, given the operand types.

First operand	Column value (second operand)	Derived value (second operand)	Special register (second operand)	Constant (second operand)	Host variable (second operand)
Column value	Second	Second	Second	Second	Second
Derived Value	First	Second	Second	Second	Second
Special Register	First	First	Second	Second	Second
Constant	First	First	First	Second	Second
Host Variable	First	First	First	First	Second

A host variable containing data in a foreign encoding scheme is always effectively converted to the native encoding scheme before it is used in any operation. The previous rules are based on the assumption that this conversion has already occurred.

An error occurs if a character of the string cannot be converted or the CCSID Conversion Selection Table is used but does not contain any information about the pair of CCSIDs. A warning occurs if a character of the string is converted to the substitution character.

## DBCS code schemes

IBM supports two DBCS code schemes: one for the host systems and the other for personal computers.

The DBCS code scheme for host systems has the following code-range characteristics:

- First byte: hex 41 to hex FE
- Second byte: hex 41 to hex FE
- Double-byte blank: hex 4040

## Shift-control characters

When the host code scheme is used, the system uses shift-control characters to identify the beginning and end of a string of double-byte characters. The shift-out (SO) character, hex 0E, indicates the beginning of a double-byte character string. The shift-in (SI) character, hex 0F, indicates the end of a double-byte character string.

Each shift-control character occupies the same amount of space as one alphanumeric character. By contrast, double-byte characters occupy the same amount of space as two alphanumeric characters.

When double-byte characters are stored in a graphic field or a variable of graphic data type, there is no need to use shift control characters to surround the double-byte characters.

## Incorrect and undefined double-byte code

Incorrect double-byte code has a double-byte code value that is not in the valid double-byte code range. This is in contrast to undefined double-byte code where the double-byte code is valid, but no graphic symbol has been defined for the code.

## Supported DBCS code ranges

The IBM i operating system supports the following DBCS character-set code ranges:

- Japanese character-set code range
- Korean character-set code range
- Simplified Chinese character-set code range
- Traditional Chinese character-set code range

## Related concepts

[Double-byte character set support](#)

[Database file management](#)

## DBCS font tables

DBCS font tables contain the images of the double-byte extended characters used on the IBM i operating system. The system uses these images to display and print extended characters when they are not resident on the device.

The following DBCS font tables are objects that you can save or restore. These font tables are distributed with the DBCS national language versions of the IBM i licensed program:

### QIGC2424

A Japanese DBCS font table used to display and print extended characters in a 24-by-24 dot matrix image. The system uses the table with Japanese displays, printers attached to displays, 5227 Model 1 Printer, and the 5327 Model 1 Printer.

### QIGC2424C

A Traditional Chinese DBCS font table used to print extended characters in a 24-by-24 dot matrix image. The system uses the table with the 5227 Model 3 Printer and the 5327 Model 3 Printer.

### QIGC2424K

A Korean DBCS font table used to print extended characters in a 24-by-24 dot matrix image. The system uses the table with the 5227 Model 2 Printer and the 5327 Model 2 Printer.

**QIGC2424S**

A Simplified Chinese DBCS font table used to print extended characters in a 24-by-24 dot matrix image. The system uses the table with the 5227 Model 5 Printer.

**QIGC3232**

A Japanese DBCS font table used to print characters in a 32-by-32 dot matrix image. The system uses the table with the 5583 Printer and the 5337 Model 1 Printer.

**QIGC3232S**

A Simplified Chinese DBCS font table used to print characters in a 32-by-32 dot matrix image. The system uses the table with the 5337 Model R05 Printer.

All DBCS font tables have an object type of \*IGCTBL. You can find instructions for adding user-defined characters to DBCS font tables in the *ADTS/400: Character Generator Utility*, SC09-1769-00 book.

**DBCS font table commands**

The following commands allow you to manage and use DBCS font tables:

- Check DBCS Font Table (CHKIGCTBL)
- Copy DBCS Font Table (CPYIGCTBL)
- Delete DBCS Font Table (DLTIGCTBL)
- Start Font Management Aid (STRFMA)

**Locating an existing font table**

Use the Check DBCS Font Table (CHKIGCTBL) command to find out if a DBCS font table exists in your system.

For example, to find out if the table QIGC2424 exists, enter:

```
CHKIGCTBL IGCTBL(QIGC2424)
```

If the table does not exist, the system responds with a message. If the table does exist, the system returns without a message.

Check for the existence of a table when adding a new type of DBCS workstation to make sure that the table used by the device exists in the system.

**Related reference**

[Check DBCS Font Table \(CHKIGCTBL\) command](#)

[Copy DBCS Font Table \(CPYIGCTBL\) command](#)

[Delete DBCS Font Table \(DLTIGCTBL\) command](#)

[Start Font Management Aid \(STRFMA\) command](#)

**Copying a DBCS font table**

You can use the Copy DBCS Font Table (CPYIGCTBL) command to copy a DBCS font table to or from a tape, diskette, or physical file.

The DBCS font tables are saved when you use the Save System (SAVSYS) command so you do not have to use the CPYIGCTBL command when performing normal system backup.

A physical file used to save and restore table information must have a minimum record length of 74 bytes.

**Copying a table onto a tape, a diskette, or a physical file**

You should copy a DBCS font table onto a tape, a diskette, or a physical file in the following instances:

- Before deleting that table
- After new user-defined characters are added to the tables

- When planning to use the tables on another system

To copy a DBCS font table onto a tape, a diskette, or a physical file, follow these steps:

1. If copying a DBCS font table onto a tape or diskettes, make sure that the tape or diskettes are initialized to the \*DATA format. If necessary, initialize the tape or diskettes by specifying the FMT(\*DATA) parameter on the Initialize Diskette (INZDKT) command.
2. Load the initialized tape or diskette onto the system.
3. Enter the CPYIGCTBL command as follows:
  - a. Choose the value OPTION(\*OUT).
  - b. Use the DEV parameter to select the device to which you want to copy the table. A value of \*FILE specifies that the DBCS font table is saved to a physical file.
  - c. Use the SELECT and RANGE parameters to specify which portion of the table you want copied from the system.
4. Press the Enter key. The system copies the DBCS font table onto the specified medium or into a physical file.
5. Remove the tape or the diskette after the system finishes copying the table.

### **Copying a DBCS font table from a tape, a diskette, or a physical file**

Use the Copy DBCS Font Table (CPYIGCTBL) command to copy a DBCS font table from a tape, a diskette, or a physical file onto the system. The system automatically creates the DBCS font table again when copying its contents if the following conditions are true:

- The specified table does not already exist in the system.
- The medium or physical file from which you are copying the table contains all of the IBM-defined double-byte characters.
- SELECT(\*ALL) or SELECT(\*SYS) is specified on the CPYIGCTBL command.

### **Related reference**

[Copy DBCS Font Table \(CPYIGCTBL\) command](#)

[Save System \(SAVSYS\) command](#)

### **Deleting a DBCS font table**

You can delete an unused DBCS font table to free storage space on your system. Use the Delete DBCS Font Table (DLTIGCTBL) command to delete a DBCS font table from the IBM i operating system.

For example, if you do not plan to use Japanese printer 5583 or 5337 with your system, font table QIGC3232 is not needed and can be deleted.

To delete a table, follow these steps:

1. Optional: Copy the table onto a tape, a diskette, or a physical file. If you do not copy the table before deleting it, you will not have a copy of the table for future use.
2. Vary off all devices using that table.
3. Enter the DLTIGCTBL command.

For example, to delete the DBCS font table QIGC3232, enter:

```
DLTIGCTBL IGCTBL(QIGC3232)
```

4. Press the Enter key.  
The system sends an inquiry message to the system operator message queue for you to confirm your intention to delete a DBCS table.
5. Respond to the inquiry message.  
The system sends you a message when it has deleted the table.

**Note:** Do not delete a DBCS font table if any device using that table is currently varied on. Also, make sure that the affected controller is not varied on. If you try to delete the table while the device and controller are varied on, the system reports any devices attached to the same controller as those devices, and the controller as damaged the next time you try to print or display extended characters on an affected device. If such damage is reported, follow these steps:

- a) Vary off the affected devices, using the Vary Configuration (VRYCFG) command.
- b) Vary off the affected controller.
- c) Vary on the affected controller.
- d) Vary on the affected devices.
- e) Continue normal work.

### **Related reference**

[Delete DBCS Font Table \(DLTIGCTBL\) command](#)

[Vary Configuration \(VRYCFG\) command](#)

### **DBCS font files**

The IBM i operating system provides DBCS font files. *DBCS font files* are physical files that contain frequently used double-byte characters. When using the character generator utility, you can use the characters in these files as the base for a new, user-defined character.

These files are supplied with read-only authority as they are not to be changed. If you do not use character generator utility or the Advanced DBCS Printer Support licensed program, you can delete these files to save space. They all exist in the QSYS library.

The following DBCS font files are distributed with the DBCS national language versions of the IBM i licensed program. They are used as a reference for the character generator utility (CGU) and the Advanced DBCS Printer Support licensed program.

#### **QCGF2424**

A Japanese DBCS font file used to store a copy of the Japanese DBCS basic character images.

#### **QCGF2424K**

A Korean DBCS font file used to store a copy of the Korean DBCS basic character images.

#### **QCGF2424C**

A Traditional Chinese DBCS font file used to store a copy of the Traditional Chinese DBCS basic character images.

#### **QCGF2424S**

A Simplified Chinese DBCS font file used to store a copy of the Simplified Chinese DBCS basic character images.

### **DBCS sort tables**

DBCS sort tables contain the sort information and collating sequences of all the double-byte characters used on the system. The sort utility on the system uses these tables to sort double-byte characters.

DBCS sort tables are objects that you can save, restore and delete. Using the character generator utility you can also add, delete and change entries in these tables corresponding to the image entries in the DBCS font tables. For Japanese use only, you can also copy the DBCS master sort table to and from a data file.

The following DBCS sort tables are distributed with the DBCS national language versions of the IBM i licensed program:

#### **QCGMSTR**

A Japanese DBCS master sort table used to store the sort information for the Japanese double-byte character set.

#### **QCGACTV**

A Japanese DBCS active sort table used to store the sort collating sequences for the Japanese double-byte character set.



**QCGMSTRC**

A Traditional Chinese DBCS master sort table used to store the sort information for the Traditional Chinese double-byte character set.

**QCGACTVC**

A Traditional Chinese DBCS active sort table used to store the sort collating sequences for the Traditional Chinese double-byte character set.

**QCGACTVK**

A Korean DBCS active sort table used to map Hanja characters to Hangeul characters with equivalent pronunciation.

**QCGMSTRS**

A Simplified Chinese DBCS master sort table used to store the sort information for the Simplified Chinese double-byte character set.

**QCGACTVS**

A Simplified Chinese DBCS active sort table used to store the sort collating sequences for the Simplified Chinese double-byte character set.

You can sort Japanese, Korean, Simplified Chinese, and Traditional Chinese double-byte characters. Each of these languages have two DBCS sort tables, a DBCS master sort table and a DBCS active sort table, except for Korean which has only a DBCS active sort table. The DBCS master sort table contains sort information for all defined DBCS characters. The DBCS active sort table for Japanese, Simplified Chinese, and Traditional Chinese is created from the master sort table information and contains the collating sequences for the double-byte characters of that given language. These collating sequences have a purpose similar to the EBCDIC and ASCII collating sequences for the single-byte alphanumeric character set. For Korean characters, the Hangeul characters are assigned both their collating sequence as well as their DBCS codes according to their pronunciation. Hence, a separate collating sequence is not required, and each of the Hanja characters is mapped to a Hangeul character of the same pronunciation using the DBCS active sort table QCGACTVK.

All DBCS sort tables have an object type of \*IGCSRT.

**Commands for DBCS sort tables**

The following commands allow you to manage and use DBCS sort tables.

- Check Object (CHKOBJ)
- Save Object (SAVOBJ)
- Restore Object (RSTOBJ)

**Using existing DBCS sort tables**

You can save the tables to a tape or diskette, delete them from the system, and restore them to the system. You can also add sort information for each user-defined character, and add that character to the DBCS collating sequence, as you create it using the character generator utility.

**Finding existing DBCS sort table**

Use the Check Object (CHKOBJ) command to find out if a DBCS sort table exists on your system. For example, to find out if the table QCGMSTR exists, enter the following command:

```
CHKOBJ OBJ(QSYS/QCGMSTR) OBJTYPE(*IGCSRT)
```

If the table does not exist, the system responds with a message. If the table does exist, the system returns without a message.

Check for the existence of a DBCS active sort table when you want to sort double-byte characters for the first time. The DBCS active table for the DBCS language must exist to sort the characters.

**Related reference**

[Check Object \(CHKOBJ\) command](#)

[Save Object \(SAVOBJ\) command](#)

[Restore Object \(RSTOBJ\) command](#)

### ***Saving and restoring a DBCS sort table***

To save DBCS sort tables to a tape or diskette, use the Save Object (SAVOBJ) command. To restore DBCS sort tables from a tape or diskette, use the Restore Object (RSTOBJ) command.

### **Saving a DBCS sort table to tape or diskette**

Save a DBCS sort table onto tape or diskette in the following instances:

- Before deleting that table
- After information is added, updated, or changed in the tables using the character generator utility
- When planning to use the tables on another system

Use the Save Object (SAVOBJ) command to save a DBCS sort table onto tape or diskette. Specify \*IGCSRT for the object type.

The DBCS sort tables are saved when you use the SAVSYS command so you do not have to use the SAVOBJ command when performing normal system backup.

### **Restoring a DBCS sort table from tape or diskette**

Use the RSTOBJ command to restore a DBCS sort table from a tape or a diskette onto the system. The tables on the tape or diskette must previously have been saved using the SAVOBJ command. Specify \*IGCSRT for the object type. The system automatically re-creates the DBCS sort table when the specified table does not already exist in the system.

These tables must be restored to the QSYS library for the system to know they exist. For that reason, RSTOBJ restores \*IGCSRT objects only to the QSYS library and only if the objects do not already exist there.

### **Related reference**

[Save Object \(SAVOBJ\) command](#)

[Save System \(SAVSYS\) command](#)

[Restore Object \(RSTOBJ\) command](#)

### ***Deleting a DBCS sort table***

You can delete an unused DBCS sort table to free disk space, but you should always first save a copy of the table using the Save Object (SAVOBJ) command. Use the Delete DBCS Sort Table (DLTIGCSRT) command to delete a DBCS sort table from the system.

You should delete the DBCS master sort table for a DBCS language if either of the following conditions are true:

- You will not be creating any new characters for that language using the character generator utility.
- You will not be using the sort utility to sort characters for that language.

You should delete the DBCS active sort table for a DBCS language if you will not be using the sort utility to sort characters for that language. The DBCS active sort table must be on the system to use the sort utility for this language.

To delete a table, follow these steps:

1. Optional: Save the table onto tape or diskettes.

If you do not save the table onto removable media before deleting it, you will not have a copy of the table for future use.

2. Enter the DLTIGCSRT command. For example, to delete the DBCS sort table QCGACTV, enter:

```
DLTIGCSRT IGCSRT(QCGACTV)
```

3. Press the Enter key. The system sends you a message when it has deleted the table.

## Related reference

[Save Object \(SAVOBJ\) command](#)

## DBCS field definition

Consider the characteristics of DBCS data when you define a DBCS field.

- Each DBCS character is 2 bytes long.
- The length of a DBCS character string is always even.
- Shift-out (SO) and shift-in (SI) control characters are required at the beginning and end of the DBCS character string, except for graphic-data type fields. Together, these characters are 2 bytes long.
- The system treats DBCS data the same as character data, and therefore cannot perform arithmetic operations on it.
- The following DBCS data types can be used to identify DBCS fields:
  - J (DBCS-only) for fields that can contain only bracketed DBCS data.
  - E (DBCS-either) for fields that can contain bracketed DBCS or SBCS data, but not both.
  - O (DBCS-open) for fields that can contain both SBCS and bracketed DBCS data.
  - G (DBCS-graphic) for fields that can contain graphic data without the SO and SI control characters.

Data type O is allowed in all types of files. Data types J and E are allowed only in database and display files. Data type G is allowed in database, display, and printer files. In most cases, the operating system automatically inserts shift-out and shift-in characters. An exception is when data is written into a data type G field in a database file.

## Related reference

[DDS concepts](#)

## Related information

[Application Display Programming PDF](#)

## Working with locales

Locales are used primarily in ILE-based application programs. Additionally, the Retrieve Locale Information (OPM, QLGRTVLC; ILE, QlgRetrieveLocaleInformation) API retrieves one or all categories of a locale.

### Benefits of using locales in global applications

Applications can be created independent of language, cultural data, or specific characters. Locales can be accessed to provide this type of support to any integrated language environment-based application.

For example, the LC\_TIME category within a locale can be defined in any of the following ways, or in any combination that is convenient for the environment in which the application runs:

- HH:MM:SS
- MM:SS:HH
- SS:MM:HH

### Creating locales

Locales are created using the Create Locale (CRTLOCALE) command. The source file used to create the locale is named QLOCALESRC, in the QSYSLOCALE library. This library is loaded with option 21 of the operating system. These source files cannot be changed. Instead, they must be copied and then edited if changes are required.

## Related concepts

[Locales](#)

A *locale* is an object that can determine how data is processed, printed, and displayed.

#### Installing and enabling locales

If you are installing a new release, you can request that library QSYSLOCALE be installed on the system at that time.

#### System-supplied locales and recommended CCSIDs

The system-supplied locale source definition file members are in the optionally installable library QSYSLOCALE in the QLOCALESRC source file. The source file members are encoded in CCSID 37 and are read only.

#### Example: Creating and enabling a locale

This example contains the steps necessary for creating and enabling a locale.

#### **Related reference**

Application programming interfaces (APIs)

Create Locale (CRTLOCALE) command

#### **Related information**

Retrieve Locale Information (QLGRTVLC, QlgRetrieveLocaleInformation) API

#### **Locale restrictions**

Certain restrictions apply when you use locales to set job attributes.

- The locale CCSID must be an EBCDIC single-byte CCSID for an SBCS system.
- The locale CCSID must be an EBCDIC, single-byte character set (SBCS), or mixed-byte CCSID for a DBCS-capable system.
- The locale object must exist in the QSYS file system.
- The DATFMT, DATSEP, TIMSEP, and DECFMT parameters within the locale must be valid values supported as job attributes.
- If you want sort sequence support from the locale, you must use the CPYSYSCOL keyword.

#### **Related concepts**

Work management

LC\_COLLATE category

The LC\_COLLATE category defines character or string collation information. Within LC\_COLLATE you can specify a sort sequence to use using the cpysyscol keyword. The *cpysyscol* keyword value is used in place of the LC\_COLLATE category definitions.

#### **Locale categories**

This table describes the locale categories that are supported on the IBM i operating system.

<b>Locale category</b>	<b>Description</b>
LC_COLLATE	Defines character or string collation information
LC_CTYPE	Defines character classification, case conversion, and other character attributes.
LC_MESSAGES	Defines the format for affirmative and negative responses.
LC_MONETARY	Defines rules and symbols for formatting monetary numeric information.
LC_NUMERIC	Defines a list of rules and symbols for formatting non-monetary numeric information.
LC_TIME	Defines a list of rules and symbols for formatting time and date information.
LC_TOD	Defines rules for daylight saving time and time zone information.

**Note:** A locale source file cannot contain duplicate categories.

### Locale category source definitions

The category source definition consists of:

- The category header (*category name*), where the category name must be all uppercase characters.
- The associated keyword/value pairs that comprise the category body. Keywords can be all uppercase, all lowercase, or mixed case characters.
- The category trailer (which consists of END *category-name*)

For example:

```
LC_CTYPE
source for LC_CTYPE category
END LC_CTYPE
```

Lines preceding the first category header can be used to change the comment character and the escape character. The `comment_char` (the default is #) and `escape_char` (the default is \) keywords can be used to change these characters. The following example shows how to change the comment character and escape character to \* and / :

```
comment_char <asterisk>
escape_char <slash>
```

**Note:** This example uses symbolic names to represent the '\*' and '/' characters.

The source for all categories is specified using keywords, strings, character literals, and character symbols:

#### Keywords

Each keyword identifies either a definition or a rule. The remainder of the statement containing the keyword contains the operands to the keyword. Operands are separated from the keyword by one or more blank characters. A statement may be continued on the next line by placing an `escape_char` as the last character before the newline or linefeed character that ends the line.

Lines containing the `comment_char` in the first column are treated as comment lines. Comment lines cannot be continued on a subsequent line using an escape character. \ is the default escape character. However, the escape character can be defined to be any character by the user.

#### Strings

Strings must be enclosed in quotation marks. Quotation marks within strings can be represented in two ways:

- Literally. The escape character can be followed by quotation marks.
- A symbolic name. For example, <quotation-mark>.

A string can be continued on the next line by placing an `escape_char` as the last character before the newline or linefeed character that ends the line.

A string is a sequence of character symbols or literals enclosed by quotation ("" ) characters. For example:

```
"<A><B><C>"
```

#### Character literals

A character literal is the character itself.

#### Character symbols

A character symbol begins with the < (less-than) character, followed by non-control characters, and ends with the > (greater-than) character. For example, <A> is a valid character symbol (symbolic name). Any character symbol referenced in the source file should be one of the predefined system-

supplied symbols. The system-supplied symbolic names are in the source file member QLGSYMBOL in the QLOCALESRC source file in the optionally installable library QSYSLOCALE.

In the event that the system does not contain a predefined symbolic name for a character, the UCS-2 level-1 format is allowed. The UCS-2 format is based on the character set defined in ISO/IEC 10646. The UCS-2 format may also be used in place of the predefined symbolic names. Here is an example of the UCS-2 symbolic name format:

```
<Uxxxx>
```

where 'xxxx' are 4 hexadecimal digits. For example, <U0041>. The hexadecimal number 0041 within this symbolic name is the UCS-2 code point that represents the character 'A'.

Each locale category must be explicitly defined in a locale definition source file.

### Related concepts

#### [System-supplied locales and recommended CCSIDs](#)

The system-supplied locale source definition file members are in the optionally installable library QSYSLOCALE in the QLOCALESRC source file. The source file members are encoded in CCSID 37 and are read only.

#### [Example: POSIX locale](#)

This example shows the POSIX (or C) locale categories and the source.

#### [Examples: Locale programming](#)

These examples illustrate how locales work in the IBM i operating system, and how you can work with locales and use locales for a multilingual system environment.

### **LC\_COLLATE category**

The LC\_COLLATE category defines character or string collation information. Within LC\_COLLATE you can specify a sort sequence to use using the *cpysyscol* keyword. The *cpysyscol* keyword value is used in place of the LC\_COLLATE category definitions.

A collation element is the unit of comparison for collation. A collation element may be a character or a sequence of characters. Every collation element in the locale has a set of weights, which determine if the collation element collates before, equal to, or after the other collation elements in the locale. Each collation element is assigned collation weights by the CRTLOCALE command when the locale definition source file is created. These collation weights are then used by applications programs that compare strings.

Every character defined in the CCSID that is specified in the CRTLOCALE command is itself a collating element. Additional collating elements can be defined using the collating-element statement. The syntax is:

```
collating-element character-symbol from string
```

The LC\_COLLATE category begins with the LC\_COLLATE keyword and ends with the END LC\_COLLATE keyword.

The following keywords are recognized in the LC\_COLLATE category:

#### **cpysyscol**

This statement specifies that a system collating sequence table is to be used for the collation information for the category. If the locale is intended to be used to set the sort sequence table for the job, then it is required that the CPYSYSCOL keyword be used. If the CPYSYSCOL keyword is specified, no other keyword can be specified. The syntax for the CPYSYSCOL keyword is:

```
CPYSYSCOL sort sequence path name;langid
```

The *sort sequence path name* is a string specifying a fully expanded path name of an existing sort sequence table to use as the definition for this category. The path name delimiter must be a slash (/). Other valid values are strings containing one of the following sort sequence tables:

**\*JOB**

The sort sequence of the job.

**\*LANGIDUNQ**

The unique-weighted sort sequence table that is associated with the language identifier requested parameter.

**\*LANGIDSHR**

The shared-weighted sort sequence table that is associated with the language identifier requested parameter.

**\*HEX**

The sort sequence according to the hexadecimal value of the characters.

The *langid* is a string specifying the language identifier of the sort sequence table to be used. All *langid* must be in uppercase. Valid values are strings containing one of the following language identifiers:

**\*JOB**

Use the language identifier of the job.

**language id**

A valid 3-character language identifier. For example, Danish can be DAN.

**Collating-element**

The collating-element statement specifies multi-character collating elements. The syntax for the collating-element statement is:

```
collating-element symbolic-name from string
```

The symbolic-name value defines a collating element that is a string of one or more characters as a single collating element. The symbolic-name value cannot duplicate any system predefined symbolic name, or any other symbolic name defined in this collation definition. The string value specifies a string of two or more characters or character symbols that define the symbolic-name value. The following are examples of the syntax for the collating-element statement:

```
collating-element <ch> from "<c><h>"
collating-element <e-acute> from "<acute><e>"
collating-element <11> from "<1><1>"
```

A symbolic-name value defined by the collating-element statement is recognized only with the LC\_COLLATE category.

**Order\_start**

The order\_start statement may be followed by one or more collation order statements, assigning collation weights to collating elements. This statement is required. The syntax for the order\_start statement is:

```
order_start sort-rules;sort-rules;...sort-rules collation-order-statements order_end
```

The sort-rules have the following syntax:

```
directive, directive,...directive
```

where directive is one of the directives; **forward**, **backward**, and **position**.

The sort-rules directives are optional. If present, they define the rules to apply during string comparison. The number of specified sort-rules directives defines the number of weights each collating element is assigned (that is, the number of collation orders in the locale). If no sort-rules directives are present, one forward directive is assumed.

If present, the first sort-rules directive applies when comparing strings using primary weight, the second when comparing strings using the secondary weight, and so on. Each set of sort-rules directives is separated by a ; (semicolon). A sort-rules directive consists of one or more comma-separated directives. The following directives are supported:

**Forward**

Specifies that collation weight comparisons proceed from the beginning of a string toward the end of the string.

**Backward**

Specifies that collation weight comparisons proceed from the end of a string toward the beginning of the string.

**Position**

Specifies that collation weight comparisons consider the relative position of non-ignored elements in the string. That is, if strings compare equal, the element with the shortest distance from the starting point of the string collates first.

The forward and backward directives are mutually exclusive. The following example shows the syntax for the sort-rules directives:

```
order_start forward;backward
```

**Order\_end**

This keyword ends collating order entries introduced by the order\_start keyword.

The order of the characters and elements specified between the order\_start and order\_end keywords defines the character order used in range expressions and regular expressions. If no weights are assigned to the characters, then the character order also becomes the collation sequence weight.

**Special symbols**

Special symbols are required to be all upper-case characters. The following special symbols can be used in the LC\_COLLATE category:

- IGNORE

The optional operands for each collation element are used to define the primary, secondary, or subsequent weights for the collating element. The special symbol IGNORE is used to indicate a collating element that is to be ignored when strings are compared.

- UNDEFINED

All characters in the character set must be placed in the collation order, either explicitly or implicitly, by using the Undefined symbol. The UNDEFINED symbol includes all coded character set values not specified explicitly. These characters are inserted in the character collation order at the point indicated by the Undefined symbol in the order of their character code page values. If a collating weight is not explicitly specified for the UNDEFINED symbol, then by default, all of the undefined characters are assigned the same collating weight equal to the relative order of the first undefined character in the collating sequence. If no UNDEFINED special symbol exists and the collation order does not specify all collation elements from the coded character set, a warning is issued and all undefined characters are placed at the end of the character collation order and be given the same collating weight.

**Example 1**

Here is an example of a collation order statement in the LC\_COLLATE locale definition source file category.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

The text that follows the LC\_COLLATE keywords has been added for clarity and does not appear in the locale source file.

```
order_start forward;backward
# The order_start has two sort rules specified:
# forward and backward

UNDEFINED IGNORE;IGNORE
# The UNDEFINED special symbol indicates that
# all characters in the CCSID of the locale
# that are not specified in the definition
# are ignored for collation purposes.
```



```

<LOW>
#       <LOW> is a collating symbol that is ordered
#       after all undefined characters. For example, if there
#       were only two undefined characters, then the <LOW> symbol
#       would be third in the order.

#       All collating elements between <space> and <a> have the
#       same primary equivalence class and individual secondary
#       weights based on their coded character set values.

<a>     <a>;<a>
<a-acute> <a>;<a-acute>
<a-grave> <a>;<a-grave>
<A>     <a>;<A>
<A-acute> <a>;<A-acute>
<A-grave> <a>;<A-grave>
#       All characters between <a> and <A-grave> belong to the
#       same primary equivalence class because they have the same
#       primary weight.

<ch>    <ch>;<ch>
<Ch>    <ch>;<Ch>
#       The <c><h> multi-character collating element is
#       represented by the <ch> collating symbol and belongs to the
#       same primary equivalence class as the <Ch> multi-character
#       collating element.

<s>     <s>;<s>
<eszet> "<s><s>";<s>
#       A one-to-many mapping is indicated by the <eszet>
#       character collated as an <s><s> string. That is, one
#       <eszet> character is expanded to <s><s> characters before
#       comparing.

<HIGH>
order_end

```

## Example 2

Here is an example of a CPYSYSCOL statement in the LC\_COLLATE locale definition source file category.

```

LC_COLLATE
CPYSYSCOL "//QSYS.LIB//QLA10025S.TBL";"ENU"
END LC_COLLATE

```

## Related concepts

### Character sorting

Traditionally, information is displayed in sorted order to enable users to easily find the items they are looking for. However, users of different languages might have very different expectations of what a sorted list should look like.

### Locale restrictions

Certain restrictions apply when you use locales to set job attributes.

### Language identifiers and associated default CCSIDs

This table shows the language identifiers and the job default CCSID (DFTCCSID) values associated with those identifiers.

## **LC\_CTYPE category**

The LC\_CTYPE category defines character classification, case conversion, and other character attributes.

The LC\_CTYPE category begins with an LC\_CTYPE category header and ends with an END LC\_CTYPE category trailer.

All operands for LC\_CTYPE category statements are defined as lists of characters. Each list consists of one or more semicolon-separated characters or symbolic character names.

The following keywords are recognized in the LC\_CTYPE category. In the descriptions, the term *automatically included* means that an error does not occur if the referenced characters are included or

omitted. The characters are provided if they are missing, and are accepted if they are present. In the event that the *automatically included* characters do not exist in the CCSID that you want to create the locale, a warning is issued by the CRTLOCALE command.

**upper**

Defines uppercase letter characters. No character defined by the `cntrl`, `digit`, `punct`, or `space` keyword can be specified. At a minimum, the uppercase letters A through Z are automatically included.

**lower**

Defines lowercase letter characters. No character defined by the `cntrl`, `digit`, `punct`, or `space` keyword can be specified. At a minimum, the lowercase letters a through z are automatically included.

**alpha**

Defines all letter characters. No character defined by the `cntrl`, `digit`, `punct`, or `space` keyword can be specified. Characters defined by the `upper` and `lower` keywords are automatically included in this character class.

**digit**

Defines numeric digit characters. Only the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 can be specified.

**space**

Defines white space characters. No character defined by the `upper`, `lower`, `alpha`, `digit`, `graph`, or `xdigit` keyword can be specified. At a minimum, the `<space>`, `<form-feed>`, `<newline>`, `<carriage return>`, `<tab>`, `<vertical-tab>` characters, and any characters defined by the `blank` keyword, are automatically included.

**cntrl**

Defines control characters. No character defined by the `upper`, `lower`, `alpha`, `digit`, `punct`, `graph`, `print`, or `xdigit` keywords can be specified.

**punct**

Defines punctuation characters. A character defined as the `<space>` character and characters defined by the `upper`, `lower`, `alpha`, `digit`, `cntrl`, or `xdigit` keyword cannot be specified.

**graph**

Defines printable characters, excluding the `<space>` character. If this keyword is not specified, characters defined by the `upper`, `lower`, `alpha`, `digit`, `xdigit`, and `punct` keywords are automatically included in this character class. No character defined by the `cntrl` keyword can be specified.

**print**

Defines printable characters, including the `<space>` character. If this keyword is not specified, the `<space>` character and characters defined by the `upper`, `lower`, `alpha`, `digit`, `xdigit`, and `punct` keywords are automatically included in this character class. No character defined by the `cntrl` keyword can be specified.

**xdigit**

Defines hexadecimal digit characters. Only the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 and the letters A, B, C, D, E, F, a, b, c, d, e, and f can be specified. If not specified, the `xdigit` class defaults to the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 and the letters A, B, C, D, E, F, a, b, c, d, e, and f.

**blank**

Defines blank characters. If this keyword is not specified, the `<space>` and `<horizontal-tab>` characters are included in this character class.

**toupper**

Defines the mapping of lowercase characters to uppercase characters. Operands for this keyword consist of semicolon-separated character pairs. Each character pair is enclosed in ( ) (parentheses) and separated from the next pair by a , (comma). The first character in each pair is considered lowercase; the second character is considered uppercase. Only characters defined by the `lower` and `upper` keywords can be specified.

**tolower**

Defines the mapping of uppercase characters to lowercase characters. Operands for this keyword consist of semicolon-separated character pairs. Each character pair is enclosed in ( ) (parentheses) and separated from the next pair by a , (comma). The first character in each pair is considered

uppercase; the second character is considered lowercase. Only characters defined by the lower and upper keywords can be specified.

**Note:** The **tolower** keyword is optional. If this keyword is not specified, the mapping defaults to the reverse mapping of the **toupper** keyword, if specified. If the **toupper** keyword is not specified, the mapping defaults to the **C** locale.

## Example

Here is an example of the LC\_CTYPE category in a locale definition source file.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```
LC_CTYPE

#"alpha" is by default "upper" and "lower"
#"print" is by default "alpha", "digit", "punct", and the space character
#"graph" is by default "alnum" and "punct"
#"tolower" is by default the reverse mapping of "toupper"
#

upper <A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;<L>;<M>;\
<N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>
#
lower <a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;\
<n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>
#
digit <zero>;<one>;<two>;<three>;<four>;<five>;<six>;\
<seven>;<eight>;<nine>
#
space <tab>;<newline>;<vertical-tab>;<form-feed>;\
<carriage-return>;<space>
#

cntrl <alert>;<backspace>;<tab>;<newline>;<vertical-tab>;\
<form-feed>;<carriage-return>;<NUL>;<SOH>;<STX>;\
<ETX>;<EOT>;<ENQ>;<ACK>;<SO>;<SI>;<DLE>;<DC1>;<DC2>;\
<DC3>;<DC4>;<NAK>;<SYN>;<ETB>;<CAN>;<EM>;<SUB>;\
<ESC>;<IS4>;<IS3>;<IS2>;<IS1>;<DEL>
#
punct <exclamation-mark>;<quotation-mark>;<number-sign>;\
<dollar-sign>;<percent-sign>;<ampersand>;<asterisk>;\
<apostrophe>;<left-parenthesis>;<right-parenthesis>;\
<plus-sign>;<comma>;<hyphen>;<period>;<slash>;\
<colon>;<semicolon>;<less-than-sign>;<equals-sign>;\
<greater-than-sign>;<question-mark>;<commercial-at>;\
<left-square-bracket>;<backslash>;<circumflex>;\
<right-square-bracket>;<underline>;<grave-accent>;\
<left-curly-bracket>;<vertical-line>;<tilde>;\
<right-curly-bracket>
#

xdigit <zero>;<one>;<two>;<three>;<four>;<five>;<six>;\
<seven>;<eight>;<nine>;<A>;<B>;<C>;<D>;<E>;<F>;\
<a>;<b>;<c>;<d>;<e>;<f>
#
blank <space>;<tab>
#
toupper (<a>,<A>);(<b>,<B>);(<c>,<C>);(<d>,<D>);(<e>,<E>);\
(<f>,<F>);(<g>,<G>);(<h>,<H>);(<i>,<I>);(<j>,<J>);\
(<k>,<K>);(<l>,<L>);(<m>,<M>);(<n>,<N>);(<o>,<O>);\
(<p>,<P>);(<q>,<Q>);(<r>,<R>);(<s>,<S>);(<t>,<T>);\
(<u>,<U>);(<v>,<V>);(<w>,<W>);(<x>,<X>);(<y>,<Y>);\
(<z>,<Z>)
#
END LC_CTYPE
```

### **LC\_MESSAGES category**

The LC\_MESSAGES category of a locale definition source file defines the format for affirmative and negative system responses. This category begins with an LC\_MESSAGES category header and ends with an END LC\_MESSAGES category trailer.

All operands for the LC\_MESSAGES category are defined as strings or *extended regular expressions* enclosed by quotation marks ("").

**Note:** These operands are separated from the keyword they define by one or more blanks. Two adjacent quotation marks ("" ) indicate an undefined value. The following keywords are recognized in the LC\_MESSAGES category:

#### **yesexpr**

Specifies an extended regular expression that describes the acceptable affirmative response to a question expecting an affirmative or negative response.

#### **noexpr**

Specifies an extended regular expression that describes the acceptable negative response to a question expecting an affirmative or negative response.

#### **yesstr**

A fixed string of acceptable affirmative response.

#### **nostr**

A fixed string of acceptable negative response.

### **Extended regular expressions**

The following special characters are used to form extended regular expressions:

<b>Character</b>	<b>Function</b>
<b>+</b>	Specifies that a string matches if one or more occurrences of the character or extended regular expression that precedes the + (plus) are within the string.
<b>?</b>	Specifies that a string matches if zero or one occurrences of the character or extended regular expression that precedes the ? (question mark) are within the string.
<b> </b>	Specifies that a string matches if either of the strings separated by the   (vertical line) are within the string.
<b>()</b>	Groups strings together in regular expressions.
<b>{m}</b>	Specifies that a string matches if exactly m occurrences of the pattern are within the string.
<b>{m,}</b>	Specifies that a string matches if at least m occurrences of the pattern are within the string.
<b>{m, n}</b>	Specifies that a string matches if between m and n, inclusive, occurrences of the pattern are within the string ( where m <= n).
<b>[String]</b>	Signifies that the regular expression matches any characters specified by the string variable within the square brackets.

## Character

[^ String]

^

\$

.(period)

\*(asterisk)

\(backslash)

## Function

A ^ (caret) within the [ ] (square brackets) and at the beginning of the specified string indicates that the regular expression does not match any characters within the square brackets.

Signifies the beginning of a field or record.

Signifies the end of a field or record.

Signifies any one character except the terminal new-line character at the end of a space.

Signifies zero or more of any characters.

The escape character. When preceding any of the characters that have special meaning in extended regular expressions, the escape character removes any special meaning for the character.

Character class expressions may also be specified in the extended regular expression. The following character class expressions are supported in all locales:

```
[ :alnum: ]
[ :alpha: ]
[ :blank: ]
[ :cntrl: ]
[ :digit: ]
[ :graph: ]
[ :lower: ]
[ :print: ]
[ :punct: ]
[ :space: ]
[ :upper: ]
[ :xdigit: ]
```

## Example

Here is an example of the LC\_MESSAGES category in a locale definition source file.

```
LC_MESSAGES
#
yesexpr "[yY]"
noexpr "[nN]"
yesstr "yes"
nostr "no"
#
END LC_MESSAGES
```

### **LC\_MONETARY category**

The LC\_MONETARY category of a locale definition source file defines rules and symbols for formatting monetary numeric information. This category begins with an LC\_MONETARY category header and ends with an END LC\_MONETARY category trailer.

All operands for the LC\_MONETARY category keywords are defined as string or integer values. String values are bounded by quotation marks (""). All values are separated from the keyword they define by one or more spaces. Two adjacent quotation marks indicate an undefined string value. A -1 indicates an undefined integer value. The following keywords are recognized in the LC\_MONETARY category:

#### **int\_curr\_symbol**

Specifies the string used for the international currency symbol. The operand for the int\_curr\_symbol keyword is a four-character string. The first three characters contain the alphabetic international-currency symbol. The fourth character specifies a character separator between the international currency symbol and a monetary quantity.

**currency\_symbol**

Specifies the string used for the local currency symbol.

**mon\_decimal\_point**

Specifies the string used for the decimal delimiter used to format monetary quantities.

**mon\_thousands\_sep**

Specifies the string used for grouping digits to the left of the decimal delimiter in formatted monetary quantities.

**mon\_grouping**

Defines the size of each group of digits in formatted monetary quantities. The operand for the `mon_grouping` keyword consists of a sequence of semicolon-separated integers. Each integer specifies the number of digits in a group. The initial integer defines the size of the group immediately to the left of the decimal delimiter. The following integers define succeeding groups to the left of the previous group. If the last digit is not -1, subsequent grouping is performed using the previous digit. If the last digit is -1, grouping is only performed for the number of groups specified.

Here is an example of the interpretation of the `mon_grouping` keyword. Assuming that the value to be formatted is 123456789 and the operand for the `mon_thousands_sep` keyword is comma (,), the following results occur:

**mon\_grouping Value**

Formatted Value

**3;-1**  
123456,789

**3**  
123,456,789

**3;2**  
12,34,56,789

**3;2;-1**  
1234,56,789

**positive\_sign**

Specifies the string used to indicate a nonnegative-valued formatted monetary quantity.

**negative\_sign**

Specifies the string used to indicate a negative-valued formatted monetary quantity.

**int\_frac\_digits**

Specifies an integer value representing the number of fractional digits (those after the decimal delimiter) to be displayed in a formatted monetary quantity using the `int_curr_symbol` value.

**frac\_digits**

Specifies an integer value representing the number of fractional digits (those after the decimal delimiter) to be displayed in a formatted monetary quantity using the `currency_symbol` value.

**p\_cs\_precedes**

Specifies an integer value indicating whether the `int_curr_symbol` or `currency_symbol` string precedes or follows the value for a non-negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that the currency symbol follows the monetary quantity.

**1**  
Indicates that the currency symbol precedes the monetary quantity.

**p\_sep\_by\_space**

Specifies an integer value indicating whether the `int_curr_symbol` or `currency_symbol` string is separated by a space from a non-negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that no space separates the currency symbol from the monetary quantity.

**1**  
Indicates that a space separates the currency symbol from the monetary quantity.

**2**  
Indicates that a space separates the currency symbol and the `positive_sign` string, if adjacent.

**n\_cs\_precedes**

Specifies an integer value indicating whether the `int_curr_symbol` or `currency_symbol` string precedes or follows the value for a negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that the currency symbol follows the monetary quantity.

**1**  
Indicates that the currency symbol precedes the monetary quantity.

**n\_sep\_by\_space**

Specifies an integer value indicating whether the `int_curr_symbol` or `currency_symbol` string is separated by a space from a negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that no space separates the currency symbol from the monetary quantity.

**1**  
Indicates that a space separates the currency symbol from the monetary quantity.

**2**  
Indicates that a space separates the currency symbol and the **negative\_sign** string, if adjacent.

**p\_sign\_posn**

Specifies an integer value indicating the positioning of the `positive_sign` string for a non-negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that parenthesis enclose both the monetary quantity and the `int_curr_symbol` or `currency_symbol` string.

**1**  
Indicates that the `positive_sign` string precedes the quantity and the `int_curr_symbol` or `currency_symbol` string.

**2**  
Indicates that the `positive_sign` string follows the quantity and the `int_curr_symbol` or `currency_symbol` string.

**3**  
Indicates that the `positive_sign` string immediately precedes the `int_curr_symbol` or `currency_symbol` string.

**4**  
Indicates that the `positive_sign` string immediately follows the `int_curr_symbol` or `currency_symbol` string.

**n\_sign\_posn**

Specifies an integer value indicating the positioning of the `negative_sign` string for a negative formatted monetary quantity. The following integer values are recognized:

**0**  
Indicates that parenthesis enclose both the monetary quantity and the `int_curr_symbol` or `currency_symbol` string.

**1**  
Indicates that the `negative_sign` string precedes the quantity and the `int_curr_symbol` or `currency_symbol` string.

- 2 Indicates that the `negative_sign` string follows the quantity and the `int_curr_symbol` or `currency_symbol` string.
- 3 Indicates that the `negative_sign` string immediately precedes the `int_curr_symbol` or `currency_symbol` string.
- 4 Indicates that the `negative_sign` string immediately follows the `int_curr_symbol` or `currency_symbol` string.

### Example

Here is an example of the `LC_MONETARY` category listed in a locale definition source file.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```
LC_MONETARY
#
int_curr_symbol      "<U><S><D>"
currency_symbol     "<dollar-sign>"
mon_decimal_point   "<period>"
mon_thousands_sep  "<comma>"
mon_grouping        3;-1
positive_sign       "<plus-sign>"
negative_sign       "<hyphen>"
int_frac_digits     2
frac_digits         2
p_cs_precedes       1
p_sep_by_space      2
n_cs_precedes       1
n_sep_by_space      2
p_sign_posn         3
n_sign_posn         3
#
END LC_MONETARY
```

### Related concepts

Example: [Producing unique monetary formats](#)

A unique customized monetary format can be produced by changing the value of a single statement.

### ***LC\_NUMERIC category***

`LC_NUMERIC` category of a locale definition source file defines rules and symbols for formatting non-monetary numeric information.

The `LC_NUMERIC` category of a locale definition source file defines rules and symbols for formatting non-monetary numeric information. This category begins with an `LC_NUMERIC` category header and terminates with an `END LC_NUMERIC` category trailer.

All operands for the `LC_NUMERIC` category keywords are defined as string or integer values. String values are bounded by quotation marks (""). All values are separated from the keyword they define by one or more spaces. Two adjacent quotation marks indicate an undefined string value. A -1 indicates an undefined integer value. The following keywords are recognized in the `LC_NUMERIC` category:

#### **decimal\_point**

Specifies a string containing the decimal delimiter character used to format numeric, non-monetary quantities.

#### **thousands\_sep**

Specifies the string separator used for grouping digits to the left of the decimal delimiter in formatted numeric, non-monetary quantities.

#### **grouping**

Defines the size of each group of digits in formatted monetary quantities. The operand for the grouping keyword consists of a sequence of semicolon-separated integers. Each integer specifies the number of digits in a group. The initial integer defines the size of the group immediately to the left of



the decimal delimiter. The following integers define succeeding groups to the left of the previous group. Grouping is performed for each integer specified for the grouping keyword. If the last digit is not -1, subsequent grouping is performed using the previous digit. If the last digit is -1, grouping is only performed for the number of groups specified.

Here is an example of the interpretation of the grouping statement. Assuming that the value to be formatted is 123456789 and the operand for the thousands\_sep keyword is comma (,), the following results occur.

Grouping value	Formatted value
3	123,456,789
3;-1	123456,789
3;2	12,34,56,789
3;2;-1	1234,56,789

### Example

Here is an example of the LC\_NUMERIC category in a locale definition source file.

```
LC_NUMERIC
#
decimal_point "<period>"
thousands_sep "<comma>"
grouping      3;-1
#
END LC_NUMERIC
```

### LC\_TIME category

The LC\_TIME category of a locale definition source file defines rules and symbols for formatting time and date information. This category begins with an LC\_TIME category header and ends with an END LC\_TIME category trailer.

All operands for the LC\_TIME category keywords are defined as string or integer values. String values are bounded by quotation marks (""). All values are separated from the keyword they define by one or more spaces. Two adjacent quotation marks indicate an undefined string value. A -1 indicates an undefined integer value. Field descriptors are used by commands and subroutines that query the LC\_TIME category to represent elements of time and date formats. The following keywords are recognized in the LC\_TIME category:

#### abday

Defines the abbreviated weekday names corresponding to the %a field descriptor. Recognized values consist of seven semicolon-separated strings. The first string corresponds to the abbreviated name for the first day of the week (Sun), the second to the abbreviated name for the second day of the week, and so on.

#### day

Defines the full spelling of the weekday names corresponding to the %A field descriptor. Recognized values consist of seven semicolon-separated strings. The first string corresponds to the full spelling of the name of the first day of the week (Sunday), the second to the name of the second day of the week, and so on.

#### abmon

Defines the abbreviated month names corresponding to the %b field descriptor. Recognized values consist of 12 semicolon-separated strings. The first string corresponds to the abbreviated name for the first month of the year (Jan), the second to the abbreviated name for the second month of the year, and so on.

#### mon

Defines the full spelling of the month names corresponding to the %B field descriptor. Recognized values consist of 12 semicolon-separated strings. The first string corresponds to the full spelling of

the name for the first month of the year (January), the second to the full spelling of the name for the second month of the year, and so on.

#### **d\_t\_fmt**

Defines the string used for the standard date and time format corresponding to the %c field descriptor. The string can contain any combination of characters, field descriptors, or escape sequences. See [“Escape sequences”](#) on page 208 for additional information.

#### **d\_fmt**

Defines the string used for the standard date format corresponding to the %x field descriptor. The string can contain any combination of characters, field descriptors, or escape sequences. Here is an example of how the d\_fmt keyword can be constructed:

##### **%D**

The %D indicates a %m/%d/%y date format. If you are using this format and have chosen to set the job attribute from the locale, then a slash (/) is extracted for the DATSEP job attribute. \*MDY is extracted for the DATFMT job attribute.

##### **%j**

The %j indicates a Julian date format. If you are using this format and have chosen to set the job attribute from the locale, then no DATSEP job is extracted. However, \*JUL is extracted for the DATFMT job attribute.

##### **%d-%m-%y**

If you are using this format and have chosen to set the job attribute from the locale, then the compiler extracts the hyphen (-) for the DATSEP job attribute and \*DMY for the DATFMT job attribute.

##### **%y.%m.%d**

If you are using this format and have chosen to set the job attribute from the locale, then the compiler extracts the period (.) for the DATSEP job attribute and \*YMD for the DATFMT job attribute.

##### **%m/%d/%Y**

If you are using this format and have chosen to set the job attribute from the locale, then the compiler extracts the slash (/) for the DATSEP job attribute. No DATFMT job attribute is extracted.

**Note:** If the locale is to contain a valid IBM i date format and date separator, then the d\_fmt value must be defined such that it contains valid IBM i date format and date separators. For example, if the value is specified as %m/%d/%y, then \*MDY will be extracted for the IBM i date format and a slash (/) will be extracted for the IBM i date format. A warning is issued by the CRTLOCALE command if an IBM i date format or date separator cannot be extracted.

#### **t\_fmt**

Defines the string used for the standard time format corresponding to the %X field descriptor. The string can contain any combination of characters, field descriptors, or escape sequences. Here is an example of how the t\_fmt keyword can be constructed:

##### **%H:%M:%S**

The compiler extracts a colon (:) for the TIMSEP job attribute.

##### **%H.%M.%S**

The compiler extracts a period (.) for the TIMSEP job attribute.

##### **%H %M %S**

The compiler extracts a blank space for the TIMSEP job attribute.

##### **%H,%M,%S**

The compiler extracts a comma (,) for the TIMSEP job attribute.

##### **%T**

%T implies a %H:%M:%S (hours, minutes, seconds) time format with a colon (:) as the TIMSEP job attribute.

##### **%H&%M&%S;**

A valid TIMSEP job attribute cannot be determined.

**Note:** If the locale is to contain a valid IBM i time separator, then the `t_fmt` value must be defined such that it contains a valid IBM i time separator. For example, if the value is specified as `%H:%M:%S`, then a `:` (colon) will be extracted for the IBM i date format. A warning is issued by the `CRTLOCALE` command if an IBM i time separator cannot be extracted.

### **am\_pm**

Defines the strings used to represent *ante meridiem* (before noon) and *post meridiem* (after noon) corresponding to the `%p` field descriptor. Recognized values consist of two strings separated by a `;` (semicolon). The first string corresponds to the *ante meridiem* designation, the last string to the *post meridiem* designation.

### **t\_fmt\_ampm**

Defines the string used for the standard 12-hour time format that includes an `am_pm` value (`%p` field descriptor). This statement corresponds to the `%r` field descriptor. The string can contain any combination of characters and field descriptors.

### **era**

Defines how the years are counted and displayed for each era in a locale, corresponding to the `%E` field descriptor modifier. For each era, there must be one string in the following format:

```
direction:offset:start_date:end_date:era_name:era_format
```

The variables for the era-string format are defined as follows:

#### **direction**

Specifies a minus sign (`-`) or plus sign (`+`) character. The plus character indicates that years count in the positive direction when moving from the start date to the end date. The minus character indicates that years count in the negative direction when moving from the start date to the end date.

#### **offset**

Specifies a number representing the first year of the era.

#### **start\_date**

Specifies the starting date of the era in the `yyyy/mm/dd` format, where `yyyy` is the year, `mm` is the month, and `dd` is the day. Years before the year 1 AD are represented as negative numbers. For example, an era beginning March 5th in the year 100 BC is represented as `-100/03/05`.

#### **end\_date**

Specifies the ending date of the era in the same form used for the `start_date` variable or one of the two special values `-*` or `+`. A `-*` value indicates that the ending date of the era extends backward to the beginning of time. A `+` value indicates that the ending date of the era extends forward to the end of time. Therefore, the ending date can be chronologically before or after the starting date of the era. For example, the strings for the Christian eras AD and BC are entered as follows:

```
+ :0:0000/01/01:++:AD:%o %N
+ :1:-0001/12/31:-*:BC:%o %N
```

#### **era\_name**

Specifies a string representing the name of the era that is substituted for the `%EC` field descriptor.

#### **era\_format**

Specifies a string for formatting the `%EY` field descriptor.

An **era** value consists of one string for each era. If more than one era is specified, each era string is separated by a `;` (semicolon).

### **era\_d\_fmt**

Defines the string used to represent the date in alternate-era format corresponding to the `%Ex` field descriptor. The string can contain any combination of characters and field descriptors.

### **era\_t\_fmt**

Defines the string used to represent the time in alternate-era format corresponding to the `%EX` field descriptor. The string can contain any combination of characters and field descriptors.

## era\_d\_t\_fmt

Defines the string used to represent the date and time in alternate-era format corresponding to the %Ec field descriptor. The string can contain any combination of characters and field descriptors.

## alt\_digits

Defines alternate strings for digits corresponding to the %O field descriptor. Recognized values consist of a group of strings separated by ; (semicolons). The first string represents the alternate string for zero, the second string represents the alternate string for one, and so on. A maximum of 100 alternate strings can be specified.

## Escape sequences

Here are escape sequences allowed for the d\_t\_fmt, d\_fmt, and t\_fmt keyword values:

>

\\	Represents the backslash character.
\a	Represents the alert character.
\b	Represents the backspace character.
\f	Represents the form-feed character.
\n	Represents the newline character.
\r	Represents the carriage-return character.
\t	Represents the tab character.
\v	Represents the vertical-tab character.

## Example

Here is an example of the LC\_TIME category in a locale definition source file.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```
LC_TIME
#
#Abbreviated weekday names (%a)
abday
"<S><u><n>";"<M><o><n>";"<T><u><e>";"<W><e><d>";\
    "<T><h><u>";"<F><r><i>";"<S><a><t>"
#
#Full weekday names (%A)
day
"<S><u><n><d><a><y>";"<M><o><n><d><a><y>";\
"<T><u><e><s><d><a><y>";"<W><e><d><n><e><s><d><a><y>";\
"<T><h><u><r><s><d><a><y>";"<F><r><i><d><a><y>";\
"<S><a><t><u><r><d><a><y>"
#
#Abbreviated month names (%b)
abmon
"<J><a><n>";"<F><e><b>";"<M><a><r>";"<A><p><r>";\
    "<M><a><y>";"<J><u><n>";"<J><u><l>";"<A><u><g>";\
    "<S><e><p>";"<O><c><t>";"<N><o><v>";"<D><e><c>"
#
#Full month names (%B)
mon
"<J><a><n><u><a><r><y>";"<F><e><b><r><u><a><r><y>";\
"<M><a><r><c><h>";"<A><p><r><i><l>";"<M><a><y>";\
"<J><u><n><e>";"<J><u><l><y>";"<A><u><g><u><s><t>";\
"<S><e><p><t><e><m><b><e><r>";"<O><c><t><o><b><e><r>";\
"<N><o><v><e><m><b><e><r>";"<D><e><c><e><m><b><e><r>"
#
#Date and time format (%c)
d_t_fmt "%a %b %d %H:%M:%S %Y"
#
```

```

#Date format (%x)
d_fmt      "%m/%d/%y"
#
#Time format (%X)
t_fmt      "%H:%M:%S"
#
#Equivalent of AM/PM (%p)
am_pm      "<A><M>"; "<P><M>"
#
#12-hour time format (%r)
t_fmt_ampm "%I:%M:%S %p"
#
era        "+:0:0000/01/01:+:AD:%EC"; \
"+:1:-0001/12/31:-*:BC:%Ey";
era_d_fmt  ""
alt_digits
"<0><t><h>"; "<1><s><t>"; "<2><n><d>"; "<3><r><d>"; \
"<4><t><h>"; "<5><t><h>"; "<6><t><h>"; "<7><t><h>"; \
"<8><t><h>"; "<9><t><h>"; "<1><0><t><h>"
#
END LC_TIME

```

## Related concepts

### Example: [Creating and enabling a locale](#)

This example contains the steps necessary for creating and enabling a locale.

### **LC\_TOD category**

The LC\_TOD category defines the rules used to define the start and end time of daylight saving time (DST), the difference between local time and Greenwich Mean time, the time zone name, and the DST name. This category is an IBM extension and must appear after all other category definitions in the source file.

#### Notes:

1. System locale files (such as /QSYS.LIB/EN\_US.LOCALE) that IBM provides do not have LC\_TOD information filled in. This is because the same locale source (for example, EN\_US) can be used for many different time zones and, therefore, no sets of values can be used as the default for every time zone.
2. The values specified in the LC\_TOD category are only reflected in the C functions that process time. These values do not affect a job's time, which is set by a time zone object.

The LC\_TOD locale time definitions are used only by applications that use locale-sensitive APIs, which are often application code that is based on C and C++ programming languages. Traditional system code (for example, DSPJOB) does not use the locale LC\_TOD information.

RPG and COBOL applications have several ways if they need to use the LC\_TOD information in a program:

- They can use the locale-sensitive APIs found in C run time.
- They can use the `QlgRetrieveLocaleInformation` API to retrieve locale information and do their own date and time formatting independent of the C runtime support.
- They can use system APIs, such as Convert Date and Time Format (QWCCVTDT), to map system time to local time. The QWCCVTDT API supports the specification of a time zone object when performing date and time conversions.

All the operands for the LC\_TOD category are defined as string or integer values. String values are bounded by quotation marks (""). All values are separated from the keyword they define by one or more spaces. Two adjacent quotation marks indicate an undefined string value. A 0 (zero) indicates an undefined integer value. The following keywords are recognized in the LC\_TOD category.

#### **tzdiff**

The time zone difference in minutes. It is the difference between the local time and Greenwich mean time.

#### **tname**

The string used for the time zone name.

#### **dstname**

The string used for the DST name.

**dststart**

A set of 4 integers that represents the DST start date. The operand for the `dststart` keyword consists of a sequence of four integers separated by comma.

**dststart keyword format**

*month,week,day,time*

The variables for the `dststart` format are defined as follows:

**month**

An integer value that represents the month of the year when DST takes effect. This value ranges from 1 to 12, with 1 corresponding to January, and 12 corresponding to December.

**week**

An integer value that represents the week of the month when DST takes effect. This value ranges from -4 to 4, with -4 corresponding to the fourth week of the month counting from the end of the month and 4 corresponding to the fourth week of the month counting from the beginning of the month.

**Note:** You can specify zero for *week*. This means that the day indicates the day of the month rather than the day of the week. For example, the string 4,0,23,0 means that DST starts the fourth month of the year, the twenty-third day of the month, and 0 seconds after midnight local standard time.

**day**

An integer value that represents the day of the month when DST takes effect. If the *week* keyword is not 0 (zero), then this is the day of the week when DST takes effect. This value ranges from 1 to the last day of the month or 1 to the last day of the week.

**time**

An integer value that represents the number of seconds after 12 midnight, local standard time, when DST takes effect. This value ranges from 0 to 86399.

**dstend**

A set of 4 integers that represents the DST end date. The operand for the `dstend` keyword consists of a sequence of four integers that are separated by comma.

**dstend keyword format**

*month,week,day,time*

The variables for the `dstend` format are defined as follows:

**month**

An integer value that represents the month of the year when DST ends. This value ranges from 1 to 12, with 1 corresponding to January, and 12 corresponding to December.

**week**

An integer value that represents the week of the month when DST ends. This value ranges from -4 to 4, with -4 corresponding to the fourth week of the month counting from the end of the month and 4 corresponding to the fourth week of the month counting from the beginning of the month.

**day**

An integer value that represents the day of the month when DST ends. If the *week* keyword is not 0 (zero), then this is the day of the week when DST ends. This value ranges from 1 to the last day of the month or 1 to the last day of the week.

**time**

An integer value that represents the number of seconds after 12 midnight, local standard time, when DST ends. This value ranges from 0 to 86399.

**dstshift**

An integer value that represents the DST shift value in seconds.

## Example

Here is an example of the LC\_TOD category in a locale definition source file.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```
LC_TOD
#
tzdiff    360
tname     "<C><S><T>"
dstname   "<C><D><T>"

#Set daylight saving time to start on 3rd week of October at
#midnight on Saturday.
dststart  10,3,6,0

#Set daylight saving time to end on April 23, at midnight.

dstend    4,0,23,0
dstshift  3600
#
END LC_TOD
```

## Locale symbolic names

The IBM i operating system supports locale symbolic names based on predefined names from the X/Open Standard portable character set.

In addition, the IBM i operating system supports a 5-character alphanumeric symbolic name for all characters, in the following situations:

- The first character of the symbolic name is a Latin capital letter U. This character identifies that the name is derived from the ISO/IEC 10646 Universal Coded Character Set.
- The second through fifth characters of the symbolic name represent the code point of the character in the ISO/IEC 10646 Universal Coded Character Set 2 Level 1. This portion of the symbolic name is assigned by code point for ease of creating and changing locales.

As an example, the question mark (?) character provides the following correlation between symbolic naming, UCS2-1 code point, and an IBM-assigned code point:

- The ? character is symbolically represented by <question-mark>
- It is at code point U003F in the ISO 10646 code page
- It is at code point 6F in IBM code page 500.

Mapping of locale symbolic names provides a list of all symbolic names supported on IBM i. The table also provides the UCS2-1 (ISO 10646) code points, their corresponding IBM code page or code points, and a graphic representation of each character.

## Related concepts

### Portable character set

The X/Open portable character set is a superset of the IBM invariant character set (00640), which includes symbols not represented in the IBM invariant character set 00640.

### Mapping of locale symbolic names

This table lists locale symbolic names.

## Examples: Locale programming

These examples illustrate how locales work in the IBM i operating system, and how you can work with locales and use locales for a multilingual system environment.

In addition to these examples, Locale categories provides programming examples for each of the different locale categories.

## Related concepts

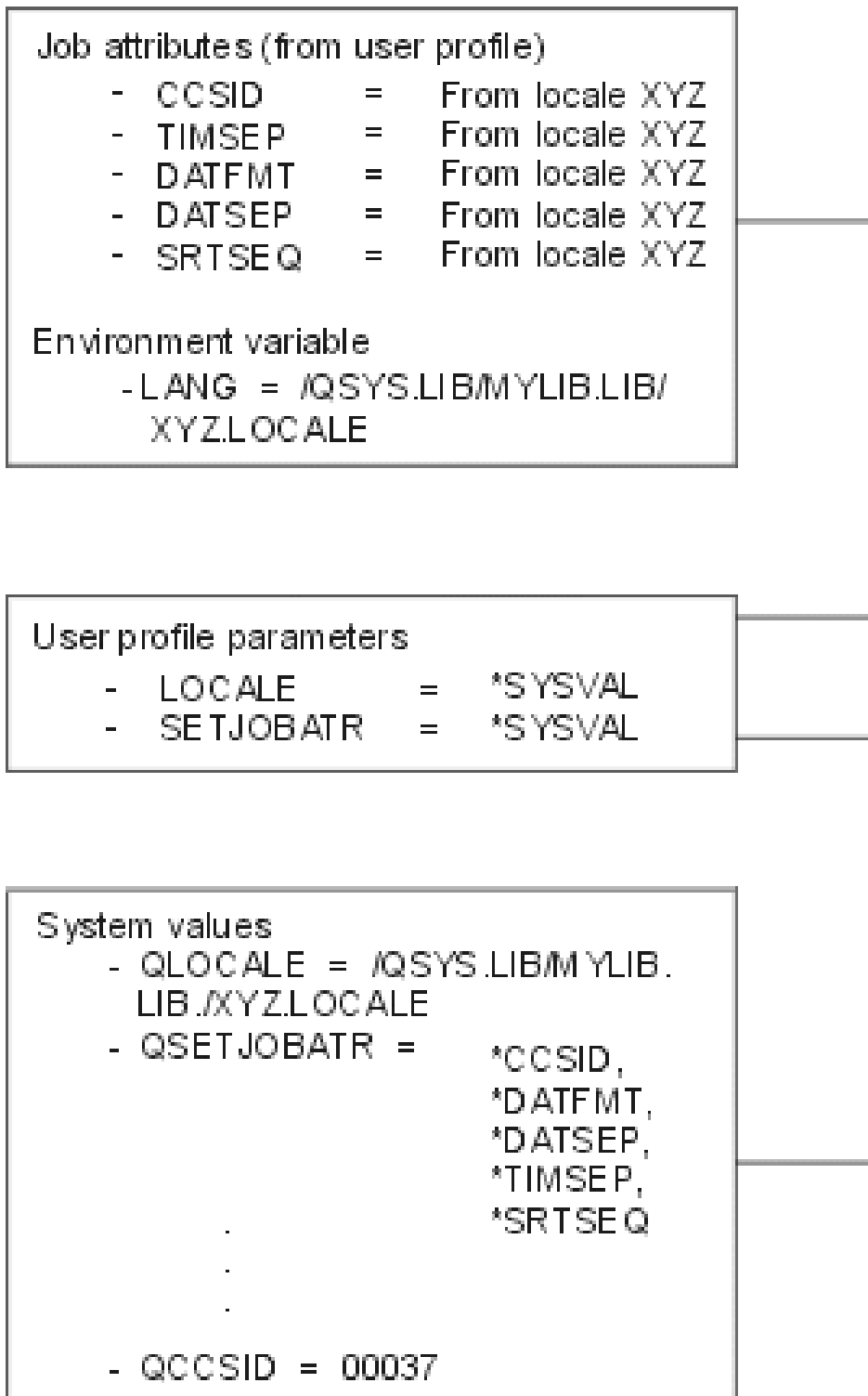
### Locale categories

This table describes the locale categories that are supported on the IBM i operating system.

**Example: How locales work**

These two examples focus on the LOCALE and SETJOBATR parameter values specified on the user profile.

The first example illustrates using locales to establish job attributes. The user profile parameters LOCALE and SETJOBATR have values of \*SYSVAL. This means that the job attributes at job start time come from the QLOCALE value based on the values in QSETJOBATR.





If a job runs based on the information in the figure, the following conditions are true:

- The locale used is XYZ.

This is because the user profile parameter value for LOCALE was \*SYSVAL. The \*SYSVAL value is XYZ.

- The CCSID is based on the value specified when locale XYZ was created.

This value is specified when the LOCALE object is created using the CRTLOCALE command.

- The time separator is derived from locale XYZ.

This value is derived from the LC\_TIME category specified in LOCALE XYZ.

- The date format separator is derived from locale XYZ.

This value is derived from the LC\_TIME category specified in LOCALE XYZ.

- The data separator is derived from locale XYZ.

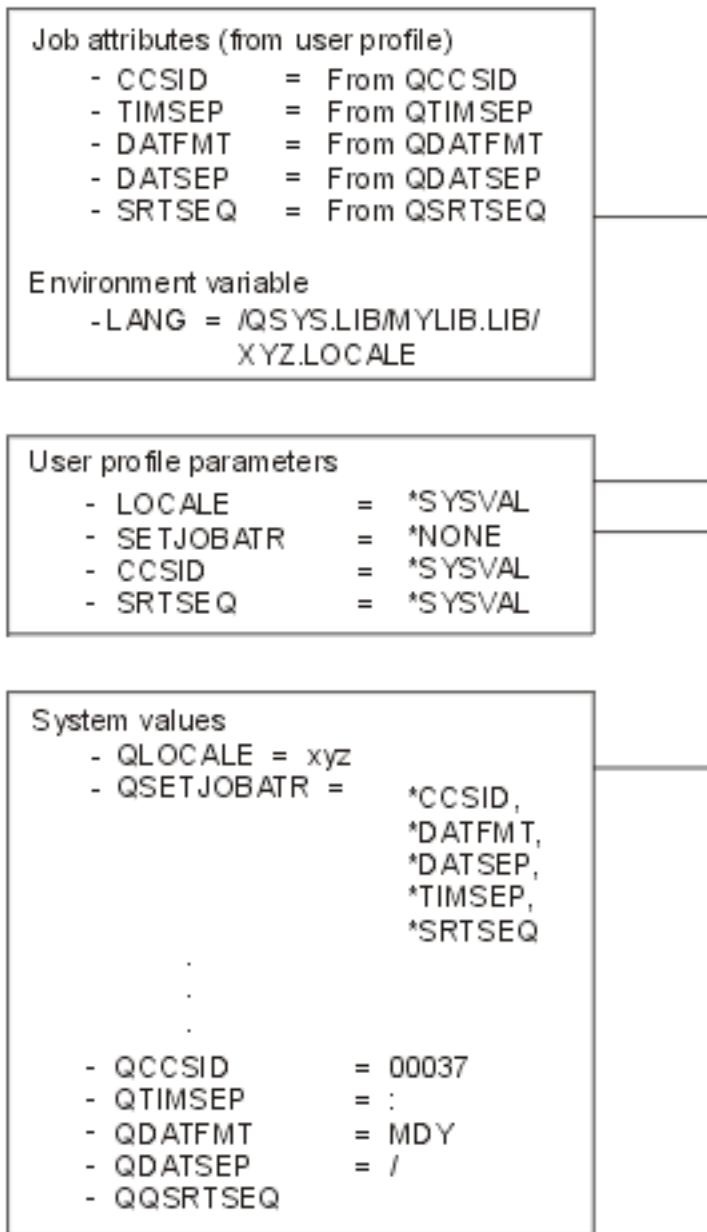
This value is derived from the LC\_TIME category specified in LOCALE XYZ.

- The decimal format character is derived from locale XYZ.

This value is derived from the LC\_NUMERIC category specified in LOCALE XYZ.

In the second example the user profile LOCALE parameter value is \*SYSVAL and the SETJOBATR parameter values is \*NONE. This means that the LOCALE value is determined by looking at the system value QLOCALE. When the SETJOBATR value is \*NONE, job attributes are determined by the values in the user profile.

Remember, because the user profile SETJOBATR parameter was \*NONE, the system's search resulted in using the values specified for QCCSID, QTIMSEP, QDATFMT, QSRTSEQ, and QDATSEP.



If a job runs based on the information in this example, the following conditions are true:

- The locale used is XYZ.

This is because the user profile parameter value for LOCALE was \*SYSVAL. The \*SYSVAL value is XYZ.

- The CCSID is 00037.

This is because the user profile SETJOBATR parameter value was \*NONE. The system search ended with the value for QCCSID being used.

- The time separator is a colon (:).

This is because the user profile SETJOBATR parameter value was \*NONE. The system search ended with the value for QTIMSEP being used.

- The date format separator is a slash (/).

This is because the user profile SETJOBATR parameter value was \*NONE. The system search ended with the value for QDATSEP being used.

- The date format is month/day/year (MDY).

This is because the user profile SETJOBATR parameter value was \*NONE. The system search ended with the value for QDATFMT being used.

- The decimal format character is a period. Zero suppression is performed.

This is because the user profile SETJOBATR parameter value was \*NONE. The system search ended with the value for QDECFMT being used.

### **Example: Creating and enabling a locale**

This example contains the steps necessary for creating and enabling a locale.

To create and enable a locale, follow these steps:

1. Create (or have) a library and a source physical file.
2. Copy an existing locale source file definition member (to a library and source physical file).
3. Edit the copied locale source file member if you need to customize any of the categories within the locale source.
4. Create the locale object.
5. Enable the locale object by using system values or parameters on the user profile.

**Note:** By using the code examples, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

### **Step 1. Creating a library and a source physical file**

A library and a source physical file are needed to store the locale source file member. Locale-source-definition files that are system supplied include a list of the locale source file members that are included in the IBM i operating system.

To create a library called localelib, follow these steps:


1. On a command line, type CRTLIB and press F4 (Prompt).
2. Type localelib for the name of the library and press Enter.

To create a source physical file (localesrc) in library localelib, follow these steps:

1. On a command line, type CRTSRCPF and press F4 (Prompt).
2. Type localesrc for the file name and press Enter.

### **Step 2. Copying an existing locale source definition**

Locale-source-definition file members that are IBM supplied are located in library QSYSLOCALE, source physical file QLOCALESRC. See System-supplied locale source definition files for a list of the IBM-supplied locale source files.

IBM provides sources for many sample locales. Besides the locales that IBM provides, an open source repository that contains locale sources is available. See the POSIX type files at the [Unicode Common Locale Data Repository \(CLDR\) Web site](#)  for more information.

In this example, member EN\_US, a locale for the English language, is copied.

To copy a locale source definition, follow these steps:

1. On a command line, type CPYF and press F4 (Prompt).
2. Type the values shown on the following display.

```

Copy File (CPYF)
Type choices, press Enter.
From file . . . . . QLOCALESRC      Name
Library . . . . . QSYSLOCALE      Name, *LIBL,*CURLIB
To file . . . . . localesrc_      Name, *PRINT
Library . . . . . localelib_     Name, *LIBL,*CURLIB
From member . . . . . EN_US       Name, generic*,*FIRST, *ALL
To member or label . . . . . EN_US_----- Name, *FIRST,*FROMMBR
Replace or add records . . . . . *ADD_----- *NONE, *ADD,*REPLACE
Create file . . . . . *YES       *NO, *YES
Record format field mapping . . . *MAP__      *NONE, *NOCHK,*CVTSRC

```

The values entered copy the EN\_US member to the source physical file localesrc in library localelib.

**Note:** When you copy a file that is tagged with a CCSID, you need to use the FMTOPT(\*MAP) parameter to ensure that the copied source is converted to the CCSID of the to-file. Scroll ahead to see the FMTOPT parameter.

### Step 3. Editing the copied locale source definition

If you want to use the IBM-supplied locale as it is included, you do not need to change it. You can continue with step 4, Creating the locale object.

In this example, edit the EN\_US member to set the time-of-day keywords used in the LC\_TOD category. To know more about the LC\_TOD category, see [“LC\\_TOD category” on page 209](#).

This example shows how to use Source Entry Utility (SEU) to edit the locale. You can also use an equivalent editor.

To edit the locale using SEU, follow these steps:

1. On a command line, type STRSEU (Start Source Entry Utility) and press F4 (Prompt).
2. Type the source file name (localesrc), library name (localelib), and source member name (EN\_US) as shown on the following display.

```

Start Source Entry Utility (STRSEU)
Type choices, press Enter.
Source file . . . . . localesrc      Name, *PRV
Library . . . . . localelib_     Name, *LIBL, *CURLIB, *PRV
Source member . . . . . EN_US_----- Name, *PRV, *SELECT

```

3. Press Enter. The following display appears.

```

COLUMNS . . . : 1 71          EDIT      localelib/localesrc
SEU==> F_LC_TOD      EN_US
FMT ** ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** BEGINNING OF DATA*****
5967.00 comment_char <percent-sign>
5968.00 escape_char <slash>
5969.00
5970.00 %
5971.00 % 5761SS1          (C) COPYRIGHT IBM CORP. 1991,2008
5972.00 % ALL RIGHTS RESERVED.
5973.00 % US GOVERNMENT USERS RESTRICTED RIGHTS -
5974.00 % USE, DUPLICATION OR DISCLOSURE RESTRICTED
5975.00 % BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
5976.00 %
5977.00 % LICENSED MATERIALS-PROPERTY OF IBM
5978.00 %
5979.00 % FILE NAME :      EN_US
5980.00 %
5981.00 % COUNTRY/REGION: UNITED STATES
5982.00 %

F3=EXIT  F4=PROMPT  F5=REFRESH  F9=RETRIEVE  F10=CURSOR  F11=TOGGLE
F16=REPEAT FIND  F17=REPEAT CHANGE  F24=MORE KEYS
(C) COPYRIGHT IBM CORP.1991, 2008.

```

4. Use the SEU search function to locate LC\_TOD. After the search is completed, the following display appears.

As you can see, all LC\_TOD category keywords have values of 0 and no descriptive names are declared for tname and dstname.

```

COLUMNS . . . : 1 71          EDIT      localelib/localesrc
SEU==>          EN_US
FMT ** ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
6519.00
6520.00 LC_TOD
6521.00
6522.00 tzdiff      0
6523.00 tname       ""
6524.00 dstname     ""
6525.00 dststart    0,0,0,0
6526.00 dstend     0,0,0,0
6527.00 dstshift   0
6528.00
6529.00 END LC_TOD
***** END OF DATA*****

F3=EXIT  F4=PROMPT  F5=REFRESH  F9=RETRIEVE  F10=CURSOR  F11=TOGGLE
F16=REPEAT FIND  F17=REPEAT CHANGE  F24=MORE KEYS
STRING LC_TOD FOUND.

```

5. To create the correct LC\_TOD values for CST and CDT in the locale (for 2007 and later years), type the following values for the LC\_TOD keywords:

**tzdiff**

Type 360.

This is the difference in the number of minutes between Greenwich meantime and the central time zone of the United States.

**tname**

Type CST.

**dstname**

Type CDT.

**dststart**

Type 3,2,1,7200.

This string of integers means that daylight saving time starts the third month of the year, the beginning of the second week, the first day of the week, and 7200 seconds (120 minutes) after midnight local standard time.

#### **dstend**

Type 10,3,6,0.

This string of integers means that daylight saving time ends the tenth month of the year, the beginning of the third week, the sixth day of the week, and 0 seconds after midnight local standard time.

#### **dstshift**

Type 3600.

For detailed information about the LC\_TOD keywords, see [“LC\\_TOD category” on page 209](#).

The SEU edit screen should match the following display.

6. Press Enter to make the changes to the EN\_US locale member.

```
COLUMNS . . . :    1  71          EDIT      localelib/localesrc
SEU==>          EN_US
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+...6 ...+... 7
6519.00
6520.00 LC_TOD
6521.00
6522.00 tzdiff    360
6523.00 tname     "CST"
6524.00 dstname   "CDT"
6525.00 dststart  3,2,1,7200
6526.00 dstend    11,1,1,7200
6527.00 dstshift  3600
6528.00
6529.00 END LC_TOD
***** END OF DATA *****

F3=EXIT  F4=PROMPT  F5=REFRESH  F9=RETRIEVE  F10=CURSOR  F11=TOGGLE
F16=REPEAT FIND   F17=REPEAT CHANGE  F24=MORE KEYS
STRING LC_TOD FOUND.
```

7. Save the member and exit SEU.

You have now copied the IBM-supplied locale source member and customized the LC\_TOD category.

### **Step 4. Creating the locale object**

To create the locale object, follow these steps:

1. On a command line, type CRTLOCALE and press F4 (Prompt). The Create Locale (CRTLOCALE) display appears.
2. In the Locale name field, type the locale path name (the path name includes the location and the name of the locale source member).
3. In the Source file path name field, type the source file path name (the location and name of the source physical file and the member name).

#### **Notes:**

- a. Make sure that the locale source file you are using has the same characters defined as does the CCSID you specify on the CRTLOCALE command. If they do not have the same characters available, unpredictable results can occur. For example, the EN\_US source file has lowercase letters defined. However, the lowercase letters are not available in CCSID 290.

- b. When you create locale objects, it is preferred to make the CCSID part of the locale object name. For example, EN\_US created with CCSID 37 can be named EN\_US37.

```

Create Locale (CRTLOCALE)
Type choices, press Enter.
Locale name . . . . . > '/qsys.lib/localelib.lib/en_us.locale'_____
----- file path name . . . . . > '/qsys.lib/localelib.lib/localesrc.file/en
_us.mbr'
Coded character set ID . . . . . > 37_____ 1-65533, *JOB
Generation severity level . . . . 10_____ 10, 20
Text 'description' . . . . . my version of locale EN_US - contains my c
hanges_

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

4. Press Enter to complete the creation of the locale object named EN\_US in the library LOCALELIB.

### Step 5. Enabling the locale object

You can enable locales system-wide by using the QLOCALE system value or for individual users by changing their user profile. To enable the locale system-wide, make EN\_US the value for the QLOCALE system value. In this example, the locale support is enabled for one user.

To enable the locale for one user, follow these steps:

1. On a command line, type CHGUSRPRF and press F4 (Prompt).
2. Specify the user ID of the user and press Enter.

In the following portion of the Change User Profile display, the LOCALE parameter now has a value indicating that EN\_US is the specified locale to be used by the user ID.

```

For more values
Locale . . . . . QSYS.LIB/LOCALELIB.LIB/EN_US.LOCALE_____
-----

```

After the user profile has been changed, any jobs initiated by the user ID have the EN\_US locale associated with those jobs. The LANG environment variable is also initialized to the name of the locale.

### Related concepts

#### Working with locales

Locales are used primarily in ILE-based application programs. Additionally, the Retrieve Locale Information (OPM, QLGRTVLC; ILE, QlgRetrieveLocaleInformation) API retrieves one or all categories of a locale.

#### System-supplied locales and recommended CCSIDs

The system-supplied locale source definition file members are in the optionally installable library QSYSLOCALE in the QLOCALESRC source file. The source file members are encoded in CCSID 37 and are read only.

#### LC\_TIME category

The LC\_TIME category of a locale definition source file defines rules and symbols for formatting time and date information. This category begins with an LC\_TIME category header and ends with an END LC\_TIME category trailer.

**Example: Producing unique monetary formats**

A unique customized monetary format can be produced by changing the value of a single statement.

For example, the following table shows the results of using all combinations of defined values for the p\_cs\_precedes, p\_sep\_by\_space, and p\_sign\_posn statements:

p_cs_precedes value	p_sign_posn value	p_sep_by_space=2	p_sep_by_space=3	p_sep_by_space=4
p_cs_precedes = 1	p_sign_posn = 0	(\$1.25)	(\$ 1.25)	(\$1.25)
	p_sign_posn = 1	+ \$1.25	+\$ 1.25	+\$1.25
	p_sign_posn = 2	\$1.25 +	\$ 1.25+	\$1.25+
	p_sign_posn = 3	+ \$1.25	+\$ 1.25	+\$1.25
	p_sign_posn = 4	\$ +1.25	+\$ 1.25	+\$1.25
p_cs_precedes = 0	p_sign_posn = 0	(1.25 \$)	(1.25 \$)	(1.25\$)
	p_sign_posn = 1	+1.25 \$	+1.25 \$	+1.25\$
	p_sign_posn = 2	1.25\$ +	1.25 \$+	1.25\$+
	p_sign_posn = 3	1.25+ \$	1.25 +\$	1.25+\$
	p_sign_posn = 4	1.25\$ +	1.25 \$+	1.25\$+

**Related concepts**

LC\_MONETARY category

The LC\_MONETARY category of a locale definition source file defines rules and symbols for formatting monetary numeric information. This category begins with an LC\_MONETARY category header and ends with an END LC\_MONETARY category trailer.

**Example: Locales as part of a multilingual environment**

The IBM i operating system, through the use of locales, user profiles, and subsystems, can provide a multilingual environment. Users of a system that is set up for multilingual environments can work with their national language and all of its cultural conventions (for example, the character used to separate hours, minutes, and seconds).

Assume that the system used in this example has its primary language defined as English and the secondary national language versions (NLVs) for French and Spanish have been installed.

Follow the steps in this example to perform the following tasks:

- Create the locales for English, French, and Spanish.
- Create user profiles for users named: English, French, and Spanish.
- Create separate subsystems for French and Spanish language users.

**Step 1. Creating locales**

1. Type CRTLOCALE and press the Prompt key (F4).
2. Enter these values for the following fields:
  - Locale name: qsys.lib/localelib.lib/en\_us.locale
  - Source file path name: qsys.lib/qsyslocale.lib/qlocalsrc.file/en\_us.mbr
  - Coded character set ID: 37



- Generation severity level: 20
- Text 'description': US English locale

3. Press Enter.

Repeat the CRTLOCALE command for the FRENCH and SPANISH locales, using these values for the following fields.

For the French locale:

- Locale name: qsys.lib/localelib.lib/fr\_fr.locale
- Source file path name: qsys.lib/qsyslocale.lib/qlocalesrc.file/fr\_fr.mbr
- Coded character set ID: 297
- Generation severity level: 20
- Text 'description': French locale

For the Spanish locale:

- Locale name: qsys.lib/localelib.lib/es\_es.locale
- Source file path name: qsys.lib/qsyslocale.lib/qlocalesrc.file/es\_es.mbr
- Coded character set ID: 284
- Generation severity level: 20
- Text 'description': Spanish locale

You have created three locales (EN\_US (English US), FR\_FR (French), and ES\_ES (Spanish)). They are stored in library localelib.lib.

## Step 2. Creating the user profile

In this example, three user profiles are created; each one will use one of the locales that was just created. The user profile names are English, French, and Spanish.

1. Type CRTUSRPRF and prompt (F4).
2. Type ENGLISH for the User profile parameter value.
3. Scroll forward until you see the Locale job attributes parameter and the Locale parameter.
4. Type:
  - /qsys.lib/localelib.lib/en\_us.locale for the Locale parameter value.
  - Type + for the Locale job attributes parameter value and press Enter. Type:

```
*CCSID
*DATFMT
*DATSEP
*TIMSEP
*SRTSEQ
*DECFMT
```

**Note:** At job start, the system finds the actual job attribute values defined in the locale object. The job attributes found in the locale override the values specified in the user profile fields for the CCSID and SRTSEQ parameters. They also override the date format, date separator, and time separator job attributes specified in any system value.

5. Press Enter. You have now created the user profile for a user named ENGLISH.

Repeat the CRTUSRPRF command for user IDs FRENCH and SPANISH. The next two displays provide the correct Locale parameter and Locale job attribute information for creating the user profiles for FRENCH and SPANISH.

```

Locale job attributes . . . . . > *CCSID          *SYSVAL,*NONE, *CCSID...
> *DATFMT
> *DATSEP
> *TIMSEP
> *SRTSEQ
Locale . . . . . >' /qsys.lib/localelib.lib/fr_fr.locale'

```

```

Locale job attributes . . . . . > *CCSID          *SYSVAL,*NONE, *CCSID...
> *DATFMT
> *DATSEP
> *TIMSEP
> *SRTSEQ
Locale . . . . . >' /qsys.lib/localelib.lib/es_es.locale'

```

### Step 3. Creating subsystems for each national language version

Subsystems can be tailored to provide users an environment in which they see their own national language with data presented in the cultural format and conventions they are used to seeing.

**Note:** Because the primary language of the system is English, it is not necessary to create a subsystem for English.

1. Type CRTSBSD and prompt (F4).
2. Specify values for the following parameters to ensure that the subsystem is enabled for a specific national language (such as French and Spanish in our example).

- Subsystem description

This can be any name you choose.

- Text 'description'

The description can be anything you want it to be.

- Sign-on display file and Library

This often is QDSIGNON. The important information here is to know the name of the library where the national language version (French in this example) is stored.

- Subsystem library

It specifies a library that is entered ahead of other libraries in the library list of jobs started in this subsystem. This parameter allows you to use a secondary language library causing messages and displays to appear in your spoken language.

**Note:** The correct values for Sign-on display file library and Subsystem library parameters are determined by adding QSYS to the national language version feature code. For example, the French national language library is named QSYS2928.

The following screen shows the correct values to ensure that users of the FRENCH subsystem interact with the computer in the French language.

```

Create Subsystem Description (CRTSBSD)
Type choices, press Enter.
Subsystem description . . . . . SBSDB          > FRENCH
Library . . . . .                               *CURLIB

```

```

Text 'description' . . . . . TEXT          > 'Subsystem for French users'
Additional Parameters
Sign-on display file . . . . . SGNDSPF    > QDSIGNON
Library . . . . .                   > QSYS2928
Subsystem library . . . . . SYSLIBLE     > QSYS2928
More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

3. Press Enter.

#### Step 4. More information about subsystems

Creation of a subsystem requires additional work such as:

- Setting subsystem attributes
- Adding workstation entries
- Adding job queue entries
- Adding communications entries (if your national language users are attached over communications lines)
- Adding autostart job entries if you want to use this feature
- Adding prestart job entries if you want to use this feature
- Creating a class
- Adding routing entries

How to perform the tasks in the previous list is not described in this example.

#### Related concepts

##### National language version feature codes

This table lists the available national language version feature codes on the IBM i operating system. When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code.

##### Work management

#### **Example: POSIX locale**

This example shows the POSIX (or C) locale categories and the source.

It is published in its entirety because:

- It provides a locale example with source provided for all categories.
- If you have not set a locale value in your C application program, the default POSIX locale is then used.

In either case, in the following listing, you are able to look at the locale categories and view the source.

**Note:** By using the code example, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```

comment_char <percent-sign>
escape_char  <slash>

%
% 5716SS1 (C) COPYRIGHT IBM(R) CORP. 1991,1996
% ALL RIGHTS RESERVED.
% US GOVERNMENT USERS RESTRICTED RIGHTS -
% USE, DUPLICATION OR DISCLOSURE RESTRICTED
% BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
%
% LICENSED MATERIALS-PROPERTY OF IBM
%
% FILE NAME : POSIX

```

```

%
% COUNTRY/REGION: POSIX DEFAULT LOCALE
%
% LANGUAGES(S): NOT SPECIFIED
%
% DESCRIPTION: LOCALE SOURCE DEFINITION FILE.
%

LC_CTYPE

upper <A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;<L>;<M>;/
<N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>

lower <a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
<n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>

space <tab>;<newline>;<vertical-tab>;<form-feed>;<carriage-return>;/
<space>

cntrl <NUL>;<SOH>;<STX>;<ETX>;<EOT>;<ENQ>;<ACK>;<alert>;<backspace>;/
<tab>;<newline>;<vertical-tab>;<form-feed>;<carriage-return>;/
<SO>;<SI>;<DLE>;<DC1>;<DC2>;<DC3>;<DC4>;<NAK>;<SYN>;<ETB>;/
<CAN>;<EM>;<SUB>;<ESC>;<IS4>;<IS3>;<IS2>;<IS1>;<DEL>

punct <exclamation-mark>;<quotation-mark>;<number-sign>;/
<dollar-sign>;<percent-sign>;<ampersand>;<apostrophe>;/
<left-parenthesis>;<right-parenthesis>;<asterisk>;<plus-sign>;/
<comma>;<hyphen>;<period>;<slash>;/
<colon>;<semicolon>;<less-than-sign>;/
<equals-sign>;<greater-than-sign>;<question-mark>;/
<commercial-at>;/
<left-square-bracket>;<backslash>;/
<right-square-bracket>;<circumflex>;/
<underscore>;<grave-accent>;/
<left-curly-bracket>;<vertical-line>;<right-curly-bracket>;/
<tilde>

digit <zero>;<one>;<two>;<three>;<four>;/
<five>;<six>;<seven>;<eight>;<nine>

xdigit <zero>;<one>;<two>;<three>;<four>;/
<five>;<six>;<seven>;<eight>;<nine>;/
<A>;<B>;<C>;<D>;<E>;<F>;/
<a>;<b>;<c>;<d>;<e>;<f>

blank <space>;/
<tab>

toupper (<a>, <A>); (<b>, <B>); (<c>, <C>); (<d>, <D>); (<e>, <E>);/
(<f>, <F>); (<g>, <G>); (<h>, <H>); (<i>, <I>); (<j>, <J>);/
(<k>, <K>); (<l>, <L>); (<m>, <M>); (<n>, <N>); (<o>, <O>);/
(<p>, <P>); (<q>, <Q>); (<r>, <R>); (<s>, <S>); (<t>, <T>);/
(<u>, <U>); (<v>, <V>); (<w>, <W>); (<x>, <X>); (<y>, <Y>);/
(<z>, <Z>)

tolower (<A>, <a>); (<B>, <b>); (<C>, <c>); (<D>, <d>); (<E>, <e>);/
(<F>, <f>); (<G>, <g>); (<H>, <h>); (<I>, <i>); (<J>, <j>);/
(<K>, <k>); (<L>, <l>); (<M>, <m>); (<N>, <n>); (<O>, <o>);/
(<P>, <p>); (<Q>, <q>); (<R>, <r>); (<S>, <s>); (<T>, <t>);/
(<U>, <u>); (<V>, <v>); (<W>, <w>); (<X>, <x>); (<Y>, <y>);/
(<Z>, <z>)

END LC_CTYPE

LC_COLLATE

order_start

<NUL>
<SOH>
<STX>
<ETX>
<EOT>
<ENQ>
<ACK>
<alert>
<backspace>
<tab>
<newline>

```

<vertical-tab>  
<form-feed>  
<carriage-return>  
<SO>  
<SI>  
<DLE>  
<DC1>  
<DC2>  
<DC3>  
<DC4>  
<NAK>  
<SYN>  
<ETB>  
<CAN>  
<EM>  
<SUB>  
<ESC>  
<IS4>  
<IS3>  
<IS2>  
<IS1>  
<space>  
<exclamation-mark>  
<quotation-mark>  
<number-sign>  
<dollar-sign>  
<percent-sign>  
<ampersand>  
<apostrophe>  
<left-parenthesis>  
<right-parenthesis>  
<asterisk>  
<plus-sign>  
<comma>  
<hyphen>  
<period>  
<slash>  
<zero>  
<one>  
<two>  
<three>  
<four>  
<five>  
<six>  
<seven>  
<eight>  
<nine>  
<colon>  
<semicolon>  
<less-than-sign>  
<equals-sign>  
<greater-than-sign>  
<question-mark>  
<commercial-at>  
<A>  
<B>  
<C>  
<D>  
<E>  
<F>  
<G>  
<H>  
<I>  
<J>  
<K>  
<L>  
<M>  
<N>  
<O>  
<P>  
<Q>  
<R>  
<S>  
<T>  
<U>  
<V>  
<W>  
<X>  
<Y>  
<Z>  
<left-square-bracket>  
<backslash>

```

<right-square-bracket>
<circumflex>
<underscore>
<grave-accent>
<a>
<b>
<c>
<d>
<e>
<f>
<g>
<h>
<i>
<j>
<k>
<l>
<m>
<n>
<o>
<p>
<q>
<r>
<s>
<t>
<u>
<v>
<w>
<x>
<y>
<z>
<left-curly-bracket>
<vertical-line>
<right-curly-bracket>
<tilde>
<DEL>
UNDEFINED

order_end

END LC_COLLATE

LC_MONETARY

int_curr_symbol ""
currency_symbol ""
mon_decimal_point ""
mon_thousands_sep ""
mon_grouping -1
positive_sign ""
negative_sign ""
int_frac_digits -1
frac_digits -1
p_cs_precedes -1
p_sep_by_space -1
n_cs_precedes -1
n_sep_by_space -1
p_sign_posn -1
n_sign_posn -1

END LC_MONETARY

LC_NUMERIC

decimal_point "<period>" thousands_sep
"" grouping -1

END LC_NUMERIC

LC_TIME

abday "<S><u><n>";/
"<M><o><n>";/
"<T><u><e>";/
"<W><e><d>";/
"<T><h><u>";/
"<F><r><i>";/
"<S><a><t>"

day "<S><u><n><d><a><y>";/
"<M><o><n><d><a><y>";/
"<T><u><e><s><d><a><y>";/
"<W><e><d><n><e><s><d><a><y>";/

```

```

"<T><h><u><r><s><d><a><y>" ;/
"<F><r><i><d><a><y>" ;/
"<S><a><t><u><r><d><a><y>"

abmon "<J><a><n>" ;/
"<F><e><b>" ;/
"<M><a><r>" ;/
"<A><p><r>" ;/
"<M><a><y>" ;/
"<J><u><n>" ;/
"<J><u><l>" ;/
"<A><u><g>" ;/
"<S><e><p>" ;/
"<O><c><t>" ;/
"<N><o><v>" ;/
"<D><e><c>"

mon "<J><a><n><u><a><r><y>" ;/
"<F><e><b><r><u><a><r><y>" ;/
"<M><a><r><c><h>" ;/
"<A><p><r><i><l>" ;/
"<M><a><y>" ;/
"<J><u><n><e>" ;/
"<J><u><l><y>" ;/
"<A><u><g><u><s><t>" ;/
"<S><e><p><t><e><m><b><e><r>" ;/
"<O><c><t><o><b><e><r>" ;/
"<N><o><v><e><m><b><e><r>" ;/
"<D><e><c><e><m><b><e><r>"

d_t_fmt "%a %b %d %H:%M:%S %Z %Y"

d_fmt "%m/%d/%y"

t_fmt "%H:%M:%S"

am_pm "<A><M>" ; "<P><M>"

t_fmt_ampm "%I:%M:%S %p"

END LC_TIME

LC_MESSAGES

yesexpr "[yY][eE][sS] | [yY]"
noexpr "[nN][oO] | [nN]"
yesstr "yes"
nostr "no"

END LC_MESSAGES

LC_TOD

tzdiff 0
tname ""
dstname ""
dststart 0,0,0,0
dstend 0,0,0,0
dstshift 0

END LC_TOD

```

## Related concepts

### Locale categories

This table describes the locale categories that are supported on the IBM i operating system.

#### **Example: EN\_US locale**

In this example, you can find the locale categories and see the source.

**Note:** By using the code example, you agree to the terms of the [“Code license and disclaimer information”](#) on page 588.

```

comment_char <percent-sign>
escape_char <slash>

```

```

%
% 5716SS1          (C) COPYRIGHT IBM(R) CORP. 1991,1996
% ALL RIGHTS RESERVED.
% US GOVERNMENT USERS RESTRICTED RIGHTS -
% USE, DUPLICATION OR DISCLOSURE RESTRICTED
% BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
%
% LICENSED MATERIALS-PROPERTY OF IBM
%
% FILE NAME   :   EN_US
%
% COUNTRY/REGION: UNITED STATES
%
% LANGUAGES(S):   ENGLISH
%
% DESCRIPTION:   LOCALE SOURCE DEFINITION FILE.
%

```

LC\_CTYPE

```

upper  <A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;<L>;<M>;/
<N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>;/
<A-acute>;<A-grave>;<A-circumflex>;<A-diaresis>;/
<A-tilde>;<A-ring>;<AE>;<C-cedilla>;<Eth>;<E-acute>;/
<E-grave>;<E-circumflex>;<E-diaresis>;<I-acute>;/
<I-grave>;<I-circumflex>;<I-diaresis>;<N-tilde>;/
<O-acute>;<O-grave>;<O-circumflex>;<O-diaresis>;/
<O-tilde>;<O-slash>;<Thorn>;<U-acute>;<U-grave>;/
<U-circumflex>;<U-diaresis>;<Y-acute>

lower  <a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
<n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>;/
<a-acute>;<a-grave>;<a-circumflex>;<a-diaresis>;/
<a-tilde>;<a-ring>;<ae>;<c-cedilla>;<eth>;<e-acute>;/
<e-grave>;<e-circumflex>;<e-diaresis>;<i-acute>;/
<i-grave>;<i-circumflex>;<i-diaresis>;<n-tilde>;/
<o-acute>;<o-grave>;<o-circumflex>;<o-diaresis>;/
<o-tilde>;<o-slash>;<s-sharp>;<thorn>;<u-acute>;/
<u-grave>;<u-circumflex>;<u-diaresis>;<y-acute>;/
<y-diaresis>

space  <tab>;<newline>;<vertical-tab>;<form-feed>;<carriage-return>;/
<space>

cntrl  <NUL>;<SOH>;<STX>;<ETX>;<EOT>;<ENQ>;<ACK>;<alert>;<backspace>;/
<tab>;<newline>;<vertical-tab>;<form-feed>;<carriage-return>;/
<SO>;<SI>;<DLE>;<DC1>;<DC2>;<DC3>;<DC4>;<NAK>;<SYN>;<ETB>;/
<CAN>;<EM>;<SUB>;<ESC>;<IS4>;<IS3>;<IS2>;<IS1>;<DEL>;/
<DS>;<SOS>;<FS>;<WUS>;<BYP>;<NL>;<RNL>;<POC>;<SA>;<SFE>;<SM>;/
<CSP>;<MFA>;<SPS>;<RPT>;<CU1>;<DCS>;<PU1>;<UBS>;<IR>;<PP>;/
<TRN>;<NBS>;<GE>;<SBS>;<IT>;<RFF>;<CU3>;<SEL>;<RES>;<PM>;<EO>

graph  <exclamation-mark>;<quotation-mark>;<number-sign>; /
<dollar-sign>;<percent-sign>;<ampersand>;<apostrophe>; /
<left-parenthesis>;<right-parenthesis>;<asterisk>;<plus-sign>;/
<comma>;<hyphen-minus>;<period>;<slash>;/
<zero>;<one>;<two>;<three>;<four>;<five>;<six>;<seven>;/
<eight>;<nine>;<colon>;<semicolon>;<less-than-sign>; /
<equals-sign>;<greater-than-sign>;<question-mark>;/
<commercial-at>;<A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;/
<L>;<M>;<N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>;/
<left-square-bracket>;<backslash>;/
<right-square-bracket>;<circumflex>;/
<underscore>;<grave-accent>;/
<a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
<n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>;/
<left-brace>;<vertical-line>;<right-brace>;/
<tilde>;<C-cedilla>;<u-diaresis>;<e-acute>;<a-circumflex>;/
<a-diaresis>;<a-grave>;<a-ring>;<c-cedilla>;<e-circumflex>;/
<e-diaresis>;<e-grave>;<i-diaresis>;<i-circumflex>;/
<i-grave>;<A-diaresis>;<A-ring>;<E-acute>;<ae>;<AE>;/
<o-circumflex>;<o-diaresis>;<o-grave>;<u-circumflex>;/
<u-grave>;<y-diaresis>;<o-diaresis>;<U-diaresis>;<o-slash>;/
<sterling>;<o-slash>;<multiply>;<a-acute>;<i-acute>;/
<o-acute>;<u-acute>;<n-tilde>;<N-tilde>;<feminine>;/
<masculine>;<question-down>;<registered>;<not>;<one-half>;/
<one-quarter>;<exclamation-down>;<guillemot-left>;/
<guillemot-right>;<A-acute>;<A-circumflex>;<A-grave>;/
<copyright>;<cent>;<yen>;<a-tilde>;<A-tilde>;<currency>;/
<eth>;<Eth>;<E-circumflex>;<E-diaresis>;<E-grave>;/
<I-acute>;<I-circumflex>;<I-diaresis>;<broken-bar>;/

```



```

<I-grave>;<O-acute>;<s-sharp>;<O-circumflex>;/
<O-grave>;<o-tilde>;<O-tilde>;<mu>;<thorn>;<Thorn>;<U-acute>;/
<U-circumflex>;<U-grave>;<y-acute>;<Y-acute>;<macron>;/
<acute>;<hyphen>;<plus-minus>;<three-quarters>;<paragraph>;/
<section>;<divide>;<cedilla>;<degree>;<diaeresis>;<dot>;/
<one-superior>;<three-superior>;<two-superior>

print <space>;<exclamation-mark>;<quotation-mark>;<number-sign>; /
<dollar-sign>;<percent-sign>;<ampersand>;<apostrophe>; /
<left-parenthesis>;<right-parenthesis>;<asterisk>;<plus-sign>;/
<comma>;<hyphen-minus>;<period>;<slash>;/
<zero>;<one>;<two>;<three>;<four>;<five>;<six>;<seven>;/
<eight>;<nine>;<colon>;<semicolon>;<less-than-sign>; /
<equals-sign>;<greater-than-sign>;<question-mark>;/
<commercial-at>;<A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;/
<L>;<M>;<N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>;/
<left-square-bracket>;<backslash>;/
<right-square-bracket>;<circumflex>;/
<underscore>;<grave-accent>;/
<a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
<n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>;/
<left-brace>;<vertical-line>;<right-brace>;/
<tilde>;<C-cedilla>;<u-diaeresis>;<e-acute>;<a-circumflex>;/
<a-diaeresis>;<a-grave>;<a-ring>;<c-cedilla>;<e-circumflex>;/
<e-diaeresis>;<e-grave>;<i-diaeresis>;<i-circumflex>;/
<i-grave>;<A-diaeresis>;<A-ring>;<E-acute>;<ae>;<AE>;/
<o-circumflex>;<o-diaeresis>;<o-grave>;<u-circumflex>;/
<u-grave>;<y-diaeresis>;<O-diaeresis>;<U-diaeresis>;<o-slash>;/
<sterling>;<O-slash>;<multiply>;<a-acute>;<i-acute>;/
<o-acute>;<u-acute>;<n-tilde>;<N-tilde>;<feminine>;/
<masculine>;<question-down>;<registered>;<not>;<one-half>;/
<one-quarter>;<exclamation-down>;<guillemot-left>;/
<guillemot-right>;<A-acute>;<A-circumflex>;<A-grave>;/
<copyright>;<cent>;<yen>;<a-tilde>;<A-tilde>;<currency>;/
<eth>;<Eth>;<E-circumflex>;<E-diaeresis>;<E-grave>;/
<I-acute>;<I-circumflex>;<I-diaeresis>;<broken-bar>;/
<I-grave>;<O-acute>;<s-sharp>;<O-circumflex>;/
<O-grave>;<o-tilde>;<O-tilde>;<mu>;<thorn>;<Thorn>;<U-acute>;/
<U-circumflex>;<U-grave>;<y-acute>;<Y-acute>;<macron>;/
<acute>;<hyphen>;<plus-minus>;<three-quarters>;<paragraph>;/
<section>;<divide>;<cedilla>;<degree>;<diaeresis>;<dot>;/
<one-superior>;<three-superior>;<two-superior>

punct <exclamation-mark>;<quotation-mark>;<number-sign>; /
<dollar-sign>;<percent-sign>;<ampersand>;<apostrophe>; /
<left-parenthesis>;<right-parenthesis>;<asterisk>;<plus-sign>;/
<comma>;<hyphen-minus>;<period>;<slash>;/
<colon>;<semicolon>;<less-than-sign>; /
<equals-sign>;<greater-than-sign>;<question-mark>;/
<commercial-at>;/
<left-square-bracket>;<backslash>;/
<right-square-bracket>;<circumflex>;/
<underscore>;<grave-accent>;/
<left-brace>;<vertical-line>;<right-brace>;/
<tilde>

digit <zero>;<one>;<two>;<three>;<four>;/
<five>;<six>;<seven>;<eight>;<nine>

xdigit <zero>;<one>;<two>;<three>;<four>;/
<five>;<six>;<seven>;<eight>;<nine>;/
<A>;<B>;<C>;<D>;<E>;<F>;/
<a>;<b>;<c>;<d>;<e>;<f>

blank <space>;/
<tab>

toupper (<a>,<A>);(<b>,<B>);(<c>,<C>);(<d>,<D>);(<e>,<E>);/
(<f>,<F>);(<g>,<G>);(<h>,<H>);(<i>,<I>);(<j>,<J>);/
(<k>,<K>);(<l>,<L>);(<m>,<M>);(<n>,<N>);(<o>,<O>);/
(<p>,<P>);(<q>,<Q>);(<r>,<R>);(<s>,<S>);(<t>,<T>);/
(<u>,<U>);(<v>,<V>);(<w>,<W>);(<x>,<X>);(<y>,<Y>);/
(<z>,<Z>);(<a-acute>,<A-acute>);(<a-grave>,<A-grave>);/
(<a-circumflex>,<A-circumflex>);(<a-diaeresis>,<A-diaeresis>);/
(<a-tilde>,<A-tilde>);(<a-ring>,<A-ring>);(<ae>,<AE>);/
(<c-cedilla>,<C-cedilla>);(<eth>,<Eth>);(<e-acute>,<E-acute>);/
(<e-grave>,<E-grave>);(<e-circumflex>,<E-circumflex>);/
(<e-diaeresis>,<E-diaeresis>);(<i-acute>,<I-acute>);/
(<i-grave>,<I-grave>);(<i-circumflex>,<I-circumflex>);/
(<i-diaeresis>,<I-diaeresis>);(<n-tilde>,<N-tilde>);/
(<o-acute>,<O-acute>);(<o-grave>,<O-grave>);/
(<o-circumflex>,<O-circumflex>);(<o-diaeresis>,<O-diaeresis>);/

```

```

(<o-tilde>,<O-tilde>);(<o-slash>,<O-slash>);(<thorn>,<Thorn>);/
(<u-acute>,<U-acute>);(<u-grave>,<U-grave>);/
(<u-circumflex>,<U-circumflex>);(<u-diaresis>,<U-diaresis>);/
(<y-acute>,<Y-acute>);(<y-diaresis>,<Y>)

tolower (<A>,<a>);(<B>,<b>);(<C>,<c>);(<D>,<d>);(<E>,<e>);/
(<F>,<f>);(<G>,<g>);(<H>,<h>);(<I>,<i>);(<J>,<j>);/
(<K>,<k>);(<L>,<l>);(<M>,<m>);(<N>,<n>);(<O>,<o>);/
(<P>,<p>);(<Q>,<q>);(<R>,<r>);(<S>,<s>);(<T>,<t>);/
(<U>,<u>);(<V>,<v>);(<W>,<w>);(<X>,<x>);(<Y>,<y>);/
(<Z>,<z>);(<A-acute>,<a-acute>);(<A-grave>,<a-grave>);/
(<A-circumflex>,<a-circumflex>);(<A-diaresis>,<a-diaresis>);/
(<A-tilde>,<a-tilde>);(<A-ring>,<a-ring>);(<AE>,<ae>);/
(<C-cedilla>,<c-cedilla>);(<Eth>,<eth>);(<E-acute>,<e-acute>);/
(<E-grave>,<e-grave>);(<E-circumflex>,<e-circumflex>);/
(<E-diaresis>,<e-diaresis>);(<I-acute>,<i-acute>);/
(<I-grave>,<i-grave>);(<I-circumflex>,<i-circumflex>);/
(<I-diaresis>,<i-diaresis>);(<N-tilde>,<n-tilde>);/
(<O-acute>,<o-acute>);(<O-grave>,<o-grave>);/
(<O-circumflex>,<o-circumflex>);(<O-diaresis>,<o-diaresis>);/
(<O-tilde>,<o-tilde>);(<O-slash>,<o-slash>);(<Thorn>,<thorn>);/
(<U-acute>,<u-acute>);(<U-grave>,<u-grave>);/
(<U-circumflex>,<u-circumflex>);(<U-diaresis>,<u-diaresis>);/
(<Y-acute>,<y-acute>)

```

END LC\_CTYPE

LC\_COLLATE

order\_start

```

<NUL>
<SOH>
<STX>
<ETX>
<SEL>
<tab>
<RNL>
<DEL>
<GE>
<SPS>
<RPT>
<vertical-tab>
<form-feed>
<carriage-return>
<SO>
<SI>
<DLE>
<DC1>
<DC2>
<DC3>
<RES>
<NL>
<backspace>
<POC>
<CAN>
<EM>
<UBS>
<CU1>
<IS4>
<IS3>
<IS2>
<IS1>
<DS>
<SOS>
<FS>
<WUS>
<BYP>
<newline>
<ETB>
<ESC>
<SA>
<SFE>
<SM>
<CSP>
<MFA>
<ENQ>
<ACK>
<alert>
<SYN>
<IR>

```

<PP>  
<TRN>  
<NBS>  
<EOT>  
<SBS>  
<IT>  
<RFF>  
<CU3>  
<DC4>  
<NAK>  
<SUB>  
<EO>  
<space>  
<underscore>  
<macron>  
<hyphen>  
<hyphen-minus>  
<comma>  
<semicolon>  
<colon>  
<exclamation-mark>  
<exclamation-down>  
<question-mark>  
<question-down>  
<slash>  
<period>  
<acute>  
<grave-accent>  
<circumflex>  
<diaeresis>  
<tilde>  
<dot>  
<cedilla>  
<apostrophe>  
<quotation-mark>  
<guillemot-left>  
<guillemot-right>  
<left-parenthesis>  
<right-parenthesis>  
<left-square-bracket>  
<right-square-bracket>  
<left-brace>  
<right-brace>  
<section>  
<paragraph>  
<copyright>  
<registered>  
<commercial-at>  
<currency>  
<cent>  
<dollar-sign>  
<sterling>  
<yen>  
<asterisk>  
<backslash>  
<ampersand>  
<number-sign>  
<percent-sign>  
<plus-sign>  
<plus-minus>  
<divide>  
<multiply>  
<less-than-sign>  
<equals-sign>  
<greater-than-sign>  
<not>  
<vertical-line>  
<broken-bar>  
<degree>  
<mu>  
<nobreakspace>  
<zero>  
<one-quarter>  
<one-half>  
<three-quarters>  
<one>  
<one-superior>  
<two>  
<two-superior>  
<three>  
<three-superior>  
<four>

<five>  
<six>  
<seven>  
<eight>  
<nine>  
<a>  
<A>  
<a-acute>  
<A-acute>  
<feminine>  
<a-grave>  
<A-grave>  
<a-circumflex>  
<A-circumflex>  
<a-ring>  
<A-ring>  
<a-diaresis>  
<A-diaresis>  
<a-tilde>  
<A-tilde>  
<ae>  
<AE>  
<b>  
<B>  
<c>  
<C>  
<c-cedilla>  
<C-cedilla>  
<d>  
<D>  
<eth>  
<Eth>  
<e>  
<E>  
<e-acute>  
<E-acute>  
<e-grave>  
<E-grave>  
<e-circumflex>  
<E-circumflex>  
<e-diaresis>  
<E-diaresis>  
<f>  
<F>  
<g>  
<G>  
<h>  
<H>  
<i-dotless>  
<i>  
<I>  
<i-acute>  
<I-acute>  
<i-grave>  
<I-grave>  
<i-circumflex>  
<I-circumflex>  
<i-diaresis>  
<I-diaresis>  
<j>  
<J>  
<k>  
<K>  
<l>  
<L>  
<m>  
<M>  
<n>  
<N>  
<n-tilde>  
<N-tilde>  
<o>  
<O>  
<masculine>  
<o-acute>  
<O-acute>  
<o-grave>  
<O-grave>  
<o-circumflex>  
<O-circumflex>  
<o-diaresis>  
<O-diaresis>

```

<o-tilde>
<O-tilde>
<o-slash>
<O-slash>
<p>
<q>
<Q>
<r>
<R>
<s>
<S>
<s-sharp>
<t>
<T>
<thorn>
<Thorn>
<u>
<U>
<u-acute>
<U-acute>
<u-grave>
<U-grave>
<u-circumflex>
<U-circumflex>
<u-diaresis>
<U-diaresis>
<v>
<V>
<w>
<W>
<x>
<X>
<y>
<Y>
<y-acute>
<Y-acute>
<y-diaresis>
<z>
<Z>
UNDEFINED

order_end

END LC_COLLATE

LC_MONETARY

int_curr_symbol    "<U><S><D><space>"
currency_symbol    "<dollar-sign>"
mon_decimal_point  "<period>"
mon_thousands_sep "<comma>"
mon_grouping       3
positive_sign      ""
negative_sign      "<hyphen-minus>"
int_frac_digits    2
frac_digits        2
p_cs_precedes      1
p_sep_by_space     0
n_cs_precedes      1
n_sep_by_space     0
p_sign_posn        2
n_sign_posn        2

END LC_MONETARY

LC_NUMERIC

decimal_point      "<period>"
thousands_sep     "<comma>"
grouping           3

END LC_NUMERIC

LC_TIME

abday    "<S><u><n>";/
" <M><o><n>";/
" <T><u><e>";/
" <W><e><d>";/

```

```

"<T><h><u>";/
"<F><r><i>";/
"<S><a><t>"

day      "<S><u><n><d><a><y>";/
"<M><o><n><d><a><y>";/
"<T><u><e><s><d><a><y>";/
"<W><e><d><n><e><s><d><a><y>";/
"<T><h><u><r><s><d><a><y>";/
"<F><r><i><d><a><y>";/
"<S><a><t><u><r><d><a><y>"

abmon    "<J><a><n>";/
"<F><e><b>";/
"<M><a><r>";/
"<A><p><r>";/
"<M><a><y>";/
"<J><u><n>";/
"<J><u><l>";/
"<A><u><g>";/
"<S><e><p>";/
"<O><c><t>";/
"<N><o><v>";/
"<D><e><c>"

mon
"<J><a><n><u><a><r><y>";/
"<F><e><b><r><u><a><r><y>";/
"<M><a><r><c><h>";/
"<A><p><r><i><l>";/
"<M><a><y>";/
"<J><u><n><e>";/
"<J><u><l><y>";/
"<A><u><g><u><s><t>";/
"<S><e><p><t><e><m><b><e><r>";/
"<O><c><t><o><b><e><r>";/
"<N><o><v><e><m><b><e><r>";/
"<D><e><c><e><m><b><e><r>"

d_t_fmt  "%a %b %e %H:%M:%S %Z %Y"
d_fmt    "%m//%d//%y"
t_fmt    "%H:%M:%S"
am_pm    "<A><M>"; "<P><M>"

END LC_TIME

LC_MESSAGES

yesexpr  "[yY][eE][sS]|[yY]"
noexpr   "[nN][oO]|[nN]"
yesstr   "yes:y:Y"
nostr    "no:n:N"

END LC_MESSAGES

LC_TOD

tzdiff   0
tname    ""
dstname  ""
dststart 0,0,0,0
dstend   0,0,0,0
dstshift 0

END LC_TOD

```

## Globalization reference information

Here is a comprehensive collection of supporting information about the concepts and tasks described in the Globalization topic collection.

### National language version feature codes

This table lists the available national language version feature codes on the IBM i operating system. When you order an IBM i licensed program, you identify the national language version you want by specifying a language feature code.

National language version	Primary language feature code	Secondary language feature code
Albanian	2995	5595
Arabic	2954	5554
Belgian Dutch	2963	5563
Belgian English	2909	5509
Belgian French	2966	5566
Brazilian Portuguese	2980	5580
Bulgarian	2974	5574
Canadian French	2981	5581
Croatian	2912	5512
Czech	2975	5575
Danish	2926	5526
Dutch Netherlands	2923	5523
English	2924	5524
English Uppercase Support for Double-Byte Character Set (DBCS)	2938	5538
English Uppercase and Lowercase Support for Double-Byte Character Set (DBCS)	2984	5584
Estonian	2902	5502
Farsi	2998	5598
Finnish	2925	5525
French	2928	5528
French Multinational Character Set	2940	5540
German	2929	5529
German Multinational Character Set	2939	5539
Greek	2957	5557
Hebrew	2961	5561

National language version	Primary language feature code	Secondary language feature code
Hungarian	2976	5576
Icelandic	2958	5558
Italian	2932	5532
Italian Multinational Character Set	2942	5542
Japanese Universal	2930	5530
Japanese Kanji	2962	5562
Korean	2986	5586
Lao	2906	5506
Latvian	2904	5504
Lithuanian	2903	5503
Macedonian	2913	5513
Norwegian	2933	5533
Polish	2978	5578
Portuguese	2922	5522
Portuguese Multinational Character Set	2996	5596
Romanian	2992	5592
Russian	2979	5579
Serbian Cyrillic	2914	5514
Simplified Chinese	2989	5589
Slovakian	2994	5594
Slovenian	2911	5511
Spanish	2931	5531
Swedish	2937	5537
Thai	2972	5572
Traditional Chinese	2987	5587
Turkish	2956	5556
Vietnamese	2905	5505

### Related concepts

#### National language version

A *national language version (NLV)* is a version of the IBM i operating system that contains a predefined set of language-dependent values, such as date format, time format, and sort sequence, for a particular language.

#### Setting up IBM i with a national language version



The steps to install and configure a national language version on the IBM i operating system include selecting and installing hardware, installing software, and configuring your environment to run in a globalized setting.

Configuring secondary languages

A secondary language consists of textual data for all licensed programs supported for a national language version.

Example: Locales as part of a multilingual environment

The IBM i operating system, through the use of locales, user profiles, and subsystems, can provide a multilingual environment. Users of a system that is set up for multilingual environments can work with their national language and all of its cultural conventions (for example, the character used to separate hours, minutes, and seconds).

**Related information**

Installing, upgrading, or deleting IBM i and related software

**Country and region identifiers**

This table lists the country and region identifiers.

Country and region name	Country and region ID
Afghanistan	AF
Albania	AL
Algeria	DZ
American Samoa	AS
Andorra	AD
Angola	AO
Anguilla	AI
Antarctica	AQ
Antigua and Barbuda	AG
Arabic speaking countries	AA
Argentina	AR
Armenia	AM
Aruba	AW
Australia	AU
Austria	AT
Azerbaijan	AZ
Bahamas	BS
Bahrain	BH
Bangladesh	BD
Barbados	BB
Belarus	BY
Belgium	BE
Belize	BZ
Benin	BJ

<b>Country and region name</b>	<b>Country and region ID</b>
Bermuda	BM
Bhutan	BT
Bolivia	BO
Bosnia/Herzegovina	BA
Botswana	BW
Bouvet island	BV
Brazil	BR
British Indian ocean territory	IO
Brunei Darussalam	BN
Bulgaria	BG
Burkina Faso	BF
Burundi	BI
Burma	BU
Cambodia	KH
Cameroon, United Republic of	CM
Canada	CA
Cape Verde	CV
Cayman islands	KY
Central African Republic	CF
Chad	TD
Chile	CL
China	CN
China (Hong Kong S.A.R.)	HK
China (Macao S.A.R.)	MO
Christmas island	CX
Cocos (Keeling) islands	CC
Colombia	CO
Comoros	KM
Congo	CG
Cook islands	CK
Costa Rica	CR
Ivory coast	CI
Croatia	HR
Cuba	CU
Cyprus	CY

<b>Country and region name</b>	<b>Country and region ID</b>
Czech Republic	CZ
Denmark	DK
Djibouti	DJ
Dominica	DM
Dominican Republic	DO
East Timor	TP
Ecuador	EC
Egypt	EG
El Salvador	SV
Equatorial Guinea	GQ
Estonia	EE
Ethiopia	ET
Falkland islands (Malvinas)	FK
Faroe islands	FO
Fiji	FJ
Finland	FI
France	FR
French Guiana	GF
French Polynesia	PF
French southern territories	TF
Gabon	GA
Gambia	GM
Georgia	GE
Germany	DE
Ghana	GH
Gibraltar	GI
Greece	GR
Greenland	GL
Grenada	GD
Guadeloupe	GP
Guam	GU
Guatemala	GT
Guinea	GN
Guinea-Bissau	GW
Guyana	GY

<b>Country and region name</b>	<b>Country and region ID</b>
Haiti	HT
Heard and McDonald islands	HM
Honduras	HN
Hungary	HU
Iceland	IS
India	IN
Indonesia	ID
Iran (Islamic Republic of)	IR
Iraq	IQ
Ireland	IE
Israel	IL
Italy	IT
Jamaica	JM
Japan	JP
Jordan	JO
Kazakhstan	KK
Kenya	KE
Kiribati	KI
Korea, Democratic People's Republic of	KP
Korea, Republic of	KR
Kuwait	KW
Kyrgyzstan	KG
Lao People's Democratic Republic	LA
Latvia	LV
Lebanon	LB
Lesotho	LS
Liberia	LR
Libyan Arab Jamahiriya	LY
Liechtenstein	LI
Lithuania	LT
Luxembourg	LU
Macedonia	MK
Madagascar	MG
Malawi	MW
Malaysia	MY

<b>Country and region name</b>	<b>Country and region ID</b>
Maldives	MV
Mali	ML
Malta	MT
Marshall islands	MH
Martinique	MQ
Mauritania	MR
Mauritius	MU
Mexico	MX
Micronesia	FM
Moldava, Republic of	MD
Monaco	MC
Mongolia	MN
Montenegro	ME
Montserrat	MS
Morocco	MA
Mozambique	MZ
Myanmar	MM
Namibia	NA
Nauru	NR
Nepal	NP
Netherlands	NL
Netherlands Antilles	AN
New Caledonia	NC
Neutral zone	NT
New Zealand	NZ
Nicaragua	NI
Niger	NE
Nigeria	NG
Niue	NU
Norfolk island	NF
Northern Mariana islands	MP
Norway	NO
Oman	OM
Pakistan	PK
Palau	PW

<b>Country and region name</b>	<b>Country and region ID</b>
Panama	PA
Papua New Guinea	PG
Paraguay	PY
Peru	PE
Philippines	PH
Pitcairn	PN
Poland	PL
Portugal	PT
Puerto Rico	PR
Qatar	QA
Reunion	RE
Romania	RO
Russia	RU
Rwanda	RW
Saint. Helena	SH
Saint Kitts and Nevis	KN
Saint Lucia	LC
Saint Pierre and Miquelon	PM
Saint Vincent and the Grenadines.	VC
Western Samoa	WS
San Marino	SM
Sao Tome and Principe	ST
Saudi Arabia	SA
Senegal	SN
Seychelles	SC
Sierra Leone	SL
Serbia	SQ
Singapore	SG
Slovakia	SK
Slovenia	SI
Solomon islands	SB
Somalia	SO
South Africa	ZA
Spain	ES
Sri Lanka	LK

<b>Country and region name</b>	<b>Country and region ID</b>
Sudan	SD
Suriname	SR
Svalbard and Jan Mayen islands	SJ
Swaziland	SZ
Sweden	SE
Switzerland	CH
Syrian Arab Republic	SY
Taiwan	TW
Tajikistan	TJ
Tanzania, United Republic of	TZ
Thailand	TH
Togo	TG
Tokelau	TK
Tonga	TO
Trinidad and Tobago	TT
Tunisia	TN
Turkmenistan	TM
Turkey	TR
Turks and Caicos islands	TC
Tuvalu	TV
Uganda	UG
Ukraine	UA
United Arab Emirates	AE
United Kingdom	GB
United States Minor Outlying Islands	UM
United States of America	US
Uruguay	UY
Uzbekistan	UZ
Vanuatu	VU
Vatican City State	VA
Venezuela	VE
Vietnam	VN
Virgin islands (British)	VG
Virgin islands (U.S.)	VI
Wallis and Futuna islands	WF

Country and region name	Country and region ID
Western Sahara	EH
Yemen, Republic of	YE
Countries of the former Yugoslavia	YU
Zaire	ZR
Zambia	ZM
Zimbabwe	ZW

### Related concepts

Country or region identifier (QCNTYID) system value

The country or region identifier (QCNTYID) system value indicates the default country or region identifier for the system.

ILE RPG sort sequence

The ILE RPG feature, an option of the IBM Rational Development Studio for i licensed program, provides the possibility for a user to specify a sort sequence table and to use the table in comparison operations that are performed with nonnumeric data.

## Default system values for national language versions

Jobs and functions on the IBM i operating system use system values as default values.

The values for each national language version are listed by keyword. These are the values you see when you display the CPX8416 message for a particular language library. The tables contain Internet values that are needed to configure a Web browser for a specific language. The Internet values are also listed by keyword.

### Values that are provided in the tables

Keyword	Description
QCCSID	Character set identifier. This is the recommended QCCSID value if you want to use CDRA support. For all national language versions (NLVs), the default QCCSID value is 65535 unless otherwise indicated.
QCHRID	Character set and code page
QCNTYID	Country or region identifier
QCURSYM	Currency symbol. The values given are accurate; however, the system supports only 1 character in that return field.
QDATFMT	Date format
QDATSEP	Date separator
QDECfmt	Decimal format
QIGC	DBCS version indicator
QIGCCDEFNT	DBCS font name
QKBDTYPE	Keyboard type
QLANGID	Language identifier
QLEAPADJ	Leap year adjustment



Keyword	Description
QLOCALE	Locale. On newly-installed systems, the system sets the QLOCALE system value to a default locale based on the primary NLV that is installed. For example, the value will be set to /QSYS.LIB/DA_DK.LOCALE if you install the 2926 NLV.
QSRTSEQ	Sort sequence
QTIMSEP	Time separator
Internet CCSID	Client character set environment.
Client encoding nomenclature	NLTC (National Language Technical Center) value and Document Encoding. Client encoding nomenclature provides a guideline for configuring a client for a specific language and setting up your internet Web browser.

### Related concepts

#### System values

The system values of the primary language on the system are used as system-wide cultural and linguistic defaults. Therefore, if you change the primary language on the system, each varying system value resets to the default system value of the new primary language.

#### Configuring the primary language

A primary language consists of program code, textual data for each licensed program ordered, and default national language cultural values.

#### Packaging and installation process

You need to consider the running code, translated textual data, and installation documents when packaging applications. Here are some suggestions for simplifying the packaging and installation of your application.

#### Cultural-dependent design

Different countries might have different standards, which you must consider when developing an NLS-enabled application. This culturally sensitive information must be placed outside the program the same way as the textual data is handled.

#### System values for other languages with no national language version

Some of the system values are associated with languages and countries that do not have a national language version. You should set these values immediately after initially installing the IBM i operating system.

### Albanian (Feature 2995)

The table shows the default system values for the Albanian (Feature 2995) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	ALI	Lek	-	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	colon (:)	00500	AL	SQ_AL	SQI	ISO-8859-1 Latin 1

**Arabic (Feature 2954)**

The table shows the default system values for the Arabic (Feature 2954) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00235 00420	J	CLB	Dollar (\$)	Slash (/)	01089

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Comma (,)	00420	AA	AR_AA	ARA	ISO-8859-6

**Belgian Dutch MNCS (Feature 2963)**

The table shows the default system values for the Belgian Dutch multinational character set (MNCS) (Feature 2963) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	BLI	F	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00500	BE	NL_BE_M	NLB	ISO-8859-1 Latin 1

**Belgium English (Feature 2909)**

The table shows the default system values for the Belgium English (Feature 2909) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	BLI	F	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00500	BE	NE_BE	ENB	ISO-8859-1 Latin 1

**Belgian French MNCS (Feature 2966)**

The table shows the default system values for the Belgian French multinational character set (MNCS) (Feature 2966) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	BLI	F	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client Encoding nomenclature
DMY	Colon (:)	00500	BE	FR_BE_M	FRB	ISO-8859-1 Latin 1

#### Brazilian Portuguese (Feature 2980)

The table shows the default system values for the Brazilian Portuguese (Feature 2980) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00037	J	BRB	Cruzeiro (\$)	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00037	BR	PT_BR	PTB	ISO-8859-1 Latin 1

#### Bulgarian (Feature 2974)

The table shows the default system values for the Bulgarian (Feature 2974) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01150 01025	J	BGB	Lv	-	00915

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	colon (:)	01025	BG	BG_BG	BGR	ISO-8859-5

#### Canadian French MNCS (Feature 2981)

The table shows the default system values for the Canadian French multinational character set (MNCS) (Feature 2981) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	CAI	Dollar (\$)	Hyphen (-)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00500	CA	FR_CA_M	FRC	ISO-8859-1 Latin 1

**Croatian (Feature 2912)**

The table shows the default system values for the Croatian (Feature 2912) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	Blank	YGI	Croatian Kuna (K)	Hyphen (-)	00912

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	HR	HR_HR	HRV	ISO-8859-2 Latin 2

**Czech (Feature 2975)**

The table shows the default system values for the Czech (Feature 2975) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	J	CSB	Ceske Koruna (K)	Hyphen (-)	00912

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	CZ	CS_CZ	CSY	ISO-8859-2 Latin 2

**Danish (Feature 2926)**

The table shows the default system values for the Danish (Feature 2926) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00277	J	DMB	Colon (:)	Hyphen (-)	00819

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Period (.)	00277	DK	DA_DK	DAN	ISO-8859-1 Latin 1

**Dutch Netherlands (Feature 2923)**

The table shows the default system values for the Dutch Netherlands (Feature 2923) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00037	J	NEB	Dollar \$	Hyphen (-)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00037	NL	NL_NL	NLD	ISO-8859-1 Latin 1

### English Uppercase and Lowercase (Feature 2924)

The table shows the default system values for the English Uppercase and Lowercase (Feature 2924) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00037	Blank	USB	Dollar (\$)	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
MDY	Colon (:)	00037	US	EN_US	ENU	ISO-8859-1 Latin 1

### Related concepts

Information in message CPX8416

If your application is translated into other languages, use message CPX8416 from the QCPFMSG message file to get the correct setting for some cultural values for the other languages. The message exists for your primary language and all installed secondary language libraries.

### English Uppercase DBCS (Feature 2938)

The table shows the default system values for the English Uppercase DBCS (Feature 2938) national language version.

If English Uppercase is installed, you must respond to messages in uppercase rather than lowercase.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00037	Blank	JKB	Yen sign	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID <sup>1</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
MDY	Colon (:)	65535	US	EN_US	ENP	ISO-8859-1 Latin 1

<sup>1</sup> Because this national language version is not specific to any country or region, 65535 is used. You should change this system value to an appropriate CCSID after installing your system, or change the job or user profile CCSID attribute. Here are the recommended CCSID values and CHRID changes that occur.

- 05026 is the CCSID for Japanese with no lowercase characters. The CHRID is set to 1172 290.
- 01399 is the CCSID for Japanese with both uppercase and lowercase characters. The CHRID is set to 01172 01027.

System and product files not explicitly assigned a CCSID value will be assigned CCSID 05035 if DBCS capable and CCSID 01027 for SBCS only files.

### English Uppercase and Lowercase DBCS (Feature 2984)

The table shows the default system values for the English Uppercase and Lowercase DBCS (Feature 2984) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01175 00037	Blank	TAB	Dollar (\$)	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID <sup>1</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
MDY	Colon (:)	65535	US	EN_US	ENU	ISO-8859-1 Latin 1

<sup>1</sup> Because this national language version is not specific to any country or region, 65535 is used. You should change this system value to an appropriate CCSID after installing your system, or change the job or user profile CCSID attribute. Here are the recommended CCSID values and CHRID changes.

- 00937 is the CCSID for Traditional Chinese. The CHRID is set to 1175 00037
- 00935 is the CCSID for Simplified Chinese. The CHRID is set to 01174 00836.
- 00933 is the CCSID for Korean. The CHRID is set to 01173 00833.

### Estonian (Feature 2902)

The table shows the default system values for the Estonian (Feature 2902) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01307 01122	J	ESB	kr	-	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	01122	EE	ET_EE	EST	ISO-8859-1 Latin 1

### Farsi (Feature 2998)

The table shows the default system values for the Farsi (Feature 2998) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01219 01097	J	IRB	Farsi Riyal	Slash (/)	

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	01097	IR	AR_AA	FAR	

### Finnish (Feature 2925)

The table shows the default system values for the Finnish (Feature 2925) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00278	J	FNB	F	Period (.)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Period (.)	00278	FI	FI_FI	FIN	ISO-8859-1 Latin 1

#### French (Feature 2928)

The table shows the default system values for the French (Feature 2928) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00297	J	FAB	F	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00297	FR	FR_FR	FRA	ISO-8859-1 Latin 1

#### French MNCS (Feature 2940)

The table shows the default system values for the French multinational character set (MNCS) (Feature 2940) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	SFI	F	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00500	CH	FR_FR_M	FRS	ISO-8859-1 Latin 1

#### German (Feature 2929)

The table shows the default system values for the German (Feature 2929) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00273	J	AGB	Dollar (\$)	Period (.)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00273	DE	DE_DE	DEU	ISO-8859-1 Latin 1

**German MNCS (Feature 2939)**

The table shows the default system values for the German multinational character set (MNCS) (Feature 2939) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	AGI	Dollar (\$)	Period (.)	00819

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Comma (,)	00500	CH	DE_DE_M	DES	ISO-8859-1 Latin 1

**Greek (Feature 2957)**

The table shows the default system values for the Greek (Feature 2957) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00925 00875	J	GNB	Dollar (\$)	Slash (/)	00813

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00875	GR	EL_GR	ELL	ISO-8859-7 Greek

**Hebrew (Feature 2961)**

The table shows the default system values for the Hebrew (Feature 2961) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00941 00424	Blank	NCB	Shin (GCGID HS210000, code point X'69' on code page 00424)	Slash (/)	00916

QDATfmt	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00424	IL	HE_IL	HEB	ISO-8859-8

**Hungarian (Feature 2976)**

The table shows the default system values for the Hungarian (Feature 2976) national language version.

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	J	HNB	Forint (F)	Hyphen (-)	00912



QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	HU	HU_HU	HUN	ISO-8859-2 Latin 2

#### Icelandic (Feature 2958)

The table shows the default system values for the Icelandic (Feature 2958) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00871	J	ICB	Dollar (\$)	Hyphen (-)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00871	IS	IS_IS	ISL	ISO-8859-1 Latin 1

#### Italian (Feature 2932)

The table shows the default system values for the Italian (Feature 2932) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00280	Blank	ITB	Lira sign	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Period (.)	00280	IT	IT_IT	ITA	ISO-8859-1 Latin 1

#### Italian MNCS (Feature 2942)

The table shows the default system values for the Italian MNCS (Feature 2942) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	Blank	ITI	Lira sign	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Comma (,)	00500	CH	IT_IT_M	ITS	ISO-8859-1 Latin 1

#### Japanese Universal (Feature 2930)

The table shows the default system values for the Japanese Universal (Feature 2930) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
1172 1027	Blank	JPB	Yen sign	Hyphen (-)	00942

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	01399	JP	JA_JP_1399	JPN	ShiftJIS

### Japanese (Katakana) (Feature 2962)

The table shows the default system values for the Japanese (Katakana) (Feature 2962) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01172 00290	Blank	JKB	Yen sign	Hyphen (-)	00942

QDATFMT	QTIMSEP	QCCSID <sup>1</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	05026	JP	JA_JP_5026	JPN	ShiftJIS

<sup>1</sup> This is the recommended QCCSID value if you want to use CDRA support. For all NLVs, the default QCCSID value is 65535.

### Korean (Feature 2986)

The table shows the default system values for the Korean (Feature 2986) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01173 00833	Blank	KOB	WON sign	Period (.)	00949

QDATFMT	QTIMSEP	QCCSID <sup>1</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00933	KR	KO_KR	KOR	EUC Korean

<sup>1</sup> This is the recommended QCCSID value if you want to use CDRA support. For all NLVs, the default QCCSID value is 65535.

### Laotian (Feature 2906)

The table shows the default system values for the Laotian (Feature 2906) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
1341 1132	J	LAB	KIP	Slash (/)	

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	01132	LA	EN_US	LAO	

**Latvian (Feature 2904)**

The table shows the default system values for the Latvian (Feature 2904) national language version. The QCCSID value shows the original default for this language, however there is a newer CCSID 1156 that may be used to get the most current support.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01305 01112	J	LVB	Ls	Period (.)	00921

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	01112	LV	LV_LV	LVA	ISO-8859-4

**Lithuanian (Feature 2903)**

The table shows the default system values for the Lithuanian (Feature 2903) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01305 01112	J	LTB	Lt	Period (.)	00921

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	01112	LT	LT_LT	LTU	ISO-8859-4

**Macedonian (Feature 2913)**

The table shows the default system values for the Macedonian (Feature 2913) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01150 01025	J	MKB	Den	-	00915

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	01025	MK	MK_MK	MKD	Cyrillic (Win1251)

**Norwegian (Feature 2933)**

The table shows the default system values for the Norwegian (Feature 2933) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00277	J	NWB	Dollar (\$)	Period (.)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00277	NO	NO_NO	NON	ISO-8859-1 Latin 1

**Polish (Feature 2978)**

The table shows the default system values for the Polish (Feature 2978) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	J	PLB	Polish Zloty (Z)	Hyphen (-)	00912

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	PL	PL_PL	PLK	ISO-8859-2 Latin 2

**Portuguese (Feature 2922)**

The table shows the default system values for the Portuguese (Feature 2922) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00037	J	PRB	Escudo (\$)	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00037	PT	PT_PT	PTG	ISO-8859-1 Latin 1

**Portuguese MNCS (Feature 2996)**

The table shows the default system values for the Portuguese multinational character set (MNCS) (Feature 2996) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00500	J	PRI	Escudo (\$)	Hyphen (-)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00500	PT	PT_PT_M	PTG	ISO-8859-1 Latin 1

**Romanian (Feature 2992)**

The table shows the default system values for the Romanian (Feature 2992) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	J	RMB	lei	Period (.)	00912

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	RO	RO_RO	ROM	ISO-8859-2 Latin 2

**Russian (Feature 2979)**

The table shows the default system values for the Russian (Feature 2979) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01150 01025	J	RUB	Ruble (GCGID KR02000 - character X'DE' on code page 1025)	Slash (/)	1251

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Period (.)	01025	RU	RU_RU	RUS	Cyrillic (Win1251)

**Serbian (Feature 2914)**

The table shows the default system values for the Serbian (Feature 2914) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01150 01025	J	SQB	Din	-	00915

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Period (.)	01025	SQ	SR_SP	SRB	ISO-8859-5

**Simplified Chinese (Feature 2989)**

The table shows the default system values for the Simplified Chinese (Feature 2989) national language version. The QCCSID value shows the original default for this language, however there is a newer CCSID 1388 that may be used to get the most current support.

QCHRID <sup>1</sup>	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01174 00836	Blank	RCB	Dollar (\$)	Period (.)	01381

QDATFMT	QTIMSEP	QCCSID <sup>2</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00935	CN	ZH_CN	CHS	GB SimpChin

<sup>1</sup> For Version 1 Release 2 and Release 3, the character set and code page is 00101 00037.

<sup>2</sup> This is the recommended QCCSID value if you want to use CDRA support.

**Slovakian (Feature 2994)**

The table shows the default system values for the Slovakian (Feature 2994) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	J	SKB	Slovak Koruna (K)	Hyphen (-)	00912

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	SK	SK_SK	SKY	ISO-8859-2 Latin 2

### Slovenian (Feature 2911)

The table shows the default system values for the Slovenian (Feature 2911) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00959 00870	Blank	YGI	Slovenian Tolar (T)	Hyphen (-)	00912

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00870	SI	SL_SI	SLO	ISO-8859-2 Latin 2

### Spanish (Feature 2931)

The table shows the default system values for the Spanish (Feature 2931) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00284	J	SPB	Dollar (\$)	Slash (/)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	00284	ES	ES_ES	ESP	ISO-8859-1 Latin 1

### Swedish (Feature 2937)

The table shows the default system values for the Swedish (Feature 2937) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
00697 00278	J	SWB	Dollar (\$)	Hyphen (-)	00819

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Period (.)	00278	SE	SV_SE	SVE	ISO-8859-1 Latin 1

### Thai (Feature 2972)

The table shows the default system values for the Thai (Feature 2972) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01279 00838	Blank	THB	Baht sign	Slash (/)	01066

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	09030	TH	TH_TH	THA	IBM 874

#### Traditional Chinese (Feature 2987)

The table shows the default system values for the Traditional Chinese (Feature 2987) national language version. The QCCSID value shows the original default for this language, however there is a newer CCSID 1377 that may be used to get the most current support.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01175 00037	Blank	TAB	Dollar (\$)	Slash (/)	00950

QDATFMT	QTIMSEP	QCCSID <sup>1</sup>	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
YMD	Colon (:)	00937	TW	ZH_TW	CHT	BIG5

<sup>1</sup> This is the recommended QCCSID value if you want to use CDRA support. For all NLVs, the default QCCSID value is 65535.

#### Turkish (Feature 2956)

The table shows the default system values for the Turkish (Feature 2956) national language version. The QCCSID value shows the original default for this language, however there is a newer CCSID 1175 that should be used to get the most current support.

**Note:** Turkish support has had several CCSIDs in the past. Historically it was 905, then changed to 1026, then changed to 1155 when the euro was added and is now 1175 to support the Turkish Lira Currency Symbol.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
01152 01026	J	TKB	Dollar (\$)	Slash (/)	00920

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	01026	TR	TR_TR	TRK	ISO-8859-9 Turkish

#### Vietnamese (Feature 2905)

The table shows the default system values for the Vietnamese (Feature 2905) national language version.

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	Internet CCSID
1336 1130	J	VNB	DONG	Slash (/)	1258

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLOCALE	QLANGID	Client encoding nomenclature
DMY	Colon (:)	01130	VN	VI_VN	VIE	Windows 1258

### System values for other languages with no national language version

Some of the system values are associated with languages and countries that do not have a national language version. You should set these values immediately after initially installing the IBM i operating system.

You must also change the message, CPX8416, in the QCPFMSG message file in library QSYS to reflect the changed system values.

- [Afrikaans \(South Africa\)](#)
- [Australian English \(Australia\)](#)
- [Byelorussia \(Belarus\)](#)
- [Irish Gaelic \(Ireland\)](#)
- [Serbian Latin \(Serbia\)](#)
- [Spanish \(Argentina\)](#)
- [UK English \(United Kingdom\)](#)

#### Afrikaans (South Africa)

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP
00697 00037	J	USB	rand	hyphen (-)

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
YMD	colon (:)	00037	ZA	AFR

#### Australian English (Australia)

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP
00697 00037	J	USB	dollar	slash (/)

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
DMY	period (.)	00037	AU	ENA

#### Byelorussia (Belarus)

QCHRID	QDECfmt	QKBDTYPE	QCURSYM	QDATSEP
01150 01025		RUB		

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
		01025	BY	BEL



**Irish Gaelic (Ireland)**

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP
00697 00285	J	UKB	punt	slash (/)

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
DMY	colon (:)	00285	IE	GAE

**Serbian Latin (Serbia)**

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP
00859 00870	blank	YGI		

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
		00870	SQ	SRL

**Spanish (Argentina)**

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP
00697 00284	J	SSB	Austral (\$)	slash (/)

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
DMY	period (.)	00284	AR	ESP

**UK English (United Kingdom)**

QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP
00697 00285	J	UKB	pound	slash (/)

QDATFMT	QTIMSEP	QCCSID	QCNTYID	QLANGID
DMY	colon (:)	00285	GB	ENG

**Related concepts**

[Default system values for national language versions](#)

Jobs and functions on the IBM i operating system use system values as default values.

**Related tasks**

[Handling languages with no NLV support](#)

If you need to support a language that does not have a supported national language version, follow these steps.

## Keyboard reference information

This reference information includes keyboard layouts, SBCS keyboard and display part numbers, the special-character keyboard set, and the keyboard types and SBCS code pages for each national language.

### Keyboard layouts

These keyboard layout samples are provided for your information. The special-character keyboard set is available only with the enhanced keyboard.

### Related concepts

[Determining the number of supported keyboard types](#)

[Workstation considerations](#)

In a multilingual environment, different workstations support different languages on the same system.

### ***Albanian IBM Enhanced Keyboard***

The figure shows the Albanian IBM Enhanced Keyboard.



### ***Arabic IBM Enhanced Keyboard***

The figure shows the Arabic IBM Enhanced Keyboard.



### ***Austrian German IBM Enhanced Keyboard***

The figure shows the Austrian German IBM Enhanced Keyboard.



**Belgian Multinational IBM Enhanced Keyboard**

The figure shows the Belgian Multinational IBM Enhanced Keyboard.



**Brazilian Portuguese IBM Enhanced Keyboard**

The figure shows the Brazilian Portuguese IBM Enhanced Keyboard.



This U.S. keyboard is used to enter Brazilian Portuguese characters. Accented characters for use in Brazil are generated using key combinations as shown in the following table. To produce the accented characters, press the key and character at the same time.

Accent	Key	Valid characters	Example result
Accent acute #	' (apostrophe)	a, e, i, o, u, A, E, I, O, U	ã
Accent grave `	` (accent grave)	a, e, i, o, u, A, E, I, O, U	ä
Diaeresis ""	" (quotation mark)	a, e, i, o, u, A, E, I, O, U	ë
Tilde ~	~ (tilde)	a, o, n, A, O, N	ñ
Circumflex ^	^ (caret)	a, e, i, o, u, A, E, I, O, U	â

### **Bulgarian Cyrillic IBM Enhanced Keyboard**

The figure shows the Bulgarian Cyrillic IBM Enhanced Keyboard.



### **Canadian French IBM Enhanced Keyboard**

The figure shows the Canadian French IBM Enhanced Keyboard.



### **Croatian IBM Enhanced Keyboard**

The figure shows the Croatian IBM Enhanced Keyboard.



### **Czech IBM Enhanced Keyboard**

The figure shows the Czech IBM Enhanced Keyboard.



**Danish IBM Enhanced Keyboard**

The figure shows the Danish IBM Enhanced Keyboard.



**Dutch IBM Enhanced Keyboard**

The figure shows the Dutch IBM Enhanced Keyboard.



**Farsi IBM Enhanced Keyboard**

The figure shows the Farsi IBM Enhanced Keyboard.



**Finnish/Swedish IBM Enhanced Keyboard**

The figure shows the Finnish/Swedish IBM Enhanced Keyboard.



**French (AZERTY) IBM Enhanced Keyboard**

The figure shows the French IBM Enhanced Keyboard.



**Greek IBM Enhanced Keyboard**

The figure shows the Greek IBM Enhanced Keyboard.



**Hebrew IBM Enhanced Keyboard**

The figure shows the Hebrew IBM Enhanced Keyboard.



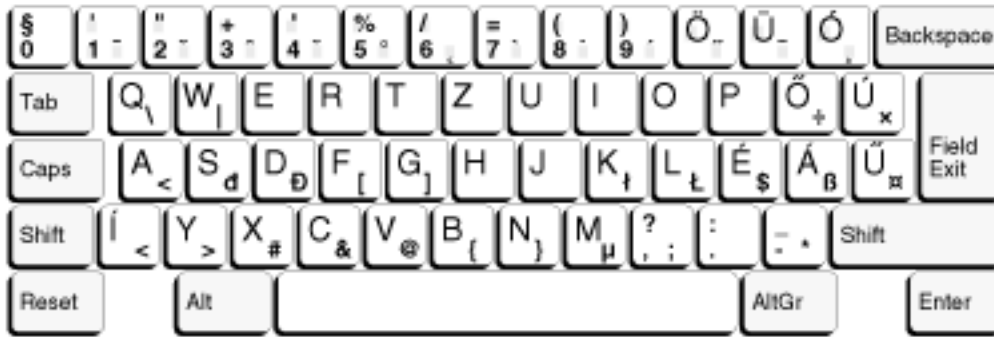
**Hebrew, Latin IBM Enhanced Keyboard**

The figure shows the Hebrew, Latin IBM Enhanced Keyboard.



### **Hungarian IBM Enhanced Keyboard**

The figure shows the Hungarian IBM Enhanced Keyboard.



### **Italian IBM Enhanced Keyboard**

The figure shows the Italian IBM Enhanced Keyboard.



### **Japanese IBM Enhanced Keyboard**

The figure shows the Japanese IBM Enhanced Keyboard.



### **Korean IBM Enhanced Keyboard**

The figure shows the Korean IBM Enhanced Keyboard.



### ***Macedonian IBM Enhanced Keyboard***

The figure shows the Macedonian IBM Enhanced Keyboard.



### ***Norwegian IBM Enhanced Keyboard***

The figure shows the Norwegian IBM Enhanced Keyboard.



### ***Polish IBM Enhanced Keyboard***

The figure shows the Polish IBM Enhanced Keyboard.





### Portuguese IBM Enhanced Keyboard

The figure shows the Portuguese IBM Enhanced Keyboard.



### Romanian IBM Enhanced Keyboard

The figure shows the Romanian IBM Enhanced Keyboard.



### Russian IBM Enhanced Keyboard

The figure shows the Russian IBM Enhanced Keyboard.



### Serbian Cyrillic IBM Enhanced Keyboard

The figure shows the Serbian Cyrillic IBM Enhanced Keyboard.



### **Slovakian IBM Enhanced Keyboard**

The figure shows the Slovakian IBM Enhanced Keyboard.



### **Slovenian IBM Enhanced Keyboard**

The figure shows the Slovenian IBM Enhanced Keyboard.



### **Spanish-Speaking IBM Enhanced Keyboard**

The figure shows the Spanish-Speaking IBM Enhanced Keyboard.



### **Spanish IBM Enhanced Keyboard**

The figure shows the Spanish IBM Enhanced Keyboard.



**Swiss-Bilingual-French IBM Enhanced Keyboard**

The figure shows the Swiss-Bilingual-French IBM Enhanced Keyboard.



**Swiss-Bilingual-German IBM Enhanced Keyboard**

The figure shows the Swiss-Bilingual-German IBM Enhanced Keyboard.



**Traditional Chinese IBM Enhanced Keyboard**

The figure shows the Traditional Chinese IBM Enhanced Keyboard.



RVOC279-0

### **Turkish IBM Enhanced Keyboard**

The figure shows the Turkish IBM Enhanced Keyboard.



### **U.K. English IBM Enhanced Keyboard**

The figure shows the U.K. English IBM Enhanced Keyboard.



### **U.S. English IBM Enhanced Keyboard**

The figure shows the U.S. English IBM Enhanced Keyboard.



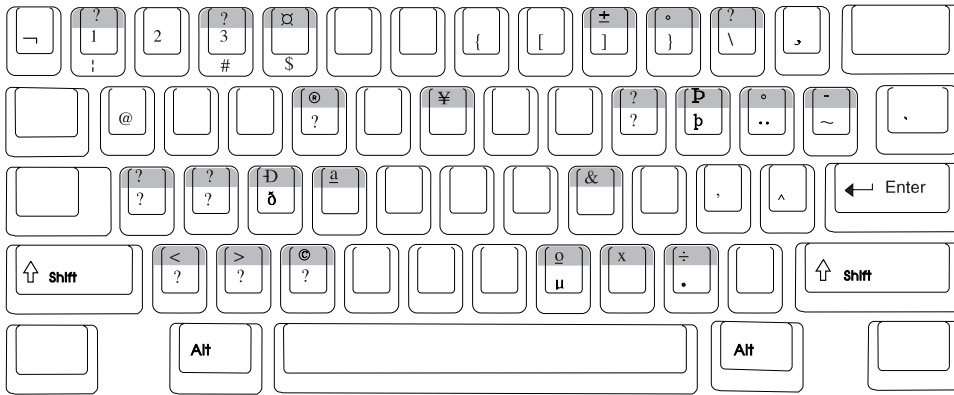
### **Special-character keyboard set**

The special-character keyboard set is available with the enhanced keyboard on most display stations. It allows a user to enter special characters that otherwise might not be available (labeled) on the keyboard.

The following figure shows all the characters in the special keyboard set and the keys on the enhanced keyboard to which each character is assigned. A special character can be assigned to a lowercase, uppercase, or ALT position on a key.

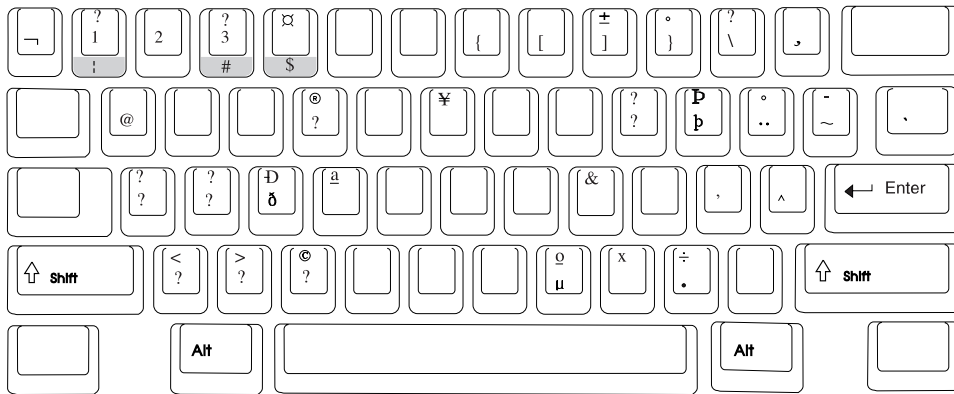
You can order a special template package, SCX21-9950, that contains the special-character keyboard set.






RV2C053-1

The following figure highlights the special characters assigned to the ALT key positions.

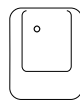


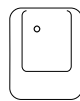
RV2C054-1

### Example: Special character


For example, if you wanted to create the  for German on an English keyboard, you should follow these steps:



1. Press and hold the ALT key, and then press the Shift key.



2. Press the Shift key with the  RV2C055-0 key. The diacritic mode symbol appears at the bottom of the screen and is waiting for the next keystroke to complete the character.



3. You now press the Shift key and the  RV2C056-0 key.

This produces the . If you press this key without also pressing the shift key, you will get . The system accepts only uppercase or lowercase A.

### National language keyboard types and SBCS code pages

This table lists the keyboard types and code pages for each national language supported by the IBM i operating system. The Create Device Display (CRTDEV DSP) command uses the KBDTYPE parameter.

Language	KBDTYPE	EBCDIC character set	EBCDIC SBCS code page	EBCDIC CCSID
Albanian	ALI	00697	00500	00500
Arabic	CLB	00235	00420	00420
Austrian/ German	AGB	00697	00273	00273
Austrian/German (MNCS)	AGI	00697	00500	00500
Belgian MNCS	BLI	00697	00500	00500
Brazilian Portuguese	BRB	00697	00037	00037
Bulgarian	BGB	01150	01025	01025
Canadian French	CAB	00341	00260	65535
Canadian French MNCS	CAI	00697	00500	00500
Croatian	YGI	00959	00870	00870
Cyrillic	CYB	00960	00880	00880
Czech	CSB	00959	00870	00870
Danish	DMB	00697	00277	00277
Danish MNCS	DMI	00697	00500	00500
Finnish/Swedish	FNB	00697	00278	00278
Finnish/Swedish MNCS	FNI	00697	00500	00500
French (Azerty)	FAB	00697	00297	00297
French (Azerty) MNCS	FAI	00697	00500	00500
French (Qwerty)	FQB	00697	00297	00297
French (Qwerty) MNCS	FQI	00697	00500	00500
Greek (See note 2.)	GNB	00925	00875	00875
Hebrew	NCB	00941	00424	00424
Hungarian	HNB	00959	00870	00870
Icelandic	ICB	00697	00871	00871
Icelandic MNCS	ICI	00697	00500	00500
International	INB	00697	00500	00500
International MNCS	INB	00697	00500	00500
Farsi (Iran)	IRB	01219	01097	01097

<b>Language</b>	<b>KBTYPE</b>	<b>EBCDIC character set</b>	<b>EBCDIC SBCS code page</b>	<b>EBCDIC CCSID</b>
Italian	ITB	00697	00280	00280
Italian MNCS	ITI	00697	00500	00500
Japanese-English	JEB	00697	00281	65535
Japanese- English MNCS	JEI	00697	00500	00500
Japanese Kanji and Katakana	JKB	01172	00290	05026
Japanese Kanji and US English	JUB	00697	00037	See note 3.
Japanese Katakana	KAB	00332	00290	00290
Japanese Latin Extended	JPB	01172	01027	01027
Korean	KOB	01173	00833	00833
Latin 2	ROB	00959	00870	00870
Macedonian	MKB	01150	01025	01025
Dutch (Netherlands)	NEB	00697	00037	00037
Dutch (Netherlands) MNCS	NEI	00697	00500	00500
Norwegian	NWB	00697	00277	00277
Norwegian MNCS	NWI	00697	00500	00500
Polish	PLB	00959	00870	00870
Portuguese	PRB	00697	00037	00037
Portuguese MNCS	PRI	00697	00500	00500
Romanian	RMB	00959	00870	00870
Russian	RUB	01150	01025	01025
Serbian, Cyrillic	SQB	01150	01025	01025
Serbian, Latin	YGI	00959	00870	00870
Simplified Chinese	RCB	01174	00836	00836
Slovakian	SKB	00959	00870	00870
Slovenian	YGI	00959	00870	00870
Spanish	SPB	00697	00284	00284
Spanish MNCS	SPI	00697	00500	00500
Spanish Speaking	SSB	00697	00284	00284
Spanish Speaking MNCS	SSI	00697	00500	00500



Language	KBDTYPE	EBCDIC character set	EBCDIC SBCS code page	EBCDIC CCSID
Swedish	SWB	00697	00278	00278
Swedish MNCS	SWI	00697	00500	00500
French (Switzerland) MNCS	SFI	00697	00500	00500
German (Switzerland) MNCS	SGI	00697	00500	00500
Thai	THB	01176	00838	00838
Traditional Chinese	TAB	01175	00037	00937
Turkish (Qwerty)	TKB	01152	01026	01026
Turkish (F)	TRB	01152	01026	01026
English (United Kingdom)	UKB	00697	00285	00285
English (United Kingdom) MNCS	UKI	00697	00500	00500
English (United States and Canada)	USB	00697	00037	00037
English (United States and Canada) MNCS	USI	00697	00500	00500

**Notes:**

1. For KBDTYPE GKB, the EBCDIC code page is 00423.
2. Recommend SBCS CCSID 00037.

**Related concepts**

Language indicator for keyboard type (QKBDTYPE) system value

The language indicator for the keyboard type (QKBDTYPE) system value specifies the language character set for the keyboard.

Recommendations and guidelines for using CCSIDs

These recommendations are useful when you write globalized applications.

**Related reference**

Create Device Display (CRTDEV DSP) command

**Code pages**

Several IBM code pages match the International Standard ISO/IEC 8859. The IBM i operating system supports parts of the ISO Standard with equivalent IBM code pages.

ISO/IEC 8859 consists of the following parts, under the general title *Information processing - 8-bit single-byte coded graphic character sets*:



- Part 1: Latin alphabet No. 1, 8859-1
- Part 2: Latin alphabet No. 2, 8859-2
- Part 3: Latin alphabet No. 3, 8859-3
- Part 4: Latin alphabet No. 4, 8835-4

- Part 5: Latin/Cyrillic alphabet, 8859-5
- Part 6: Latin/Arabic alphabet, 8858-6
- Part 7: Latin/Greek alphabet, 8859-7
- Part 8: Latin/Hebrew alphabet, 8859-8
- Part 9: Latin alphabet No. 5, 8859-9

IBM i supports Parts 1, 2, and 9 of the ISO Standard with equivalent IBM code pages. Equivalent IBM code pages have characters that map to the ISO Standard. The equivalent IBM code pages contain all of the characters that are contained in the ISO standard. Equivalent IBM code pages are not identical to the ISO Standard. Some of the characters in the equivalent IBM code pages are at different code points than the same characters in the ISO standard.

IBM i supports Parts 1, 2, 5, 7, 8, and 9 of the ISO Standard with identical IBM code pages. Identical IBM code pages are the same as the ISO standard.

You can find a list of code pages on the following websites:

- [IBM Code Page Identifiers](#) 
- [IBM i globalization](#) 

#### ISO standards and IBM EBCDIC code pages

Equivalent EBCDIC code page	ISO standard
00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00819, 00871	8859-1
00870	8859-2
00905	8859-9
01026	8859-9

#### ISO standards and IBM ASCII code pages

Equivalent ASCII code page	ISO standard
00852	8859-2
00857	8859-5

#### ISO standards and identical IBM code page

Identical ASCII code page	ISO standard
00813	8859-7
00819	8859-1
00916	8859-8
00920	8859-9

#### Notes:

1. Hexadecimal value 40 represents the space character on the EBCDIC code pages.

- Hexadecimal value 20 represents the space character on the PC code pages.
- Hexadecimal value FF represents the eight ones (11111111) control character.
- The 8-digit alphanumeric label under each character in the code page chart is the graphic character global identifier (GCGID). The label is used with Sort sequence tables.

### Related concepts

[IBM i globalization](#)

[Sort sequence tables](#)

A sort sequence table is an object that contains the weight of each single-byte graphic character within a specified coded character set identifier (CCSID). The system-recognized identifier for the sort sequence table object type is \*TBL.

## Character sets

This reference information is about various character sets on the IBM i operating system.

### Country extended character set 00697

The figure shows country extended character set 00697.

A	B	C	D	E	F	G	H	I	J	K	L	M
LA020000	LB020000	LC020000	LD020000	LE020000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA010000	LB010000	LC010000	LD010000	LE010000	LF010000	LG010000	LH010000	LI010000	LJ010000	LK010000	LL010000	LM010000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN010000	LO010000	LP010000	LQ010000	LR010000	LS010000	LT010000	LU010000	LV010000	LW010000	LX010000	LY010000	LZ010000
Á	Â	Ã	Ä	Å	Æ	Ç	Ð	É	Ê	Ë	Ë	Ë
LA120000	LA140000	LA160000	LA180000	LA200000	LA280000	LA320000	LA420000	LD820000	LE120000	LE140000	LE160000	LE180000
í	î	ï	Ï	Ñ	Ó	Ô	Õ	Ö	Ø	Ø	Þ	Ú
LI020000	LI140000	LI160000	LI180000	LI200000	LO120000	LO140000	LO160000	LO180000	LC020000	LO820000	LT640000	LU120000
Û	Ü	Û	Ý									
LU140000	LU160000	LU180000	LY120000									
á	â	ã	ä	å	æ	ç	ð	é	ê	ë	ë	ë
LA110000	LA130000	LA150000	LA170000	LA190000	LA270000	LA310000	LC410000	LD830000	LE110000	LE130000	LE150000	LE170000
ì	í	î	ï	ñ	ó	ô	õ	ö	ø	ø	þ	þ
LI110000	LI130000	LI150000	LI170000	LI190000	LD110000	LD130000	LD150000	LD170000	LD190000	LD190000	LS010000	LT030000
ú	û	ü	ü	ý	ÿ							
LU110000	LU130000	LU150000	LU170000	LY110000	LY170000							
0	1	2	3	4	5	6	7	8	9			
ND010000	ND010000	ND020000	ND030000	ND040000	ND050000	ND060000	ND070000	ND080000	ND090000			
¼	½	¾	¾	¾	+	+	<	=	>	÷	×	
ND011000	ND021000	ND031000	NF040000	NF040000	NF050000	SA010000	SA020000	SA030000	SA040000	SA050000	SA060000	SA070000
£	£	£	£	¥	¥	¥	¥	¥	¥	¥	¥	#
SC010000	SC020000	SC030000	SC040000	SC050000	SD110000	SD130000	SD150000	SD170000	SD190000	SD410000	SD830000	SM030000
%	&	*	@	[	\	]	{		}	~	~	μ
SM020000	SM030000	SM040000	SM050000	SM060000	SM070000	SM080000	SM110000	SM130000	SM140000	SM150000	SM170000	SM190000
°	°	§	¶	©	®	®	®	®	®	®	®	(
SM200000	SM210000	SM240000	SM250000	SM320000	SM030000	SM050000	SM060000	SM080000	SM090000	SM100000	SM140000	SM190000
)	,	—	-	.	/	:	:	?	¿	«	»	(KSP)
SP070000	SP080000	SP090000	SP100000	SP110000	SP120000	SP130000	SP140000	SP150000	SP160000	SP170000	SP180000	SP200000
␣	(MITY)											
SP200000												

Character Set 00697

### Graphic character conversion tables

Table (\*TBL) objects support non-CCSID conversions from one code page to another. The system-supplied table objects are located in the QUSRSYS library.

To see the supported code pages, run the following command:

```
WRKOBJ OBJ(QUSRSYS/*ALL) OBJTYPE(*TBL)
```

The table description shows the *from* code page and the *to* code page and character set. For example, if the from code page is 1112 and the to code page is 500, the description shows the following text:

```
CHRID(*N 1112) to CHRID(697 500)
```

This conversion method is supported on the IBM i operating system, but is not enhanced.

To convert data, a preferred method is to use the systems CCSID support. For details, see [Character Conversion APIs](#).

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
QA3R	(*N 1025)	Cyrillic, Multilingual	Uppercase table	
QA3S	(*N 1026)	Turkey Latin 5	Uppercase table	
QA3T	(*N 1027)	Japan extended	Uppercase table	
QA5R	(*N 1097)	Farsi	Uppercase table	
QA57	(*N 1112)	Baltic, Multilingual	Uppercase table	
QA6G	(*N 1122)	Estonian	Uppercase table	
QA6O	(*N 1130)	Vietnamese	Uppercase table	
QA6Q	(*N 1133)	Lao	Uppercase table	
Q037	(*N 037)	USA/Canada	Uppercase table	
Q256	(*N 256)	Multinational #1	Uppercase table	
Q260	(*N 260)	Canada French	Uppercase table	
Q273	(*N 273)	Austria/Germany	Uppercase table	
Q277	(*N 277)	Denmark/Norway	Uppercase table	
Q278	(*N 278)	Finland/Sweden	Uppercase table	
Q280	(*N 280)	Italy	Uppercase table	
Q281	(*N 281)	Japan (Latin)	Uppercase table	
Q284	(*N 284)	Spain/Latin America	Uppercase table	
Q285	(*N 285)	United Kingdom	Uppercase table	
Q290	(*N 290)	Japan Katakana	Uppercase table	
Q297	(*N 297)	France	Uppercase table	
Q420	(*N 420)	Arabic Bilingual	Uppercase table	
Q423	(*N 423)	Greece	Uppercase table	
Q424	(*N 424)	Israel (Hebrew)	Uppercase table	
Q437	(*N 437)	PC-USA	Uppercase table	
Q500	(*N 500)	Multinational #5	Uppercase table	
Q833	(*N 833)	Korea	Uppercase table	
Q836	(*N 836)	People's Republic of China	Uppercase table	
Q838	(*N 838)	Thai Extended	Uppercase table	
Q850	(*N 850)	PC-Multilingual	Uppercase table	
Q851	(*N 851)	PC-Greece (old)	Uppercase table	
Q857	(*N 857)	PC-Turkey	Uppercase table	

Table name	From	From text	To	To text
Q860	(*N 860)	PC-Portugal	Uppercase table	
Q861	(*N 861)	PC-Iceland	Uppercase table	
Q862	(*N 862)	PC-Israel	Uppercase table	
Q863	(*N 863)	PC-Canadian French	Uppercase table	
Q864	(*N 864)	PC-Arabic	Uppercase table	
Q865	(*N 865)	PC-Nordic	Uppercase table	
Q870	(*N 870)	Latin 2, Multilingual	Uppercase table	
Q871	(*N 871)	Iceland	Uppercase table	
Q875	(*N 875)	Greece	Uppercase table	
Q880	(*N 880)	Cyrillic, Multilingual (old)	Uppercase table	
Q891	(*N 891)	PC-Korea	Uppercase table	
Q897	(*N 897)	PC-Japan	Uppercase table	
Q903	(*N 903)	PC-People's Republic of China	Uppercase table	
Q904	(*N 904)	PC-People's Republic of China	Uppercase table	
Q905	(*N 905)	PC-Turkey	Uppercase table	
Q037A0MA5K	(*N 037)	USA/Canada	( 1272 1090)	VT100 Line drawing set
Q037A05A5U	(*N 037)	USA/Canada	( 1290 1100)	VT220 Multinational
Q037A6G897	(*N 037)	USA/Canada	( 1122 897)	PC-Japan
Q037A69A3R	(*N 037)	USA/Canada	( 1150 1025)	Cyrillic, Multilingual
Q037A7AA3S	(*N 037)	USA/Canada	( 1152 1026)	Turkey Latin 5
Q037A7RA3B	(*N 037)	USA/Canada	( 1169 1009)	International Alphabet 5
Q037A7UA3T	(*N 037)	USA/Canada	( 1172 1027)	Japan extended
Q037A7U290	(*N 037)	USA/Canada	( 1172 290)	Japan Katakana
Q037A7V833	(*N 037)	USA/Canada	( 1173 833)	Korea
Q037A7W836	(*N 037)	USA/Canada	( 1174 836)	People's Republic of China
Q037A7Y838	(*N 037)	USA/Canada	( 1176 838)	Thai Extended
Q037A7Y874	(*N 037)	USA/Canada	( 1176 874)	PC-Thai Extended
Q037A86A5R	(*N 037)	USA/Canada	( 1219 1097)	Farsi
Q037BAJA57	(*N 037)	USA/Canada	( 1305 1112)	Baltic, Multilingual

Table name	From	From text	To	To text
Q037BALA6G	(*N 037)	USA/Canada	( 1307 1122)	Estonian
Q037BBEA6O	(*N 037)	USA/Canada	( 1336 1130)	Vietnamese
Q037BBJA6Q	(*N 037)	USA/Canada	( 1341 1133)	Lao
Q037101037	(*N 037)	USA/Canada	( 101 037)	USA/Canada
Q037103A59	(*N 037)	USA/Canada	( 103 1114)	Taiwan Industry (Big 5) code
Q037235420	(*N 037)	USA/Canada	( 235 420)	Arabic Bilingual
Q037337256	(*N 037)	USA/Canada	( 337 256)	Multinational #1
Q037337437	(*N 037)	USA/Canada	( 337 437)	PC-USA
Q037337850	(*N 037)	USA/Canada	( 337 850)	PC-Multilingual
Q037337860	(*N 037)	USA/Canada	( 337 860)	PC-Portugal
Q037337863	(*N 037)	USA/Canada	( 337 863)	PC-Canadian French
Q037337904	(*N 037)	USA/Canada	( 337 904)	PC-People's Republic of China
Q037936836	(*N 037)	USA/Canada	( 936 836)	People's Republic of China
Q037941424	(*N 037)	USA/Canada	( 941 424)	Israel (Hebrew)
Q037959870	(*N 037)	USA/Canada	( 959 870)	Latin 2, Multilingual
Q037960880	(*N 037)	USA/Canada	( 960 880)	Cyrillic, Multilingual (old)
Q037965905	(*N 037)	USA/Canada	( 965 905)	PC-Turkey
Q038337256	(*N 038)	USA/ASCII	( 337 256)	Multinational #1
Q256A69A3R	(*N 256)	Multinational #1	( 1150 1025)	Cyrillic, Multilingual
Q256A7AA3S	(*N 256)	Multinational #1	( 1152 1026)	Turkey Latin 5
Q256A7V833	(*N 256)	Multinational #1	( 1173 833)	Korea
Q256A7W836	(*N 256)	Multinational #1	( 1174 836)	People's Republic of China
Q256A7X037	(*N 256)	Multinational #1	( 1175 037)	USA/Canada
Q256001256	(*N 256)	Multinational #1	( 001 256)	Multinational #1
Q256101037	(*N 256)	Multinational #1	( 101 037)	USA/Canada
Q256101367	(*N 256)	Multinational #1	( 101 367)	ASCII
Q256103038	(*N 256)	Multinational #1	( 103 038)	USA/ASCII
Q256218423	(*N 256)	Multinational #1	( 218 423)	Greece
Q256265273	(*N 256)	Multinational #1	( 265 273)	Austria/Germany
Q256269274	(*N 256)	Multinational #1	( 269 274)	Belgium
Q256273275	(*N 256)	Multinational #1	( 273 275)	Brazil

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q256277276	(*N 256)	Multinational #1	( 277 276)	Canada (French)
Q256281277	(*N 256)	Multinational #1	( 281 277)	Denmark/Norway
Q256285278	(*N 256)	Multinational #1	( 285 278)	Finland/Sweden
Q256288297	(*N 256)	Multinational #1	( 288 297)	France
Q256289279	(*N 256)	Multinational #1	( 289 279)	France
Q256293280	(*N 256)	Multinational #1	( 293 280)	Italy
Q256297281	(*N 256)	Multinational #1	( 297 281)	Japan (Latin)
Q256301282	(*N 256)	Multinational #1	( 301 282)	Portugal
Q256305283	(*N 256)	Multinational #1	( 305 283)	Spain
Q256309284	(*N 256)	Multinational #1	( 309 284)	Spain/Latin America
Q256313285	(*N 256)	Multinational #1	( 313 285)	United Kingdom
Q256332290	(*N 256)	Multinational #1	( 332 290)	Japan Katakana
Q256337037	(*N 256)	Multinational #1	( 337 037)	USA/Canada
Q256337273	(*N 256)	Multinational #1	( 337 273)	Austria/Germany
Q256337274	(*N 256)	Multinational #1	( 337 274)	Belgium
Q256337275	(*N 256)	Multinational #1	( 337 275)	Brazil
Q256337276	(*N 256)	Multinational #1	( 337 276)	Canada (French)
Q256337277	(*N 256)	Multinational #1	( 337 277)	Denmark/Norway
Q256337278	(*N 256)	Multinational #1	( 337 278)	Finland/Sweden
Q256337280	(*N 256)	Multinational #1	( 337 280)	Italy
Q256337281	(*N 256)	Multinational #1	( 337 281)	Japan (Latin)
Q256337282	(*N 256)	Multinational #1	( 337 282)	Portugal
Q256337283	(*N 256)	Multinational #1	( 337 283)	Spain
Q256337284	(*N 256)	Multinational #1	( 337 284)	Spain/Latin America
Q256337285	(*N 256)	Multinational #1	( 337 285)	United Kingdom
Q256337297	(*N 256)	Multinational #1	( 337 297)	France
Q256337420	(*N 256)	Multinational #1	( 337 420)	Arabic Bilingual
Q256337833	(*N 256)	Multinational #1	( 337 833)	Korea
Q256338257	(*N 256)	Multinational #1	( 338 257)	Multinational #2
Q256339258	(*N 256)	Multinational #1	( 339 258)	Multinational #3
Q256340259	(*N 256)	Multinational #1	( 340 259)	Symbols, Set #7
Q256341260	(*N 256)	Multinational #1	( 341 260)	Canada French
Q256697871	(*N 256)	Multinational #1	( 697 871)	Iceland
Q256925875	(*N 256)	Multinational #1	( 925 875)	Greece

Table name	From	From text	To	To text
Q256933833	(*N 256)	Multinational #1	( 933 833)	Korea
Q256936836	(*N 256)	Multinational #1	( 936 836)	People's Republic of China
Q256938838	(*N 256)	Multinational #1	( 938 838)	Thai Extended
Q256941424	(*N 256)	Multinational #1	( 941 424)	Israel (Hebrew)
Q256959870	(*N 256)	Multinational #1	( 959 870)	Latin 2, Multilingual
Q256960880	(*N 256)	Multinational #1	( 960 880)	Cyrillic, Multilingual (old)
Q257337256	(*N 257)	Multinational #2	( 337 256)	Multinational #1
Q258337256	(*N 258)	Multinational #3	( 337 256)	Multinational #1
Q259337256	(*N 259)	Symbols, Set #7	( 337 256)	Multinational #1
Q260337256	(*N 260)	Canada French	( 337 256)	Multinational #1
Q273A7RA3B	(*N 273)	Austria/Germany	( 1169 1009)	International Alphabet 5
Q273337256	(*N 273)	Austria/Germany	( 337 256)	Multinational #1
Q273337437	(*N 273)	Austria/Germany	( 337 437)	PC-USA
Q273337850	(*N 273)	Austria/Germany	( 337 850)	PC-Multilingual
Q274337256	(*N 274)	Belgium	( 337 256)	Multinational #1
Q275337256	(*N 275)	Brazil	( 337 256)	Multinational #1
Q276337256	(*N 276)	Canada (French)	( 337 256)	Multinational #1
Q277A7RA3B	(*N 277)	Denmark/Norway	( 1169 1009)	International Alphabet 5
Q277337256	(*N 277)	Denmark/Norway	( 337 256)	Multinational #1
Q277337850	(*N 277)	Denmark/Norway	( 337 850)	PC-Multilingual
Q277337865	(*N 277)	Denmark/Norway	( 337 865)	PC-Nordic
Q277697284	(*N 277)	Denmark/Norway	( 697 284)	Spain/Latin America
Q278A7RA3B	(*N 278)	Finland/Sweden	( 1169 1009)	International Alphabet 5
Q278337256	(*N 278)	Finland/Sweden	( 337 256)	Multinational #1
Q278337437	(*N 278)	Finland/Sweden	( 337 437)	PC-USA
Q278337850	(*N 278)	Finland/Sweden	( 337 850)	PC-Multilingual
Q279337256	(*N 279)	France	( 337 256)	Multinational #1
Q280A7RA3B	(*N 280)	Italy	( 1169 1009)	International Alphabet 5
Q280337256	(*N 280)	Italy	( 337 256)	Multinational #1
Q280337437	(*N 280)	Italy	( 337 437)	PC-USA



<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q280337850	(*N 280)	Italy	( 337 850)	PC-Multilingual
Q281337256	(*N 281)	Japan (Latin)	( 337 256)	Multinational #1
Q282337256	(*N 282)	Portugal	( 337 256)	Multinational #1
Q282337850	(*N 282)	Portugal	( 337 850)	PC-Multilingual
Q282337860	(*N 282)	Portugal	( 337 860)	PC-Portugal
Q283337256	(*N 283)	Spain	( 337 256)	Multinational #1
Q284A7RA3B	(*N 284)	Spain/Latin America	( 1169 1009)	International Alphabet 5
Q284A7W836	(*N 284)	Spain/Latin America	( 1174 836)	People's Republic of China
Q284337256	(*N 284)	Spain/Latin America	( 337 256)	Multinational #1
Q284337437	(*N 284)	Spain/Latin America	( 337 437)	PC-USA
Q284337850	(*N 284)	Spain/Latin America	( 337 850)	PC-Multilingual
Q284697277	(*N 284)	Spain/Latin America	( 697 277)	Denmark/Norway
Q285337256	(*N 285)	United Kingdom	( 337 256)	Multinational #1
Q285337437	(*N 285)	United Kingdom	( 337 437)	PC-USA
Q285337850	(*N 285)	United Kingdom	( 337 850)	PC-Multilingual
Q290A7RA3B	(*N 290)	Japan Katakana	( 1169 1009)	International Alphabet 5
Q290A7UA3T	(*N 290)	Japan Katakana	( 1172 1027)	Japan extended
Q290A7UA38	(*N 290)	Japan Katakana	( 1172 1041)	PC-Japan extended
Q290337256	(*N 290)	Japan Katakana	( 337 256)	Multinational #1
Q290337897	(*N 290)	Japan Katakana	( 337 897)	PC-Japan
Q290697037	(*N 290)	Japan Katakana	( 697 037)	USA/Canada
Q290697500	(*N 290)	Japan Katakana	( 697 500)	Multinational #5
Q297A7RA3B	(*N 297)	France	( 1169 1009)	International Alphabet 5
Q297337256	(*N 297)	France	( 337 256)	Multinational #1
Q297337437	(*N 297)	France	( 337 437)	PC-USA
Q297337850	(*N 297)	France	( 337 850)	PC-Multilingual
Q367A7RA3B	(*N 367)	ASCII	( 1169 1009)	International Alphabet 5
Q367337256	(*N 367)	ASCII	( 337 256)	Multinational #1
Q367697500	(*N 367)	ASCII	( 697 500)	Multinational #5

Table name	From	From text	To	To text
Q420235864	(*N 420)	Arabic Bilingual	( 235 864)	PC-Arabic
Q420337256	(*N 420)	Arabic Bilingual	( 337 256)	Multinational #1
Q420697037	(*N 420)	Arabic Bilingual	( 697 037)	USA/Canada
Q420697500	(*N 420)	Arabic Bilingual	( 697 500)	Multinational #5
Q423A7RA3B	(*N 423)	Greece	( 1169 1009)	International Alphabet 5
Q423218851	(*N 423)	Greece	( 218 851)	PC-Greece (old)
Q423697256	(*N 423)	Greece	( 697 256)	Multinational #1
Q423925875	(*N 423)	Greece	( 925 875)	Greece
Q423998869	(*N 423)	Greece	( 998 869)	PC-Greece
Q424697037	(*N 424)	Israel (Hebrew)	( 697 037)	USA/Canada
Q424697256	(*N 424)	Israel (Hebrew)	( 697 256)	Multinational #1
Q424697500	(*N 424)	Israel (Hebrew)	( 697 500)	Multinational #5
Q424941862	(*N 424)	Israel (Hebrew)	( 941 862)	PC-Israel
Q437A0ZA5S	(*N 437)	PC-USA	( 1285 1098)	PC-Farsi
Q437A69A3R	(*N 437)	PC-USA	( 1150 1025)	Cyrillic, Multilingual
Q437A69915	(*N 437)	PC-USA	( 1150 915)	8-bit ASCII/ISO Cyrillic
Q437A7X037	(*N 437)	PC-USA	( 1175 037)	USA/Canada
Q437A8C866	(*N 437)	PC-USA	( 1190 866)	PC-Cyrillic #2
Q437A9I852	(*N 437)	PC-USA	( 1232 852)	PC-Latin 2
Q437A9L855	(*N 437)	PC-USA	( 1235 855)	PC-Cyrillic
Q437A9N857	(*N 437)	PC-USA	( 1237 857)	PC-Turkey
Q437101037	(*N 437)	PC-USA	( 101 037)	USA/Canada
Q437337A5R	(*N 437)	PC-USA	( 337 1097)	Farsi
Q437337037	(*N 437)	PC-USA	( 337 037)	USA/Canada
Q437337273	(*N 437)	PC-USA	( 337 273)	Austria/Germany
Q437337278	(*N 437)	PC-USA	( 337 278)	Finland/Sweden
Q437337280	(*N 437)	PC-USA	( 337 280)	Italy
Q437337284	(*N 437)	PC-USA	( 337 284)	Spain/Latin America
Q437337285	(*N 437)	PC-USA	( 337 285)	United Kingdom
Q437337297	(*N 437)	PC-USA	( 337 297)	France
Q437337500	(*N 437)	PC-USA	( 337 500)	Multinational #5
Q437959870	(*N 437)	PC-USA	( 959 870)	Latin 2, Multilingual

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q437960880	(*N 437)	PC-USA	( 960 880)	Cyrillic, Multilingual (old)
Q500A0MA5K	(*N 500)	Multinational #5	( 1272 1090)	VT100 Line drawing set
Q500A00A5Z	(*N 500)	Multinational #5	( 1295 1105)	VT220 Norwegian/Danish
Q500A05A5U	(*N 500)	Multinational #5	( 1290 1100)	VT220 Multinational
Q500A06A5V	(*N 500)	Multinational #5	( 1291 1101)	VT220 British
Q500A07A5W	(*N 500)	Multinational #5	( 1292 1102)	VT220 Dutch
Q500A08A5X	(*N 500)	Multinational #5	( 1293 1103)	VT220 Finnish
Q500A69A3R	(*N 500)	Multinational #5	( 1150 1025)	Cyrillic, Multilingual
Q500A7AA3S	(*N 500)	Multinational #5	( 1152 1026)	Turkey Latin 5
Q500A7RA3B	(*N 500)	Multinational #5	( 1169 1009)	International Alphabet 5
Q500A7UA3T	(*N 500)	Multinational #5	( 1172 1027)	Japan extended
Q500A7U290	(*N 500)	Multinational #5	( 1172 290)	Japan Katakana
Q500A7W836	(*N 500)	Multinational #5	( 1174 836)	People's Republic of China
Q500A7X037	(*N 500)	Multinational #5	( 1175 037)	USA/Canada
Q500A7Y838	(*N 500)	Multinational #5	( 1176 838)	Thai Extended
Q500A7Y874	(*N 500)	Multinational #5	( 1176 874)	PC-Thai Extended
Q500A8EA3M	(*N 500)	Multinational #5	( 1192 1020)	VT220 Canadian/French
Q500A8FA3N	(*N 500)	Multinational #5	( 1193 1021)	VT220 Switzerland
Q500A8HA3P	(*N 500)	Multinational #5	( 1195 1023)	VT220 Spain
Q500A86A5R	(*N 500)	Multinational #5	( 1219 1097)	Farsi
Q500BAAA51	(*N 500)	Multinational #5	( 1296 1106)	VT220 Swedish
Q500BABA52	(*N 500)	Multinational #5	( 1297 1107)	VT220 Norwegian/Danish Alt
Q500BAJA57	(*N 500)	Multinational #5	( 1305 1112)	Baltic, Multilingual
Q500BALA6G	(*N 500)	Multinational #5	( 1307 1122)	Estonian
Q500BBEA60	(*N 500)	Multinational #5	( 1336 1130)	Vietnamese
Q500BBJA6Q	(*N 500)	Multinational #5	( 1341 1133)	Loa
Q500103367	(*N 500)	Multinational #5	( 103 367)	ASCII
Q500235420	(*N 500)	Multinational #5	( 235 420)	Arabic Bilingual
Q500265A3D	(*N 500)	Multinational #5	( 265 1011)	VT220 Germany
Q500289A5Y	(*N 500)	Multinational #5	( 289 1104)	VT220 French

Table name	From	From text	To	To text
Q500293A3E	(*N 500)	Multinational #5	( 293 1012)	VT220 Italy
Q500337437	(*N 500)	Multinational #5	( 337 437)	PC-USA
Q500337836	(*N 500)	Multinational #5	( 337 836)	People's Republic of China
Q500337850	(*N 500)	Multinational #5	( 337 850)	PC-Multilingual
Q500337860	(*N 500)	Multinational #5	( 337 860)	PC-Portugal
Q500337861	(*N 500)	Multinational #5	( 337 861)	PC-Iceland
Q500337863	(*N 500)	Multinational #5	( 337 863)	PC-Canadian French
Q500337865	(*N 500)	Multinational #5	( 337 865)	PC-Nordic
Q500697037	(*N 500)	Multinational #5	( 697 037)	USA/Canada
Q500697280	(*N 500)	Multinational #5	( 697 280)	Italy
Q500925875	(*N 500)	Multinational #5	( 925 875)	Greece
Q500936836	(*N 500)	Multinational #5	( 936 836)	People's Republic of China
Q500941424	(*N 500)	Multinational #5	( 941 424)	Israel (Hebrew)
Q500959870	(*N 500)	Multinational #5	( 959 870)	Latin 2, Multilingual
Q500960880	(*N 500)	Multinational #5	( 960 880)	Cyrillic, Multilingual (old)
Q500965905	(*N 500)	Multinational #5	( 965 905)	PC-Turkey
Q500981851	(*N 500)	Multinational #5	( 981 851)	PC-Greece (old)
Q500998869	(*N 500)	Multinational #5	( 998 869)	PC-Greece
Q813998869	(*N 813)	8-bit ASCII/ISO Greece	( 998 869)	PC-Greece
Q819BBEA60	(*N 819)	8-bit ASCII/ISO Latin 1	( 1336 1130)	Vietnamese
Q819BBJA6Q	(*N 819)	8-bit ASCII/ISO Latin 1	( 1341 1133)	Lao
Q833A0SA5I	(*N 833)	Korea	( 1278 1088)	PC-Korean
Q833A7RA3B	(*N 833)	Korea	( 1169 1009)	International Alphabet 5
Q833337256	(*N 833)	Korea	( 337 256)	Multinational #1
Q833337891	(*N 833)	Korea	( 337 891)	PC-Korea
Q833933256	(*N 833)	Korea	( 933 256)	Multinational #1
Q836A7RA3B	(*N 836)	People's Republic of China	( 1169 1009)	International Alphabet 5
Q836A7V833	(*N 836)	People's Republic of China	( 1173 833)	Korea

Table name	From	From text	To	To text
Q836A7X037	(*N 836)	People's Republic of China	( 1175 037)	USA/Canada
Q836101037	(*N 836)	People's Republic of China	( 101 037)	USA/Canada
Q836103A50	(*N 836)	People's Republic of China	( 103 1115)	People's Republic of China GB
Q836337256	(*N 836)	People's Republic of China	( 337 256)	Multinational #1
Q836337500	(*N 836)	People's Republic of China	( 337 500)	Multinational #5
Q836337903	(*N 836)	People's Republic of China	( 337 903)	PC-People's Republic of China
Q836697037	(*N 836)	People's Republic of China	( 697 037)	USA/Canada
Q836697284	(*N 836)	People's Republic of China	( 697 284)	Spain/Latin America
Q836936500	(*N 836)	People's Republic of China	( 936 500)	Multinational #5
Q838A7Y874	(*N 838)	Thai Extended	( 1176 874)	PC-Thai Extended
Q838337037	(*N 838)	Thai Extended	( 337 037)	USA/Canada
Q838697500	(*N 838)	Thai Extended	( 697 500)	Multinational #5
Q850A0ZA5S	(*N 850)	PC-Multilingual	( 1285 1098)	PC-Farsi
Q850A6G897	(*N 850)	PC-Multilingual	( 1122 897)	PC-Japan
Q850A69A3R	(*N 850)	PC-Multilingual	( 1150 1025)	Cyrillic, Multilingual
Q850A69915	(*N 850)	PC-Multilingual	( 1150 915)	8-bit ASCII/ISO Cyrillic
Q850A7UA38	(*N 850)	PC-Multilingual	( 1172 1041)	PC-Japan extended
Q850A7Y874	(*N 850)	PC-Multilingual	( 1176 874)	PC-Thai Extended
Q850A8C866	(*N 850)	PC-Multilingual	( 1190 866)	PC-Cyrillic #2
Q850A84862	(*N 850)	PC-Multilingual	( 1217 862)	PC-Israel
Q850A9I852	(*N 850)	PC-Multilingual	( 1232 852)	PC-Latin 2
Q850A9L855	(*N 850)	PC-Multilingual	( 1235 855)	PC-Cyrillic
Q850A9N857	(*N 850)	PC-Multilingual	( 1237 857)	PC-Turkey
Q850A9U864	(*N 850)	PC-Multilingual	( 1244 864)	PC-Arabic
Q850BBEA6O	(*N 850)	PC-Multilingual	( 1336 1130)	Vietnamese
Q850BBJA6Q	(*N 850)	PC-Multilingual	( 1341 1133)	Lao
Q850337A5R	(*N 850)	PC-Multilingual	( 337 1097)	Farsi
Q850337037	(*N 850)	PC-Multilingual	( 337 037)	USA/Canada
Q850337273	(*N 850)	PC-Multilingual	( 337 273)	Austria/Germany

Table name	From	From text	To	To text
Q850337277	(*N 850)	PC-Multilingual	( 337 277)	Denmark/Norway
Q850337278	(*N 850)	PC-Multilingual	( 337 278)	Finland/Sweden
Q850337280	(*N 850)	PC-Multilingual	( 337 280)	Italy
Q850337282	(*N 850)	PC-Multilingual	( 337 282)	Portugal
Q850337284	(*N 850)	PC-Multilingual	( 337 284)	Spain/Latin America
Q850337285	(*N 850)	PC-Multilingual	( 337 285)	United Kingdom
Q850337297	(*N 850)	PC-Multilingual	( 337 297)	France
Q850337500	(*N 850)	PC-Multilingual	( 337 500)	Multinational #5
Q850337871	(*N 850)	PC-Multilingual	( 337 871)	Iceland
Q850959870	(*N 850)	PC-Multilingual	( 959 870)	Latin 2, Multilingual
Q850960880	(*N 850)	PC-Multilingual	( 960 880)	Cyrillic, Multilingual (old)
Q851218423	(*N 851)	PC-Greece (old)	( 218 423)	Greece
Q851925875	(*N 851)	PC-Greece (old)	( 925 875)	Greece
Q852A51850	(*N 852)	PC-Latin 2	( 1106 850)	PC-Multilingual
Q852A69A3R	(*N 852)	PC-Latin 2	( 1150 1025)	Cyrillic, Multilingual
Q852A8Y437	(*N 852)	PC-Latin 2	( 1212 437)	PC-USA
Q852959870	(*N 852)	PC-Latin 2	( 959 870)	Latin 2, Multilingual
Q852960880	(*N 852)	PC-Latin 2	( 960 880)	Cyrillic, Multilingual (old)
Q853965905	(*N 853)	PC-Latin 3	( 965 905)	PC-Turkey
Q855A51850	(*N 855)	PC-Cyrillic	( 1106 850)	PC-Multilingual
Q855A69A3R	(*N 855)	PC-Cyrillic	( 1150 1025)	Cyrillic, Multilingual
Q855A69915	(*N 855)	PC-Cyrillic	( 1150 915)	8-bit ASCII/ISO Cyrillic
Q855A8Y437	(*N 855)	PC-Cyrillic	( 1212 437)	PC-USA
Q855959870	(*N 855)	PC-Cyrillic	( 959 870)	Latin 2, Multilingual
Q855960880	(*N 855)	PC-Cyrillic	( 960 880)	Cyrillic, Multilingual (old)
Q857A51850	(*N 857)	PC-Turkey	( 1106 850)	PC-Multilingual
Q857A7AA3S	(*N 857)	PC-Turkey	( 1152 1026)	Turkey Latin 5
Q857A8Y437	(*N 857)	PC-Turkey	( 1212 437)	PC-USA
Q857965905	(*N 857)	PC-Turkey	( 965 905)	PC-Turkey
Q860337037	(*N 860)	PC-Portugal	( 337 037)	USA/Canada

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q860337282	(*N 860)	PC-Portugal	( 337 282)	Portugal
Q860337500	(*N 860)	PC-Portugal	( 337 500)	Multinational #5
Q861337500	(*N 861)	PC-Iceland	( 337 500)	Multinational #5
Q861337871	(*N 861)	PC-Iceland	( 337 871)	Iceland
Q862A51850	(*N 862)	PC-Israel	( 1106 850)	PC-Multilingual
Q862941424	(*N 862)	PC-Israel	( 941 424)	Israel (Hebrew)
Q863337037	(*N 863)	PC-Canadian French	( 337 037)	USA/Canada
Q863337500	(*N 863)	PC-Canadian French	( 337 500)	Multinational #5
Q864A51850	(*N 864)	PC-Arabic	( 1106 850)	PC-Multilingual
Q864235420	(*N 864)	PC-Arabic	( 235 420)	Arabic Bilingual
Q865337277	(*N 865)	PC-Nordic	( 337 277)	Denmark/Norway
Q865337500	(*N 865)	PC-Nordic	( 337 500)	Multinational #5
Q866A51850	(*N 866)	PC-Cyrillic #2	( 1106 850)	PC-Multilingual
Q866A69A3R	(*N 866)	PC-Cyrillic #2	( 1150 1025)	Cyrillic, Multilingual
Q866A8Y437	(*N 866)	PC-Cyrillic #2	( 1212 437)	PC-USA
Q869218423	(*N 869)	PC-Greece	( 218 423)	Greece
Q869337256	(*N 869)	PC-Greece	( 337 256)	Multinational #1
Q869337500	(*N 869)	PC-Greece	( 337 500)	Multinational #5
Q869925813	(*N 869)	PC-Greece	( 925 813)	8-bit ASCII/ISO Greece
Q869925875	(*N 869)	PC-Greece	( 925 875)	Greece
Q869981851	(*N 869)	PC-Greece	( 981 851)	PC-Greece (old)
Q870A69A3R	(*N 870)	Latin 2, Multilingual	( 1150 1025)	Cyrillic, Multilingual
Q870A69855	(*N 870)	Latin 2, Multilingual	( 1150 855)	PC-Cyrillic
Q870A69915	(*N 870)	Latin 2, Multilingual	( 1150 915)	8-bit ASCII/ISO Cyrillic
Q870A7RA3B	(*N 870)	Latin 2, Multilingual	( 1169 1009)	International Alphabet 5
Q870A9I852	(*N 870)	Latin 2, Multilingual	( 1232 852)	PC-Latin 2
Q870337256	(*N 870)	Latin 2, Multilingual	( 337 256)	Multinational #1
Q870697037	(*N 870)	Latin 2, Multilingual	( 697 037)	USA/Canada

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q870697500	(*N 870)	Latin 2, Multilingual	( 697 500)	Multinational #5
Q870697850	(*N 870)	Latin 2, Multilingual	( 697 850)	PC-Multilingual
Q870919437	(*N 870)	Latin 2, Multilingual	( 919 437)	PC-USA
Q870959852	(*N 870)	Latin 2, Multilingual	( 959 852)	PC-Latin 2
Q870959912	(*N 870)	Latin 2, Multilingual	( 959 912)	8-bit ASCII/ISO Latin 2
Q870960880	(*N 870)	Latin 2, Multilingual	( 960 880)	Cyrillic, Multilingual (old)
Q871A7RA3B	(*N 871)	Iceland	( 1169 1009)	International Alphabet 5
Q871337850	(*N 871)	Iceland	( 337 850)	PC-Multilingual
Q871337861	(*N 871)	Iceland	( 337 861)	PC-Iceland
Q871697256	(*N 871)	Iceland	( 697 256)	Multinational #1
Q874A51850	(*N 874)	PC-Thai Extended	( 1106 850)	PC-Multilingual
Q874A7Y500	(*N 874)	PC-Thai Extended	( 1176 500)	Multinational #5
Q874A7Y838	(*N 874)	PC-Thai Extended	( 1176 838)	Thai Extended
Q874337037	(*N 874)	PC-Thai Extended	( 337 037)	USA/Canada
Q875A7RA3B	(*N 875)	Greece	( 1169 1009)	International Alphabet 5
Q875218423	(*N 875)	Greece	( 218 423)	Greece
Q875337256	(*N 875)	Greece	( 337 256)	Multinational #1
Q875337500	(*N 875)	Greece	( 337 500)	Multinational #5
Q875925813	(*N 875)	Greece	( 925 813)	8-bit ASCII/ISO Greece
Q875981851	(*N 875)	Greece	( 981 851)	PC-Greece (old)
Q875998869	(*N 875)	Greece	( 998 869)	PC-Greece
Q880A69A3R	(*N 880)	Cyrillic, Multilingual (old)	( 1150 1025)	Cyrillic, Multilingual
Q880A69855	(*N 880)	Cyrillic, Multilingual (old)	( 1150 855)	PC-Cyrillic
Q880A69915	(*N 880)	Cyrillic, Multilingual (old)	( 1150 915)	8-bit ASCII/ISO Cyrillic
Q880A7RA3B	(*N 880)	Cyrillic, Multilingual (old)	( 1169 1009)	International Alphabet 5
Q880337256	(*N 880)	Cyrillic, Multilingual (old)	( 337 256)	Multinational #1



<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q880697037	(*N 880)	Cyrillic, Multilingual (old)	( 697 037)	USA/Canada
Q880697500	(*N 880)	Cyrillic, Multilingual (old)	( 697 500)	Multinational #5
Q880697850	(*N 880)	Cyrillic, Multilingual (old)	( 697 850)	PC-Multilingual
Q880919437	(*N 880)	Cyrillic, Multilingual (old)	( 919 437)	PC-USA
Q880959852	(*N 880)	Cyrillic, Multilingual (old)	( 959 852)	PC-Latin 2
Q880959870	(*N 880)	Cyrillic, Multilingual (old)	( 959 870)	Latin 2, Multilingual
Q880959912	(*N 880)	Cyrillic, Multilingual (old)	( 959 912)	8-bit ASCII/ISO Latin 2
Q891337833	(*N 891)	PC-Korea	( 337 833)	Korea
Q897A51850	(*N 897)	PC-Japan	( 1106 850)	PC-Multilingual
Q897A7UA3T	(*N 897)	PC-Japan	( 1172 1027)	Japan extended
Q897337290	(*N 897)	PC-Japan	( 337 290)	Japan Katakana
Q897358037	(*N 897)	PC-Japan	( 358 037)	USA/Canada
Q897640037	(*N 897)	PC-Japan	( 640 037)	USA/Canada
Q897697037	(*N 897)	PC-Japan	( 697 037)	USA/Canada
Q903A7W836	(*N 903)	PC-People's Republic of China	( 1174 836)	People's Republic of China
Q903337836	(*N 903)	PC-People's Republic of China	( 337 836)	People's Republic of China
Q904337037	(*N 904)	PC-People's Republic of China	( 337 037)	USA/Canada
Q905A7AA3S	(*N 905)	PC-Turkey	( 1152 1026)	Turkey Latin 5
Q905337037	(*N 905)	PC-Turkey	( 337 037)	USA/Canada
Q905697500	(*N 905)	PC-Turkey	( 697 500)	Multinational #5
Q905965853	(*N 905)	PC-Turkey	( 965 853)	PC-Latin 3
Q905965857	(*N 905)	PC-Turkey	( 965 857)	PC-Turkey
Q912A69A3R	(*N 912)	8-bit ASCII/ISO Latin 2	( 1150 1025)	Cyrillic, Multilingual
Q912959870	(*N 912)	8-bit ASCII/ISO Latin 2	( 959 870)	Latin 2, Multilingual
Q912960880	(*N 912)	8-bit ASCII/ISO Latin 2	( 960 880)	Cyrillic, Multilingual (old)
Q915A51850	(*N 915)	8-bit ASCII/ISO Cyrillic	( 1106 850)	PC-Multilingual

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
Q915A69A3R	(*N 915)	8-bit ASCII/ISO Cyrillic	( 1150 1025)	Cyrillic, Multilingual
Q915A8Y437	(*N 915)	8-bit ASCII/ISO Cyrillic	( 1212 437)	PC-USA
Q915A9L855	(*N 915)	8-bit ASCII/ISO Cyrillic	( 1235 855)	PC-Cyrillic
Q915959870	(*N 915)	8-bit ASCII/ISO Cyrillic	( 959 870)	Latin 2, Multilingual
Q915960880	(*N 915)	8-bit ASCII/ISO Cyrillic	( 960 880)	Cyrillic, Multilingual (old)
Q920A7AA3S	(*N 920)	ASCII-Turkey	( 1152 1026)	Turkey Latin 5
QA3BA69A3R	(*N 1009)	International Alphabet 5	( 1150 1025)	Cyrillic, Multilingual
QA3BA7AA3S	(*N 1009)	International Alphabet 5	( 1152 1026)	Turkey Latin 5
QA3BA7W836	(*N 1009)	International Alphabet 5	( 1174 836)	People's Republic of China
QA3B103367	(*N 1009)	International Alphabet 5	( 103 367)	ASCII
QA3B218423	(*N 1009)	International Alphabet 5	( 218 423)	Greece
QA3B332290	(*N 1009)	International Alphabet 5	( 332 290)	Japan Katakana
QA3B697037	(*N 1009)	International Alphabet 5	( 697 037)	USA/Canada
QA3B697273	(*N 1009)	International Alphabet 5	( 697 273)	Austria/Germany
QA3B697277	(*N 1009)	International Alphabet 5	( 697 277)	Denmark/Norway
QA3B697278	(*N 1009)	International Alphabet 5	( 697 278)	Finland/Sweden
QA3B697280	(*N 1009)	International Alphabet 5	( 697 280)	Italy
QA3B697284	(*N 1009)	International Alphabet 5	( 697 284)	Spain/Latin America
QA3B697297	(*N 1009)	International Alphabet 5	( 697 297)	France
QA3B697500	(*N 1009)	International Alphabet 5	( 697 500)	Multinational #5
QA3B697871	(*N 1009)	International Alphabet 5	( 697 871)	Iceland
QA3B925875	(*N 1009)	International Alphabet 5	( 925 875)	Greece

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
QA3B933833	(*N 1009)	International Alphabet 5	( 933 833)	Korea
QA3B936836	(*N 1009)	International Alphabet 5	( 936 836)	People's Republic of China
QA3B959870	(*N 1009)	International Alphabet 5	( 959 870)	Latin 2, Multilingual
QA3B960880	(*N 1009)	International Alphabet 5	( 960 880)	Cyrillic, Multilingual (old)
QA3D697500	(*N 1011)	VT220 Germany	( 697 500)	Multinational #5
QA3E697500	(*N 1012)	VT220 Italy	( 697 500)	Multinational #5
QA3M697500	(*N 1020)	VT220 Canadian/ French	( 697 500)	Multinational #5
QA3N697500	(*N 1021)	VT220 Switzerland	( 697 500)	Multinational #5
QA3P697500	(*N 1023)	VT220 Spain	( 697 500)	Multinational #5
QA3QA93A33	(*N 1024)	CCITT T.61 (EBCDIC)	( 1252 1036)	CCITT T.61 IBM PC
QA3RA69855	(*N 1025)	Cyrillic, Multilingual	( 1150 855)	PC-Cyrillic
QA3RA69915	(*N 1025)	Cyrillic, Multilingual	( 1150 915)	8-bit ASCII/ISO Cyrillic
QA3RA7RA3B	(*N 1025)	Cyrillic, Multilingual	( 1169 1009)	International Alphabet 5
QA3RA8C866	(*N 1025)	Cyrillic, Multilingual	( 1190 866)	PC-Cyrillic #2
QA3RA9L855	(*N 1025)	Cyrillic, Multilingual	( 1235 855)	PC-Cyrillic
QA3R337256	(*N 1025)	Cyrillic, Multilingual	( 337 256)	Multinational #1
QA3R697037	(*N 1025)	Cyrillic, Multilingual	( 697 037)	USA/Canada
QA3R697500	(*N 1025)	Cyrillic, Multilingual	( 697 500)	Multinational #5
QA3R697850	(*N 1025)	Cyrillic, Multilingual	( 697 850)	PC-Multilingual
QA3R919437	(*N 1025)	Cyrillic, Multilingual	( 919 437)	PC-USA
QA3R959852	(*N 1025)	Cyrillic, Multilingual	( 959 852)	PC-Latin 2
QA3R959870	(*N 1025)	Cyrillic, Multilingual	( 959 870)	Latin 2, Multilingual
QA3R959912	(*N 1025)	Cyrillic, Multilingual	( 959 912)	8-bit ASCII/ISO Latin 2
QA3R960880	(*N 1025)	Cyrillic, Multilingual	( 960 880)	Cyrillic, Multilingual (old)
QA3SA7A857	(*N 1026)	Turkey Latin 5	( 1152 857)	PC-Turkey
QA3SA7A920	(*N 1026)	Turkey Latin 5	( 1152 920)	ASCII-Turkey
QA3SA7RA3B	(*N 1026)	Turkey Latin 5	( 1169 1009)	International Alphabet 5
QA3S337037	(*N 1026)	Turkey Latin 5	( 337 037)	USA/Canada

<b>Table name</b>	<b>From</b>	<b>From text</b>	<b>To</b>	<b>To text</b>
QA3S337256	(*N 1026)	Turkey Latin 5	( 337 256)	Multinational #1
QA3S337500	(*N 1026)	Turkey Latin 5	( 337 500)	Multinational #5
QA3S965905	(*N 1026)	Turkey Latin 5	( 965 905)	PC-Turkey
QA3TA6G897	(*N 1027)	Japan extended	( 1122 897)	PC-Japan
QA3TA7UA38	(*N 1027)	Japan extended	( 1172 1041)	PC-Japan extended
QA3TA7U290	(*N 1027)	Japan extended	( 1172 290)	Japan Katakana
QA3T697037	(*N 1027)	Japan extended	( 697 037)	USA/Canada
QA3T697500	(*N 1027)	Japan extended	( 697 500)	Multinational #5
QA33A93A3Q	(*N 1036)	CCITT T.61 IBM PC	( 1252 1024)	CCITT T.61 (EBCDIC)
QA38A51850	(*N 1041)	PC-Japan extended	( 1106 850)	PC-Multilingual
QA38A7UA3T	(*N 1041)	PC-Japan extended	( 1172 1027)	Japan extended
QA38A7U290	(*N 1041)	PC-Japan extended	( 1172 290)	Japan Katakana
QA5IA7V833	(*N 1088)	PC-Korean	( 1173 833)	Korea
QA5KA7X037	(*N 1090)	VT100 Line drawing set	( 1175 037)	USA/Canada
QA5K697037	(*N 1090)	VT100 Line drawing set	( 697 037)	USA/Canada
QA5K697500	(*N 1090)	VT100 Line drawing set	( 697 500)	Multinational #5
QA5RA0ZA5S	(*N 1097)	Farsi	( 1285 1098)	PC-Farsi
QA5R337437	(*N 1097)	Farsi	( 337 437)	PC-USA
QA5R337850	(*N 1097)	Farsi	( 337 850)	PC-Multilingual
QA5R697037	(*N 1097)	Farsi	( 697 037)	USA/Canada
QA5R697500	(*N 1097)	Farsi	( 697 500)	Multinational #5
QA5SA86A5R	(*N 1098)	PC-Farsi	( 1219 1097)	Farsi
QA5S919437	(*N 1098)	PC-Farsi	( 919 437)	PC-USA
QA5S980850	(*N 1098)	PC-Farsi	( 980 850)	PC-Multilingual
QA5UA7X037	(*N 1100)	VT220 Multinational	( 1175 037)	USA/Canada
QA5U697037	(*N 1100)	VT220 Multinational	( 697 037)	USA/Canada
QA5U697500	(*N 1100)	VT220 Multinational	( 697 500)	Multinational #5
QA5V697500	(*N 1101)	VT220 British	( 697 500)	Multinational #5
QA5W697500	(*N 1102)	VT220 Dutch	( 697 500)	Multinational #5
QA5X697500	(*N 1103)	VT220 Finnish	( 697 500)	Multinational #5
QA5Y697500	(*N 1104)	VT220 French	( 697 500)	Multinational #5

Table name	From	From text	To	To text
QA5Z697500	(*N 1105)	VT220 Norwegian/ Danish	( 697 500)	Multinational #5
QA50A7W836	(*N 1115)	People's Republic of China GB	( 1174 836)	People's Republic of China
QA51697500	(*N 1106)	VT220 Swedish	( 697 500)	Multinational #5
QA52697500	(*N 1107)	VT220 Norwegian/ Danish Alt	( 697 500)	Multinational #5
QA57697037	(*N 1112)	Baltic, Multilingual	( 697 037)	USA/Canada
QA57697500	(*N 1112)	Baltic, Multilingual	( 697 500)	Multinational #5
QA59697037	(*N 1114)	Taiwan Industry (Big 5) code	( 697 037)	USA/Canada
QA6G697037	(*N 1122)	Estonian	( 697 037)	USA/Canada
QA6G697500	(*N 1122)	Estonian	( 697 500)	Multinational #5
QA6OA51850	(*N 1130)	Vietnamese	( 1106 850)	PC-Multilingual
QA6OBC8A99	(*N 1130)	Vietnamese	( BC8 1258)	MS Window, Vietnamese
QA6O697037	(*N 1130)	Vietnamese	( 697 037)	USA/Canada
QA6O697500	(*N 1130)	Vietnamese	( 697 500)	Multinational #5
QA6O697819	(*N 1130)	Vietnamese	( 697 819)	8-bit ASCII/ISO Latin 1
QA6QA51850	(*N 1133)	Lao	( 1106 850)	PC-Multilingual
QA6QBBJA6R	(*N 1133)	Lao	( 1341 1133)	8-bit ASCII/ISO Lao
QA6Q697037	(*N 1133)	Lao	( 697 037)	USA/Canada
QA6Q697500	(*N 1133)	Lao	( 697 500)	Multinational #5
QA6Q697819	(*N 1133)	Lao	( 697 819)	8-bit ASCII/ISO Latin 1
QA6RBBJA6Q	(*N 1133)	8-bit ASCII/ISO Lao	( 1341 1133)	Lao
QA99BBEA6O	(*N 1258)	MS Window, Vietnamese	( 1336 1130)	Vietnamese

### Related tasks

[Job default coded character set identifier](#)

A job attribute, job default CCSID (DFTCCSID), is created for jobs with a CCSID of 65535. The DFTCCSID value is used by a system code when a CCSID other than 65535 is needed.

### International DP 94 00103 (ASCII)

The figure shows International DP 94 00103 (ASCII).

A	B	C	D	E	F	G	H	I	J	K	L	M
LA020000	LE020000	LC020000	LD020000	LE020000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA010000	LB010000	LC010000	LD010000	LE010000	LF010000	LG010000	LH010000	LI010000	LJ010000	LK010000	LL010000	LM010000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN010000	LO010000	LP010000	LQ010000	LR010000	LS010000	LT010000	LU010000	LV010000	LW010000	LX010000	LY010000	LZ010000
0	1	2	3	4	5	6	7	8	9			
ND010000	ND010000	ND020000	ND030000	ND040000	ND050000	ND060000	ND070000	ND080000	ND090000			
+	<	=	>	\$	'	^	~					
SA010000	SA030000	SA040000	SA050000	SC030000	SD130000	SD150000	SD190000					
#	%	&	*	@	[	\	]	{		}	!	"
SM010000	SM020000	SM030000	SM040000	SM050000	SM060000	SM070000	SM080000	SM110000	SM130000	SM140000	SP020000	SP040000
'	(	)	,	_	-	.	/	:	;	?		
SP050000	SP060000	SP070000	SP080000	SP090000	SP100000	SP110000	SP120000	SP130000	SP140000	SP150000		

Character Set 00103

### Character set 01169 (International Alphabet 5)

The figure shows character set 01169 (International Alphabet 5).

A	B	C	D	E	F	G	H	I	J	K	L	M
LA020000	LB020000	LC020000	LD020000	LE020000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA010000	LB010000	LC010000	LD010000	LE010000	LF010000	LG010000	LH010000	LI010000	LJ010000	LK010000	LL010000	LM010000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN010000	LO010000	LP010000	LQ010000	LR010000	LS010000	LT010000	LU010000	LV010000	LW010000	LX010000	LY010000	LZ010000
0	1	2	3	4	5	6	7	8	9			
ND010000	ND010000	ND020000	ND030000	ND040000	ND050000	ND060000	ND070000	ND080000	ND090000			
+	<	=	>	∑	'	^	~	#	%	&	*	@
SA010000	SA030000	SA040000	SA050000	SC010000	SD130000	SD150000	SD190000	SM110000	SM220000	SM030000	SM040000	SM120000
[	\	]	{		}	!	"	'	(	)	,	_
SM050000	SM070000	SM080000	SM110000	SM130000	SM140000	SP020000	SP040000	SP050000	SP060000	SP070000	SP080000	SP090000
-	.	/	:	;	?							
SP100000	SP110000	SP120000	SP130000	SP140000	SP150000							

Character Set 01169

### Invariant character set (and its exceptions)

An *invariant character set* is a character set, such as the syntactic character set, whose code point assignments do not change from code page to code page. The table illustrates the invariant character set (character set 00640) on the IBM i operating system.

With only a few exceptions, each EBCDIC code page contains a common set of graphic characters. Within an encoding scheme, the common characters can be found at the same code points. The following EBCDIC code pages are exceptions:

- EBCDIC code page 290 (for Japanese support) has Katakana characters at the code points where lowercase a through z are in the invariant character set
- EBCDIC code pages 905, 1026, 1155 & 1175 (for Turkish support) where the quotation mark (") is not at the same code point as it is in the invariant character set
- EBCDIC code page 420 does not contain the invariant character SM040000 (asterisk (\*)). However, code page 420 has a similar character named SM040007 ( \* ) at the same code point where SM040000 normally exists.
- EBCDIC code page 420 does not contain the invariant character SM020000 (percent sign (%)). However, code page 420 has a similar character named SM020007 ( % ) at the same code point where SM020000 normally exists.

**Invariant character set symbols**

<b>GCGID</b>	<b>Description</b>	<b>Graphic character</b>
LA010000	Latin small letter a	a
LA020000	Latin capital letter A	A
LB010000	Latin small letter b	b
LB020000	Latin capital letter B	B
LC010000	Latin small letter c	c
LC020000	Latin capital letter C	C
LD010000	Latin small letter d	d
LD020000	Latin capital letter D	D
LE010000	Latin small letter e	e
LE020000	Latin capital letter E	E
LF010000	Latin small letter f	f
LF020000	Latin capital letter F	F
LG010000	Latin small letter g	g
LG020000	Latin capital letter G	G
LH010000	Latin small letter h	h
LH020000	Latin capital letter H	H
LI010000	Latin small letter i	i
LI020000	Latin capital letter I	I
LJ010000	Latin small letter j	j
LJ020000	Latin capital letter J	J
LK010000	Latin small letter k	k
LK020000	Latin capital letter K	K
LL010000	Latin small letter l	l
LL020000	Latin capital letter L	L
LM010000	Latin small letter m	m
LM020000	Latin capital letter M	M
LN010000	Latin small letter n	n
LN020000	Latin capital letter N	N
LO010000	Latin small letter o	o
LO020000	Latin capital letter O	O
LP010000	Latin small letter p	p
LP020000	Latin capital letter P	P
LQ010000	Latin small letter q	q
LQ020000	Latin capital letter Q	Q

<b>GCGID</b>	<b>Description</b>	<b>Graphic character</b>
LR010000	Latin small letter r	r
LR020000	Latin capital letter R	R
LS010000	Latin small letter s	s
LS020000	Latin capital letter S	S
LT010000	Latin small letter t	t
LT020000	Latin capital letter T	T
LU010000	Latin small letter u	u
LU020000	Latin capital letter U	U
LV010000	Latin small letter v	v
LV020000	Latin capital letter V	V
LW010000	Latin small letter w	w
LW020000	Latin capital letter W	W
LX010000	Latin small letter x	x
LX020000	Latin capital letter X	X
LY010000	Latin small letter y	y
LY020000	Latin capital letter Y	Y
LZ010000	Latin small letter z	z
LZ020000	Latin capital letter Z	Z
ND100000	Arabic number zero	0
ND010000	Arabic number one	1
ND020000	Arabic number two	2
ND030000	Arabic number three	3
ND040000	Arabic number four	4
ND050000	Arabic number five	5
ND060000	Arabic number six	6
ND070000	Arabic number seven	7
ND080000	Arabic number eight	8
ND090000	Arabic number nine	9
SA010000	Plus sign	+
SA030000	Less than sign	<
SA040000	Equal sign	=
SA050000	Greater than sign	>
SM020000	Percent sign	%
SM030000	Ampersand	&
SM040000	Asterisk	*



<b>GCGID</b>	<b>Description</b>	<b>Graphic character</b>
SP040000	Straight quotation mark	"
SP050000	Straight single quotation mark	'
SP060000	Left parenthesis	(
SP070000	Right parenthesis	)
SP080000	Comma	,
SP090000	Underscore	_
SP100000	Hyphen	-
SP110000	Period	.
SP120000	Slash right	/
SP130000	Colon	:
SP140000	Semicolon	;
SP150000	Question mark	?

### **Related concepts**

#### Application part names

When you want to enable your application for different languages and countries, consider the environments of the target systems in your naming conventions.

#### Coding globalized applications with high-level languages

Your major goal must be to have only one general set of running code that is common for all language versions and to make your programs table-driven as much as possible.

### **Monocase tables**

Here is a list of monocase tables on the IBM i operating system.

<b>Code page</b>	<b>Table object for monocase</b>	<b>Description</b>
00037	Q037	USA/Canada (EBCDIC)
00256	Q256	International 1 (EBCDIC)
00260	Q260	Canadian French (EBCDIC)
00273	Q273	Germany/Austria (EBCDIC)
00277	Q277	Denmark, Norway (EBCDIC)
00278	Q278	Finland, Sweden (EBCDIC)
00280	Q280	Italy (EBCDIC)
00281	Q281	Japan Latin (EBCDIC)
00284	Q284	Spain/Latin America (EBCDIC)
00285	Q285	United Kingdom (EBCDIC)
00290	Q290	Japanese Katakana extended
00297	Q297	France (EBCDIC)
00420	Q420	Arabic Bilingual (EBCDIC)
00423	Q423	Greece (EBCDIC)
00424	Q424	Israel (Hebrew)

Code page	Table object for monospace	Description
00437	Q437	USA (IBM personal computer)
00500	Q500	Multilingual #5
00833	Q833	Korean extended (EBCDIC)
00836	Q836	Simplified Chinese extended (EBCDIC)
00838	Q838	Thai extended (EBCDIC)
00850	Q850	Multilingual (IBM personal computer)
00851	Q851	Greece (IBM personal computer)
00857	Q857	Turkey (ISO 8859-5)
00860	Q860	Portugal (IBM personal computer)
00861	Q861	Iceland (IBM personal computer)
00862	Q862	Israel (IBM personal computer)
00863	Q863	Canadian French (IBM personal computer)
00864	Q864	Arabic (IBM personal computer)
00865	Q865	Nordic (IBM personal computer)
00870	Q870	Multilingual (ISO 8859-2)
00871	Q871	Iceland (EBCDIC)
00875	Q875	Greece (EBCDIC)
00880	Q880	Cyrillic, Multilingual
00891	Q891	Korea (IBM personal computer)
00897	Q897	Japan PC #1 (IBM Personal Computer)
00903	Q903	People's Republic of China (IBM personal computer)
00904	Q904	Taiwan (IBM personal computer)
00905	Q905	PC-Turkey (ISO 8859-9)
01025	QA3R	Cyrillic, multilingual (EBCDIC)
01026	QA3S	Turkey (ISO 8859-9)
01027	QA3T	Japanese (Latin) extended (EBCDIC)
01097	QA5R	Farsi bilingual (EBCDIC)

#### Portable character set

The X/Open portable character set is a superset of the IBM invariant character set (00640), which includes symbols not represented in the IBM invariant character set 00640.

The portable character set includes the following 13 symbols that are not represented in the IBM invariant character set 00640.

<b>GCGID</b>	<b>Description</b>	<b>Graphic character</b>
SC030000	Dollar	\$
SD110000	Accent acute	´
SD150000	Caret	^
SD190000	Tilde	~
SM010000	Number sign	#
SM050000	At sign	@
SM060000	Left bracket	[
SM070000	Back slash	\
SM080000	Right bracket	]
SM110000	Left brace	{
SM130000	Logical or	
SM140000	Right brace	}
SP020000	Exclamation point	!

The portable character set also includes the space character and control characters representing the horizontal tab, the vertical tab, and form feed.

#### **Related concepts**

##### Locale symbolic names

The IBM i operating system supports locale symbolic names based on predefined names from the X/Open Standard portable character set.

#### **Syntactic/invariant character set 00640**

With only a few exceptions, each EBCDIC code page contains a common set of graphic characters. Within an encoding scheme, the common characters can be found at the same code points. However, there are exceptions.

The exceptions are as follows:

- EBCDIC code page 290 has Katakana characters at the code points where lowercase a through z are in the invariant character set.
- EBCDIC code page 905 where the quotation mark (") is not at the same code point as it is in the invariant character set.
- EBCDIC code page 1026 where the quotation mark (") is not at the same code point as it is in the invariant character set.
- EBCDIC code page 420 does not contain the invariant character SM040000 (asterisk (\*)). However, code page 420 has a similar character named SM040007 ( \* ) at the same code point where SM040000 normally exists.
- EBCDIC code page 420 does not contain the invariant character SM020000 (percent sign (%)). However, code page 420 has a similar character named SM020007 ( % ) at the same code point where SM020000 normally exists.

A	B	C	D	E	F	G	H	I	J	K	L	M
LA620000	LB202000	LC024000	LD206000	LE028000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA010000	LB010000	LC010000	LD010000	LE010000	LF010000	LG010000	LH010000	LI010000	LJ010000	LK010000	LL010000	LM010000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN010000	LO010000	LP010000	LQ010000	LR010000	LS010000	LT010000	LU010000	LV010000	LW010000	LX010000	LY010000	LZ010000
0	1	2	3	4	5	6	7	8	9			
ND030000	ND040000	ND050000	ND060000	ND070000	ND080000	ND090000	ND0A0000	ND0B0000	ND0C0000	ND0D0000		
+	<	=	>	%	&	*	"	'	(	)	,	_
SA010000	SA020000	SA030000	SA040000	SA050000	SA060000	SA070000	SA080000	SA090000	SA0A0000	SA0B0000	SA0C0000	SA0D0000
-	.	/	:	;	?							
SP100000	SP110000	SP120000	SP130000	SP140000	SP150000							

Character Set 00640

### T.61 Character Set 01252

This figure shows T.61 Character Set 01252.

A	B	C	D	E	F	G	H	I	J	K	L	M
LA620000	LB202000	LC024000	LD206000	LE028000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA010000	LB010000	LC010000	LD010000	LE010000	LF010000	LG010000	LH010000	LI010000	LJ010000	LK010000	LL010000	LM010000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN010000	LO010000	LP010000	LQ010000	LR010000	LS010000	LT010000	LU010000	LV010000	LW010000	LX010000	LY010000	LZ010000
Æ	Ð	Ĥ	Ų	Ł	Ł	Œ	Œ	ƒ	ƒ	ŋ	œ	ø
LA520000	LD620000	LH820000	LI520000	LL820000	LL640000	LM820000	LO620000	LO420000	LT820000	LT640000		
æ	đ	ĥ	ħ	ij	ı	κ	ı	ı	ı	ı	œ	ø
LA510000	LD610000	LD630000	LH810000	LJ510000	LJ690000	LJ610000	LL810000	LL830000	LN610000	LN630000	LO510000	LO610000
ß	t	þ										
LS610000	LT810000	LT830000										
0	1	2	3	4	5	6	7	8	9			
ND030000	ND040000	ND050000	ND060000	ND070000	ND080000	ND090000	ND0A0000	ND0B0000	ND0C0000			
²	³	½	¼	¾	+	±	<	=	>	÷	×	∞
ND021000	ND031000	NF010000	NF340000	NF500000	SA010000	SA020000	SA030000	SA040000	SA050000	SA060000	SA070000	SC010000
£	\$	¢	¥	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ
SC020000	SC030000	SC040000	SC050000	SD118000	SD138000	SD158000	SD178000	SD198000	SD218000	SD238000	SD258000	SD278000
·	-	ˆ	ˆ	ˆ	#	%	&	*	@	[	]	
SD298000	SD318000	SD418000	SD438000	SD630000	SM010000	SM020000	SM030000	SM040000	SM050000	SM060000	SM070000	SM130000
µ	Ω	∞	∞	∞	§	¶	!	!	!"	!"	!"	!"
SM170000	SM180000	SM190000	SM200000	SM210000	SM240000	SM250000	SP020000	SP030000	SP040000	SP050000	SP060000	SP070000
ˆ	ˆ	ˆ	-	.	/	:	:	?	ˆ	«	»	
SP080000	SP090000	SP098000	SP100000	SP110000	SP120000	SP130000	SP140000	SP150000	SP160000	SP170000	SP180000	

Character Set 01252

## T.61 Character Repertoire 01253

The figure shows T.61 Character Repertoire 01253.

A	B	C	D	E	F	G	H	I	J	K	L	M
LA020000	LB020000	LC020000	LD020000	LE020000	LF020000	LG020000	LH020000	LI020000	LJ020000	LK020000	LL020000	LM020000
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
LN020000	LO020000	LP020000	LQ020000	LR020000	LS020000	LT020000	LU020000	LV020000	LW020000	LX020000	LY020000	LZ020000
a	b	c	d	e	f	g	h	i	j	k	l	m
LA030000	LB030000	LC030000	LD030000	LE030000	LF030000	LG030000	LH030000	LI030000	LJ030000	LK030000	LL030000	LM030000
n	o	p	q	r	s	t	u	v	w	x	y	z
LN030000	LO030000	LP030000	LQ030000	LR030000	LS030000	LT030000	LU030000	LV030000	LW030000	LX030000	LY030000	LZ030000
Á	Â	Ã	Ä	Å	Ă	Ą	Ȧ	Æ	Č	Ĉ	Ċ	
LA120000	LA140000	LA160000	LA180000	LA200000	LA240000	LA280000	LA320000	LA440000	LA520000	LC120000	LC160000	LC220000
Ć	Ĉ	Ċ	Đ	Ð	Ê	Ë	Ĕ	Ė	Ę	Ě	Ĝ	
LC300000	LC420000	LD020000	LD200000	LE120000	LE140000	LE160000	LE180000	LE200000	LE320000	LE380000	LE440000	LG160000
Ĝ	Ğ	Ġ	Ĥ	Ħ	Í	Ĭ	Ī	Ĵ	Ķ	Ĳ	Ĵ	Ĵ
LG200000	LG300000	LG420000	LH180000	LH200000	LI120000	LI140000	LI160000	LI180000	LI200000	LI300000	LI320000	LI440000
Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ	Ĵ
LJ120000	LJ160000	LK420000	LI120000	LI200000	LI420000	LI480000	LI600000	LI120000	LI200000	LI280000	LI420000	LI480000
Ó	Ò	Ô	Ö	Õ	Ö	Œ	Ø	Ř	Ŕ	Ŗ	Š	
LO120000	LO140000	LO160000	LO180000	LO200000	LO280000	LO320000	LO380000	LO420000	LI120000	LI220000	LI420000	LS120000
Š	Š	Š	Ť	Ť	Ť	Ť	Ť	Ť	Ť	Ť	Ť	Ť
LS160000	LS220000	LS420000	LT120000	LT200000	LT200000	LT400000	LU120000	LU140000	LU160000	LU180000	LU200000	LU240000
Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů	Ů
LU260000	LU280000	LU320000	LU440000	LW160000	LY120000	LY160000	LY180000	LZ120000	LZ220000	LZ300000		
á	â	ã	ä	å	ă	ą	ȧ	æ	č	ĉ	ċ	ċ
LA110000	LA130000	LA150000	LA180000	LA190000	LA200000	LA270000	LA310000	LA440000	LA510000	LC110000	LC160000	LC210000
ĉ	ç	ď	đ	ð	é	è	ê	ë	ě	ê	e	ę
LC200000	LC410000	LD010000	LD600000	LD630000	LE110000	LE130000	LE150000	LE170000	LE210000	LE260000	LE310000	LE430000
ĝ	ĝ	ĝ	ĝ	ĥ	h	í	ì	î	ï	ĵ	ĵ	ĵ
LG150000	LG230000	LG280000	LG400000	LH150000	LH100000	LI160000	LI300000	LI150000	LI170000	LI180000	LI310000	LI430000
ĵ	ı	ĵ	ķ	κ	ı	ı	ı	ı	ı	ı	ı	ı
LJ150000	LJ160000	LJ190000	LK410000	LJ160000	LI150000	LI210000	LI410000	LI160000	LI300000	LI110000	LI160000	LI210000
ŋ	ŋ	ŋ	ó	ò	ô	ö	ö	ö	ö	œ	ø	í
LI410000	LI610000	LI620000	LO110000	LO130000	LO150000	LO170000	LO190000	LO250000	LO160000	LO100000	LO610000	LI110000
ř	ŕ	š	š	š	š	š	š	š	š	š	š	š
LR210000	LR410000	LS110000	LS150000	LS210000	LS410000	LS610000	LT210000	LT410000	LT610000	LT630000	LU110000	LU130000
û	ü	ü	ü	ü	ü	ü	ü	ü	ü	ü	ü	ü
LU150000	LU170000	LU190000	LU200000	LU250000	LU270000	LU310000	LU430000	LU150000	LU160000	LY150000	LY170000	LZ110000
ž	ž	0	1	2	3	4	5	6	7	8	9	
LZ210000	LZ290000	ND100000	ND010000	ND200000	ND300000	ND400000	ND500000	ND600000	ND700000	ND800000	ND900000	
<sup>2</sup>	<sup>3</sup>	½	¼	¼	+	±	<	=	>	÷	×	
ND021000	ND031000	NF010000	NF040000	NF050000	SA010000	SA020000	SA030000	SA240000	SA050000	SA060000	SA070000	
ƒ	£	§	€	¥	'	`	^	..	~	v	v	"
SC210000	SC020000	SC030000	SC040000	SC050000	SD110000	SD130000	SD150000	SD170000	SD190000	SD210000	SD250000	SD260000
°	°	°	°	°	°	°	°	°	°	°	°	°
SD270000	SD290000	SD310000	SD410000	SD430000	SD630000							
#	%	&	*	@	[	]		μ	Ω	°	°	°
SM110000	SM120000	SM130000	SM140000	SM150000	SM160000	SM170000	SM180000	SM190000	SM190000	SM200000	SM210000	SM220000
§	¶	!	i	"	'	(	)	+	=	-	.	/
SM230000	SM250000	SP020000	SP030000	SP040000	SP050000	SP060000	SP070000	SP130000	SP160000	SP100000	SP110000	SP120000

Character Set 01253 (Sheet 1 of 2)

:	:	?	¿	«	»
SP130000	SP140000	SP150000	SP160000	SP170000	SP180000

Character Set 01253 (Sheet 2 of 2)

## T.61 graphic character conversions

The table contains a list of T.61 conversions supported on the IBM i operating system.

The \*N character in the From Value field means any character set is used.

These conversion tables are used to convert data to and from character set 01253 on code page 01024 to another supported character set and code page.

Table name	From description	From value	To description	To value
International alphabet 5	(*N 01009)	CCITT T.61 (EBCDIC)	(01253 01024)	

<b>Table name</b>	<b>From description</b>	<b>From value</b>	<b>To description</b>	<b>To value</b>
CCITT T.61 (EBCDIC)	(*N 01024)	Cyrillic, multilingual	(01150 01025)	
CCITT T.61 (EBCDIC)	(*N 01024)	Turkey EBCDIC	(01152 01026)	
CCITT T.61 (EBCDIC)	(*N 01024)	International alphabet 5	(01169 01009)	
CCITT T.61 (EBCDIC)	(*N 01024)	People's Republic of China	(01174 00836)	
CCITT T.61 (EBCDIC)	(*N 01024)	ASCII	(00103 00367)	
CCITT T.61 (EBCDIC)	(*N 01024)	Greece	(00218 00423)	
CCITT T.61 (EBCDIC)	(*N 01024)	Japan Katakana	(00332 00290)	
CCITT T.61 (EBCDIC)	(*N 01024)	USA/Canada	(00697 00037)	
CCITT T.61 (EBCDIC)	(*N 01024)	Austria/Germany	(00697 00273)	
CCITT T.61 (EBCDIC)	(*N 01024)	Denmark/Norway	(00697 00277)	
CCITT T.61 (EBCDIC)	(*N 01024)	Finland/Sweden	(00697 00278)	
CCITT T.61 (EBCDIC)	(*N 01024)	Italy	(00697 00280)	
CCITT T.61 (EBCDIC)	(*N 01024)	Spain/Latin America	(00697 00284)	
CCITT T.61 (EBCDIC)	(*N 01024)	France	(00697 00297)	
CCITT T.61 (EBCDIC)	(*N 01024)	Multinational #5	(00697 00500)	
CCITT T.61 (EBCDIC)	(*N 01024)	Iceland	(00697 00871)	
CCITT T.61 (EBCDIC)	(*N 01024)	Greece	(00925 00875)	
CCITT T.61 (EBCDIC)	(*N 01024)	Korea	(00933 00833)	
CCITT T.61 (EBCDIC)	(*N 01024)	People's Republic of China	(00936 00836)	
CCITT T.61 (EBCDIC)	(*N 01024)	Latin 2, multilingual	(00959 00870)	
CCITT T.61 (EBCDIC)	(*N 01024)	Cyrillic, multilingual (old)	(00960 00880)	

<b>Table name</b>	<b>From description</b>	<b>From value</b>	<b>To description</b>	<b>To value</b>
Cyrillic, multilingual	(*N 01025)	CCITT T.61 (EBCDIC)	(01253 001024)	
Turkey EBCDIC	(*N 01026)	CCITT T.61 (EBCDIC)	(01253 01024)	
USA/Canada	(*N 00037)	CCITT T.61 (EBCDIC)	(01253 01024)	
Austria/Germany	(*N 00273)	CCITT T.61 (EBCDIC)	(01253 01024)	
Denmark/Norway	(*N 00277)	CCITT T.61 (EBCDIC)	(01253 01024)	
Finland/Sweden	(*N 00278)	CCITT T.61 (EBCDIC)	(01253 01024)	
Italy	(*N 00280)	CCITT T.61 (EBCDIC)	(01253 01024)	
Spain/Latin America	(*N 00284)	CCITT T.61 (EBCDIC)	(01253 01024)	
Japan Katakana	(*N 00290)	CCITT T.61 (EBCDIC)	(01253 01024)	
France	(*N 00297)	CCITT T.61 (EBCDIC)	(01253 01024)	
ASCII	(*N 00367)	CCITT T.61 (EBCDIC)	(01253 01024)	
Greece	(*N 00423)	CCITT T.61 (EBCDIC)	(01253 01024)	
Multinational #5	(*N 00500)	CCITT T.61 (EBCDIC)	(01253 01024)	
Korea	(*N 00833)	CCITT T.61 (EBCDIC)	(01253 01024)	
People's Republic of China	(*N 00836)	CCITT T.61 (EBCDIC)	(01253 01024)	
Latin 2, multilingual	(*N 00870)	CCITT T.61 (EBCDIC)	(01253 01024)	
Iceland	(*N 00871)	CCITT T.61 (EBCDIC)	(01253 01024)	
Greece	(*N 00875)	CCITT T.61 (EBCDIC)	(01253 01024)	
Cyrillic, multilingual (old)	(*N 00880)	CCITT T.61 (EBCDIC)	(01253 01024)	

## CCSID reference information

Coded character set identifier (CCSID) is a 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

Character Data Representation Architecture (CDRA) has the concept of a growing CCSID. This CCSID is one where the code page is not full and new characters are added over time as needed. For more information about this concept, see [the CDRA documentation](#). To indicate a growing CCSID, CDRA uses a value of 65535 in the character set part of a CCSID. On IBM i, a growing CCSID is indicated by a value of 32000 instead.

### Related concepts

#### [Character data integrity](#)

The Character Data Representation Architecture (CDRA) system of tags uses coded character set identifiers (CCSIDs) to maintain data integrity when character data is passed from system to system or from user to user. CCSIDs assign a value that uniquely identifies the coded graphic character representation used for character data.

#### [Automatic character set and code page conversion](#)

The IBM i operating system provides automatic conversion between character set and code pages for all applications that are enabled for national language support.

#### [Coding globalized applications that use bidirectional data](#)

When you are developing NLV-enabled applications, you should consider some specific restrictions on bidirectional languages.

### CCSID values defined on IBM i

This table lists the coded character set identifiers (CCSIDs) that are defined on the IBM i operating system.

CCSID	Encoding	Description
00037	1100	US, Canada, Netherlands, Portugal, Brazil, New Zealand, Australia
00256	1100	Netherlands
00273	1100	Austria, Germany
00277	1100	Denmark, Norway
00278	1100	Finland, Sweden
00280	1100	Italy
00284	1100	Spanish, Latin America
00285	1100	United Kingdom
00290	1100	Japan Katakana
00297	1100	France
00300	1200	Japan English
00301	2200	Japanese PC Data
00367	5100	ANSI X3.4 ASCII standard; USA
00420	1100	Arabic-speaking countries
00423	1100	Greece
00424	1100	Hebrew
00425	1100	Arabic-speaking countries
00437	2100	PC Data; PC Base; USA



<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
00500	1100	Belgium, Canada, Switzerland, International Latin-1
00720	2100	MS-DOS Arabic
00737	2100	MS-DOS Greek PC-Data
00775	2100	MS-DOS Baltic PC-Data
00813	4100	ISO 8859-7; Greek/Latin
00819	4100	ISO 8859-1; Latin Alphabet No. 1
00833	1100	Korea (extended range)
00834	1200	Korea host double byte (including 1880 UDC)
00835	1200	Traditional Chinese host double byte (including 6204 UDC)
00836	1100	Simplified Chinese (extended range)
00837	1200	Simplified Chinese
00838	1100	Thailand (extended range)
00850	2100	PC Data; MLP 222 Latin Alphabet 1
00851	2100	PC Data; Greek
00852	2100	PC Data; Latin-2 Multilingual
00855	2100	PC Data; ROECE Cyrillic
00857	2100	PC Data; Turkey Latin #5
00858	2100	PC Data: MLP 222; Latin Alphabet Number 1 w/euro; Latin-1 Countries
00860	2100	PC Data; Portugal
00861	2100	PC Data; Iceland
00862	2100	PC Data; Hebrew
00863	2100	PC Data; Canada
00864	2100	PC Data; Arabic
00865	2100	PC Data; Denmark, Norway
00866	2100	PC Data; Cyrillic #2 - Personal Computer
00868	2100	PC Data: Urdu
00869	2100	PC Data; Greek
00870	1100	Latin-2 Multilingual
00871	1100	Iceland
00874	2100	Thai PC Data
00875	1100	Greece
00878	4105	Russian Internet KOI8-R Cyrillic
00880	1100	Cyrillic Multilingual
00891	2100	Korean PC Data (non-extended)
00897	2100	Japanese PC Data (non-extended)

<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
00903	2100	Simplified Chinese PC Data (non-extended)
00904	2100	Traditional Chinese PC Data
00905	1100	Turkey Latin-3
00912	4100	ISO 8859-2; ROECE Latin-2 Multilingual
00914	4100	Latin 4 - ISO 8859-4
00915	4100	ISO 8859-5; Cyrillic; 8-bit ISO
00916	4100	ISO 8859-8; Hebrew
00918	1100	Urdu EBCDIC
00920	4100	ISO 8859-9; Latin 5
00921	4100	Baltic, 8-bit (ISO 8859-13)
00922	4100	Estonia, 8-bit (ISO)
00923	4100	ISO 8859-15: Latin Alphabet with euro
00924	1100	Latin 9 EBCDIC
00926	2200	Korean PC Data DBCS, UDC 1880
00927	2200	Traditional Chinese PC Data DBCS, UDC 6204
00928	2200	Simplified Chinese PC Data DBCS, UDC 1880
00930	1301	Japan Katakana (extended range) 4370 UDC (User Defined Characters)
00932	2300	Japan PC Data Mixed
00933	1301	Korea (extended range), 1880 UDC
00934	2300	Korean PC Data
00935	1301	Simplified Chinese (extended range)
00936	2300	Simplified Chinese (non-extended)
00937	1301	Traditional Chinese (extended range)
00938	2300	Traditional Chinese (non-extended)
00939	1301	Japan English (extended range) 4370 UDC
00941	2200	Japanese DBCS PC for Open environment (Multi-vendor code): 6878 JIS X 0208-1990 characters, 386 IBM selected characters, 1880 IBM UDC (X'F040' to X'F9FC')
00942	2300	Japanese PC Data Mixed
00943	2300	Japanese PC Data Mixed for Open environment (Multi-vendor code): 6878 JIS X 0208-1990 characters, 386 IBM selected DBCS characters, 1880 UDC (X'F040' to X'F9FC')
00944	2300	Korean PC Data Mixed
00946	2300	Simplified Chinese PC Data Mixed
00947	2200	ASCII Double-byte

<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
00948	2300	Traditional Chinese PC Data Mixed 6204 UDC (User Defined Characters)
00949	2300	Republic of Korea National Standard Graphic Character Set (KS) PC Data mixed-byte including 1800 UDC
00950	2300	Traditional Chinese PC Data Mixed for Big5
00951	2200	Republic of Korea National Standard Graphic Character Set (KS) PC Data double-byte including 1800 UDC
00954	4403	Japanese EUC; G0 - JIS X201 Roman set (00895); G1 - JIS X208-1990 set (00952); G2 - JIS X201 Katakana set (04992); G3 - JIS X212 set (00953)
00956	5404	JIS X201 Roman for CP 00895; JIS X208-1983 for CP 00952
00957	5404	JIS X201 Roman for CP 00895; JIS X208-1978 for CP 00955
00958	5404	ASCII for CP 00367; JIS X208-1983 for CP 00952
00959	5404	ASCII for CP 00367; JIS X208-1978 for CP 00955
00964	4403	G0 - ASCII for CP 00367; G1- CNS 11643 plane 1 for CP 960
00965	5404	ASCII for CP 00367; CNS 11643 plane 1 for CP 960
00970	4403	G0 ASCII for CP 00367; G1 KSC X5601-1989 (including 188 UDCs) for CP 971
00971	8200	Korean EUC, G1 - KS C5601-1989 (including 188 UDC)
01008	4100	Arabic 8-bit ISO/ASCII
01009	5100	ISO-7: IRV
01010	5100	ISO-7; France
01011	5100	ISO-7; Germany
01012	5100	ISO-7; Italy
01013	5100	ISO-7; United Kingdom
01014	5100	ISO-7; Spain
01015	5100	ISO-7; Portugal
01016	5100	ISO-7; Norway
01017	5100	ISO-7; Denmark
01018	5100	ISO-7; Finland and Sweden
01019	5100	ISO-7; Belgium and Netherlands
01025	1100	Cyrillic Multilingual
01026	1100	Turkey Latin 5 CECP
01027	1100	Japan English (extended range)
01040	2100	Korean Latin PC Data extended
01041	2100	Japanese PC Data extended
01042	2100	Simplified Chinese PC Data extended

<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
01043	2100	Traditional Chinese PC Data extended
01046	2100	PC Data - Arabic Extended
01051	4100	HP Emulation(for use with Latin 1). GCGID SF150000 is mapped to a control X'7F'
01088	2100	Korean PC Data single-byte
01089	4100	ISO 8859-6: Arabic (string type 5)
01097	1100	Farsi
01098	2100	Farsi (IBM-PC)
01112	1100	Baltic, Multilingual
01114	2100	Traditional Chinese, Taiwan Industry Graphic Character Set (Big5)
01115	2100	Simplified Chinese National Standard (GB), personal computer SBCS
01122	1100	Estonia
01123	1100	Cyrillic Ukraine EBCDIC
01124	4100	Cyrillic Ukraine 8-Bit
01125	2100	Cyrillic Ukraine PC-Data
01126	2100	Windows Korean PC Data Single-Byte
01129	4100	ISO-8 Vietnamese
01130	1100	EBCDIC Vietnamese
01131	2100	Cyrillic Belarus PC-Data
01132	1100	EBCDIC Lao
01133	4100	ISO-8 Lao
01137	1100	Devanagari EBCDIC
01140	1100	ECECP: USA, Canada, Netherlands, Portugal, Brazil, Australia, New Zealand
01141	1100	ECECP: Austria, Germany
01142	1100	ECECP: Denmark, Norway
01143	1100	ECECP: Finland, Sweden
01144	1100	ECECP: Italy
01145	1100	ECECP: Spain, Latin America (Spanish)
01146	1100	ECECP: United Kingdom
01147	1100	ECECP: France
01148	1100	ECECP: International 1
01149	1100	ECECP: Iceland
01153	1100	Latin-2 - EBCDIC Multilingual with euro
01154	1100	Cyrillic Multilingual with euro
01155	1100	Turkey Latin 5 with euro

<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
01156	1100	Baltic, Multilingual with euro
01157	1100	Estonia EBCDIC with euro
01158	1100	Cyrillic Ukraine EBCDIC with euro
01160	1100	Thai host with euro
01164	1100	EBCDIC Vietnamese with euro
01166	1100	Cyrillic multilingual with Euro for Kazakhstan
01175	1100	Turkey with Euro and Turkish Lira
01200	7200	Unicode: UTF-16, big endian
01208	7807	Unicode: UTF-8
01250	4105	Windows, Latin 2
01251	4105	Windows, Cyrillic
01252	4105	Windows, Latin 1
01253	4105	Windows, Greek
01254	4105	Windows, Turkish
01255	4105	Windows, Hebrew
01256	4105	Windows, Arabic
01257	4105	Windows, Baltic Rim
01258	4105	MS Windows, Vietnamese
01275	4105	Apple Latin-1
01280	4105	Apple Greek
01281	4105	Apple Turkey
01282	4105	Apple Central European (Latin-2)
01283	4105	Apple Cyrillic
01362	2200	Windows Korean PC DBCS-PC, including 11 172 full hangul
01363	2300	Windows Korean PC Mixed, including 11 172 full hangul
01364	1301	Korean host mixed extended including 11 172 full hangul
01371	1301	Traditional Chinese host mixed including 6204 UDC, Extended SBCS including SBCS and DBCS euro (CCSID 5467 level)
01377	1301	Hong Kong Traditional Chinese mixed host enhancement for HKSCS-2004 (Mapping is HKSCS-2004 to Unicode 17584 level)
01380	2200	Simplified Chinese, People's Republic of China National Standard (GB), personal computer DBCS
01381	2300	Simplified Chinese, People's Republic of China National Standard (GB) personal computer mixed SBCS and DBCS
01382	8200	Simplified Chinese DBCS PC GB 2312-80 set, including 31 IBM selected and 1360 UDC.

<b>CCSID</b>	<b>Encoding</b>	<b>Description</b>
01383	4403	Simplified Chinese, EUC <ul style="list-style-type: none"> <li>• G0 set; ASCII</li> <li>• G1 set; GB 2312-80 set (1382)</li> </ul>
01385	2200	Simplified Chinese DBCS-PC GBK, all GBK character set and others
01386	2300	Simplified Chinese PC Data GBK mixed, all GBK character set and others
01388	1301	Simplified Chinese DBCS- GB 18030 Host with UDCs and Uygur extension.
01399	1301	Japanese Latin-Kanji Host Mixed including 4370 UDC, Extended SBCS (includes SBCS and DBCS euro)
04396	1200	Japanese Host DB including 1880
04930	1200	Korean DBCS-Host extended including 11 172 full hangul
04933	1200	Simplified Chinese DBCS Host (GBK), all GBK character set and others
04948	2100	Latin 2 PC Data Multilingual
04951	2100	Cyrillic PC Data Multilingual
04952	2100	Hebrew PC Data
04953	2100	Turkey PC Data Latin 5
04960	2100	Arabic PC Data
04965	2100	Greek PC Data
04970	2100	Thai PC Data Single-Byte
04971	1100	Greek (including euro)
05026	1301	Japan Katakana (extended range) 1880 UDC
05035	1301	Japan English (extended range) 1880 UDC
05050	4403	G0 - JIS X201 Roman for CP 895; G1 JIS X208-1990 for CP 952
05052	5404	JIS X201 Roman for CP 895; JIS X208-1983 for CP 952
05053	5404	JIS X201 Roman for CP 895; JIS X208-1978 for CP 955
05054	5404	ASCII for CP 367; JIS X208-1983 for CP 952
05055	5404	ASCII for CP 367; JIS X208-1978 for CP 955
05123	1100	Japanese Latin Host Extended SBCS (includes euro)
05210	2100	Simplified Chinese PC Data Single-Byte (GBK), growing CS
5233	1100	Devanagari EBCDIC, including Indian Rupee
05348	4105	Windows, Latin 1 with euro
08612	01100	Arabic (base shapes only)
09030	1100	Thai Host Extended SBCS
09056	2100	PC Data: Arabic PC Storage/Interchange
09066	2100	Thai PC Data Extended SBCS

CCSID	Encoding	Description
12708	1100	Arabic (base shapes, Lamaleph ligatures and Hindi digits) (string type 7)
13121	1100	Korean Host Extended SBCS
13124	1100	Simplified Chinese Host Data Single-Byte (GBK) equivalent to Simplified Chinese Host Data Single-Byte (GB) except growing CS
13488	7200	Unicode: UTF-16 as defined in the Unicode Standard. Fixed CS as defined by Unicode 2.0. Big endian
16684	1200	Japanese Latin Host Double-Byte including 4370 UDC (includes euro)
17354	5404	G0 - ASCII for CP 00367; G1 - KSC X5601-1989 (including 188 UDCs) for CP 00971
25546	5409	Korean 2022-KR TCP, ASCII, KS C5601-1989 (includes 188 UDC, RFC1557 using SO/SI)
28709	1100	Traditional Chinese (extended range)
33722	4403	Japanese EUC <ul style="list-style-type: none"> <li>• G0; JIS X201 Roman set (00895)</li> <li>• G1; JIS X208-1990 set (00952)</li> <li>• G2; JIS X201 Katakana set (04992)</li> <li>• G3; JIS X212 set (09145)</li> </ul>
57345	5404	All Japanese 2022 characters
61175	1100	IBM i specific Turkey with Euro and Turkish Lira but with the quotation mark (") characters that are swapped to the invariant character positions. This CCSID can be useful for SQL and related applications.  An example of conversions that involve the IBM i specific CCSID 61175: <ul style="list-style-type: none"> <li>• from CCSID 1175 to CCSID 61175: 'FC'x to '7F'x</li> <li>• from CCSID 1175 to CCSID 13488: 'FC'x to '0022'x</li> <li>• from CCSID 61175 to CCSID 13488: '7F'x to '0022'x</li> </ul>
61952	7200	IBM i specific (old CCSID for UCS). Use of 13488 is recommended instead.
62210	4100	IBM i specific ISO 8859-8; Hebrew, string type 4.
62211	1100	IBM i specific EBCDIC; Hebrew, string type 5
62215	4105	IBM i specific MS Windows; Hebrew, string type 4
62218	2100	IBM i specific PC data; Arabic, string type 4
62222	4100	IBM i specific ISO 8859-9; Hebrew, string type 6
62223	4105	IBM i specific MS Windows; Hebrew, string type 6
62224	1100	IBM i specific EBCDIC; Arabic, string type 6
62228	4105	IBM i specific MS Windows; Arabic, string type 6
62235	1100	IBM i specific EBCDIC; Hebrew, string type 6

CCSID	Encoding	Description
62238	4100	IBM i specific ISO 8859-9; Hebrew, string type 10
62239	4105	IBM i specific MS Windows; Hebrew, string type 10
62245	1100	IBM i specific EBCDIC; Hebrew, string type 10
65534		Look at lower level CCSID
65535		Special value indicating data is hex and should not be converted. This is the default for the QCCSID system value.

### Related information

IBM Globalization: Coded character set identifiers

[IBM Coded Character Sets and Related Resources Web site](#)

### Bidirectional Language String Types and Associated Attributes

Below are the string types and short description used on the system for bidi CCSIDs.

Typ e	Text Type 1	Numeric Shaping <sup>2</sup>	Orientation	Text Shaping <sup>3</sup>	Symmetric Swapping <sup>4</sup>
4	V	pass-through	LTR	Shaped	Off
5	I	Arabic	LTR	Unshaped	On
6	I	Arabic	RTL	Unshaped	On
8	V	pass-through	RTL	Shaped	Off
10	I	pass-through	Contextual Left	Unshaped	On

#### Notes:

**1**

Text type: V = Visual; I = Implicit

**2**


Numeric Shaping: In Arabic, it is common to use Hindi numbers instead of Arabic numbers. "1" "2" etc. are the Arabic version of the numbers.

**3**

Text Shaping: Specifies the shaping; that is, choosing (or composing) the correct shape of the input or output text.

**4**

Symmetrical Swapping: The Swapping descriptor specifies whether symmetric swapping is applied to the text. A list of symmetric swapping characters is given in the ISO/IEC 10646 standard. For example, the string "(1)" without might become ")1("

More information on this topic can be found at [IBM i Globalization](#) 

### Supported CCSID mappings

The IBM i operating system can dynamically create CCSID mappings on demand by using Unicode. The following CCSID mappings are predefined for better performance or mapping customization.

**Note:** Some of the mappings that involve mapping from a mixed encoding (such as CCSID 937 ) to a single byte encoding (such as CCSID 37) might return errors if any ideographic characters exist in the data.



<b>From CCSID</b>	<b>To CCSID</b>
00037	00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00367, 00420, 00423, 00424, 00425, 00437, 00500, 00720, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00901, 00902, 00903, 00904, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00924, 00930, 00933, 00935, 00937, 00939, 00948, 01025, 01026, 01027, 01041, 01043, 01051, 01088, 01089, 01097, 01098, 01112, 01114, 01115, 01122, 01123, 01124, 01126, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01364, 01388, 01399, 04909, 04970, 04971, 05026, 05035, 05123, 05348, 08612, 09030, 12708, 13121, 13124, 13488, 28709, 61952, 62211, 62224, 62235, 62245, 62251
00256	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00500, 00737, 00775, 00819, 00833, 00836, 00838, 00850, 00870, 00871, 00875, 00880, 00905, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01112, 01122, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 08612, 09030, 13121, 13124, 13488, 28709, 61952
00273	00037, 00256, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01250, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00277	00037, 00256, 00273, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00278	00037, 00256, 00273, 00277, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00280	00037, 00256, 00273, 00277, 00278, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952

<b>From CCSID</b>	<b>To CCSID</b>
00284	00037, 00256, 00273, 00277, 00278, 00280, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00285	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00290	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00737, 00775, 00819, 00833, 00836, 00850, 00871, 00895, 00897, 00930, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 01025, 01027, 01041, 01112, 01122, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 09030, 13121, 13124, 13488, 28709, 61952
00297	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 09030, 13121, 13124, 13488, 28709, 61952
00300	00301, 00930, 00939, 00941, 01200, 01208, 04396, 05026, 05035, 13488, 61952
00301	00300, 01200, 01208, 04396, 13488, 16684, 61952
00367	00037, 00500, 00833, 01200, 01208, 13121, 13124, 13488, 61952
00420	00037, 00256, 00425, 00500, 00720, 00737, 00775, 00819, 00850, 00864, 00937, 01008, 01046, 01089, 01112, 01122, 01200, 01208, 01252, 01256, 04960, 08612, 09030, 09056, 12708, 13488, 28709, 61952, 62218, 62224, 62228, 62248, 62251
00423	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00937, 01025, 01026, 01027, 01041, 01042, 01043, 01112, 01122, 01200, 01208, 01252, 01253, 04965, 04971, 05123, 09030, 13488, 28709, 61952
00424	00037, 00256, 00500, 00737, 00775, 00819, 00850, 00862, 00916, 00937, 01112, 01122, 01200, 01208, 01252, 01255, 04952, 09030, 13488, 28709, 61952, 62210, 62211, 62215, 62222, 62223, 62235, 62238, 62239, 62245
00425	00037, 00420, 00500, 00819, 00864, 01046, 01089, 01140, 01148, 01200, 01208, 01252, 01256, 08612, 12708, 13488, 61952, 62224, 62228

<b>From CCSID</b>	<b>To CCSID</b>
00437	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00852, 00855, 00857, 00860, 00861, 00863, 00865, 00866, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00937, 01025, 01026, 01027, 01042, 01051, 01097, 01098, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01257, 01275, 01280, 01281, 01282, 01283, 04971, 05123, 05348, 13121, 13124, 13488, 28709, 61952
00500	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00367, 00420, 00423, 00424, 00425, 00437, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00869, 00870, 00871, 00875, 00880, 00891, 00897, 00901, 00902, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00924, 00930, 00933, 00935, 00937, 00939, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01025, 01026, 01027, 01051, 01088, 01089, 01097, 01098, 01112, 01114, 01115, 01122, 01123, 01124, 01126, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01250, 01251, 01252, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01364, 01388, 01399, 04909, 04970, 04971, 05026, 05035, 05123, 05348, 08612, 09030, 12708, 13121, 13124, 13488, 28709, 61952, 62211, 62224, 62235, 62245, 62251
00720	00037, 00420, 00864, 01200, 01208, 01256, 13488, 61952
00737	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00437, 00500, 00833, 00836, 00838, 00850, 00869, 00870, 00871, 00875, 00880, 00905, 01025, 01026, 01027, 01097, 01200, 01208, 01252, 01253, 04971, 05123, 13121, 13124, 13488, 28709, 61952
00775	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00437, 00500, 00833, 00836, 00838, 00850, 00870, 00871, 00875, 00880, 00905, 01025, 01026, 01027, 01097, 01112, 01122, 01200, 01208, 01252, 01257, 05123, 13121, 13124, 13488, 28709, 61952
00813	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 01200, 01208, 01252, 04909, 04971, 05123, 13488, 28709, 61952
00819	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00425, 00437, 00500, 00813, 00833, 00836, 00838, 00850, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00924, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01041, 01042, 01043, 01051, 01088, 01089, 01097, 01098, 01112, 01114, 01122, 01126, 01130, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01364, 01388, 01399, 04971, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952, 62211, 62235, 62245, 62251

<b>From CCSID</b>	<b>To CCSID</b>
00833	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00367, 00437, 00500, 00737, 00775, 00819, 00836, 00850, 00871, 00891, 00930, 00933, 00935, 00937, 00939, 00944, 00949, 01027, 01040, 01088, 01112, 01122, 01126, 01200, 01208, 01252, 01363, 01364, 01388, 01399, 05026, 05035, 05123, 09030, 13121, 13124, 13488, 28709, 61952
00834	00926, 00933, 00951, 00971, 01200, 01208, 01362, 01364, 04930, 13488, 61952
00835	00927, 00937, 00947, 01200, 01208, 13488, 61952
00836	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00437, 00500, 00737, 00775, 00819, 00833, 00850, 00871, 00903, 00930, 00933, 00935, 00937, 00939, 00946, 01027, 01042, 01112, 01114, 01115, 01122, 01200, 01208, 01252, 01364, 01381, 01386, 01388, 01399, 05026, 05035, 05123, 09030, 13121, 13124, 13488, 28709, 61952
00837	00928, 00935, 01200, 01208, 01380, 01382, 01385, 01388, 13488, 61952
00838	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00937, 01025, 01026, 01027, 01041, 01042, 01043, 01112, 01122, 01160, 01200, 01208, 01252, 04970, 05123, 09030, 13488, 28709, 61952
00850	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00924, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01041, 01051, 01089, 01097, 01098, 01112, 01122, 01130, 01132, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01200, 01208, 01250, 01251, 01252, 01254, 01255, 01256, 01257, 01275, 01280, 01281, 01282, 01283, 01364, 01388, 01399, 04971, 05026, 05035, 05123, 05348, 08612, 09030, 09056, 13121, 13124, 13488, 28709, 61952, 62211, 62235, 62245
00851	01200, 01208, 13488, 61952
00852	00037, 00273, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00869, 00870, 00874, 00875, 00897, 00903, 00912, 01153, 01200, 01208, 01252, 04948, 13488, 61952
00855	00037, 00437, 00500, 00819, 00850, 00915, 01025, 01200, 01208, 01252, 13488, 61952
00857	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00863, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00905, 00912, 00916, 01025, 01026, 01042, 01200, 01208, 01252, 04953, 13488, 28709, 61952
00858	00901, 00902, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01399, 04909, 04971, 05123, 05348
00860	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00861, 00863, 00865, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00937, 01025, 01027, 01041, 01042, 01043, 01200, 01208, 01252, 05123, 13488, 28709, 61952

<b>From CCSID</b>	<b>To CCSID</b>
00861	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00860, 00863, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 01149, 01200, 01208, 01252, 05123, 13488, 28709, 61952
00862	00037, 00424, 00500, 00819, 00850, 00916, 01200, 01208, 01252, 01255, 13488, 61952, 62211, 62235, 62245
00863	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00857, 00860, 00861, 00865, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00937, 01025, 01026, 01027, 01041, 01042, 01043, 01200, 01208, 01252, 05123, 13488, 28709, 61952
00864	00037, 00420, 00425, 00500, 00720, 00819, 00850, 01200, 01208, 01252, 08612, 09056, 13488, 61952, 62251
00865	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00860, 00863, 00871, 00937, 01200, 01208, 01252, 13488, 28709, 61952
00866	00037, 00437, 00500, 00819, 00850, 01025, 01200, 01208, 01252, 13488, 61952
00868	00918, 01200, 01208, 13488, 61952
00869	00037, 00423, 00500, 00737, 00813, 00819, 00838, 00850, 00852, 00870, 00874, 00875, 00880, 00897, 00903, 00912, 01025, 01027, 01041, 01042, 01043, 01200, 01208, 01252, 04971, 05123, 13488, 28709, 61952
00870	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00852, 00857, 00861, 00863, 00869, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00915, 00916, 00920, 00937, 01025, 01026, 01112, 01122, 01147, 01153, 01200, 01208, 01250, 01252, 01282, 04948, 04951, 09030, 13488, 28709, 61952
00871	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00870, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 05348, 09030, 13121, 13124, 13488, 28709, 61952
00874	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00813, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00875, 00880, 00897, 00903, 00912, 00916, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 01200, 01208, 05123, 13488, 28709, 61952
00875	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00880, 00897, 00903, 00912, 00916, 00920, 00937, 01025, 01026, 01027, 01041, 01043, 01112, 01122, 01200, 01208, 01252, 01253, 01280, 04965, 04971, 05123, 09030, 13488, 28709, 61952

<b>From CCSID</b>	<b>To CCSID</b>
00878	01200, 01208, 13488, 61952
00880	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00897, 00903, 00912, 00915, 00916, 00920, 00937, 01025, 01026, 01027, 01041, 01042, 01043, 01112, 01122, 01200, 01208, 01251, 01252, 01283, 04948, 04951, 05123, 09030, 13488, 28709, 61952
00891	00500, 00833, 01200, 01208, 13121, 13488, 61952
00897	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00903, 00912, 00916, 00920, 00930, 00939, 01025, 01026, 01027, 01042, 01043, 01200, 01208, 01252, 01399, 05026, 05035, 05123, 13488, 28709, 61952
00901	00037, 00500, 00858, 01140, 01148, 01156, 01200, 01208, 05348, 13488, 61952
00902	00037, 00500, 00858, 01140, 01148, 01157, 01200, 01208, 05348, 13488, 61952
00903	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00813, 00819, 00836, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00912, 00916, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 05123, 13124, 28709
00904	00037, 28709
00905	00037, 00256, 00500, 00737, 00775, 00819, 00850, 00857, 00920, 00937, 01026, 01112, 01122, 01200, 01208, 01252, 01254, 04953, 09030, 13488, 28709, 61952
00912	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00852, 00857, 00860, 00861, 00863, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00916, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 01153, 01200, 01208, 01252, 05123, 13488, 28709, 61952
00914	00037, 00437, 00500, 00819, 00850, 01200, 01208, 01252, 01257, 13488, 61952
00915	00037, 00437, 00500, 00819, 00850, 00855, 00870, 00880, 01025, 01200, 01208, 01252, 13488, 61952
00916	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00424, 00437, 00500, 00813, 00819, 00838, 00850, 00857, 00860, 00861, 00862, 00863, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00920, 01025, 01026, 01027, 01041, 01042, 01043, 01200, 01208, 01252, 01255, 05123, 13488, 28709, 61952, 62211, 62235, 62245
00918	00037, 00437, 00500, 00819, 00850, 00868, 01200, 01208, 01252, 13488, 61952
00920	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00813, 00819, 00838, 00850, 00860, 00861, 00863, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00905, 00912, 00916, 01025, 01026, 01155, 01200, 01208, 01252, 13488, 28709, 61952
00921	00037, 00437, 00500, 00819, 00850, 01112, 01200, 01208, 01252, 01257, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
00922	00037, 00437, 00500, 00819, 00850, 01122, 01200, 01208, 01252, 01257, 13488, 61952
00923	00924, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 04971, 13488, 61952
00924	00037, 00500, 00819, 00850, 00923, 01200, 01208, 13488, 61952
00926	00834, 01200, 01208, 13488, 61952
00927	00835, 01200, 01208, 13488, 61952
00928	00837, 01200, 01208, 13488, 61952
00930	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00500, 00819, 00833, 00836, 00850, 00871, 00895, 00897, 00932, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 00956, 00957, 00958, 00959, 01027, 01041, 01200, 01208, 01252, 01364, 01388, 01399, 04396, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 13121, 13124, 13488, 16684, 28709, 33722, 61952
00932	00290, 00930, 00939, 01027, 01200, 01208, 01399, 05026, 05035, 13488, 61952
00933	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00819, 00833, 00834, 00836, 00850, 00871, 00930, 00934, 00935, 00937, 00939, 00944, 00949, 00970, 01027, 01040, 01200, 01208, 01252, 01363, 01364, 01388, 01399, 05026, 05035, 05123, 13121, 13124, 13488, 17354, 25546, 28709, 61952
00934	00833, 00933, 01200, 01208, 01364, 13488, 61952
00935	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00819, 00833, 00836, 00837, 00850, 00871, 00930, 00933, 00936, 00937, 00939, 00946, 01027, 01042, 01200, 01208, 01252, 01364, 01381, 01383, 01386, 01388, 01399, 05026, 05035, 05123, 13121, 13124, 13488, 28709, 61952
00936	00836, 00935, 01388, 13124
00937	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00437, 00500, 00819, 00833, 00835, 00836, 00838, 00850, 00860, 00863, 00865, 00870, 00871, 00875, 00880, 00905, 00930, 00933, 00935, 00938, 00939, 00948, 00950, 00964, 00965, 01025, 01026, 01027, 01043, 01114, 01140, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 13121, 13124, 13488, 28709, 61952
00938	00937, 01200, 01208, 13488, 28709, 61952
00939	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00500, 00819, 00833, 00836, 00850, 00871, 00895, 00897, 00930, 00932, 00933, 00935, 00937, 00942, 00943, 00954, 00956, 00957, 00958, 00959, 01027, 01041, 01200, 01208, 01252, 01364, 01388, 01399, 04396, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 13121, 13124, 13488, 16684, 28709, 33722, 61952
00941	00300, 01200, 01208, 04396, 13488, 16684, 61952
00942	00290, 00930, 00939, 01027, 01041, 01200, 01208, 01399, 05026, 05035, 05123, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
00943	00290, 00930, 00939, 01027, 01200, 01208, 01399, 05026, 05035, 05123, 13488, 61952
00944	00833, 00933, 01040, 01200, 01208, 01364, 13121, 13488, 61952
00946	00836, 00935, 01042, 01200, 01208, 01388, 13124, 13488, 61952
00947	00835, 01200, 01208, 13488, 61952
00948	00037, 00937, 01043, 01200, 01208, 13488, 28709, 61952
00949	00833, 00933, 01200, 01208, 01364, 13121, 13488, 61952
00950	00937, 01200, 01208, 13488, 28709, 61952
00951	00834, 01200, 01208, 04930, 13488, 61952
00954	00290, 00930, 00939, 01027, 01200, 01208, 01399, 05026, 05035, 13488, 61952
00956	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
00957	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
00958	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
00959	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
00964	00937, 01200, 01208, 13488, 61952
00965	00937, 01200, 01208, 13488, 61952
00970	00933, 01200, 01208, 01364, 13488, 61952
00971	00834, 04930
01008	00420, 08612
01009	01200, 01208, 13488, 61952
01010	00500, 01200, 01208, 13488, 61952
01011	00500, 01200, 01208, 13488, 61952
01012	00500, 01200, 01208, 13488, 61952
01013	00500, 01200, 01208, 13488, 61952
01014	00500, 01200, 01208, 13488, 61952
01015	00500, 01200, 01208, 13488, 61952
01016	00500, 01200, 01208, 13488, 61952
01017	00500, 01200, 01208, 13488, 61952
01018	00500, 01200, 01208, 13488, 61952
01019	00500, 01200, 01208, 13488, 61952
01025	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00855, 00857, 00860, 00861, 00863, 00866, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00915, 00916, 00920, 00937, 01026, 01027, 01042, 01112, 01122, 01131, 01154, 01200, 01208, 01251, 01252, 01283, 04948, 04951, 05123, 09030, 13488, 28709, 61952



<b>From CCSID</b>	<b>To CCSID</b>
01026	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00838, 00850, 00857, 00861, 00863, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00905, 00912, 00916, 00920, 00937, 01025, 01042, 01112, 01122, 01155, 01200, 01208, 01252, 01254, 01281, 04953, 09030, 13488, 28709, 61952
01027	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00860, 00861, 00863, 00869, 00871, 00874, 00875, 00880, 00895, 00897, 00903, 00912, 00916, 00930, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 01025, 01041, 01042, 01112, 01122, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 09030, 13121, 13124, 13488, 28709, 61952
01040	00833, 00933, 00944, 01364, 13121
01041	00037, 00290, 00423, 00813, 00819, 00838, 00850, 00860, 00861, 00863, 00869, 00874, 00875, 00880, 00903, 00912, 00916, 00930, 00939, 00942, 01027, 01042, 01399, 05026, 05035, 05123
01042	00423, 00437, 00813, 00819, 00836, 00838, 00857, 00860, 00861, 00863, 00869, 00874, 00880, 00897, 00903, 00912, 00916, 00935, 00946, 01025, 01026, 01027, 01041, 01043, 01388, 05123, 13124
01043	00037, 00423, 00813, 00819, 00838, 00860, 00861, 00863, 00869, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00937, 00948, 01042, 28709
01046	00420, 00425, 01200, 01208, 08612, 13488, 61952, 62251
01051	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00871, 01200, 01208, 01252, 05348, 13488, 61952
01088	00037, 00500, 00819, 00833, 13121, 28709
01089	00037, 00420, 00425, 00500, 00819, 00850, 01200, 01208, 13488, 61952, 62251
01097	00037, 00437, 00500, 00737, 00775, 00819, 00850, 01098, 01112, 01122, 01200, 01208, 01252, 01256, 09030, 13488, 61952
01098	00037, 00437, 00500, 00819, 00850, 01097, 01200, 01208, 01252, 13488, 61952
01112	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00500, 00775, 00819, 00833, 00836, 00838, 00850, 00870, 00871, 00875, 00880, 00905, 00921, 01025, 01026, 01027, 01097, 01122, 01156, 01200, 01208, 01252, 01257, 05123, 09030, 13121, 13124, 13488, 28709, 61952
01114	00037, 00500, 00819, 00836, 00937, 13124, 28709
01115	00037, 00500, 00836, 13124
01122	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00500, 00775, 00819, 00833, 00836, 00838, 00850, 00870, 00871, 00875, 00880, 00905, 00922, 01025, 01026, 01027, 01097, 01112, 01157, 01200, 01208, 01252, 01257, 05123, 09030, 13121, 13124, 13488, 28709, 61952
01123	00037, 00500, 01124, 01125, 01158, 01200, 01208, 01251, 13488, 61952
01124	00037, 00500, 01123, 01200, 01208, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
01125	01123, 01200, 01208, 13488, 61952
01126	00037, 00500, 00819, 00833, 01252, 13121
01129	01200, 01208, 13488, 61952
01130	00037, 00500, 00819, 00850, 01164, 01200, 01208, 01252, 01258, 13488, 61952
01131	00037, 00500, 01025, 01200, 01208, 13488, 61952
01132	00037, 00500, 00819, 00850, 01133, 01200, 01208, 01252, 13488, 61952
01133	01132
01137	00037, 00500, 00819, 01200, 01208, 01252, 13488, 61952
01140	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00425, 00437, 00500, 00819, 00850, 00858, 00871, 00901, 00902, 00923, 00937, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952, 62251
01141	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01142	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01141, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01143	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01141, 01142, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01144	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01141, 01142, 01143, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01145	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01141, 01142, 01143, 01144, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01146	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00871, 00923, 01140, 01141, 01142, 01143, 01144, 01145, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
01147	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00870, 00871, 00923, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01148	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00425, 00437, 00500, 00819, 00850, 00858, 00871, 00901, 00902, 00923, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952, 62251
01149	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00437, 00500, 00819, 00850, 00858, 00861, 00871, 00923, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01252, 01399, 04909, 04971, 05123, 05348, 13488, 61952
01153	00037, 00500, 00819, 00850, 00852, 00870, 00912, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01250, 04971, 13488, 61952
01154	00037, 00500, 00819, 01025, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01251, 04971, 05123, 13488, 61952
01155	00037, 00500, 00819, 00920, 01026, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01254, 04971, 13488, 61952
01156	00037, 00500, 00819, 00901, 01112, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01257, 04971, 05123, 13488, 61952
01157	00037, 00500, 00819, 00902, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01257, 04971, 05123, 13488, 61952
01158	00037, 00500, 00819, 01123, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01251, 13488, 61952
01160	00037, 00500, 00819, 00838, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 05123, 13488, 61952
01164	00037, 00500, 00819, 01130, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01258, 13488, 61952

From CCSID	To CCSID
01200	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00301, 00367, 00420, 00423, 00424, 00425, 00437, 00500, 00720, 00737, 00775, 00813, 00819, 00833, 00834, 00835, 00836, 00837, 00838, 00850, 00851, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00868, 00869, 00870, 00871, 00874, 00875, 00878, 00880, 00891, 00897, 00901, 00902, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00923, 00924, 00926, 00927, 00928, 00930, 00932, 00933, 00934, 00935, 00937, 00938, 00939, 00941, 00942, 00943, 00944, 00946, 00947, 00948, 00949, 00950, 00951, 00954, 00956, 00957, 00958, 00959, 00964, 00965, 00970, 01009, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01025, 01026, 01027, 01046, 01051, 01089, 01097, 01098, 01112, 01122, 01123, 01124, 01125, 01129, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01208, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01362, 01363, 01364, 01380, 01381, 01383, 01385, 01386, 01388, 01392, 01399, 04396, 04909, 04930, 04933, 04948, 04951, 04952, 04960, 04971, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 05348, 08612, 09030, 09056, 09066, 12708, 13121, 13124, 13488, 16684, 17354, 28709, 33722, 61952, 62211, 62224, 62235, 62245, 62248, 62251
01208	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00301, 00367, 00420, 00423, 00424, 00425, 00437, 00500, 00720, 00737, 00775, 00813, 00819, 00833, 00834, 00835, 00836, 00837, 00838, 00850, 00851, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00868, 00869, 00870, 00871, 00874, 00875, 00878, 00880, 00891, 00897, 00901, 00902, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00923, 00924, 00926, 00927, 00928, 00930, 00932, 00933, 00934, 00935, 00937, 00938, 00939, 00941, 00942, 00943, 00944, 00946, 00947, 00948, 00949, 00950, 00951, 00954, 00956, 00957, 00958, 00959, 00964, 00965, 00970, 01009, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01025, 01026, 01027, 01046, 01051, 01089, 01097, 01098, 01112, 01122, 01123, 01124, 01125, 01129, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01362, 01363, 01364, 01380, 01381, 01383, 01385, 01386, 01388, 01392, 01399, 04396, 04909, 04930, 04933, 04948, 04951, 04952, 04960, 04971, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 05348, 08612, 09030, 09056, 09066, 12708, 13121, 13124, 13488, 16684, 17354, 28709, 33722, 62211, 62224, 62235, 62245, 62248, 62251
01250	00037, 00273, 00500, 00819, 00850, 00870, 01153, 01200, 01208, 01252, 01282, 13488, 61952
01251	00037, 00500, 00819, 00850, 00880, 01025, 01123, 01154, 01158, 01200, 01208, 01252, 01283, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
01252	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00425, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00869, 00870, 00871, 00875, 00880, 00897, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00930, 00933, 00935, 00937, 00939, 01025, 01026, 01027, 01051, 01097, 01098, 01112, 01122, 01126, 01130, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01250, 01251, 01254, 01255, 01256, 01257, 01275, 01280, 01281, 01282, 01283, 01364, 01388, 01399, 04971, 05026, 05035, 05123, 05348, 13121, 13124, 13488, 28709, 61952, 62211, 62235, 62245, 62251
01253	00037, 00423, 00737, 00819, 00875, 01200, 01208, 01280, 04971, 13488, 61952
01254	00037, 00500, 00819, 00850, 00905, 01026, 01155, 01200, 01208, 01252, 13488, 61952
01255	00037, 00424, 00500, 00819, 00850, 00862, 00916, 01200, 01208, 01252, 01281, 13488, 61952, 62211, 62235, 62245
01256	00037, 00420, 00425, 00500, 00720, 00819, 00850, 01097, 01200, 01208, 01252, 08612, 12708, 13488, 61952, 62224, 62251
01257	00037, 00437, 00500, 00775, 00819, 00850, 00914, 00921, 00922, 01112, 01122, 01156, 01157, 01200, 01208, 01252, 13488, 61952
01258	00037, 00500, 00819, 01130, 01164, 01200, 01208, 13488, 61952
01275	00037, 00437, 00500, 00819, 00850, 01200, 01208, 01252, 05348, 13488, 61952
01280	00037, 00437, 00500, 00819, 00850, 00875, 01200, 01208, 01252, 01253, 13488, 61952
01281	00037, 00437, 00500, 00819, 00850, 01026, 01200, 01208, 01252, 01255, 13488, 61952
01282	00037, 00437, 00500, 00819, 00850, 00870, 01200, 01208, 01250, 01252, 13488, 61952
01283	00037, 00437, 00500, 00819, 00850, 00880, 01025, 01200, 01208, 01251, 01252, 13488, 61952
01362	00834, 01200, 01208, 04930, 13488, 61952
01363	00833, 00933, 01200, 01208, 01364, 13488, 61952
01364	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00819, 00833, 00834, 00836, 00850, 00871, 00930, 00933, 00934, 00935, 00937, 00939, 00944, 00949, 00970, 01027, 01040, 01200, 01208, 01252, 01363, 01388, 01399, 05026, 05035, 05123, 13121, 13124, 13488, 28709, 61952
01380	00837, 01200, 01208, 04933, 13488, 61952
01381	00836, 00935, 01200, 01208, 01388, 13488, 61952
01382	00837, 04933
01383	00935, 01200, 01208, 01388, 13488, 61952
01385	00837, 01200, 01208, 04933, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
01386	00836, 00935, 01200, 01208, 01388, 13124, 13488, 61952
01388	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00819, 00833, 00836, 00837, 00850, 00871, 00930, 00933, 00935, 00936, 00937, 00939, 00946, 01027, 01042, 01200, 01208, 01252, 01364, 01381, 01383, 01386, 01399, 05026, 05035, 05123, 13121, 13124, 13488, 28709, 61952
01392	01200, 01208, 13488
01399	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00819, 00833, 00836, 00850, 00858, 00871, 00895, 00897, 00930, 00932, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 01027, 01041, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01364, 01388, 05026, 05035, 05050, 05123, 13121, 13124, 13488, 16684, 28709, 33722, 61952
04396	00300, 00301, 00930, 00939, 00941, 01200, 01208, 05026, 05035, 13488, 16684, 61952
04909	00037, 00500, 00813, 00858, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 04971, 13488, 61952
04930	00834, 00951, 00971, 01200, 01208, 01362, 13488, 61952
04933	01200, 01208, 01380, 01382, 01385, 13488, 61952
04948	00852, 00870, 00880, 01025, 01200, 01208, 13488, 61952
04951	00870, 00880, 01025, 01200, 01208, 13488, 61952
04952	00424, 01200, 01208, 13488, 61952
04953	00857, 00905, 01026
04960	00420, 01200, 01208, 08612, 13488, 61952
04965	00423, 00875
04970	00037, 00500, 00838
04971	00037, 00423, 00437, 00500, 00737, 00813, 00819, 00850, 00858, 00869, 00875, 00923, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01200, 01208, 01252, 01253, 04909, 05348, 13488, 61952
05026	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00500, 00819, 00833, 00836, 00850, 00871, 00895, 00897, 00930, 00932, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 00956, 00957, 00958, 00959, 01027, 01041, 01200, 01208, 01252, 01364, 01388, 01399, 04396, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 13121, 13124, 13488, 16684, 28709, 33722, 61952
05035	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00500, 00819, 00833, 00836, 00850, 00871, 00895, 00897, 00930, 00932, 00933, 00935, 00937, 00939, 00942, 00943, 00954, 00956, 00957, 00958, 00959, 01027, 01041, 01200, 01208, 01252, 01364, 01388, 01399, 04396, 05026, 05050, 05052, 05053, 05054, 05055, 05123, 13121, 13124, 13488, 16684, 28709, 33722, 61952
05050	00930, 00939, 01200, 01208, 01399, 05026, 05035, 13488, 61952
05052	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952

<b>From CCSID</b>	<b>To CCSID</b>
05053	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
05054	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
05055	00930, 00939, 01200, 01208, 05026, 05035, 13488, 61952
05123	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00423, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00858, 00860, 00861, 00863, 00869, 00871, 00874, 00875, 00880, 00897, 00903, 00912, 00916, 00930, 00933, 00935, 00937, 00939, 00942, 00943, 01025, 01027, 01041, 01042, 01112, 01122, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01154, 01156, 01157, 01160, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05348, 09030, 13121, 13124, 13488, 28709, 61952
05210	13124
05348	00037, 00273, 00277, 00278, 00280, 00284, 00285, 00437, 00500, 00819, 00850, 00858, 00871, 00901, 00902, 01051, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01200, 01208, 01252, 01275, 04971, 05123, 13488, 61952
08612	00037, 00256, 00420, 00425, 00500, 00850, 00864, 01008, 01046, 01200, 01208, 01256, 04960, 12708, 13488, 28709, 61952, 62224, 62251
09030	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00500, 00819, 00833, 00836, 00838, 00850, 00870, 00871, 00875, 00880, 00905, 01025, 01026, 01027, 01097, 01112, 01122, 01200, 01208, 05123, 09066, 13121, 13488, 28709, 61952
09056	00420, 00850, 00864, 01200, 01208, 13488, 61952
09066	01200, 01208, 09030, 13488, 61952
12708	00037, 00420, 00425, 00500, 01200, 01208, 01256, 08612, 13488, 61952, 62224, 62251
13121	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00367, 00437, 00500, 00737, 00775, 00819, 00833, 00836, 00850, 00871, 00891, 00930, 00933, 00935, 00937, 00939, 00944, 00949, 01027, 01040, 01088, 01112, 01122, 01126, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 09030, 13124, 13488, 28709, 61952
13124	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00367, 00437, 00500, 00737, 00775, 00819, 00833, 00836, 00850, 00871, 00903, 00930, 00933, 00935, 00937, 00939, 00946, 01027, 01042, 01112, 01114, 01115, 01122, 01200, 01208, 01252, 01364, 01386, 01388, 01399, 05026, 05035, 05123, 05210, 13121, 13488, 28709, 61952

<b>From CCSID</b>	<b>To CCSID</b>
13488	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00301, 00367, 00420, 00423, 00424, 00425, 00437, 00500, 00720, 00737, 00775, 00813, 00819, 00833, 00834, 00835, 00836, 00837, 00838, 00850, 00851, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00868, 00869, 00870, 00871, 00874, 00875, 00878, 00880, 00891, 00897, 00901, 00902, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00923, 00924, 00926, 00927, 00928, 00930, 00932, 00933, 00934, 00935, 00937, 00938, 00939, 00941, 00942, 00943, 00944, 00946, 00947, 00948, 00949, 00950, 00951, 00954, 00956, 00957, 00958, 00959, 00964, 00965, 00970, 01009, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01025, 01026, 01027, 01046, 01051, 01089, 01097, 01098, 01112, 01122, 01123, 01124, 01125, 01129, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01208, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01362, 01363, 01364, 01380, 01381, 01383, 01385, 01386, 01388, 01392, 01399, 04396, 04909, 04930, 04933, 04948, 04951, 04952, 04960, 04971, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 05348, 08612, 09030, 09056, 09066, 12708, 13121, 13124, 16684, 17354, 28709, 33722, 61952, 62211, 62224, 62235, 62245, 62248, 62251
16684	00301, 00930, 00939, 00941, 01200, 01208, 01399, 04396, 05026, 05035, 13488, 61952
17354	00933, 01200, 01208, 13488, 61952
25546	00933
28709	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00420, 00423, 00424, 00437, 00500, 00737, 00775, 00813, 00819, 00833, 00836, 00838, 00850, 00857, 00860, 00861, 00863, 00865, 00869, 00870, 00871, 00874, 00875, 00880, 00897, 00903, 00904, 00905, 00912, 00916, 00920, 00930, 00933, 00935, 00937, 00939, 00948, 00950, 01025, 01026, 01027, 01043, 01088, 01112, 01114, 01122, 01200, 01208, 01252, 01364, 01388, 01399, 05026, 05035, 05123, 08612, 09030, 13121, 13124, 13488, 61952
33722	00930, 00939, 01200, 01208, 01399, 05026, 05035, 13488, 61952
57345	00930, 00939, 05026, 05035
57777	01208, 13488



<b>From CCSID</b>	<b>To CCSID</b>
61952	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00300, 00301, 00367, 00420, 00423, 00424, 00425, 00437, 00500, 00720, 00737, 00775, 00813, 00819, 00833, 00834, 00835, 00836, 00837, 00838, 00850, 00851, 00852, 00855, 00857, 00860, 00861, 00862, 00863, 00864, 00865, 00866, 00868, 00869, 00870, 00871, 00874, 00875, 00878, 00880, 00891, 00897, 00901, 00902, 00905, 00912, 00914, 00915, 00916, 00918, 00920, 00921, 00922, 00923, 00924, 00926, 00927, 00928, 00930, 00932, 00933, 00934, 00935, 00937, 00938, 00939, 00941, 00942, 00943, 00944, 00946, 00947, 00948, 00949, 00950, 00951, 00954, 00956, 00957, 00958, 00959, 00964, 00965, 00970, 01009, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01025, 01026, 01027, 01046, 01051, 01089, 01097, 01098, 01112, 01122, 01123, 01124, 01125, 01129, 01130, 01131, 01132, 01137, 01140, 01141, 01142, 01143, 01144, 01145, 01146, 01147, 01148, 01149, 01153, 01154, 01155, 01156, 01157, 01158, 01160, 01164, 01200, 01250, 01251, 01252, 01253, 01254, 01255, 01256, 01257, 01258, 01275, 01280, 01281, 01282, 01283, 01362, 01363, 01364, 01380, 01381, 01383, 01385, 01386, 01388, 01399, 04396, 04909, 04930, 04933, 04948, 04951, 04952, 04960, 04971, 05026, 05035, 05050, 05052, 05053, 05054, 05055, 05123, 05348, 08612, 09030, 09056, 09066, 12708, 13121, 13124, 13488, 16684, 17354, 28709, 33722, 62211, 62224, 62235, 62245, 62248, 62251
62209	62211, 62235, 62245
62210	00424, 62211, 62235, 62245
62211	00037, 00424, 00500, 00819, 00850, 00862, 00916, 01200, 01208, 01252, 01255, 13488, 61952, 62209, 62210, 62213, 62215, 62222, 62223, 62235, 62245
62213	62211
62215	00424, 62211, 62235, 62245
62218	00420
62221	62235
62222	00424, 62211, 62235, 62245
62223	00424, 62211, 62235, 62245
62224	00037, 00420, 00425, 00500, 01200, 01208, 01256, 08612, 12708, 13488, 61952, 62251
62228	00420, 00425, 62251
62235	00037, 00424, 00500, 00819, 00850, 00862, 00916, 01200, 01208, 01252, 01255, 13488, 61952, 62209, 62210, 62211, 62215, 62221, 62222, 62223, 62245
62238	00424, 62245
62239	00424, 62245
62245	00037, 00424, 00500, 00819, 00850, 00862, 00916, 01200, 01208, 01252, 01255, 13488, 61952, 62209, 62210, 62211, 62215, 62222, 62223, 62235, 62238, 62239
62248	00420, 01200, 01208, 13488, 61952

From CCSID	To CCSID
62251	00037, 00420, 00500, 00819, 00864, 01046, 01089, 01140, 01148, 01200, 01208, 01252, 01256, 08612, 12708, 13488, 61952, 62224, 62228

### Associated CCSIDs

This table shows the associated CCSIDs for a given CCSID value and encoding scheme.

Although not represented in the table, if you request an encoding scheme that is the same as the input CCSID, you will get back the input CCSID. For example, if you request the associated CCSID of CCSID 00037 with an encoding scheme of 01100, you will get back 00037.

Input CCSID	1100	1200	1301	2100	2200	2300	4100	4105	4403	5100	5404
00037		00835	00937	00437	00947	00950	00819	01252		00367	
00256				00437			00819	01252		00367	
00273				00850			00819	01252			
00277				00865			00819	01252			
00278				00865			00819	01252			
00280				00850			00819	01252			
00284				00850			00819	01252			
00285				00850			00819	01252			
00290		04396	05026	01041	00301	00942					
00297				00850			00819	01252			
00300	00290		00930	01041	00301	00942					
00301	00290	04396	05026	01041		00942					
00367	00500			00850			00819	01252			
00420				00864			01089	01256			
00423				00869			00813	01253			
00424				00862			00916	01255			
00437	00037						00819	01252			
00500				00850			00819	01252		00367	
00720	00420						01089	01256			
00737	00875						00813	01253			
00775	01112						00921	01257			
00813	00875							01253			
00819	00500			00850				01252		00367	
00833		00834	00933	01088	00951	00949					
00834	00833		00933	01088	00951	00949					
00835	28709		00937	01043	00927	00950					
00836		00837	00935	01115	01380	01381					
00837	00836		00935	01115	01380	01381					

<b>Input CCSID</b>	<b>1100</b>	<b>1200</b>	<b>1301</b>	<b>2100</b>	<b>2200</b>	<b>2300</b>	<b>4100</b>	<b>4105</b>	<b>4403</b>	<b>5100</b>	<b>5404</b>
00838				00874							
00850	00500						00819	01252			
00851	00875						00813	01253			
00852	00870						00912	01250			
00855	01025							01251			
00857	01026						00920	01254			
00860	00037						00819	01252			
00861	00871						00819	01252			
00862	00424						00916	01255			
00863	00500						00819	01252			
00864	00420						01089	01256			
00865	00277						00819	01252			
00866	01025			00866				01251			
00869	00875						00813	01253			
00870				00852			00912	01250			
00871				00861			00819	01252			
00874	00838										
00875				00869			00813	01253			
00880				00855							
00891	00833	00834	00933		00926	00934					
00897	00290	04396	05026		00301	00932					
00903	00836	00837	00935		00928	00936					
00904	28709	00835	00937		00927	00938					
00905				00857			00920				
00912	00870			00852				01250			
00915	01025			00855				01251			
00916	00424			00862				01255			
00920	01026			00857				01254			
00921	01112			01125				01257			
00922	01122			01125				01251			
00923	00924			00850				01252		00367	
00924				00850			00923	01252		00367	
00926	00833	00834	00933	01040		00944					
00927	28709	00835	00937	01043		00950					
00928	00836	00837	00935	01042		00946					

<b>Input CCSID</b>	<b>1100</b>	<b>1200</b>	<b>1301</b>	<b>2100</b>	<b>2200</b>	<b>2300</b>	<b>4100</b>	<b>4105</b>	<b>4403</b>	<b>5100</b>	<b>5404</b>
00930	00290	00300		01041	00301	00942					05052
00932	00290	04396	05026	00897	00301						
00933	00833	00834		01088	00951	00949			00970		17354
00934	00833	00834	00933	00891	00926						
00935	00836	00837		01115	01380	01381					
00936	00836	00837	00935	00903	00928						
00937	28709	00835		01043	00927	00950			00964		00965
00938	28709	00835	00937	00904	00927						
00939	01027	00300		01041	00301	00942					05052
00942	00290	04396	05026	01041	00301						
00943	00290	04396	05026	00897	00301						
00944	00833	00834	00933	01040	00926						
00946	00836	00837	00935	01042	00928						
00947		00835	00937								
00948	28709	00835	00937	01043	00927						
00949	00833	00834	00933	01088	00951				00970		17354
00950	28709	00835	00937	01114	00947						
00951	00833	00834	00933	01088		00949			00970		17354
00956	00290	00300	05026								
00957	00290	00300	05026								
00958	00290	00300	05026								
00959	00290	00300	05026								
00964	00037	00835	00937								
00965	00037	00835	00937								
00970	13121	04930	01364								
01008	00420										
01009	00500										
01010	00500										
01011	00500										
01012	00500										
01013	00500										
01014	00500										
01015	00500										
01016	00500										
01017	00500										

<b>Input CCSID</b>	<b>1100</b>	<b>1200</b>	<b>1301</b>	<b>2100</b>	<b>2200</b>	<b>2300</b>	<b>4100</b>	<b>4105</b>	<b>4403</b>	<b>5100</b>	<b>5404</b>
01018	00500										
01019	00500										
01025				00855				01251			
01026				00857			00920	01254			
01027		04396	05035	01041	00301	00942					
01040	00833	00834	00933		00926	00944					
01041	00290	04396	05026		00301	00942					
01042	00836	00837	00935		00928	00946					
01043	28709	00835	00937		00927	00950					
01046	00420										
01051	00037			00850							
01088	00833	00834	00933		00951	00949			00970		17354
01089	00420							01256			
01097				01098							
01098	01097										
01112								01257			
01114	28709										
01115	00836	00837	00935		01380	01381					
01122				01125			01124	01257			
01123				01125			01124	01251			
01126	13121	04930	01364		01362	01363			00970		17354
01129	01130							01258			
01130							01129	01258			
01132							01133				
01133	01132										
01140				00437			00923	01252		00367	
01141				00850			00923	01252		00367	
01142				00865			00923	01252		00367	
01143				00865			00923	01252		00367	
01144				00850			00923	01252		00367	
01145				00850			00923	01252		00367	
01146				00850			00923	01252		00367	
01147				00850			00923	01252		00367	
01148				00850			00923	01252		00367	
01149				00861			00923	01252		00367	

Input CCSID	1100	1200	1301	2100	2200	2300	4100	4105	4403	5100	5404
01153				00852			00912	01250			
01154				00855				01251			
01155				00857			00920	01254			
01156								01257			
01157				01125			01124	01257			
01158				01125			01124	01251			
01160				00874							
01164							01129	01258			
01250	00870			00852							
01251	01025			00855							
01252	00500			00850			00819			00367	
01253	00875			00869							
01254	01026			00857							
01255	00424			00862							
01256	00420			00864							
01257	01112						00921				
01258	01130			01258			01129				
01275	00037			00850							
01280	00875			00869							
01281	01026			00857							
01282	00870			00852							
01283	01025			00855							
01362	13121	04930	01364	01126		01363			00970		17354
01363	13121	04930	01364	01126	01362				00970		17354
01364	13121	04930		01126	01362	01363			00970		17354
01380	00836	00837	00935	01115		01381					
01381	00836	00837	00935	01115	01380						
01386	13124	04933	01388								
01388	13124	04933		01114	01385	01386					
01399	05123	16684		01041	00301	00942			05050		05052
04396	00290		05026	01041	00301	00942					
04930	13121		01364	01126	01362	01363					
04933	13124		01388	01114	01385	01386					
04948	00870										
04951	01025										

<b>Input CCSID</b>	<b>1100</b>	<b>1200</b>	<b>1301</b>	<b>2100</b>	<b>2200</b>	<b>2300</b>	<b>4100</b>	<b>4105</b>	<b>4403</b>	<b>5100</b>	<b>5404</b>
04952	00424										
04953	01026										
04960	00420										
04965	00875										
05026	00290	04396		01041	00301	00942			05050		05052
05035	01027	04396		01041	00301	00942					05052
05050	00290	00300	05026	01041	00301	00942					
05052	00290	00300	05026	01041	00301	00942					
05053	00290	00300	05026	01041	00301	00942					
00000											
05054	00290	00300	05026	01041	00301	00942					
00000											
05055	00290	00300	05026	01041	00301	00942					
00000											
05123		16684	01399	01041	00301	00942					
00000											
09030				09066							
09066	09030										
13121		04930	01364	01126	01362	01363					
13124		04933	01388	01114	01385	01386					
00000											
16684	05123		01399	01041	00301	00942					
00000											
17354	00833	00834	00933								
25546	00833	00834	00933	01088	00951	00949					
28709		00835	00937	01043	00927	00948					
33722	00290	04366	05026								
57345	00290	00300	05026								

### Encoding schemes for the CCSIDs

This table shows the encoding scheme values (from Character Data Representation Architecture) that are used for the CCSIDs.

Encoding scheme identifier (ESID) hexadecimal	Interpretation
1100	EBCDIC, single-byte, No code extension is allowed. Number of States = 1.
1200	EBCDIC, double-byte, No code extension is allowed. Number of States = 1.
1300	EBCDIC, mixed-byte, No code extension is allowed Number of States = 1.
1301	EBCDIC, mixed single-byte and double-byte, using shift-in (SI) and shift-out (SO) code extension method Number of States = 2.
2100	IBM-PC Data, single-byte, No code extension is allowed Number of States = 1.
2200	IBM-PC Data, double-byte, No code extension is allowed Number of States = 1.
2300	IBM-PC Data, mixed single-byte and double-byte, with implicit code extension Number of States = 2.
3100	IBM-PC Display, single-byte, No code extension is allowed Number of States = 1.
3200	IBM-PC Display, double-byte, No code extension is allowed Number of States = 1.
3300	IBM-PC Display, mixed single-byte and double-byte, with implicit code extension Number of States = 2.
4100	ISO 8, single-byte, No code extension is allowed Number of States = 1.
4105	ISO 8 (ASCII code), single-byte, Graphics in C1 Note that graphic characters may be present in the area normally reserved for the C1 control codes. (ie X'80' to X'9F') Number of States = 1.
4403	IBM EUC Number of States = 2 to 4
5100	ISO 7 (ASCII code), single-byte, No code extension is allowed Number of States = 1.
5404	ISO 2022 TCP/IP Number of States = 2 to 4.
7200	UCS-2, No code extension is allowed Number of States = 1.
7808	UTF-8, No code extension is allowed Number of States = 3.

### Language identifiers and associated default CCSIDs

This table shows the language identifiers and the job default CCSID (DFTCCSID) values associated with those identifiers.

If the QTQ\_DEFAULT\_CCSID system level environment variable is not defined, then the defaults act as indicated in the following table.

If the QTQ\_DEFAULT\_CCSID system level environment variable is defined, then it consists of pairs of valid language ID and single-byte or mixed-byte EBCDIC CCSID values. You can use this environment variable to change to the euro value. If this value is not properly defined, the system issues an informational message and uses the value in the following table.

For example, the following command specifies that you want the default CCSID to be 500 for the ENU and ENP language identifiers:

```
ADDENVVAR ENVVAR(QTQ_DEFAULT_CCSID) VALUE('ENU 00500 ENP 00500') LEVEL(*SYS)
```



You should be careful to ensure that this environment is set to a valid value. If this value is not set correctly, you might not be able to log on to the system. Before you change this value, you might consider having two signed-on sessions. Then, you can test your change on one of the sessions and make changes on the other if needed.

The Job default coded character set identifier topic includes information about the QTQ\_DEFAULT\_CCSD system-level environment variable and how it is used to find the default value.

**Note:** the following table shows the original default for this language, however for some languages there is a newer CCSID you may wish to use to get the most current support.

Language ID	Language	Newer CCSID
TRK	Turkish	1175
LVB	Latvia	1156
CHT	Traditional Chinese	1377
CHS	Simplified Chinese	1388
JPN	Japanese	1399

Language identifier	Single-byte CCSID	Mixed-byte CCSID	Language
AFR	00037	None	Afrikaans
ARA	00420	None	Arabic
BEL	01025	None	Byelorussian
BGR	01025	None	Bulgarian
CAT	00284	None	Catalan
CHS	00836	00935	Simplified Chinese
CHT	00037	00937	Traditional Chinese
CSY	00870	None	Czech
DAN	00277	None	Danish
DES	00500	None	Swiss German
DEU	00273 (1141 with Euro)	None	German
ELL	00875 (4971 with Euro)	None	Greek
ENA	00037	None	Australian English
ENG	00285	None	United Kingdom English
ENP	00037	None	Uppercase English
ENU	00037	None	United States English
ESP	00284 (1145 with Euro)	None	Spanish
FAR	01097	None	Farsi
FIN	00278 (1143 with Euro)	None	Finnish
FRA	00297 (1147 with Euro)	None	French
FRB	00500 (1148 with Euro)	None	Belgian French
FRC	00500	None	Canadian French
FRS	00500	None	Swiss French

Language identifier	Single-byte CCSID	Mixed-byte CCSID	Language
GAE	00285 (1146 with Euro)	None	Irish Gaelic
HEB	00424	None	Hebrew
HRV	00870	None	Croatian
HUN	00870	None	Hungarian
ISL	00871 (1149 with Euro)	None	Icelandic
ITA	00280 (1144 with Euro)	None	Italian
ITS	00500	None	Swiss Italian
JPN	00290	05026	Japanese Katakana  The mixed value for DFTCCSID is 05026 when the job CCSID is 65535. In order for the DFTCCSID to be 05035, the job CCSID must be 05035.
KOR	00833	00933	Korean
MKD	01025	None	Macedonian
NLB	00500 (1148 with Euro)	None	Belgian Dutch
NLD	00037 (1140 with Euro)	None	Dutch
NON	00277	None	Norwegian Nynorsk
NOR	00277	None	Norwegian Bokmal
PLK	00870	None	Polish
PTB	00037	None	Brazilian Portuguese
PTG	00037 (1140 with Euro)	None	Portuguese
ROM	00870	None	Romanian
RUS	01025	None	Russian
SKY	00870	None	Slovakian
SLO	00870	None	Slovenian
SQI	00500	None	Albanian
SRB	01025	None	Serbian Cyrillic
SRL	00870	None	Serbian Latin
SVE	00278	None	Swedish
THA	00838	None	Thai
TRK	01026	None	Turkish

### Related concepts

[Recommendations and guidelines for using CCSIDs](#)

These recommendations are useful when you write globalized applications.

[Database management](#)

Database management support provides default coded character set identifier (CCSID) values for database files on the system. All database files are assigned a CCSID. At file creation time, the CCSID is either explicitly assigned through DDS, SQL, or IDDU, or implicitly assigned the job default CCSID (DFTCCSID).

#### LC\_COLLATE category

The LC\_COLLATE category defines character or string collation information. Within LC\_COLLATE you can specify a sort sequence to use using the `cpysyscol` keyword. The `cpysyscol` keyword value is used in place of the LC\_COLLATE category definitions.

#### **Related tasks**

##### Job default coded character set identifier

A job attribute, job default CCSID (DFTCCSID), is created for jobs with a CCSID of 65535. The DFTCCSID value is used by a system code when a CCSID other than 65535 is needed.

## **Locale reference information**

A *locale* is an identifier for a set of conventions governing the input, processing, and display of computerized data so that they match the requirements and expectations of a particular user community. This information covers locales on the IBM i operating system.

#### **System-supplied locales and recommended CCSIDs**

The system-supplied locale source definition file members are in the optionally installable library QSYSLOCALE in the QLOCALESRC source file. The source file members are encoded in CCSID 37 and are read only.

The following table shows the locale source file definition members that are shipped with IBM i, the recommended CCSID for creating each locale, and how the locale is supplied (as source, object, or both).

The source definition file members themselves cannot be changed. They can be copied and then edited, if needed.

#### **Notes:**

- Source is changed in V5R4 to support euro values.
- The Japanese 5035 is shipped as an object but uses the same JA\_JP source member as the Japanese.

<b>Description</b>	<b>Member</b>	<b>CCSID</b>	<b>How shipped</b>
Albania/Albanian	SQ_AL	500	Both
Algeria/Arabic	AR_DZ	420	Source
Arabic-speaking countries/Arabic	AR_AA	420	Both
Argentina/Spanish	ES_AR	284	Source
Australia/English	EN_AU	37	Source
Austria/German	DE_AT	1141	Source
Austria/German euro	DE_AT_E	1141	Source
Austria/German preeuro	DE_AT_PE	273	Source
Bahrain/Arabic	AR_BH	420	Source
Belgium/Dutch	NL_BE	1148	Both
Belgium/Dutch euro	NL_BE_E	1148	Both
Belgium/Dutch preeuro	NL_BE_PE	500	Both
Belgium/English	EN_BE	1148	Both
Belgium/English euro	EN_BE_E	1148	Source

<b>Description</b>	<b>Member</b>	<b>CCSID</b>	<b>How shipped</b>
Belgium/English preeuro	EN_BE_PE	37	Source
Belgium/French	FR_BE	1148	Both
Belgium/French euro	FR_BE_E	1148	Both
Belgium/French preeuro	FR_BE_PE	500	Both
Bolivia/Spanish	ES_BO	284	Source
Brazil/Portuguese	PT_BR	37	Both
Bulgaria/Bulgarian	BG_BG	1025	Both
Bulgaria/Bulgarian-Lotus	BG_BG_L	1025	Both
Byelorussia/Byelorussian	BE_BY	1025	Source
Canada/English	EN_CA	37	Source
Canada/French	FR_CA	500	Both
Chile/Spanish	ES_CL	284	Source
China/Simplified Chinese	ZH_CN	1388	Both
China/Simplified Chinese (old)	ZH_CN_GBK	935	Source
China (Hong Kong S.A.R.)/English	EN_HK	37	Source
China (Hong Kong S.A.R.)/Simplified Chinese	ZH_HK_S	1388	Source
China (Hong Kong S.A.R.)/Traditional Chinese	ZH_HK_T	937	Source
Columbia/Spanish	ES_CO	284	Source
Costa Rica/Spanish	ES_CR	284	Source
Croatia/Croatian	HR_HR	870	Both
Czech Republic/Czech	CS_CZ	870	Both
Denmark/Danish	DA_DK	1142	Both
Denmark/Danish euro	DA_DK_E	1142	Source
Denmark/Danish preeuro	DA_DK_PE	277	Source
Dominican Republic/Spanish	ES_DO	284	Source
Ecuador/Spanish	ES_EC	284	Source
Egypt/Arabic	AR_EG	420	Source
El Salvador/Spanish	ES_SV	284	Source

<b>Description</b>	<b>Member</b>	<b>CCSID</b>	<b>How shipped</b>
Estonia/Estonian	ET_EE	1122	Both
Finland/Finnish	FI_FI	1143	Both
Finland/Finnish euro	FI_FI_E	1143	Both
Finland/Finnish preeuro	FI_FI_PE	278	Both
France/French	FR_FR	1147	Both
France/French euro	FR_FR_E	1147	Both
France/French preeuro	FR_FR_PE	297	Both
Germany/German	DE_DE	1141	Both
Germany/German euro	DE_DE_E	1141	Both
Germany/German preeuro	DE_DE_PE	273	Both
Great Britain/English	EN_GB	285	Both
Greece/Greek	EL_GR	4971	Both
Greece/Greek euro	EL_GR_E	4971	Source
Greece/Greek preeuro	EL_GR_PE	875	Source
Guatemala/Spanish	ES_GT	284	Source
Honduras/Spanish	ES_HN	284	Source
Hungary/Hungarian	HU_HU	870	Both
Iceland/Icelandic	IS_IS	871	Both
India/English	EN_IN	37	Source
India/Hindi	HI_IN	1137	Source
India/Tamil	TA_IN	13488	Source
India/Telugu	TE_IN	13488	Source
Indonesia/Indonesian	ID_ID	500	Source
Ireland/English	EN_IE	1140	Source
Ireland/English euro	EN_IE_E	1140	Source
Ireland/English preeuro	EN_IE_PE	37	Source
Israel/Hebrew	HE_IL	424	Both
Israel/English	IW_IL	424	Both
Italy/Italian	IT_IT	1144	Both
Italy/Italian euro	IT_IT_E	1144	Both
Italy/Italian preeuro	IT_IT_PE	280	Both
Japan 13488	JA_13488	13488	Both
Japan 5035	JA_5035	5035	Object
Japan/Japanese	JA_JP	5026	Both
Jordan/Arabic	AR_JO	420	Source

<b>Description</b>	<b>Member</b>	<b>CCSID</b>	<b>How shipped</b>
Kazakhstan/Kazakh	KK_KZ	13488	Source
Kuwait/Arabic	AR_KW	420	Source
Latvia/Latvian	LV_LV	1112	Both
Lebanon/Arabic	AR_LB	420	Source
Lithuania/Lithuanian	LT_LT	1112	Both
Luxembourg/French	FR_LU	1147	Source
Luxembourg/French euro	FR_LU_E	1147	Source
Luxembourg/French preeuro	FR_LU_PE	297	Source
Luxembourg/German	DE_LU	1141	Source
Luxembourg/German euro	DE_LU_E	1141	Source
Luxembourg/German preeuro	DE_LU_PE	273	Source
Macedonia/Macedonian	MK_MK	1025	Both
Macedonia/Macedonian-Lotus	MK_MK_L	1025	Both
Malaysia/Malaysian	MS_MY	500	Source
Mexico/Spanish	ES_MX	284	Source
Morocco/Arabic	AR_MA	420	Source
Netherlands/Dutch	NL_NL	1140	Both
Netherlands/Dutch euro	NL_NL_E	1140	Both
Netherlands/Dutch preeuro	NL_NL_PE	37	Both
New Zealand/English	EN_NZ	37	Source
Nicaragua/Spanish	ES_NI	284	Source
Norway/Norwegian	NO_NO	277	Both
Norway/Norwegian (Bokmal)	NB_NO	277	Source
Oman/Arabic	AR_OM	420	Source
Panama/Spanish	ES_PA	284	Source
Paraguay/Spanish	ES_PY	284	Source
Peru/Spanish	ES_PE	284	Source
Philippines/English	EN_PH	37	Source
Poland/Polish	PL_PL	870	Both
Portugal/Portuguese	PT_PT	1140	Both

<b>Description</b>	<b>Member</b>	<b>CCSID</b>	<b>How shipped</b>
Portugal/Portuguese euro	PT_PT_E	1140	Both
Portugal/Portuguese preeuro	PT_PT_PE	37	Both
Puerto Rico/Spanish	ES_PR	284	Source
Qatar/Arabic	AR_QA	420	Source
Romania/Romanian	RO_RO	870	Both
Russia/Russian	RU_RU	1025	Both
Saudi Arabia/Arabic	AR_SA	420	Source
Serbia/Serbian, Cyrillic	SR_SP	1025	Both
Serbia/Serbian, Cyrillic-Lotus	SR_SP_L	1025	Both
Serbia/Serbian, Latin	SH_SP	870	Both
Singapore/English	EN_SG	37	Source
Singapore/Simplified Chinese	ZH_SG	1388	Source
Slovak/Slovakian	SK_SK	870	Both
Slovenia/Slovenian	SL_SI	870	Both
South Africa/English	EN_ZA	37	Source
South Korea/Korean	KO_KR	933	Both
Spain/Catalan	CA_ES	1145	Source
Spain/Catalan euro	CA_ES_E	1145	Source
Spain/Catalan preeuro	CA_PES_PE	284	Source
Spain/Spanish	ES_ES	1145	Both
Spain/Spanish euro	ES_ES_E	1145	Both
Spain/Spanish preeuro	ES_PES_PE	284	Both
Sweden/Swedish	SV_SE	1143	Both
Sweden/Swedish euro	SV_SE_E	1143	Source
Sweden/Swedish preeuro	SV_SE_PE	278	Source
Switzerland/French	FR_CH	500	Both
Switzerland/French-Lotus	FR_CH_L	500	Both
Switzerland/German	DE_CH	500	Both
Switzerland/German-Lotus	DE_CH_L	500	Both
Switzerland/Italian	IT_CH	500	Source
Syria/Arabic	AR_SY	420	Source

Description	Member	CCSID	How shipped
Taiwan/Mandarin, Traditional Chinese	ZH_TW	937	Both
Thailand/Thai	TH_TH	838	Both
Tunisia/Arabic	AR_TN	420	Source
Turkey/Turkish	TR_TR	1026	Both
Turkey/Turkish, English	TR_TR2	1026	Source
Ukraine/Ukrainian	UK_UA	1025	Both
United Arab Emirates/Arabic	AR_AE	420	Source
United Kingdom/English euro	EN_GB_E	1146	Source
United States/Spanish	ES_US	284	Source
Uruguay/Spanish	ES_UY	284	Source
USA/English	EN_US	37	Both
USA/English,Upper Case	EN_UPPER	37	Both
Venezuela/Spanish	ES_VE	284	Source
Vietnam/Vietnamese	VI_VN	1129	Source
Yemen/Arabic	AR_YE	420	Source
Yugoslavia/Serbian, Latin	SH_YU	870	Source
Yugoslavia/Serbian, Cyrillic	SR_YU	1025	Source

### Related concepts

#### Installing and enabling locales

If you are installing a new release, you can request that library QSYSLOCALE be installed on the system at that time.

#### Working with locales

Locales are used primarily in ILE-based application programs. Additionally, the Retrieve Locale Information (OPM, QLGRTVLC; ILE, QlgRetrieveLocaleInformation) API retrieves one or all categories of a locale.

#### Locale categories

This table describes the locale categories that are supported on the IBM i operating system.

#### Example: Creating and enabling a locale

This example contains the steps necessary for creating and enabling a locale.

### Mapping of locale symbolic names

This table lists locale symbolic names.

The table shows the following items:

- Some of the common locale symbolic names used by IBM i and the unicode code points they represent
- When appropriate, a comparison of code points between unicode code page and various IBM code pages
- The IBM graphic character global identifier (GCGID)



- An illustration of the GCGID

This table shows code points up through x'00FF'. The locale compiler understands most of the names defined by the Unicode organization. See the Unicode Web site (<http://www.unicode.org>) for an extended listing.

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<NUL>	0000	NULL (NUL)	037/00		
<SOH>	0001	START OF HEADING (SOH)	037/01		
<STX>	0002	START OF TEXT (STX)	037/02		
<ETX>	0003	END OF TEXT (ETX)	037/03		
<EOT>	0004	END OF TRANSMISSION (EOT)	037/37		
<ENQ>	0005	ENQUIRY (ENQ)	037/2D		
<ACK>	0006	ACKNOWLEDGE (ACK)	037/2E		
<alert>	0007	BELL	037/2F		
<BEL>	0007	BELL (BEL)	037/2F		
<BS>	0008	BACKSPACE (BS)	037/16		
<backspace>	0008	BACKSPACE	037/16		
<tab>	0009	CHARACTER TABULATION	037/05		
<HT>	0009	CHARACTER TABULATION (HT)	037/05		
<newline>	000A	LINE FEED	037/25		
<LF>	000A	LINE FEED (LF)	037/25		
<vertical-tab>	000B	LINE TABULATION	037/0B		
<VT>	000B	LINE TABULATION (VT)	037/0B		
<FF>	000C	FORM FEED (FF)	037/0C		
<form-feed>	000C	FORM FEED	037/0C		
<carriage-return>	000D	CARRIAGE RETURN	037/0D		
<SO>	000E	SHIFT OUT	037/0E		
<SI>	000F	SHIFT IN	037/0F		
<DLE>	0010	DATALINK ESCAPE (DLE)	037/10		
<DC1>	0011	DEVICE CONTROL ONE (DC1)	037/11		
<DC2>	0012	DEVICE CONTROL TWO (DC2)	037/12		
<DC3>	0013	DEVICE CONTROL THREE (DC3)	037/13		
<DC4>	0014	DEVICE CONTROL FOUR (DC4)	037/3C		

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<NAK>	0015	NEGATIVE ACKNOWLEDGE (NAK)	037/3D		
<SYN>	0016	SYNCHRONOUS IDLE (SYN)	037/32		
<ETB>	0017	END OF TRANSMISSION BLOCK (ETB)	037/26		
<CAN>	0018	CANCEL (CAN)	037/18		
<EM>	0019	END OF MEDIA	037/19		
<SUB>	001A	SUBSTITUTE (SUB)	037/3F		
<ESC>	001B	ESCAPE (ESC)	037/27		
<IS4>	001C	FILE SEPARATOR (IS4)	037/1C		
<FS>	001C	FILE SEPARATOR (IS4)	037/1C		
<IS3>	001D	GROUP SEPARATOR (IS3)	037/1D		
<GS>	001D	GROUP SEPARATOR (IS3)	037/1D		
<IS2>	001E	RECORD SEPARATOR (IS2)	037/1E		
<RS>	001E	RECORD SEPARATOR (IS2)	037/1E		
<IS1>	001F	UNIT SEPARATOR (IS1)	037/1F		
<US>	001F	UNIT SEPARATOR (IS1)	037/1F		
<DEL>	007F	DELETE (DEL)	037/07		
<space>	0020	SPACE	037/40	SP010000	
<exclamation-mark>	0021	EXCLAMATION MARK	500/4F	SP020000	!
<quotation-mark>	0022	QUOTATION MARK	500/7F	SP040000	"
<number-sign>	0023	NUMBER SIGN	500/7B	SM010000	#
<dollar-sign>	0024	DOLLAR SIGN	500/5B	SC030000	\$
<percent-sign>	0025	PERCENT SIGN	500/6C	SM020000	%
<ampersand>	0026	AMPERSAND	500/50	SM030000	&
<apostrophe>	0027	APOSTROPHE	500/7D	SP050000	'
<left-parenthesis>	0028	LEFT PARENTHESIS	500/4D	SP060000	(
<right-parenthesis>	0029	RIGHT PARENTHESIS	500/5D	SP070000	)
<asterisk>	002A	ASTERISK	500/5C	SM040000	*

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<plus-sign>	002B	PLUS SIGN	500/4E	SA010000	+
<comma>	002C	COMMA	500/6B	SP080000	,
<hyphen>	002D	HYPHEN	500/60	SP100000	-
<hyphen-minus>	002D	HYPHEN-MINUS	500/60	SP100000	-
<period>	002E	PERIOD	500/4B	SP110000	.
<full-stop>	002E	FULL STOP	500/4B	SP110000	.
<slash>	002F	SLASH	500/61	SP120000	/
<solidus>	002F	SOLIDUS	500/61	SP120000	/
<zero>	0030	DIGIT ZERO	500/F0	ND100000	0
<one>	0031	DIGIT ONE	500/F1	ND010000	1
<two>	0032	DIGIT TWO	500/F2	ND020000	2
<three>	0033	DIGIT THREE	500/F3	ND030000	3
<four>	0034	DIGIT FOUR	500/F4	ND040000	4
<five>	0035	DIGIT FIVE	500/F5	ND050000	5
<six>	0036	DIGIT SIX	500/F6	ND060000	6
<seven>	0037	DIGIT SEVEN	500/F7	ND070000	7
<eight>	0038	DIGIT EIGHT	500/F8	ND080000	8
<nine>	0039	DIGIT NINE	500/F9	ND090000	9
<colon>	003A	COLON	500/7A	SP130000	:
<semicolon>	003B	SEMICOLON	500/5E	SP140000	;
<less-than-sign>	003C	LESS-THAN SIGN	500/4C	SA030000	<
<equals-sign>	003D	EQUALS SIGN	500/7E	SA040000	=
<greater-than-sign>	003E	GREATER-THAN SIGN	500/6E	SA050000	>
<question-mark>	003F	QUESTION MARK	500/6F	SP150000	?
<commercial-at>	0040	COMMERCIAL AT	500/7C	SM050000	@
<A>	0041	LATIN CAPITAL LETTER A	500/C1	LA020000	<b>A</b>
<B>	0042	LATIN CAPITAL LETTER B	500/C2	LB020000	<b>B</b>
<C>	0043	LATIN CAPITAL LETTER C	500/C3	LC020000	<b>C</b>

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<D>	0044	LATIN CAPITAL LETTER D	500/C4	LD020000	<b>D</b>
<E>	0045	LATIN CAPITAL LETTER E	500/C5	LE020000	<b>E</b>
<F>	0046	LATIN CAPITAL LETTER F	500/C6	LF020000	<b>F</b>
<G>	0047	LATIN CAPITAL LETTER G	500/C7	LG020000	<b>G</b>
<H>	0048	LATIN CAPITAL LETTER H	500/C8	LH020000	<b>H</b>
<I>	0049	LATIN CAPITAL LETTER I	500/C9	LI020000	<b>I</b>
<J>	004A	LATIN CAPITAL LETTER J	500/D1	LJ020000	<b>J</b>
<K>	004B	LATIN CAPITAL LETTER K	500/D2	LK020000	<b>K</b>
<L>	004C	LATIN CAPITAL LETTER L	500/D3	LL020000	<b>L</b>
<M>	004D	LATIN CAPITAL LETTER M	500/D4	LM020000	<b>M</b>
<N>	004E	LATIN CAPITAL LETTER N	500/D5	LN020000	<b>N</b>
<O>	004F	LATIN CAPITAL LETTER O	500/D6	LO020000	<b>O</b>
	0050	LATIN CAPITAL LETTER P	500/D7	LP020000	<b>P</b>
<Q>	0051	LATIN CAPITAL LETTER Q	500/D8	LQ020000	<b>Q</b>
<R>	0052	LATIN CAPITAL LETTER R	500/D9	LR020000	<b>R</b>
<S>	0053	LATIN CAPITAL LETTER S	500/E2	LS020000	<b>S</b>
<T>	0054	LATIN CAPITAL LETTER T	500/E3	LT020000	<b>T</b>
<U>	0055	LATIN CAPITAL LETTER U	500/E4	LU020000	<b>U</b>
<V>	0056	LATIN CAPITAL LETTER V	500/E5	LV020000	<b>V</b>
<W>	0057	LATIN CAPITAL LETTER W	500/E6	LW020000	<b>W</b>
<X>	0058	LATIN CAPITAL LETTER X	500/E7	LX020000	<b>X</b>
<Y>	0059	LATIN CAPITAL LETTER Y	500/E8	LY020000	<b>Y</b>
<Z>	005A	LATIN CAPITAL LETTER Z	500/E9	LZ020000	<b>Z</b>
<left-square-bracket>	005B	LEFT SQUARE BRACKET	500/4A	SM060000	<b>[</b>
<backslash>	005C	BACKSLASH	500/E0	SM070000	<b>\</b>

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<reverse-solidus>	005C	REVERSE SOLIDUS	500/E0	SM070000	\
<right-square-bracket>	005D	RIGHT SQUARE BRACKET	500/5A	SM080000	]
<circumflex>	005E	CIRCUMFLEX	500/5F	SD150000	^
<circumflex-accent>	005E	CIRCUMFLEX ACCENT	500/5F	SD150000	^
<underscore>	005F	UNDERSCORE	500/6D	SP090000	_
<underline>	005F	UNDERLINE	500/6D	SP090000	_
<low-line>	005F	LOW LINE	500/6D	SP090000	_
<grave-accent>	0060	GRAVE ACCENT	500/79	SD130000	`
<a>	0061	LATIN SMALL LETTER A	500/81	LA010000	a
<b>	0062	LATIN SMALL LETTER B	500/82	LB010000	b
<c>	0063	LATIN SMALL LETTER C	500/83	LC010000	c
<d>	0064	LATIN SMALL LETTER D	500/84	LD010000	d
<e>	0065	LATIN SMALL LETTER E	500/85	LE010000	e
<f>	0066	LATIN SMALL LETTER F	500/86	LF010000	f
<g>	0067	LATIN SMALL LETTER G	500/87	LG010000	g
<h>	0068	LATIN SMALL LETTER H	500/88	LH010000	h
<i>	0069	LATIN SMALL LETTER I	500/89	LI010000	i
<j>	006A	LATIN SMALL LETTER J	500/91	LJ010000	j
<k>	006B	LATIN SMALL LETTER K	500/92	LK010000	k
<l>	006C	LATIN SMALL LETTER L	500/93	LL010000	l
<m>	006D	LATIN SMALL LETTER M	500/94	LM010000	m
<n>	006E	LATIN SMALL LETTER N	500/95	LN010000	n
<o>	006F	LATIN SMALL LETTER O	500/96	LO010000	o
	0070	LATIN SMALL LETTER P	500/97	LP010000	p
<q>	0071	LATIN SMALL LETTER Q	500/98	LQ010000	q
<r>	0072	LATIN SMALL LETTER R	500/99	LR010000	r

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<s>	0073	LATIN SMALL LETTER S	500/A2	LS010000	s
<t>	0074	LATIN SMALL LETTER T	500/A3	LT010000	t
<u>	0075	LATIN SMALL LETTER U	500/A4	LU010000	u
<v>	0076	LATIN SMALL LETTER V	500/A5	LV010000	v
<w>	0077	LATIN SMALL LETTER W	500/A6	LW010000	w
<x>	0078	LATIN SMALL LETTER X	500/A7	LX010000	x
<y>	0079	LATIN SMALL LETTER Y	500/A8	LY010000	y
<z>	007A	LATIN SMALL LETTER Z	500/A9	LZ010000	z
<left-brace>	007B	LEFT BRACE	500/C0	SM110000	{
<left-curly-bracket>	007B	LEFT CURLY BRACKET	500/C0	SM110000	{
<vertical-line>	007C	VERTICAL LINE	500/BB	SM130000	
<right-brace>	007D	RIGHT BRACE	500/D0	SM140000	}
<right-curly-bracket>	007D	RIGHT CURLY BRACKET	500/D0	SM140000	}
<tilde>	007E	TILDE	500/A1	SD190000	~
<BPH>	0082	BREAK PERMITTED HERE	037/22		
<NBH>	0083	NO BREAK HERE	037/23		
<IND>	0084	INDEX	037/24		
<NEL>	0085	NEXT LINE	037/15		
<SSA>	0086	START OF SELECTED AREA	037/06		
<ESA>	0087	END OF SELECTED AREA	037/17		
<HTS>	0088	CHARACTER TABULATION SET	037/28		
<HTJ>	0089	CHARACTER TABULATION WITH JUSTIFICATION	037/29		
<VTS>	008A	LINE TABULATION SET	037/2A		
<PLD>	008B	PARTIAL LINE DOWN	037/2B		
<PLU>	008C	PARTIAL LINE UP	037/2C		
<RI>	008D	REVERSE INDEX	037/09		
<SS2>	008E	SINGLE SHIFT TWO	037/0A		

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<SS3>	008F	SINGLE SHIFT THREE	037/1B		
<DCS>	0090	DEVICE CONTROL STRING	037/30		
<PU1>	0091	PRIVATE USE ONE	037/31		
<PU2>	0092	PRIVATE USE TWO	037/1A		
<STS>	0093	SET TRANSMIT STATE	037/33		
<CCH>	0094	CANCEL CHARACTER	037/34		
<MW>	0095	MESSAGE WAITING	037/35		
<SPS>	0096	START OF GUARDED AREA	037/36		
<EPA>	0097	END OF GUARDED AREA	037/08		
<SOS>	0098	START OF STRING	037/38		
<SCI>	009A	SINGLE CHARACTER INTRODUCER	037/39		
<CSI>	009B	CONTROL SEQUENCE INTRODUCER	037/3B		
<ST>	009C	STRING TERMINATOR	037/04		
<OSC>	009D	OPERATING SYSTEM COMMAND	037/14		
<PM>	009E	PRIVACY MESSAGE	037/3E		
<APC>	009F	APPLICATION PROGRAM COMMAND	037/FF		
<nobreakspace>	00A0	NO-BREAK SPACE	500/41	SP300000	␣
<exclamation-down>	00A1	INVERTED EXCLAMATION MARK	500/AA	SP030000	¡
<cent>	00A2	CENT SIGN	500/B0	SC040000	¢
<sterling>	00A3	POUND SIGN	500/B1	SC020000	£
<currency>	00A4	CURRENCY SIGN	500/9F	SC010000	¤
<yen>	00A5	YEN SIGN	500/B2	SC050000	¥
<yen-sign>	00A5	YEN SIGN	850/BE	SC050000	¥
<broken-bar>	00A6	BROKEN VERTICAL BAR	500/6A	SM650000	¦
<section>	00A7	SECTION SIGN	500/B5	SM240000	§
<diaeresis>	00A8	SPACING DIAERESIS	500/8D	SD170000	¨

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<copyright>	00A9	COPYRIGHT SIGN	500/B4	SM520000	©
<feminine>	00AA	FEMININE ORDINAL INDICATOR	500/9A	SM210000	♀
<guillemot-left>	00AB	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK	500/8A	SP170000	«
<not>	00AC	NOT SIGN	500/BA	SM660000	¬
<dash>	00AD	SOFT HYPHEN	500/CA	SP320000	–
<registered>	00AE	REGISTERED TRADE MARK SIGN	500/AF	SM530000	®
<macron>	00AF	SPACING MACRON	500/BC	SM150000	¯
<degree>	00B0	DEGREE SIGN	500/90	SM190000	°
<plus-minus>	00B1	PLUS-OR-MINUS SIGN	500/8F	SA020000	±
<two-superior>	00B2	SUPERSCRIPT DIGIT TWO	500/EA	ND021000	²
<three-superior>	00B3	SUPERSCRIPT DIGIT THREE	500/FA	ND031000	³
<acute>	00B4	SPACING ACUTE	500/BE	SD110000	´
<mu>	00B5	MICRO SIGN	500/A0	SM170000	μ
<paragraph>	00B6	PARAGRAPH (PILCROW) SIGN	500/B6	SM250000	¶
<dot>	00B7	MIDDLE DOT	500/B3	SD630000	·
<cedilla>	00B8	SPACING CEDILLA	500/9D	SD410000	¸
<one-superior>	00B9	SUPERSCRIPT DIGIT ONE	500/DA	ND011000	¹
<masculine>	00BA	MASCULINE ORDINAL INDICATOR	500/9B	SM200000	♂
<guillemot-right>	00BB	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK	500/8B	SP180000	»
<one-quarter>	00BC	FRACTION ONE QUARTER	500/B7	NF040000	¼
<one-half>	00BD	FRACTION ONE HALF	500/B8	NF010000	½
<three-quarters>	00BE	FRACTION THREE QUARTERS	500/B9	NF050000	¾
<question-down>	00BF	INVERTED QUESTION MARK	500/AB	SP160000	¿
<A-grave>	00C0	LATIN CAPITAL LETTER A GRAVE	500/64	LA140000	À
<A-acute>	00C1	LATIN CAPITAL LETTER A ACUTE	500/65	LA120000	Á



Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<A-circumflex>	00C2	LATIN CAPITAL LETTER A CIRCUMFLEX	500/62	LA160000	Â
<A-tilde>	00C3	LATIN CAPITAL LETTER A TILDE	500/66	LA200000	Ã
<A-diaeresis>	00C4	LATIN CAPITAL LETTER A DIAERESIS	500/63	LA180000	Ä
<A-ring>	00C5	LATIN CAPITAL LETTER A RING ABOVE	500/67	LA280000	Å
<AE>	00C6	LATIN CAPITAL LIGATURE AE	500/9E	LA520000	Æ
<C-cedilla>	00C7	LATIN CAPITAL LETTER C CEDILLA	500/68	LC420000	Ç
<E-grave>	00C8	LATIN CAPITAL LETTER E GRAVE	500/74	LE140000	È
<E-acute>	00C9	LATIN CAPITAL LETTER E ACUTE	500/71	LE120000	É
<E-circumflex>	00CA	LATIN CAPITAL LETTER E CIRCUMFLEX	500/72	LE160000	Ê
<E-diaeresis>	00CB	LATIN CAPITAL LETTER E DIAERESIS	500/73	LE180000	Ë
<I-grave>	00CC	LATIN CAPITAL LETTER I GRAVE	500/78	LI140000	Ï
<I-acute>	00CD	LATIN CAPITAL LETTER I ACUTE	500/75	LI120000	Í
<I-circumflex>	00CE	LATIN CAPITAL LETTER I CIRCUMFLEX	500/76	LI160000	Î
<I-diaeresis>	00CF	LATIN CAPITAL LETTER I DIAERESIS	500/77	LI180000	Ï
<Eth>	00D0	LATIN CAPITAL LETTER ETH (Icelandic)	500/AC	LD620000	Ð
<N-tilde>	00D1	LATIN CAPITAL LETTER N TILDE	500/69	LN200000	Ñ
<O-grave>	00D2	LATIN CAPITAL LETTER O GRAVE	500/ED	LO140000	Ò
<O-acute>	00D3	LATIN CAPITAL LETTER O ACUTE	500/EE	LO120000	Ó
<O-circumflex>	00D4	LATIN CAPITAL LETTER O CIRCUMFLEX	500/EB	LO160000	Ô
<O-tilde>	00D5	LATIN CAPITAL LETTER O TILDE	500/EF	LO200000	Õ
<O-diaeresis>	00D6	LATIN CAPITAL LETTER O DIAERESIS	500/EC	LO180000	Ö

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<multiply>	00D7	MULTIPLICATION SIGN	500/BF	SA070000	×
<O-slash>	00D8	LATIN CAPITAL LETTER O STROKE	500/80	LO620000	Ø
<U-grave>	00D9	LATIN CAPITAL LETTER U GRAVE	500/FD	LU140000	Ù
<U-acute>	00DA	LATIN CAPITAL LETTER U ACUTE	500/FE	LU120000	Ú
<U-circumflex>	00DB	LATIN CAPITAL LETTER U CIRCUMFLEX	500/FB	LU160000	Û
<U-diaeresis>	00DC	LATIN CAPITAL LETTER U DIAERESIS	500/FC	LU180000	Ü
<Y-acute>	00DD	LATIN CAPITAL LETTER Y ACUTE	500/AD	LY120000	Ý
<Thorn>	00DE	LATIN CAPITAL LETTER THORN (Icelandic)	500/AE	LT640000	Þ
<s-sharp>	00DF	LATIN SMALL LETTER SHARP S (German)	500/59	LS610000	ß
<a-grave>	00E0	LATIN SMALL LETTER A GRAVE	500/44	LA130000	à
<a-acute>	00E1	LATIN SMALL LETTER A ACUTE	500/45	LA110000	á
<a-circumflex>	00E2	LATIN SMALL LETTER A CIRCUMFLEX	500/42	LA150000	â
<a-tilde>	00E3	LATIN SMALL LETTER A TILDE	500/46	LA190000	ã
<a-diaeresis>	00E4	LATIN SMALL LETTER A DIAERESIS	500/43	LA170000	ä
<a-ring>	00E5	LATIN SMALL LETTER A RING ABOVE	500/47	LA270000	å
<ae>	00E6	LATIN SMALL LIGATURE AE	500/9C	LA510000	æ
<c-cedilla>	00E7	LATIN SMALL LETTER C CEDILLA	500/48	LC410000	ç
<e-grave>	00E8	LATIN SMALL LETTER E GRAVE	500/54	LE130000	è
<e-acute>	00E9	LATIN SMALL LETTER E ACUTE	500/51	LE110000	é
<e-circumflex>	00EA	LATIN SMALL LETTER E CIRCUMFLEX	500/52	LE150000	ê
<e-diaeresis>	00EB	LATIN SMALL LETTER E DIAERESIS	500/53	LE170000	ë

Symbolic name	Unicode (ISO 10646) code point (hexadecimal)	Description	IBM code page and code point (xxx/xx)	IBM GCGID	GCGID illustration
<i-grave>	00EC	LATIN SMALL LETTER I GRAVE	500/58	LI130000	ï
<i-acute>	00ED	LATIN SMALL LETTER I ACUTE	500/55	LI110000	í
<i-circumflex>	00EE	LATIN SMALL LETTER I CIRCUMFLEX	500/56	LI150000	î
<i-diaresis>	00EF	LATIN SMALL LETTER I DIAERESIS	500/57	LI170000	ï
<eth>	00F0	LATIN SMALL LETTER ETH (Icelandic)	500/8C	LD630000	ð
<n-tilde>	00F1	LATIN SMALL LETTER N TILDE	500/49	LN190000	ñ
<o-grave>	00F2	LATIN SMALL LETTER O GRAVE	500/CD	LO130000	ò
<o-acute>	00F3	LATIN SMALL LETTER O ACUTE	500/CE	LO110000	ó
<o-circumflex>	00F4	LATIN SMALL LETTER O CIRCUMFLEX	500/CB	LO150000	ô
<o-tilde>	00F5	LATIN SMALL LETTER O TILDE	500/CF	LO190000	õ
<o-diaresis>	00F6	LATIN SMALL LETTER O DIAERESIS	500/CC	LO170000	ö
<divide>	00F7	DIVISION SIGN	500/E1	SA060000	÷
<division>	00F7	DIVISION SIGN	500/E1	SA060000	÷
<o-slash>	00F8	LATIN SMALL LETTER O STROKE	500/70	LO610000	ø
<u-grave>	00F9	LATIN SMALL LETTER U GRAVE	500/DD	LU130000	ù
<u-acute>	00FA	LATIN SMALL LETTER U ACUTE	500/DE	LU110000	ú
<u-circumflex>	00FB	LATIN SMALL LETTER U CIRCUMFLEX	500/DB	LU150000	û
<u-diaresis>	00FC	LATIN SMALL LETTER U DIAERESIS	500/DC	LU170000	ü
<y-acute>	00FD	LATIN SMALL LETTER Y ACUTE	500/8D	LY110000	ý
<thorn>	00FE	LATIN SMALL LETTER THORN (Icelandic)	500/8E	LT630000	þ
<y-diaresis>	00FF	LATIN SMALL LETTER Y DIAERESIS	500/DF	LY170000	ÿ

## Related concepts

Locale symbolic names

The IBM i operating system supports locale symbolic names based on predefined names from the X/Open Standard portable character set.

## Related information

[Unicode Home Page](#)

## UCS-2 level-1 mapping tables

You can use the Convert Case API to convert characters encoded in universal character set 2 level 1 (UCS-2 level-1) between uppercase and lowercase.

## Related information

[Convert Case \(QLGCNVCS, QlgConvertCase\) API](#)

### Unicode uppercase to lowercase conversion mapping table

The table shows the mapping for the conversion of Unicode from uppercase to lowercase. You can refer to this table to see how the Convert Case API converts character data from uppercase to lowercase.

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0041	0061	LATIN CAPITAL LETTER A	LATIN SMALL LETTER A
0042	0062	LATIN CAPITAL LETTER B	LATIN SMALL LETTER B
0043	0063	LATIN CAPITAL LETTER C	LATIN SMALL LETTER C
0044	0064	LATIN CAPITAL LETTER D	LATIN SMALL LETTER D
0045	0065	LATIN CAPITAL LETTER E	LATIN SMALL LETTER E
0046	0066	LATIN CAPITAL LETTER F	LATIN SMALL LETTER F
0047	0067	LATIN CAPITAL LETTER G	LATIN SMALL LETTER G
0048	0068	LATIN CAPITAL LETTER H	LATIN SMALL LETTER H
0049	0069	LATIN CAPITAL LETTER I	LATIN SMALL LETTER I
004A	006A	LATIN CAPITAL LETTER J	LATIN SMALL LETTER J
004B	006B	LATIN CAPITAL LETTER K	LATIN SMALL LETTER K
004C	006C	LATIN CAPITAL LETTER L	LATIN SMALL LETTER L
004D	006D	LATIN CAPITAL LETTER M	LATIN SMALL LETTER M
004E	006E	LATIN CAPITAL LETTER N	LATIN SMALL LETTER N
004F	006F	LATIN CAPITAL LETTER O	LATIN SMALL LETTER O
0050	0070	LATIN CAPITAL LETTER P	LATIN SMALL LETTER P
0051	0071	LATIN CAPITAL LETTER Q	LATIN SMALL LETTER Q
0052	0072	LATIN CAPITAL LETTER R	LATIN SMALL LETTER R
0053	0073	LATIN CAPITAL LETTER S	LATIN SMALL LETTER S
0054	0074	LATIN CAPITAL LETTER T	LATIN SMALL LETTER T
0055	0075	LATIN CAPITAL LETTER U	LATIN SMALL LETTER U
0056	0076	LATIN CAPITAL LETTER V	LATIN SMALL LETTER V
0057	0077	LATIN CAPITAL LETTER W	LATIN SMALL LETTER W
0058	0078	LATIN CAPITAL LETTER X	LATIN SMALL LETTER X
0059	0079	LATIN CAPITAL LETTER Y	LATIN SMALL LETTER Y
005A	007A	LATIN CAPITAL LETTER Z	LATIN SMALL LETTER Z
00C0	00E0	LATIN CAPITAL LETTER A GRAVE	LATIN SMALL LETTER A GRAVE
00C1	00E1	LATIN CAPITAL LETTER A ACUTE	LATIN SMALL LETTER A GRAVE
00C2	00E2	LATIN CAPITAL LETTER A CIRCUMFLEX	LATIN SMALL LETTER A GRAVE

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
00C3	00E3	LATIN CAPITAL LETTER A TILDE	LATIN SMALL LETTER A GRAVE
00C4	00E4	LATIN CAPITAL LETTER A DIAERESIS	LATIN SMALL LETTER A GRAVE
00C5	00E5	LATIN CAPITAL LETTER A RING	LATIN SMALL LETTER A GRAVE
00C6	00E6	LATIN CAPITAL LETTER A E	LATIN SMALL LETTER A GRAVE
00C7	00E7	LATIN CAPITAL LETTER C CEDILLA	LATIN SMALL LETTER A GRAVE
00C8	00E8	LATIN CAPITAL LETTER E GRAVE	LATIN SMALL LETTER A GRAVE
00C9	00E9	LATIN CAPITAL LETTER E ACUTE	LATIN SMALL LETTER A GRAVE
00CA	00EA	LATIN CAPITAL LETTER E CIRCUMFLEX	LATIN SMALL LETTER E CIRCUMFLEX
00CB	00EB	LATIN CAPITAL LETTER E DIAERESIS	LATIN SMALL LETTER E DIAERESIS
00CC	00EC	LATIN CAPITAL LETTER I GRAVE	LATIN SMALL LETTER I GRAVE
00CD	00ED	LATIN CAPITAL LETTER I ACUTE	LATIN SMALL LETTER I ACUTE
00CE	00EE	LATIN CAPITAL LETTER I CIRCUMFLEX	LATIN SMALL LETTER I CIRCUMFLEX
00CF	00EF	LATIN CAPITAL LETTER I DIAERESIS	LATIN SMALL LETTER I DIAERESIS
00D0	00F0	LATIN CAPITAL LETTER ETH	LATIN SMALL LETTER ETH
00D1	00F1	LATIN CAPITAL LETTER N TILDE	LATIN SMALL LETTER N TILDE
00D2	00F2	LATIN CAPITAL LETTER O GRAVE	LATIN SMALL LETTER O GRAVE
00D3	00F3	LATIN CAPITAL LETTER O ACUTE	LATIN SMALL LETTER O ACUTE
00D4	00F4	LATIN CAPITAL LETTER O CIRCUMFLEX	LATIN SMALL LETTER O CIRCUMFLEX
00D5	00F5	LATIN CAPITAL LETTER O TILDE	LATIN SMALL LETTER O TILDE
00D6	00F6	LATIN CAPITAL LETTER O DIAERESIS	LATIN SMALL LETTER O DIAERESIS
00D8	00F8	LATIN CAPITAL LETTER O SLASH	LATIN SMALL LETTER O SLASH
00D9	00F9	LATIN CAPITAL LETTER U GRAVE	LATIN SMALL LETTER U GRAVE
00DA	00FA	LATIN CAPITAL LETTER U ACUTE	LATIN SMALL LETTER U ACUTE
00DB	00FB	LATIN CAPITAL LETTER U CIRCUMFLEX	LATIN SMALL LETTER U CIRCUMFLEX
00DC	00FC	LATIN CAPITAL LETTER U DIAERESIS	LATIN SMALL LETTER U DIAERESIS
00DD	00FD	LATIN CAPITAL LETTER Y ACUTE	LATIN SMALL LETTER Y ACUTE
00DE	00FE	LATIN CAPITAL LETTER THORN	LATIN SMALL LETTER THORN
0100	0101	LATIN CAPITAL LETTER A WITH MACRON	LATIN SMALL LETTER A WITH MACRON
0102	0103	LATIN CAPITAL LETTER A WITH BREVE	LATIN SMALL LETTER A WITH BREVE
0104	0105	LATIN CAPITAL LETTER A WITH OGONEK	LATIN SMALL LETTER A WITH OGONEK
0106	0107	LATIN CAPITAL LETTER C WITH ACUTE	LATIN SMALL LETTER C WITH ACUTE
0108	0109	LATIN CAPITAL LETTER C WITH CIRCUMFLEX	LATIN SMALL LETTER C WITH CIRCUMFLEX
010A	010B	LATIN CAPITAL LETTER C WITH DOT ABOVE	LATIN SMALL LETTER C WITH DOT ABOVE
010C	010D	LATIN CAPITAL LETTER C WITH CARON	LATIN SMALL LETTER C WITH CARON

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
010E	010F	LATIN CAPITAL LETTER D WITH CARON	LATIN SMALL LETTER D WITH CARON
0110	0111	LATIN CAPITAL LETTER D WITH STROKE	LATIN SMALL LETTER D WITH STROKE
0112	0113	LATIN CAPITAL LETTER E WITH MACRON	LATIN SMALL LETTER E WITH MACRON
0114	0115	LATIN CAPITAL LETTER E WITH BREVE	LATIN SMALL LETTER E WITH BREVE
0116	0117	LATIN CAPITAL LETTER E WITH DOT ABOVE	LATIN SMALL LETTER E WITH DOT ABOVE
0118	0119	LATIN CAPITAL LETTER E WITH OGONEK	LATIN SMALL LETTER E WITH OGONEK
011A	011B	LATIN CAPITAL LETTER E WITH CARON	LATIN SMALL LETTER E WITH CARON
011C	011D	LATIN CAPITAL LETTER G WITH CIRCUMFLEX	LATIN SMALL LETTER G WITH CIRCUMFLEX
011E	011F	LATIN CAPITAL LETTER G WITH BREVE	LATIN SMALL LETTER G WITH BREVE
0120	0121	LATIN CAPITAL LETTER G WITH DOT ABOVE	LATIN SMALL LETTER G WITH DOT ABOVE
0122	0123	LATIN CAPITAL LETTER G WITH CEDILLA	LATIN SMALL LETTER G WITH CEDILLA
0124	0125	LATIN CAPITAL LETTER H WITH CIRCUMFLEX	LATIN SMALL LETTER H WITH CIRCUMFLEX
0126	0127	LATIN CAPITAL LETTER H WITH STROKE	LATIN SMALL LETTER H WITH STROKE
0128	0129	LATIN CAPITAL LETTER I WITH TILDE	LATIN SMALL LETTER I WITH TILDE
012A	012B	LATIN CAPITAL LETTER I WITH MACRON	LATIN SMALL LETTER I WITH MACRON
012C	012D	LATIN CAPITAL LETTER I WITH BREVE	LATIN SMALL LETTER I WITH BREVE
012E	012F	LATIN CAPITAL LETTER I WITH OGONEK	LATIN SMALL LETTER I WITH OGONEK
0130	0069	LATIN CAPITAL LETTER I WITH DOT ABOVE	LATIN SMALL LETTER I
0132	0133	LATIN CAPITAL LIGATURE IJ	LATIN SMALL LIGATURE IJ
0134	0135	LATIN CAPITAL LETTER J WITH CIRCUMFLEX	LATIN SMALL LETTER J WITH CIRCUMFLEX
0136	0137	LATIN CAPITAL LETTER K WITH CEDILLA	LATIN SMALL LETTER K WITH CEDILLA
0139	013A	LATIN CAPITAL LETTER L WITH ACUTE	LATIN SMALL LETTER L WITH ACUTE
013B	013C	LATIN CAPITAL LETTER L WITH CEDILLA	LATIN SMALL LETTER L WITH CEDILLA
013D	013E	LATIN CAPITAL LETTER L WITH CARON	LATIN SMALL LETTER L WITH CARON
013F	0140	LATIN CAPITAL LETTER L WITH MIDDLE DOT	LATIN SMALL LETTER L WITH MIDDLE DOT
0141	0142	LATIN CAPITAL LETTER L WITH STROKE	LATIN SMALL LETTER L WITH STROKE
0143	0144	LATIN CAPITAL LETTER N WITH ACUTE	LATIN SMALL LETTER N WITH ACUTE
0145	0146	LATIN CAPITAL LETTER N WITH CEDILLA	LATIN SMALL LETTER N WITH CEDILLA

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0147	0148	LATIN CAPITAL LETTER N WITH CARON	LATIN SMALL LETTER N WITH CARON
014A	014B	LATIN CAPITAL LETTER ENG (SAMI)	LATIN SMALL LETTER ENG (SAMI)
014C	014D	LATIN CAPITAL LETTER O WITH MACRON	LATIN SMALL LETTER O WITH MACRON
014E	014F	LATIN CAPITAL LETTER O WITH BREVE	LATIN SMALL LETTER O WITH BREVE
0150	0151	LATIN CAPITAL LETTER O WITH DOUBLE ACUTE	LATIN SMALL LETTER O WITH DOUBLE ACUTE
0152	0153	LATIN CAPITAL LIGATURE OE	LATIN SMALL LIGATURE OE
0154	0155	LATIN CAPITAL LETTER R WITH ACUTE	LATIN SMALL LETTER R WITH ACUTE
0156	0157	LATIN CAPITAL LETTER R WITH CEDILLA	LATIN SMALL LETTER R WITH CEDILLA
0158	0159	LATIN CAPITAL LETTER R WITH CARON	LATIN SMALL LETTER R WITH CARON
015A	015B	LATIN CAPITAL LETTER S WITH ACUTE	LATIN SMALL LETTER S WITH ACUTE
015C	015D	LATIN CAPITAL LETTER S WITH CIRCUMFLEX	LATIN SMALL LETTER S WITH CIRCUMFLEX
015E	015F	LATIN CAPITAL LETTER S WITH CEDILLA	LATIN SMALL LETTER S WITH CEDILLA
0160	0161	LATIN CAPITAL LETTER S WITH CARON	LATIN SMALL LETTER S WITH CARON
0162	0163	LATIN CAPITAL LETTER T WITH CEDILLA	LATIN SMALL LETTER T WITH CEDILLA
0164	0165	LATIN CAPITAL LETTER T WITH CARON	LATIN SMALL LETTER T WITH CARON
0166	0167	LATIN CAPITAL LETTER T WITH STROKE	LATIN SMALL LETTER T WITH STROKE
0168	0169	LATIN CAPITAL LETTER U WITH TILDE	LATIN SMALL LETTER U WITH TILDE
016A	016B	LATIN CAPITAL LETTER U WITH MACRON	LATIN SMALL LETTER U WITH MACRON
016C	016D	LATIN CAPITAL LETTER U WITH BREVE	LATIN SMALL LETTER U WITH BREVE
016E	016F	LATIN CAPITAL LETTER U WITH RING ABOVE	LATIN SMALL LETTER U WITH RING ABOVE
0170	0171	LATIN CAPITAL LETTER U WITH DOUBLE ACUTE	LATIN SMALL LETTER U WITH DOUBLE ACUTE
0172	0173	LATIN CAPITAL LETTER U WITH OGONEK	LATIN SMALL LETTER U WITH OGONEK
0174	0175	LATIN CAPITAL LETTER W WITH CIRCUMFLEX	LATIN SMALL LETTER W WITH CIRCUMFLEX
0176	0177	LATIN CAPITAL LETTER Y WITH CIRCUMFLEX	LATIN SMALL LETTER Y WITH CIRCUMFLEX
0178	00FF	LATIN CAPITAL LETTER Y WITH DIAERESIS	LATIN SMALL LETTER Y WITH DIAERESIS
0179	017A	LATIN CAPITAL LETTER Z WITH ACUTE	LATIN SMALL LETTER Z WITH ACUTE
017B	017C	LATIN CAPITAL LETTER Z WITH DOT ABOVE	LATIN SMALL LETTER Z WITH DOT ABOVE
017D	017E	LATIN CAPITAL LETTER Z WITH CARON	LATIN SMALL LETTER Z WITH CARON

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0181	0253	LATIN CAPITAL LETTER B WITH HOOK	LATIN SMALL LETTER B WITH HOOK
0182	0183	LATIN CAPITAL LETTER B WITH TOPBAR	LATIN SMALL LETTER B WITH TOPBAR
0184	0185	LATIN CAPITAL LETTER TONE SIX	LATIN SMALL LETTER TONE SIX
0186	0254	LATIN CAPITAL LETTER OPEN O	LATIN SMALL LETTER OPEN O
0187	0188	LATIN CAPITAL LETTER C WITH HOOK	LATIN SMALL LETTER C WITH HOOK
018A	0257	LATIN CAPITAL LETTER D WITH HOOK	LATIN SMALL LETTER D WITH HOOK
018B	018C	LATIN CAPITAL LETTER D WITH TOPBAR	LATIN SMALL LETTER D WITH TOPBAR
018E	0258	LATIN CAPITAL LETTER REVERSED E	LATIN SMALL LETTER REVERSED E
018F	0259	LATIN CAPITAL LETTER SCHWA	LATIN SMALL LETTER SCHWA
0190	025B	LATIN CAPITAL LETTER OPEN E	LATIN SMALL LETTER OPEN E
0191	0192	LATIN CAPITAL LETTER F WITH HOOK	LATIN SMALL LETTER F WITH HOOK
0193	0260	LATIN CAPITAL LETTER G WITH HOOK	LATIN SMALL LETTER G WITH HOOK
0194	0263	LATIN CAPITAL LETTER GAMMA	LATIN SMALL LETTER GAMMA
0196	0269	LATIN CAPITAL LETTER IOTA	LATIN SMALL LETTER IOTA
0197	0268	LATIN CAPITAL LETTER I WITH STROKE	LATIN SMALL LETTER I WITH STROKE
0198	0199	LATIN CAPITAL LETTER K WITH HOOK	LATIN SMALL LETTER K WITH HOOK
019C	026f	LATIN CAPITAL LETTER TURNED M	LATIN SMALL LETTER TURNED M
019D	0272	LATIN CAPITAL LETTER N WITH LEFT HOOK	LATIN SMALL LETTER N WITH LEFT HOOK
019F	0275	LATIN CAPITAL LETTER O WITH MIDDLE TILDE	LATIN SMALL LETTER BARRED O
01A0	01A1	LATIN CAPITAL LETTER O WITH HORN	LATIN SMALL LETTER O WITH HORN
01A2	01A3	LATIN CAPITAL LETTER OI	LATIN SMALL LETTER OI
01A4	01A5	LATIN CAPITAL LETTER P WITH HOOK	LATIN SMALL LETTER P WITH HOOK
01A7	01A8	LATIN CAPITAL LETTER TONE TWO	LATIN SMALL LETTER TONE TWO
01A9	0283	LATIN CAPITAL LETTER ESH	LATIN SMALL LETTER ESH
01AC	01AD	LATIN CAPITAL LETTER T WITH HOOK	LATIN SMALL LETTER T WITH HOOK
01AE	0288	LATIN CAPITAL LETTER T WITH RETROFLEX HOOK	LATIN SMALL LETTER T WITH RETROFLEX HOOK
01AF	01B0	LATIN CAPITAL LETTER U WITH HORN	LATIN SMALL LETTER U WITH HORN
01B1	028A	LATIN CAPITAL LETTER UPSILON	LATIN SMALL LETTER UPSILON
01B2	028B	LATIN CAPITAL LETTER V WITH HOOK	LATIN SMALL LETTER V WITH HOOK
01B3	01B4	LATIN CAPITAL LETTER Y WITH HOOK	LATIN SMALL LETTER Y WITH HOOK
01B5	01B6	LATIN CAPITAL LETTER Z WITH STROKE	LATIN SMALL LETTER Z WITH STROKE
01B7	0292	LATIN CAPITAL LETTER EZH	LATIN SMALL LETTER EZH



Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
01B8	01B9	LATIN CAPITAL LETTER EZH REVERSED	LATIN SMALL LETTER EZH REVERSED
01BC	01BD	LATIN CAPITAL LETTER TONE FIVE	LATIN SMALL LETTER TONE FIVE
01C4	01C6	LATIN CAPITAL LETTER DZ WITH CARON	LATIN SMALL LETTER DZ WITH CARON
01C5	01C6	LATIN CAPITAL LETTER D WITH SMALL LETTER Z WITH CARON	LATIN SMALL LETTER DZ WITH CARON
01C7	01C9	LATIN CAPITAL LETTER LJ	LATIN SMALL LETTER LJ
01C8	01C9	LATIN CAPITAL LETTER L WITH SMALL LETTER J	LATIN SMALL LETTER LJ
01CA	01CC	LATIN CAPITAL LETTER NJ	LATIN SMALL LETTER NJ
01CB	01CC	LATIN CAPITAL LETTER N WITH SMALL LETTER J	LATIN SMALL LETTER NJ
01CD	01CE	LATIN CAPITAL LETTER A WITH CARON	LATIN SMALL LETTER A WITH CARON
01CF	01D0	LATIN CAPITAL LETTER I WITH CARON	LATIN SMALL LETTER I WITH CARON
01D1	01D2	LATIN CAPITAL LETTER O WITH CARON	LATIN SMALL LETTER O WITH CARON
01D3	01D4	LATIN CAPITAL LETTER U WITH CARON	LATIN SMALL LETTER U WITH CARON
01D5	01D6	LATIN CAPITAL LETTER U WITH DIAERESIS AND MACRON	LATIN SMALL LETTER U WITH DIAERESIS AND MACRON
01D7	01D8	LATIN CAPITAL LETTER U WITH DIAERESIS AND ACUTE	LATIN SMALL LETTER U WITH DIAERESIS AND ACUTE
01D9	01DA	LATIN CAPITAL LETTER U WITH DIAERESIS AND CARON	LATIN SMALL LETTER U WITH DIAERESIS AND CARON
01DB	01DC	LATIN CAPITAL LETTER U WITH DIAERESIS AND GRAVE	LATIN SMALL LETTER U WITH DIAERESIS AND GRAVE
01DE	01DF	LATIN CAPITAL LETTER A WITH DIAERESIS AND MACRON	LATIN SMALL LETTER A WITH DIAERESIS AND MACRON
01E0	01E1	LATIN CAPITAL LETTER A WITH DOT ABOVE AND MACRON	LATIN SMALL LETTER A WITH DOT ABOVE AND MACRON
01E2	01E3	LATIN CAPITAL LIGATURE AE WITH MACRON	LATIN SMALL LIGATURE AE WITH MACRON
01E4	01E5	LATIN CAPITAL LETTER G WITH STROKE	LATIN SMALL LETTER G WITH STROKE
01E6	01E7	LATIN CAPITAL LETTER G WITH CARON	LATIN SMALL LETTER G WITH CARON
01E8	01E9	LATIN CAPITAL LETTER K WITH CARON	LATIN SMALL LETTER K WITH CARON
01EA	01EB	LATIN CAPITAL LETTER O WITH OGONEK	LATIN SMALL LETTER O WITH OGONEK
01EC	01ED	LATIN CAPITAL LETTER O WITH OGONEK AND MACRON	LATIN SMALL LETTER O WITH OGONEK AND MACRON
01EE	01EF	LATIN CAPITAL LETTER EZH WITH CARON	LATIN SMALL LETTER EZH WITH CARON
01F1	01F3	LATIN CAPITAL LETTER DZ	LATIN SMALL LETTER DZ
01F4	01F5	LATIN CAPITAL LETTER G WITH ACUTE	LATIN SMALL LETTER G WITH ACUTE
01FA	01FB	LATIN CAPITAL LETTER A WITH RING ABOVE AND ACUTE	LATIN SMALL LETTER A WITH RING ABOVE AND ACUTE
01FC	01FD	LATIN CAPITAL LIGATURE AE WITH ACUTE	LATIN SMALL LIGATURE AE WITH ACUTE
01FE	01FF	LATIN CAPITAL LETTER O WITH STROKE AND ACUTE	LATIN SMALL LETTER O WITH STROKE AND ACUTE

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0200	0201	LATIN CAPITAL LETTER A WITH DOUBLE GRAVE	LATIN SMALL LETTER A WITH DOUBLE GRAVE
0202	0203	LATIN CAPITAL LETTER A WITH INVERTED BREVE	LATIN SMALL LETTER A WITH INVERTED BREVE
0204	0205	LATIN CAPITAL LETTER E WITH DOUBLE GRAVE	LATIN SMALL LETTER E WITH DOUBLE GRAVE
0206	0207	LATIN CAPITAL LETTER E WITH INVERTED BREVE	LATIN SMALL LETTER E WITH INVERTED BREVE
0208	0209	LATIN CAPITAL LETTER I WITH DOUBLE GRAVE	LATIN SMALL LETTER I WITH DOUBLE GRAVE
020A	020B	LATIN CAPITAL LETTER I WITH INVERTED BREVE	LATIN SMALL LETTER I WITH INVERTED BREVE
020C	020D	LATIN CAPITAL LETTER O WITH DOUBLE GRAVE	LATIN SMALL LETTER O WITH DOUBLE GRAVE
020E	020F	LATIN CAPITAL LETTER O WITH INVERTED BREVE	LATIN SMALL LETTER O WITH INVERTED BREVE
0210	0211	LATIN CAPITAL LETTER R WITH DOUBLE GRAVE	LATIN SMALL LETTER R WITH DOUBLE GRAVE
0212	0213	LATIN CAPITAL LETTER R WITH INVERTED BREVE	LATIN SMALL LETTER R WITH INVERTED BREVE
0214	0215	LATIN CAPITAL LETTER U WITH DOUBLE GRAVE	LATIN SMALL LETTER U WITH DOUBLE GRAVE
0216	0217	LATIN CAPITAL LETTER U WITH INVERTED BREVE	LATIN SMALL LETTER U WITH INVERTED BREVE
0386	03AC	GREEK CAPITAL LETTER ALPHA WITH TONOS	GREEK SMALL LETTER ALPHA WITH TONOS
0388	03AD	GREEK CAPITAL LETTER EPSILON WITH TONOS	GREEK SMALL LETTER EPSILON WITH TONOS
0389	03AE	GREEK CAPITAL LETTER ETA WITH TONOS	GREEK SMALL LETTER ETA WITH TONOS
038A	03AF	GREEK CAPITAL LETTER IOTA WITH TONOS	GREEK SMALL LETTER IOTA WITH TONOS
038C	03CC	GREEK CAPITAL LETTER OMICRON WITH TONOS	GREEK SMALL LETTER OMICRON WITH TONOS
038E	03CD	GREEK CAPITAL LETTER UPSILON WITH TONOS	GREEK SMALL LETTER UPSILON WITH TONOS
038F	03CE	GREEK CAPITAL LETTER OMEGA WITH TONOS	GREEK SMALL LETTER OMEGA WITH TONOS
0391	03B1	GREEK CAPITAL LETTER ALPHA	GREEK SMALL LETTER ALPHA
0392	03B2	GREEK CAPITAL LETTER BETA	GREEK SMALL LETTER BETA
0393	03B3	GREEK CAPITAL LETTER GAMMA	GREEK SMALL LETTER GAMMA
0394	03B4	GREEK CAPITAL LETTER DELTA	GREEK SMALL LETTER DELTA
0395	03B5	GREEK CAPITAL LETTER EPSILON	GREEK SMALL LETTER EPSILON
0396	03B6	GREEK CAPITAL LETTER ZETA	GREEK SMALL LETTER ZETA
0397	03B7	GREEK CAPITAL LETTER ETA	GREEK SMALL LETTER ETA
0398	03B8	GREEK CAPITAL LETTER THETA	GREEK SMALL LETTER THETA
0399	03B9	GREEK CAPITAL LETTER IOTA	GREEK SMALL LETTER IOTA
039A	03BA	GREEK CAPITAL LETTER KAPPA	GREEK SMALL LETTER KAPPA
039B	03BB	GREEK CAPITAL LETTER LAMDA	GREEK SMALL LETTER LAMDA
039C	03BC	GREEK CAPITAL LETTER MU	GREEK SMALL LETTER MU
039D	03BD	GREEK CAPITAL LETTER NU	GREEK SMALL LETTER NU
039E	03BE	GREEK CAPITAL LETTER XI	GREEK SMALL LETTER XI

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
039F	03BF	GREEK CAPITAL LETTER OMICRON	GREEK SMALL LETTER OMICRON
03A0	03C0	GREEK CAPITAL LETTER PI	GREEK SMALL LETTER PI
03A1	03C1	GREEK CAPITAL LETTER RHO	GREEK SMALL LETTER RHO
03A3	03C3	GREEK CAPITAL LETTER SIGMA	GREEK SMALL LETTER SIGMA
03A4	03C4	GREEK CAPITAL LETTER TAU	GREEK SMALL LETTER TAU
03A5	03C5	GREEK CAPITAL LETTER UPSILON	GREEK SMALL LETTER UPSILON
03A6	03C6	GREEK CAPITAL LETTER PHI	GREEK SMALL LETTER PHI
03A7	03C7	GREEK CAPITAL LETTER CHI	GREEK SMALL LETTER CHI
03A8	03C8	GREEK CAPITAL LETTER PSI	GREEK SMALL LETTER PSI
03A9	03C9	GREEK CAPITAL LETTER OMEGA	GREEK SMALL LETTER OMEGA
03AA	03CA	GREEK CAPITAL LETTER IOTA WITH DIALYTIKA	GREEK SMALL LETTER IOTA WITH DIALYTIKA
03AB	03CB	GREEK CAPITAL LETTER UPSILON WITH DIALYTIKA	GREEK SMALL LETTER UPSILON WITH DIALYTIKA
03E2	03E3	COPTIC CAPITAL LETTER SHEI	COPTIC SMALL LETTER SHEI
03E4	03E5	COPTIC CAPITAL LETTER FEI	COPTIC SMALL LETTER FEI
03E6	03E7	COPTIC CAPITAL LETTER KHEI	COPTIC SMALL LETTER KHEI
03E8	03E9	COPTIC CAPITAL LETTER HORI	COPTIC SMALL LETTER HORI
03EA	03EB	COPTIC CAPITAL LETTER GANGIA	COPTIC SMALL LETTER GANGIA
03EC	03ED	COPTIC CAPITAL LETTER SHIMA	COPTIC SMALL LETTER SHIMA
03EE	03EF	COPTIC CAPITAL LETTER DEI	COPTIC SMALL LETTER DEI
0401	0451	CYRILLIC CAPITAL LETTER IO	CYRILLIC SMALL LETTER IO
0402	0452	CYRILLIC CAPITAL LETTER DJE (SERBOCROATIAN)	CYRILLIC SMALL LETTER DJE (SERBOCROATIAN)
0403	0453	CYRILLIC CAPITAL LETTER GJE	CYRILLIC SMALL LETTER GJE
0404	0454	CYRILLIC CAPITAL LETTER UKRAINIAN IE	CYRILLIC SMALL LETTER UKRAINIAN IE
0405	0455	CYRILLIC CAPITAL LETTER DZE	CYRILLIC SMALL LETTER DZE
0406	0456	CYRILLIC CAPITAL LETTER BYELORUSSIAN_ukrainian I	CYRILLIC SMALL LETTER BYELORUSSIAN-ukrainian I
0407	0457	CYRILLIC CAPITAL LETTER YI (UKRAINIAN)	CYRILLIC SMALL LETTER YI (UKRAINIAN)
0408	0458	CYRILLIC CAPITAL LETTER JE	CYRILLIC SMALL LETTER JE
0409	0459	CYRILLIC CAPITAL LETTER LJE	CYRILLIC SMALL LETTER LJE
040A	045A	CYRILLIC CAPITAL LETTER NJE	CYRILLIC SMALL LETTER NJE
040B	045B	CYRILLIC CAPITAL LETTER TSHE (SERBOCROATIAN)	CYRILLIC SMALL LETTER TSHE (SERBOCROATIAN)
040C	045C	CYRILLIC CAPITAL LETTER KJE	CYRILLIC SMALL LETTER KJE
040E	045E	CYRILLIC CAPITAL LETTER SHORT U (BYELORUSSIAN)	CYRILLIC SMALL LETTER SHORT U (BYELORUSSIAN)
040F	045F	CYRILLIC CAPITAL LETTER DZHE	CYRILLIC SMALL LETTER DZHE
0410	0430	CYRILLIC CAPITAL LETTER A	CYRILLIC SMALL LETTER A
0411	0431	CYRILLIC CAPITAL LETTER BE	CYRILLIC SMALL LETTER BE
0412	0432	CYRILLIC CAPITAL LETTER VE	CYRILLIC SMALL LETTER VE
0413	0433	CYRILLIC CAPITAL LETTER GHE	CYRILLIC SMALL LETTER GHE
0414	0434	CYRILLIC CAPITAL LETTER DE	CYRILLIC SMALL LETTER DE
0415	0435	CYRILLIC CAPITAL LETTER IE	CYRILLIC SMALL LETTER IE

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0416	0436	CYRILLIC CAPITAL LETTER ZHE	CYRILLIC SMALL LETTER ZHE
0417	0437	CYRILLIC CAPITAL LETTER ZE	CYRILLIC SMALL LETTER ZE
0418	0438	CYRILLIC CAPITAL LETTER I	CYRILLIC SMALL LETTER I
0419	0439	CYRILLIC CAPITAL LETTER SHORT I	CYRILLIC SMALL LETTER SHORT I
041A	043A	CYRILLIC CAPITAL LETTER KA	CYRILLIC SMALL LETTER KA
041B	043B	CYRILLIC CAPITAL LETTER EL	CYRILLIC SMALL LETTER EL
041C	043C	CYRILLIC CAPITAL LETTER EM	CYRILLIC SMALL LETTER EM
041D	043D	CYRILLIC CAPITAL LETTER EN	CYRILLIC SMALL LETTER EN
041E	043E	CYRILLIC CAPITAL LETTER O	CYRILLIC SMALL LETTER O
041F	043F	CYRILLIC CAPITAL LETTER PE	CYRILLIC SMALL LETTER PE
0420	0440	CYRILLIC CAPITAL LETTER ER	CYRILLIC SMALL LETTER ER
0421	0441	CYRILLIC CAPITAL LETTER ES	CYRILLIC SMALL LETTER ES
0422	0442	CYRILLIC CAPITAL LETTER TE	CYRILLIC SMALL LETTER TE
0423	0443	CYRILLIC CAPITAL LETTER U	CYRILLIC SMALL LETTER U
0424	0444	CYRILLIC CAPITAL LETTER EF	CYRILLIC SMALL LETTER EF
0425	0445	CYRILLIC CAPITAL LETTER HA	CYRILLIC SMALL LETTER HA
0426	0446	CYRILLIC CAPITAL LETTER TSE	CYRILLIC SMALL LETTER TSE
0427	0447	CYRILLIC CAPITAL LETTER CHE	CYRILLIC SMALL LETTER CHE
0428	0448	CYRILLIC CAPITAL LETTER SHA	CYRILLIC SMALL LETTER SHA
0429	0449	CYRILLIC CAPITAL LETTER SHCHA	CYRILLIC SMALL LETTER SHCHA
042A	044A	CYRILLIC CAPITAL LETTER HARD SIGN	CYRILLIC SMALL LETTER HARD SIGN
042B	044B	CYRILLIC CAPITAL LETTER YERU	CYRILLIC SMALL LETTER YERU
042C	044C	CYRILLIC CAPITAL LETTER SOFT SIGN	CYRILLIC SMALL LETTER SOFT SIGN
042D	044D	CYRILLIC CAPITAL LETTER E	CYRILLIC SMALL LETTER E
042E	044E	CYRILLIC CAPITAL LETTER YU	CYRILLIC SMALL LETTER YU
042F	044F	CYRILLIC CAPITAL LETTER YA	CYRILLIC SMALL LETTER YA
0460	0461	CYRILLIC CAPITAL LETTER OMEGA	CYRILLIC SMALL LETTER OMEGA
0462	0463	CYRILLIC CAPITAL LETTER YAT	CYRILLIC SMALL LETTER YAT
0464	0465	CYRILLIC CAPITAL LETTER IOTIFIED E	CYRILLIC SMALL LETTER IOTIFIED E
0466	0467	CYRILLIC CAPITAL LETTER LITTLE YUS	CYRILLIC SMALL LETTER LITTLE YUS
0468	0469	CYRILLIC CAPITAL LETTER IOTIFIED LITTLE YUS	CYRILLIC SMALL LETTER IOTIFIED LITTLE YUS
046A	046B	CYRILLIC CAPITAL LETTER BIG YUS	CYRILLIC SMALL LETTER BIG YUS
046C	046D	CYRILLIC CAPITAL LETTER IOTIFIED BIG YUS	CYRILLIC SMALL LETTER IOTIFIED BIG YUS
046E	046F	CYRILLIC CAPITAL LETTER KSI	CYRILLIC SMALL LETTER KSI
0470	0471	CYRILLIC CAPITAL LETTER PSI	CYRILLIC SMALL LETTER PSI
0472	0473	CYRILLIC CAPITAL LETTER FITA	CYRILLIC SMALL LETTER FITA
0474	0475	CYRILLIC CAPITAL LETTER IZHITSA	CYRILLIC SMALL LETTER IZHITSA
0476	0477	CYRILLIC CAPITAL LETTER IZHITSA WITH DOUBLE GRAVE ACCENT	CYRILLIC SMALL LETTER IZHITSA WITH DOUBLE GRAVE ACCENT

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
0478	0479	CYRILLIC CAPITAL LETTER UK	CYRILLIC SMALL LETTER UK
047A	047B	CYRILLIC CAPITAL LETTER ROUND OMEGA	CYRILLIC SMALL LETTER ROUND OMEGA
047C	047D	CYRILLIC CAPITAL LETTER OMEGA WITH TITLO	CYRILLIC SMALL LETTER OMEGA WITH TITLO
047E	047F	CYRILLIC CAPITAL LETTER OT	CYRILLIC SMALL LETTER OT
0480	0481	CYRILLIC CAPITAL LETTER KOPPA	CYRILLIC SMALL LETTER KOPPA
0490	0491	CYRILLIC CAPITAL LETTER GHE WITH UPTURN	CYRILLIC SMALL LETTER GHE WITH UPTURN
0492	0493	CYRILLIC CAPITAL LETTER GHE WITH STROKE	CYRILLIC SMALL LETTER GHE WITH STROKE
0494	0495	CYRILLIC CAPITAL LETTER GHE WITH MIDDLE HOOK	CYRILLIC SMALL LETTER GHE WITH MIDDLE HOOK
0496	0497	CYRILLIC CAPITAL LETTER ZHE WITH DESCENDER	CYRILLIC SMALL LETTER ZHE WITH DESCENDER
0498	0499	CYRILLIC CAPITAL LETTER ZE WITH DESCENDER	CYRILLIC SMALL LETTER ZE WITH DESCENDER
049A	049B	CYRILLIC CAPITAL LETTER KA WITH DESCENDER	CYRILLIC SMALL LETTER KA WITH DESCENDER
049C	049D	CYRILLIC CAPITAL LETTER KA WITH VERTICAL STROKE	CYRILLIC SMALL LETTER KA WITH VERTICAL STROKE
049E	049F	CYRILLIC CAPITAL LETTER KA WITH STROKE	CYRILLIC SMALL LETTER KA WITH STROKE
04A0	04A1	CYRILLIC CAPITAL LETTER BASHKIR KA	CYRILLIC SMALL LETTER EASHKIR KA
04A2	04A3	CYRILLIC CAPITAL LETTER EN WITH DESCENDER	CYRILLIC SMALL LETTER EN WITH DESCENDER
04A4	04A5	CYRILLIC CAPITAL LIGATURE EN GHF	CYRILLIC SMALL LIGATURE EN GHE
04A6	04A7	CYRILLIC CAPITAL LETTER PE WITH MIDDLE HOOK (ABKHASIAN)	CYRILLIC SMALL LETTER PE WITH MIDDLE HOOK (ABKHASIAN)
04A8	04A9	CYRILLIC CAPITAL LETTER ABKHASIAN HA	CYRILLIC SMALL LETTER ABKHASIAN HA
04AA	04AB	CYRILLIC CAPITAL LETTER ES WITH DESCENDER	CYRILLIC SMALL LETTER ES WITH DESCENDER
04AC	04AD	CYRILLIC CAPITAL LETTER TE WITH DESCENDER	CYRILLIC SMALL LETTER TE WITH DESCENDER
04AE	04AF	CYRILLIC CAPITAL LETTER STRAIGHT U	CYRILLIC SMALL LETTER STRAIGHT U
04B0	04B1	CYRILLIC CAPITAL LETTER STRAIGHT U WITH STROKE	CYRILLIC SMALL LETTER STRAIGHT U WITH STROKE
04B2	04B3	CYRILLIC CAPITAL LETTER HA WITH DESCENDER	CYRILLIC SMALL LETTER HA WITH DESCENDER
04B4	04B5	CYRILLIC CAPITAL LIGATURE TE TSE (ABKHASIAN)	CYRILLIC SMALL LIGATURE TE TSE (ABKHASIAN)
04B6	04B7	CYRILLIC CAPITAL LETTER CHE WITH DESCENDER	CYRILLIC SMALL LETTER CHE WITH DESCENDER
04B8	04B9	CYRILLIC CAPITAL LETTER CHE WITH VERTICAL STROKE	CYRILLIC SMALL LETTER CHE WITH VERTICAL STROKE
04BA	04BB	CYRILLIC CAPITAL LETTER SHHA	CYRILLIC SMALL LETTER SHHA
04BC	04BD	CYRILLIC CAPITAL LETTER ABKHASIAN CHE	CYRILLIC SMALL LETTER ABKHASIAN CHE
04BE	04BF	CYRILLIC CAPITAL LETTER ABKHASIAN CHE WITH DESCENDER	CYRILLIC SMALL LETTER ABKHASIAN CHE WITH DESCENDER

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
04C1	04C2	CYRILLIC CAPITAL LETTER ZHE WITH BREVE	CYRILLIC SMALL LETTER ZHE WITH BREVE
04C3	04C4	CYRILLIC CAPITAL LETTER KA WITH HOOK	CYRILLIC SMALL LETTER KA WITH HOOK
04C7	04C8	CYRILLIC CAPITAL LETTER EN WITH HOOK	CYRILLIC SMALL LETTER EN WITH HOOK
04CB	04CC	CYRILLIC CAPITAL LETTER KHAKASSIAN CHE	CYRILLIC SMALL LETTER KHAKASSIAN CHE
04D0	04D1	CYRILLIC CAPITAL LETTER A WITH BREVE	CYRILLIC SMALL LETTER A WITH BREVE
04D2	04D3	CYRILLIC CAPITAL LETTER A WITH DIAERESIS	CYRILLIC SMALL LETTER A WITH DIAERESIS
04D4	04D5	CYRILLIC CAPITAL LIGATURE A IE	CYRILLIC SMALL LIGATURE A IE
04D6	04D7	CYRILLIC CAPITAL LETTER IE WITH BREVE	CYRILLIC SMALL LETTER IE WITH BREVE
04D8	04D9	CYRILLIC CAPITAL LETTER SCHWA	CYRILLIC SMALL LETTER SCHWA
04DA	04DB	CYRILLIC CAPITAL LETTER SCHWA WITH DIAERESIS	CYRILLIC SMALL LETTER SCHWA WITH DIAERESIS
04DC	04DD	CYRILLIC CAPITAL LETTER ZHE WITH DIAERESIS	CYRILLIC SMALL LETTER ZHE WITH DIAERESIS
04DE	04DF	CYRILLIC CAPITAL LETTER ZE WITH DIAERESIS	CYRILLIC SMALL LETTER ZE WITH DIAERESIS
04E0	04E1	CYRILLIC CAPITAL LETTER ABKHASIAN DZE	CYRILLIC SMALL LETTER ABKHASIAN DZE
04E2	04E3	CYRILLIC CAPITAL LETTER I WITH MACRON	CYRILLIC SMALL LETTER I WITH MACRON
04E4	04E5	CYRILLIC CAPITAL LETTER I WITH DIAERESIS	CYRILLIC SMALL LETTER I WITH DIAERESIS
04E6	04E7	CYRILLIC CAPITAL LETTER O WITH DIAERESIS	CYRILLIC SMALL LETTER O WITH DIAERESIS
04E8	04E9	CYRILLIC CAPITAL LETTER BARRED O	CYRILLIC SMALL LETTER BARRED O
04EA	04EB	CYRILLIC CAPITAL LETTER BARRED O WITH DIAERESIS	CYRILLIC SMALL LETTER BARRED O WITH DIAERESIS
04EE	04EF	CYRILLIC CAPITAL LETTER U WITH MACRON	CYRILLIC SMALL LETTER U WITH MACRON
04F0	04F1	CYRILLIC CAPITAL LETTER U WITH DIAERESIS	CYRILLIC SMALL LETTER U WITH DIAERESIS
04F2	04F3	CYRILLIC CAPITAL LETTER U WITH DOUBLE ACUTE	CYRILLIC SMALL LETTER U WITH DOUBLE ACUTE
04F4	04F5	CYRILLIC CAPITAL LETTER CHE WITH DIAERESIS	CYRILLIC SMALL LETTER CHE WITH DIAERESIS
04F8	04F9	CYRILLIC CAPITAL LETTER YERU WITH DIAERESIS	CYRILLIC SMALL LETTER YERU WITH DIAERESIS
0531	0561	ARMENIAN CAPITAL LETTER AYB	ARMENIAN SMALL LETTER AYB
0532	0562	ARMENIAN CAPITAL LETTER BEN	ARMENIAN SMALL LETTER BEN
0533	0563	ARMENIAN CAPITAL LETTER GIM	ARMENIAN SMALL LETTER GIM
0534	0564	ARMENIAN CAPITAL LETTER DA	ARMENIAN SMALL LETTER DA
0535	0565	ARMENIAN CAPITAL LETTER ECH	ARMENIAN SMALL LETTER ECH
0536	0566	ARMENIAN CAPITAL LETTER ZA	ARMENIAN SMALL LETTER ZA
0537	0567	ARMENIAN CAPITAL LETTER EH	ARMENIAN SMALL LETTER EH
0538	0568	ARMENIAN CAPITAL LETTER ET	ARMENIAN SMALL LETTER ET
0539	0569	ARMENIAN CAPITAL LETTER TO	ARMENIAN SMALL LETTER TO

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
053A	056A	ARMENIAN CAPITAL LETTER ZHE	ARMENIAN SMALL LETTER ZHE
053B	056B	ARMENIAN CAPITAL LETTER INI	ARMENIAN SMALL LETTER INI
053C	056C	ARMENIAN CAPITAL LETTER LIWN	ARMENIAN SMALL LETTER LIWN
053D	056D	ARMENIAN CAPITAL LETTER XEH	ARMENIAN SMALL LETTER XEH
053E	056E	ARMENIAN CAPITAL LETTER CA	ARMENIAN SMALL LETTER CA
053F	056F	ARMENIAN CAPITAL LETTER KEN	ARMENIAN SMALL LETTER KEN
0540	0570	ARMENIAN CAPITAL LETTER HO	ARMENIAN SMALL LETTER HO
0541	0571	ARMENIAN CAPITAL LETTER JA	ARMENIAN SMALL LETTER JA
0542	0572	ARMENIAN CAPITAL LETTER GHAD	ARMENIAN SMALL LETTER GHAD
0543	0573	ARMENIAN CAPITAL LETTER CHEH	ARMENIAN SMALL LETTER CHEH
0544	0574	ARMENIAN CAPITAL LETTER MEN	ARMENIAN SMALL LETTER MEN
0545	0575	ARMENIAN CAPITAL LETTER YI	ARMENIAN SMALL LETTER YI
0546	0576	ARMENIAN CAPITAL LETTER NOW	ARMENIAN SMALL LETTER NOW
0547	0577	ARMENIAN CAPITAL LETTER SHA	ARMENIAN SMALL LETTER SNA
0548	0578	ARMENIAN CAPITAL LETTER VO	ARMENIAN SMALL LETTER VO
0549	0579	ARMENIAN CAPITAL LETTER CHA	ARMENIAN SMALL LETTER CHA
054A	057A	ARMENIAN CAPITAL LETTER PEH	ARMENIAN SMALL LETTER PEH
054B	057B	ARMENIAN CAPITAL LETTER JHEH	ARMENIAN SMALL LETTER JHEH
054C	057C	ARMENIAN CAPITAL LETTER RA	ARMENIAN SMALL LETTER RA
054D	057D	ARMENIAN CAPITAL LETTER SEH	ARMENIAN SMALL LETTER SEH
054E	057E	ARMENIAN CAPITAL LETTER VEV	ARMENIAN SMALL LETTER VEV
054F	057F	ARMENIAN CAPITAL LETTER TIWN	ARMENIAN SMALL LETTER TIWN
0550	0580	ARMENIAN CAPITAL LETTER REH	ARMENIAN SMALL LETTER REH
0551	0581	ARMENIAN CAPITAL LETTER CO	ARMENIAN SMALL LETTER CO
0552	0582	ARMENIAN CAPITAL LETTER YIWN	ARMENIAN SMALL LETTER YIWN
0553	0583	ARMENIAN CAPITAL LETTER PIWR	ARMENIAN SMALL LETTER PIWP
0554	0584	ARMENIAN CAPITAL LETTER KEH	ARMENIAN SMALL LETTER KEH
0555	0585	ARMENIAN CAPITAL LETTER OH	ARMENIAN SMALL LETTER OH
0556	0586	ARMENIAN CAPITAL LETTER FEH	ARMENIAN SMALL LETTER FEH
10A0	10D0	GEORGIAN CAPITAL LETTER AN (KHUTSURI)	GEORGIAN LETTER AN
10A1	10D1	GEORGIAN CAPITAL LETTER BAN (KHUTSURI)	GEORGIAN LETTER BAN
10A2	10D2	GEORGIAN CAPITAL LETTER GAN (KHUTSURI)	GEORGIAN LETTER GAN
10A3	10D3	GEORGIAN CAPITAL LETTER DON (KHUTSURI)	GEORGIAN LETTER DON
10A4	10D4	GEORGIAN CAPITAL LETTER EN (KHUTSURI)	GEORGIAN LETTER EN
10A5	10D5	GEORGIAN CAPITAL LETTER VIN (KHUTSURI)	GEORGIAN LETTER VIN
10A6	10D6	GEORGIAN CAPITAL LETTER ZEN (KHUTSURI)	GEORGIAN LETTER ZEN
10A7	10D7	GEORGIAN CAPITAL LETTER TAN (KHUTSURI)	GEORGIAN LETTER TAN
10A8	10D8	GEORGIAN CAPITAL LETTER IN (KHUTSURI)	GEORGIAN LETTER IN

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
10A9	10D9	GEORGIAN CAPITAL LETTER KAN (KHUTSURI)	GEORGIAN LETTER KAN
10AA	10DA	GEORGIAN CAPITAL LETTER LAS (KHUTSURI)	GEORGIAN LETTER LAS
10AB	10DB	GEORGIAN CAPITAL LETTER MAN (KHUTSURI)	GEORGIAN LETTER MAN
10AC	10DC	GEORGIAN CAPITAL LETTER NAR (KHUTSURI)	GEORGIAN LETTER NAR
10AD	10DD	GEORGIAN CAPITAL LETTER ON (KHUTSURI)	GEORGIAN LETTER ON
10AE	10DE	GEORGIAN CAPITAL LETTER PAR (KHUTSURI)	GEORGIAN LETTER PAR
10AF	10DF	GEORGIAN CAPITAL LETTER ZHAR (KHUTSURI)	GEORGIAN LETTER ZHAR
10B0	10E0	GEORGIAN CAPITAL LETTER RAE (KHUTSURI)	GEORGIAN LETTER RAE
10B1	10E1	GEORGIAN CAPITAL LETTER SAN (KHUTSURI)	GEORGIAN LETTER SAN
10B2	10E2	GEORGIAN CAPITAL LETTER TAR (KHUTSURI)	GEORGIAN LETTER TAR
10B3	10E3	GEORGIAN CAPITAL LETTER UN (KHUTSURI)	GEORGIAN LETTER UN
10B4	10E4	GEORGIAN CAPITAL LETTER PHAR (KHUTSURI)	GEORGIAN LETTER PHAR
10B5	10E5	GEORGIAN CAPITAL LETTER KHAR (KHUTSURI)	GEORGIAN LETTER KHAR
10B6	10E6	GEORGIAN CAPITAL LETTER GHAN (KHUTSURI)	GEORGIAN LETTER GHAN
10B7	10E7	GEORGIAN CAPITAL LETTER QAR (KHUTSURI)	GEORGIAN LETTER QAR
10B8	10E8	GEORGIAN CAPITAL LETTER SHIN (KHUTSURI)	GEORGIAN LETTER SHIN
10B9	10E9	GEORGIAN CAPITAL LETTER CHIN (KHUTSURI)	GEORGIAN LETTER CHIN
10BA	10EA	GEORGIAN CAPITAL LETTER CAN (KHUTSURI)	GEORGIAN LETTER CAN
10BB	10EB	GEORGIAN CAPITAL LETTER JIL (KHUTSURI)	GEORGIAN LETTER JIL
10BC	10EC	GEORGIAN CAPITAL LETTER CIL (KHUTSURI)	GEORGIAN LETTER CIL
10BD	10ED	GEORGIAN CAPITAL LETTER CHAR (KHUTSURI)	GEORGIAN LETTER CHAR
10BE	10EE	GEORGIAN CAPITAL LETTER XAN (KHUTSURI)	GEORGIAN LETTER XAN
10BF	10EF	GEORGIAN CAPITAL LETTER JHAN (KHUTSURI)	GEORGIAN LETTER JHAN
10C0	10F0	GEORGIAN CAPITAL LETTER HAE (KHUTSURI)	GEORGIAN LETTER HAE
10C1	10F1	GEORGIAN CAPITAL LETTER HE (KHUTSURI)	GEORGIAN LETTER HE
10C2	10F2	GEORGIAN CAPITAL LETTER HIE (KHUTSURI)	GEORGIAN LETTER HIE
10C3	10F3	GEORGIAN CAPITAL LETTER WE (KHUTSURI)	GEORGIAN LETTER WE
10C4	10F4	GEORGIAN CAPITAL LETTER HAR (KHUTSURI)	GEORGIAN LETTER HAR



Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
10C5	10F5	GEORGIAN CAPITAL LETTER HOE (KHUTSURI)	GEORGIAN LETTER HOE
1E00	1E01	LATIN CAPITAL LETTER A WITH RING BELOW	LATIN SMALL LETTER A WITH RING BELOW
1E02	1E03	LATIN CAPITAL LETTER B WITH DOT ABOVE	LATIN SMALL LETTER B WITH DOT ABOVE
1E04	1E05	LATIN CAPITAL LETTER B WITH DOT BELOW	LATIN SMALL LETTER B WITH DOT BELOW
1E06	1E07	LATIN CAPITAL LETTER B WITH LINE BELOW	LATIN SMALL LETTER B WITH LINE BELOW
1E08	1E09	LATIN CAPITAL LETTER C WITH CEDILLA AND ACUTE	LATIN SMALL LETTER C WITH CEDILLA AND ACUTE
1E0A	1E0B	LATIN CAPITAL LETTER D WITH DOT ABOVE	LATIN SMALL LETTER D WITH DOT ABOVE
1E0C	1E0D	LATIN CAPITAL LETTER D WITH DOT BELOW	LATIN SMALL LETTER D WITH DOT BELOW
1E0E	1E0F	LATIN CAPITAL LETTER D WITH LINE BELOW	LATIN SMALL LETTER D WITH LINE BELOW
1E10	1E11	LATIN CAPITAL LETTER D WITH CEDILLA	LATIN SMALL LETTER D WITH CEDILLA
1E12	1E13	LATIN CAPITAL LETTER D WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER D WITH CIRCUMFLEX BELOW
1E14	1E15	LATIN CAPITAL LETTER E WITH MACRON AND GRAVE	LATIN SMALL LETTER E WITH MACRON AND GRAVE
1E16	1E17	LATIN CAPITAL LETTER E WITH MACRON AND ACUTE	LATIN SMALL LETTER E WITH MACRON AND ACUTE
1E18	1E19	LATIN CAPITAL LETTER E WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER E WITH CIRCUMFLEX BELOW
1E1A	1E1B	LATIN CAPITAL LETTER E WITH TILDE BELOW	LATIN SMALL LETTER E WITH TILDE BELOW
1E1C	1E1D	LATIN CAPITAL LETTER E WITH CEDILLA AND BREVE	LATIN SMALL LETTER E WITH CEDILLA AND BREVE
1E1E	1E1F	LATIN CAPITAL LETTER F WITH DOT ABOVE	LATIN SMALL LETTER F WITH DOT ABOVE
1E20	1E21	LATIN CAPITAL LETTER G WITH MACRON	LATIN SMALL LETTER G WITH MACRON
1E22	1E23	LATIN CAPITAL LETTER H WITH DOT ABOVE	LATIN SMALL LETTER H WITH DOT ABOVE
1E24	1E25	LATIN CAPITAL LETTER H WITH DOT BELOW	LATIN SMALL LETTER H WITH DOT BELOW
1E26	1E27	LATIN CAPITAL LETTER H WITH DIAERESIS	LATIN SMALL LETTER H WITH DIAERESIS
1E28	1E29	LATIN CAPITAL LETTER H WITH CEDILLA	LATIN SMALL LETTER H WITH CEDILLA
1E2A	1E2B	LATIN CAPITAL LETTER H WITH BREVE BELOW	LATIN SMALL LETTER H WITH BREVE BELOW
1E2C	1E2D	LATIN CAPITAL LETTER I WITH TILDE BELOW	LATIN SMALL LETTER I WITH TILDE BELOW
1E2E	1E2F	LATIN CAPITAL LETTER I WITH DIAERESIS AND ACUTE	LATIN SMALL LETTER I WITH DIAERESIS AND ACUTE
1E30	1E31	LATIN CAPITAL LETTER K WITH ACUTE	LATIN SMALL LETTER K WITH ACUTE
1E32	1E33	LATIN CAPITAL LETTER K WITH DOT BELOW	LATIN SMALL LETTER K WITH DOT BELOW
1E34	1E35	LATIN CAPITAL LETTER K WITH LINE BELOW	LATIN SMALL LETTER K WITH LINE BELOW

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1E36	1E37	LATIN CAPITAL LETTER L WITH DOT BELOW	LATIN SMALL LETTER L WITH DOT BELOW
1E38	1E39	LATIN CAPITAL LETTER L WITH DOT BELOW AND MACRON	LATIN SMALL LETTER L WITH DOT BELOW AND MACRON
1E3A	1E3B	LATIN CAPITAL LETTER L WITH LINE BELOW	LATIN SMALL LETTER L WITH LINE BELOW
1E3C	1E3D	LATIN CAPITAL LETTER L WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER L WITH CIRCUMFLEX BELOW
1E3E	1E3F	LATIN CAPITAL LETTER M WITH ACUTE	LATIN SMALL LETTER M WITH ACUTE
1E40	1E41	LATIN CAPITAL LETTER M WITH DOT ABOVE	LATIN SMALL LETTER M WITH DOT ABOVE
1E42	1E43	LATIN CAPITAL LETTER M WITH DOT BELOW	LATIN SMALL LETTER M WITH DOT BELOW
1E44	1E45	LATIN CAPITAL LETTER N WITH DOT ABOVE	LATIN SMALL LETTER N WITH DOT ABOVE
1E46	1E47	LATIN CAPITAL LETTER N WITH DOT BELOW	LATIN SMALL LETTER N WITH DOT BELOW
1E48	1E49	LATIN CAPITAL LETTER N WITH LINE BELOW	LATIN SMALL LETTER N WITH LINE BELOW
1E4A	1E4B	LATIN CAPITAL LETTER N WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER N WITH CIRCUMFLEX BELOW
1E4C	1E4D	LATIN CAPITAL LETTER O WITH TILDE AND ACUTE	LATIN SMALL LETTER O WITH TILDE AND ACUTE
1E4E	1E4F	LATIN CAPITAL LETTER O WITH TILDE AND DIAERESIS	LATIN SMALL LETTER O WITH TILDE AND DIAERESIS
1E50	1E51	LATIN CAPITAL LETTER O WITH MACRON AND GRAVE	LATIN SMALL LETTER O WITH MACRON AND GRAVE
1E52	1E53	LATIN CAPITAL LETTER O WITH MACRON AND ACUTE	LATIN SMALL LETTER O WITH MACRON AND ACUTE
1E54	1E55	LATIN CAPITAL LETTER P WITH ACUTE	LATIN SMALL LETTER P WITH ACUTE
1E56	1E57	LATIN CAPITAL LETTER P WITH DOT ABOVE	LATIN SMALL LETTER P WITH DOT ABOVE
1E58	1E59	LATIN CAPITAL LETTER R WITH DOT ABOVE	LATIN SMALL LETTER R WITH DOT ABOVE
1E5A	1E5B	LATIN CAPITAL LETTER R WITH DOT BELOW	LATIN SMALL LETTER R WITH DOT BELOW
1E5C	1E5D	LATIN CAPITAL LETTER R WITH DOT BELOW AND MACRON	LATIN SMALL LETTER R WITH DOT BELOW AND MACRON
1E5E	1E5F	LATIN CAPITAL LETTER R WITH LINE BELOW	LATIN SMALL LETTER R WITH LINE BELOW
1E60	1E61	LATIN CAPITAL LETTER S WITH DOT ABOVE	LATIN SMALL LETTER S WITH DOT ABOVE
1E62	1E63	LATIN CAPITAL LETTER S WITH DOT BELOW	LATIN SMALL LETTER S WITH DOT BELOW
1E64	1E65	LATIN CAPITAL LETTER S WITH ACUTE AND DOT ABOVE	LATIN SMALL LETTER S WITH ACUTE AND DOT ABOVE
1E66	1E67	LATIN CAPITAL LETTER S WITH CARON AND DOT ABOVE	LATIN SMALL LETTER S WITH CARON AND DOT ABOVE
1E68	1E69	LATIN CAPITAL LETTER S WITH DOT BELOW AND DOT ABOVE	LATIN SMALL LETTER S WITH DOT BELOW AND DOT ABOVE
1E6A	1E6B	LATIN CAPITAL LETTER T WITH DOT ABOVE	LATIN SMALL LETTER T WITH DOT ABOVE
1E6C	1E6D	LATIN CAPITAL LETTER T WITH DOT BELOW	LATIN SMALL LETTER T WITH DOT BELOW

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1E6E	1E6F	LATIN CAPITAL LETTER T WITH LINE BELOW	LATIN SMALL LETTER T WITH LINE BELOW
1E70	1E71	LATIN CAPITAL LETTER T WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER T WITH CIRCUMFLEX BELOW
1E72	1E73	LATIN CAPITAL LETTER U WITH DIAERESIS BELOW	LATIN SMALL LETTER U WITH DIAERESIS BELOW
1E74	1E75	LATIN CAPITAL LETTER U WITH TILDE BELOW	LATIN SMALL LETTER U WITH TILDE BELOW
1E76	1E77	LATIN CAPITAL LETTER U WITH CIRCUMFLEX BELOW	LATIN SMALL LETTER U WITH CIRCUMFLEX BELOW
1E78	1E79	LATIN CAPITAL LETTER U WITH TILDE AND ACUTE	LATIN SMALL LETTER U WITH TILDE AND ACUTE
1E7A	1E7B	LATIN CAPITAL LETTER U WITH MACRON AND DIAERESIS	LATIN SMALL LETTER U WITH MACRON AND DIAERESIS
1E7C	1E7D	LATIN CAPITAL LETTER V WITH TILDE	LATIN SMALL LETTER V WITH TILDE
1E7E	1E7F	LATIN CAPITAL LETTER V WITH DOT BELOW	LATIN SMALL LETTER V WITH DOT BELOW
1E80	1E81	LATIN CAPITAL LETTER W WITH GRAVE	LATIN SMALL LETTER W WITH GRAVE
1E82	1E83	LATIN CAPITAL LETTER W WITH ACUTE	LATIN SMALL LETTER W WITH ACUTE
1E84	1E85	LATIN CAPITAL LETTER W WITH DIAERESIS	LATIN SMALL LETTER W WITH DIAERESIS
1E86	1E87	LATIN CAPITAL LETTER W WITH DOT ABOVE	LATIN SMALL LETTER W WITH DOT ABOVE
1E88	1E89	LATIN CAPITAL LETTER W WITH DOT BELOW	LATIN SMALL LETTER W WITH DOT BELOW
1E8A	1E8B	LATIN CAPITAL LETTER X WITH DOT ABOVE	LATIN SMALL LETTER X WITH DOT ABOVE
1E8C	1E8D	LATIN CAPITAL LETTER X5 WITH DIAERESIS	LATIN SMALL LETTER X WITH DIAERESIS
1E8E	1E8F	LATIN CAPITAL LETTER Y WITH DOT ABOVE	LATIN SMALL LETTER Y WITH DOT ABOVE
1E90	1E91	LATIN CAPITAL LETTER Z WITH CIRCUMFLEX	LATIN SMALL LETTER Z WITH CIRCUMFLEX
1E92	1E93	LATIN CAPITAL LETTER Z WITH DOT BELOW	LATIN SMALL LETTER Z WITH DOT BELOW
1E94	1E95	LATIN CAPITAL LETTER Z WITH LINE BELOW	LATIN SMALL LETTER Z WITH LINE BELOW
1EA0	1EA1	LATIN CAPITAL LETTER A WITH DOT BELOW	LATIN SMALL LETTER A WITH DOT BELOW
1EA2	1EA3	LATIN CAPITAL LETTER A WITH HOOK ABOVE	LATIN SMALL LETTER A WITH HOOK ABOVE
1EA4	1EA5	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND ACUTE	LATIN SMALL LETTER A WITH CIRCUMFLEX AND ACUTE
1EA6	1EA7	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND GRAVE	LATIN SMALL LETTER A WITH CIRCUMFLEX AND GRAVE
1EA8	1EA9	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE	LATIN SMALL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
1EAA	1EAB	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND TILDE	LATIN SMALL LETTER A WITH CIRCUMFLEX AND TILDE
1EAC	1EAD	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND DOT BELOW	LATIN SMALL LETTER A WITH CIRCUMFLEX AND DOT BELOW
1EAE	1EAF	LATIN CAPITAL LETTER A WITH BREVE AND ACUTE	LATIN SMALL LETTER A WITH BREVE AND ACUTE

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1EB0	1EB1	LATIN CAPITAL LETTER A WITH BREVE AND GRAVE	LATIN SMALL LETTER A WITH BREVE AND GRAVE
1EB2	1EB3	LATIN CAPITAL LETTER A WITH BREVE AND HOOK ABOVE	LATIN SMALL LETTER A WITH BREVE AND HOOK ABOVE
1EB4	1EB5	LATIN CAPITAL LETTER A WITH BREVE AND TILDE	LATIN SMALL LETTER A WITH BREVE AND TILDE
1EB6	1EB7	LATIN CAPITAL LETTER A WITH BREVE AND DOT BELOW	LATIN SMALL LETTER A WITH BREVE AND DOT BELOW
1EB8	1EB9	LATIN CAPITAL LETTER E WITH DOT BELOW	LATIN SMALL LETTER E WITH DOT BELOW
1EBA	1EBB	LATIN CAPITAL LETTER E WITH HOOK ABOVE	LATIN SMALL LETTER E WITH HOOK ABOVE
1EBC	1EBD	LATIN CAPITAL LETTER E WITH TILDE	LATIN SMALL LETTER E WITH TILDE
1EBE	1EBF	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND ACUTE	LATIN SMALL LETTER E WITH CIRCUMFLEX AND ACUTE
1EC0	1EC1	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND GRAVE	LATIN SMALL LETTER E WITH CIRCUMFLEX AND GRAVE
1EC2	1EC3	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE	LATIN SMALL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
1EC4	1EC5	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND TILDE	LATIN SMALL LETTER E WITH CIRCUMFLEX AND TILDE
1EC6	1EC7	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND DOT BELOW	LATIN SMALL LETTER E WITH CIRCUMFLEX AND DOT BELOW
1EC8	1EC9	LATIN CAPITAL LETTER I WITH HOOK ABOVE	LATIN SMALL LETTER I WITH HOOK ABOVE
1ECA	1ECB	LATIN CAPITAL LETTER I WITH DOT BELOW	LATIN SMALL LETTER I WITH DOT BELOW
1ECC	1ECD	LATIN CAPITAL LETTER O WITH DOT BELOW	LATIN SMALL LETTER O WITH DOT BELOW
1ECE	1ECF	LATIN CAPITAL LETTER O WITH HOOK ABOVE	LATIN SMALL LETTER O WITH HOOK ABOVE
1ED0	1ED1	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND ACUTE	LATIN SMALL LETTER O WITH CIRCUMFLEX AND ACUTE
1ED2	1ED3	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND GRAVE	LATIN SMALL LETTER O WITH CIRCUMFLEX AND GRAVE
1ED4	1ED5	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE	LATIN SMALL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
1ED6	1ED7	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND TILDE	LATIN SMALL LETTER O WITH CIRCUMFLEX AND TILDE
1ED8	1ED9	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND DOT BELOW	LATIN SMALL LETTER O WITH CIRCUMFLEX AND DOT BELOW
1EDA	1EDB	LATIN CAPITAL LETTER O WITH HORN AND ACUTE	LATIN SMALL LETTER O WITH HORN AND ACUTE
1EDC	1EDD	LATIN CAPITAL LETTER O WITH HORN AND GRAVE	LATIN SMALL LETTER O WITH HORN AND GRAVE
1EDE	1EDF	LATIN CAPITAL LETTER O WITH HORN AND HOOK ABOVE	LATIN SMALL LETTER O WITH HORN AND HOOK ABOVE
1EE0	1EE1	LATIN CAPITAL LETTER O WITH HORN AND TILDE	LATIN SMALL LETTER O WITH HORN AND TILDE
1EE2	1EE3	LATIN CAPITAL LETTER O WITH HORN AND DOT BELOW	LATIN SMALL LETTER O WITH HORN AND DOT BELOW
1EE4	1EE5	LATIN CAPITAL LETTER U WITH DOT BELOW	LATIN SMALL LETTER U WITH DOT BELOW
1EE6	1EE7	LATIN CAPITAL LETTER U WITH HOOK ABOVE	LATIN SMALL LETTER U WITH HOOK ABOVE

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1EE8	1EE9	LATIN CAPITAL LETTER U WITH HORN AND ACUTE	LATIN SMALL LETTER U WITH HORN AND ACUTE
1EEA	1EEB	LATIN CAPITAL LETTER U WITH HORN AND GRAVE	LATIN SMALL LETTER U WITH HORN AND GRAVE
1EEC	1EED	LATIN CAPITAL LETTER U WITH HORN AND HOOK ABOVE	LATIN SMALL LETTER U WITH HORN AND HOOK ABOVE
1EEE	1EEF	LATIN CAPITAL LETTER U WITH HORN AND TILDE	LATIN SMALL LETTER U WITH HORN AND TILDE
1EF0	1EF1	LATIN CAPITAL LETTER U WITH HORN AND DOT BELOW	LATIN SMALL LETTER U WITH HORN AND DOT BELOW
1EF2	1EF3	LATIN CAPITAL LETTER Y WITH GRAVE	LATIN SMALL LETTER Y WITH GRAVE
1EF4	1EF5	LATIN CAPITAL LETTER Y WITH DOT BELOW	LATIN SMALL LETTER Y WITH DOT BELOW
1EF6	1EF7	LATIN CAPITAL LETTER Y WITH HOOK ABOVE	LATIN SMALL LETTER Y WITH HOOK ABOVE
1EF8	1EF9	LATIN CAPITAL LETTER Y WITH TILDE	LATIN SMALL LETTER Y WITH TILDE
1F08	1F09	GREEK CAPITAL LETTER ALPHA WITH PSILI	GREEK SMALL LETTER ALPHA WITH PSILI
1F09	1F0A	GREEK CAPITAL LETTER ALPHA WITH DASIA	GREEK SMALL LETTER ALPHA WITH DASIA
1F0A	1F0B	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA
1F0B	1F0C	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA
1F0C	1F0D	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA
1F0D	1F0E	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA
1F0E	1F0F	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI
1F0F	1F10	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI
1F18	1F19	GREEK CAPITAL LETTER EPSILON WITH PSILI	GREEK SMALL LETTER EPSILON WITH PSILI
1F19	1F1A	GREEK CAPITAL LETTER EPSILON WITH DASIA	GREEK SMALL LETTER EPSILON WITH DASIA
1F1A	1F1B	GREEK CAPITAL LETTER EPSILON WITH PSILI AND VARIA	GREEK SMALL LETTER EPSILON WITH PSILI AND VARIA
1F1B	1F1C	GREEK CAPITAL LETTER EPSILON WITH DASIA AND VARIA	GREEK SMALL LETTER EPSILON WITH DASIA AND VARIA
1F1C	1F1D	GREEK CAPITAL LETTER EPSILON WITH PSILI AND OXIA	GREEK SMALL LETTER EPSILON WITH PSILI AND OXIA
1F1D	1F1E	GREEK CAPITAL LETTER EPSILON WITH DASIA AND OXIA	GREEK SMALL LETTER EPSILON WITH DASIA AND OXIA
1F28	1F29	GREEK CAPITAL LETTER ETA WITH PSILI	GREEK SMALL LETTER ETA WITH PSILI
1F29	1F2A	GREEK CAPITAL LETTER ETA WITH DASIA	GREEK SMALL LETTER ETA WITH DASIA
1F2A	1F2B	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA	GREEK SMALL LETTER ETA WITH PSILI AND VARIA
1F2B	1F2C	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA	GREEK SMALL LETTER ETA WITH DASIA AND VARIA
1F2C	1F2D	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA	GREEK SMALL LETTER ETA WITH PSILI AND OXIA

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1F2D	1F25	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA	GREEK SMALL LETTER ETA WITH DASIA AND OXIA
1F2E	1F26	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI
1F2F	1F27	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI
1F38	1F30	GREEK CAPITAL LETTER IOTA WITH PSILI	GREEK SMALL LETTER IOTA WITH PSILI
1F39	1F31	GREEK CAPITAL LETTER IOTA WITH DASIA	GREEK SMALL LETTER IOTA WITH DASIA
1F3A	1F32	GREEK CAPITAL LETTER IOTA WITH PSILI AND VARIA	GREEK SMALL LETTER IOTA WITH PSILI AND VARIA
1F3B	1F33	GREEK CAPITAL LETTER IOTA WITH DASIA AND VARIA	GREEK SMALL LETTER IOTA WITH DASIA AND VARIA
1F3C	1F34	GREEK CAPITAL LETTER IOTA WITH PSILI AND OXIA	GREEK SMALL LETTER IOTA WITH PSILI AND OXIA
1F3D	1F35	GREEK CAPITAL LETTER IOTA WITH DASIA AND OXIA	GREEK SMALL LETTER IOTA WITH DASIA AND OXIA
1F3E	1F36	GREEK CAPITAL LETTER IOTA WITH PSILI AND PERISPOMENI	GREEK SMALL LETTER IOTA WITH PSILI AND PERISPOMENI
1F3F	1F37	GREEK CAPITAL LETTER IOTA WITH DASIA AND PERISPOMENI	GREEK SMALL LETTER IOTA WITH DASIA AND PERISPOMENI
1F48	1F40	GREEK CAPITAL LETTER OMICRON WITH PSILI	GREEK SMALL LETTER OMICRON WITH PSILI
1F49	1F41	GREEK CAPITAL LETTER OMICRON WITH DASIA	GREEK SMALL LETTER OMICRON WITH DASIA
1F4A	1F42	GREEK CAPITAL LETTER OMICRON WITH PSILI AND VARIA	GREEK SMALL LETTER OMICRON WITH PSILI AND VARIA
1F4B	1F43	GREEK CAPITAL LETTER OMICRON WITH DASIA AND VARIA	GREEK SMALL LETTER OMICRON WITH DASIA AND VARIA
1F4C	1F44	GREEK CAPITAL LETTER OMICRON WITH PSILI AND OXIA	GREEK SMALL LETTER OMICRON WITH PSILI AND OXIA
1F4D	1F45	GREEK CAPITAL LETTER OMICRON WITH DASIA AND OXIA	GREEK SMALL LETTER OMICRON WITH DASIA AND OXIA
1F59	1F51	GREEK CAPITAL LETTER UPSILON WITH OASIS	GREEK SMALL LETTER UPSILON WITH DASIA
1F5B	1F53	GREEK CAPITAL LETTER UPSILON WITH DASIA AND VARIA	GREEK SMALL LETTER UPSILON WITH DASIA AND VARIA
1F5D	1F55	GREEK CAPITAL LETTER UPSILON WITH DASIA AND OXIA	GREEK SMALL LETTER UPSILON WITH DASIA AND OXIA
1F5F	1F57	GREEK CAPITAL LETTER UPSILON WITH DASIA AND PERISPOMENI	GREEK SMALL LETTER UPSILON WITH DASIA AND PERISPOMENI
1F68	1F60	GREEK CAPITAL LETTER OMEGA WITH PSILI	GREEK SMALL LETTER OMEGA WITH PSILI
1F69	1F61	GREEK CAPITAL LETTER OMEGA WITH DASIA	GREEK SMALL LETTER OMEGA WITH DASIA
1F6A	1F62	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA
1F6B	1F63	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA
1F6C	1F64	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA
1F6D	1F65	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA
1F6E	1F66	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1F6F	1F67	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI
1F88	1F80	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH PSILI AND YPOGEGRAMMENI
1F89	1F81	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH DASIA AND YPOGEGRAMMENI
1F8A	1F82	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA AND YPOGEGRAMMENI
1F8B	1F83	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA AND YPOGEGRAMMENI
1F8C	1F84	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA AND YPOGEGRAMMENI
1F8D	1F85	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA AND YPOGEGRAMMENI
1F8E	1F86	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
1F8F	1F87	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
1F98	1F90	GREEK CAPITAL LETTER ETA WITH PSILI AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH PSILI AND YPOGEGRAMMENI
1F99	1F91	GREEK CAPITAL LETTER ETA WITH DASIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH DASIA AND YPOGEGRAMMENI
1F9A	1F92	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH PSILI AND VARIA AND YPOGEGRAMMENI
1F9B	1F93	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH DASIA AND VARIA AND YPOGEGRAMMENI
1F9C	1F94	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH PSILI AND OXIA AND YPOGEGRAMMENI
1F9D	1F95	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH DASIA AND OXIA AND YPOGEGRAMMENI
1F9E	1F96	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
1F9F	1F97	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
1FA8	1FA0	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH PSILI AND YPOGEGRAMMENI
1FA9	1FA1	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH DASIA AND YPOGEGRAMMENI
1FAA	1FA2	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA AND YPOGEGRAMMENI
1FAB	1FA3	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA AND YPOGEGRAMMENI

Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
1FAC	1FA4	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA AND YPOGEGRAMMENI
1FAD	1FA5	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA AND YPOGEGRAMMENI
1FAE	1FA6	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI
1FAF	1FA7	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI
1FB8	1FB0	GREEK CAPITAL LETTER ALPHA WITH VRACHY	GREEK SMALL LETTER ALPHA WITH VRACHY
1FB9	1FB1	GREEK CAPITAL LETTER ALPHA WITH MACRON	GREEK SMALL LETTER ALPHA WITH MACRON
1FD8	1FD0	GREEK CAPITAL LETTER IOTA WITH VRACHY	GREEK SMALL LETTER IOTA WITH VRACHY
1FD9	1FD1	GREEK CAPITAL LETTER IOTA WITH MACRON	GREEK SMALL LETTER IOTA WITH MACRON
1FE8	1FE0	GREEK CAPITAL LETTER UPSILON WITH VRACHY	GREEK SMALL LETTER UPSILON WITH VRACHY
1FE9	1FE1	GREEK CAPITAL LETTER UPSILON WITH MACRON	GREEK SMALL LETTER UPSILON WITH MACRON
24B6	24D0	CIRCLED LATIN CAPITAL LETTER A	CIRCLED LATIN SMALL LETTER A
24B7	24D1	CIRCLED LATIN CAPITAL LETTER B	CIRCLED LATIN SMALL LETTER B
24B8	24D2	CIRCLED LATIN CAPITAL LETTER C	CIRCLED LATIN SMALL LETTER C
24B9	24D3	CIRCLED LATIN CAPITAL LETTER D	CIRCLED LATIN SMALL LETTER D
24BA	24D4	CIRCLED LATIN CAPITAL LETTER E	CIRCLED LATIN SMALL LETTER E
24BB	24D5	CIRCLED LATIN CAPITAL LETTER F	CIRCLED LATIN SMALL LETTER F
24BC	24D6	CIRCLED LATIN CAPITAL LETTER G	CIRCLED LATIN SMALL LETTER G
24BD	24D7	CIRCLED LATIN CAPITAL LETTER H	CIRCLED LATIN SMALL LETTER H
24BE	24D8	CIRCLED LATIN CAPITAL LETTER I	CIRCLED LATIN SMALL LETTER I
24BF	24D9	CIRCLED LATIN CAPITAL LETTER J	CIRCLED LATIN SMALL LETTER J
24C0	24DA	CIRCLED LATIN CAPITAL LETTER K	CIRCLED LATIN SMALL LETTER K
24C1	24DB	CIRCLED LATIN CAPITAL LETTER L	CIRCLED LATIN SMALL LETTER L
24C2	24DC	CIRCLED LATIN CAPITAL LETTER M	CIRCLED LATIN SMALL LETTER M
24C3	24DD	CIRCLED LATIN CAPITAL LETTER N	CIRCLED LATIN SMALL LETTER N
24C4	24DE	CIRCLED LATIN CAPITAL LETTER O	CIRCLED LATIN SMALL LETTER O
24C5	24DF	CIRCLED LATIN CAPITAL LETTER P	CIRCLED LATIN SMALL LETTER P
24C6	24E0	CIRCLED LATIN CAPITAL LETTER Q	CIRCLED LATIN SMALL LETTER Q
24C7	24E1	CIRCLED LATIN CAPITAL LETTER R	CIRCLED LATIN SMALL LETTER R
24C8	24E2	CIRCLED LATIN CAPITAL LETTER S	CIRCLED LATIN SMALL LETTER S
24C9	24E3	CIRCLED LATIN CAPITAL LETTER T	CIRCLED LATIN SMALL LETTER T
24CA	24E4	CIRCLED LATIN CAPITAL LETTER U	CIRCLED LATIN SMALL LETTER U
24CB	24E5	CIRCLED LATIN CAPITAL LETTER V	CIRCLED LATIN SMALL LETTER V
24CC	24E6	CIRCLED LATIN CAPITAL LETTER W	CIRCLED LATIN SMALL LETTER W
24CD	24E7	CIRCLED LATIN CAPITAL LETTER X	CIRCLED LATIN SMALL LETTER X
24CE	24E8	CIRCLED LATIN CAPITAL LETTER Y	CIRCLED LATIN SMALL LETTER Y



Uppercase code point	Lowercase code point	Uppercase character description	Lowercase character description
24CF	24E9	CIRCLED LATIN CAPITAL LETTER Z	CIRCLED LATIN SMALL LETTER Z
FF21	FF41	FULLWIDTH LATIN CAPITAL LETTER A	FULLWIDTH LATIN SMALL LETTER A
FF22	FF42	FULLWIDTH LATIN CAPITAL LETTER B	FULLWIDTH LATIN SMALL LETTER B
FF23	FF43	FULLWIDTH LATIN CAPITAL LETTER C	FULLWIDTH LATIN SMALL LETTER C
FF24	FF44	FULLWIDTH LATIN CAPITAL LETTER D	FULLWIDTH LATIN SMALL LETTER D
FF25	FF45	FULLWIDTH LATIN CAPITAL LETTER E	FULLWIDTH LATIN SMALL LETTER E
FF26	FF46	FULLWIDTH LATIN CAPITAL LETTER F	FULLWIDTH LATIN SMALL LETTER F
FF27	FF47	FULLWIDTH LATIN CAPITAL LETTER G	FULLWIDTH LATIN SMALL LETTER G
FF28	FF48	FULLWIDTH LATIN CAPITAL LETTER H	FULLWIDTH LATIN SMALL LETTER H
FF29	FF49	FULLWIDTH LATIN CAPITAL LETTER I	FULLWIDTH LATIN SMALL LETTER I
FF2A	FF4A	FULLWIDTH LATIN CAPITAL LETTER J	FULLWIDTH LATIN SMALL LETTER J
FF2B	FF4B	FULLWIDTH LATIN CAPITAL LETTER K	FULLWIDTH LATIN SMALL LETTER K
FF2C	FF4C	FULLWIDTH LATIN CAPITAL LETTER L	FULLWIDTH LATIN SMALL LETTER L
FF2D	FF4D	FULLWIDTH LATIN CAPITAL LETTER M	FULLWIDTH LATIN SMALL LETTER M
FF2E	FF4E	FULLWIDTH LATIN CAPITAL LETTER N	FULLWIDTH LATIN SMALL LETTER N
FF2F	FF4F	FULLWIDTH LATIN CAPITAL LETTER O	FULLWIDTH LATIN SMALL LETTER O
FF30	FF50	FULLWIDTH LATIN CAPITAL LETTER P	FULLWIDTH LATIN SMALL LETTER P
FF31	FF51	FULLWIDTH LATIN CAPITAL LETTER Q	FULLWIDTH LATIN SMALL LETTER Q
FF32	FF52	FULLWIDTH LATIN CAPITAL LETTER R	FULLWIDTH LATIN SMALL LETTER R
FF33	FF53	FULLWIDTH LATIN CAPITAL LETTER S	FULLWIDTH LATIN SMALL LETTER S
FF34	FF54	FULLWIDTH LATIN CAPITAL LETTER T	FULLWIDTH LATIN SMALL LETTER T
FF35	FF55	FULLWIDTH LATIN CAPITAL LETTER U	FULLWIDTH LATIN SMALL LETTER U
FF36	FF56	FULLWIDTH LATIN CAPITAL LETTER V	FULLWIDTH LATIN SMALL LETTER V
FF37	FF57	FULLWIDTH LATIN CAPITAL LETTER W	FULLWIDTH LATIN SMALL LETTER W
FF38	FF58	FULLWIDTH LATIN CAPITAL LETTER X	FULLWIDTH LATIN SMALL LETTER X
FF39	FF59	FULLWIDTH LATIN CAPITAL LETTER Y	FULLWIDTH LATIN SMALL LETTER Y
FF3A	FF5A	FULLWIDTH LATIN CAPITAL LETTER Z	FULLWIDTH LATIN SMALL LETTER Z

## Unicode lowercase to uppercase conversion mapping table

The table shows the mapping for the conversion of Unicode from lowercase to uppercase. You can refer to this table to see how the Convert Case API converts character data from lowercase to uppercase.

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
0061	0041	LATIN SMALL LETTER A	LATIN CAPITAL LETTER A
0062	0042	LATIN SMALL LETTER B	LATIN CAPITAL LETTER B
0063	0043	LATIN SMALL LETTER C	LATIN CAPITAL LETTER C
0064	0044	LATIN SMALL LETTER D	LATIN CAPITAL LETTER D
0065	0045	LATIN SMALL LETTER E	LATIN CAPITAL LETTER E
0066	0046	LATIN SMALL LETTER F	LATIN CAPITAL LETTER F
0067	0047	LATIN SMALL LETTER G	LATIN CAPITAL LETTER G
0068	0048	LATIN SMALL LETTER H	LATIN CAPITAL LETTER H
0069	0049	LATIN SMALL LETTER I	LATIN CAPITAL LETTER I
006A	004A	LATIN SMALL LETTER J	LATIN CAPITAL LETTER J
006B	004B	LATIN SMALL LETTER K	LATIN CAPITAL LETTER K
006C	004C	LATIN SMALL LETTER L	LATIN CAPITAL LETTER L
006D	004D	LATIN SMALL LETTER M	LATIN CAPITAL LETTER M
006E	004E	LATIN SMALL LETTER N	LATIN CAPITAL LETTER N
006F	004F	LATIN SMALL LETTER O	LATIN CAPITAL LETTER O
0070	0050	LATIN SMALL LETTER P	LATIN CAPITAL LETTER P
0071	0051	LATIN SMALL LETTER Q	LATIN CAPITAL LETTER Q
0072	0052	LATIN SMALL LETTER R	LATIN CAPITAL LETTER R
0073	0053	LATIN SMALL LETTER S	LATIN CAPITAL LETTER S
0074	0054	LATIN SMALL LETTER T	LATIN CAPITAL LETTER T
0075	0055	LATIN SMALL LETTER U	LATIN CAPITAL LETTER U
0076	0056	LATIN SMALL LETTER V	LATIN CAPITAL LETTER V
0077	0057	LATIN SMALL LETTER W	LATIN CAPITAL LETTER W
0078	0058	LATIN SMALL LETTER X	LATIN CAPITAL LETTER X
0079	0059	LATIN SMALL LETTER Y	LATIN CAPITAL LETTER Y
007A	005A	LATIN SMALL LETTER Z	LATIN CAPITAL LETTER Z
00E0	00C0	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A GRAVE
00E1	00C1	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A ACUTE
00E2	00C2	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A CIRCUMFLEX
00E3	00C3	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A TILDE
00E4	00C4	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A DIAERESIS
00E5	00C5	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A RING
00E6	00C6	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER A E
00E7	00C7	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER C CEDILLA
00E8	00C8	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER E GRAVE
00E9	00C9	LATIN SMALL LETTER A GRAVE	LATIN CAPITAL LETTER E ACUTE
00EA	00CA	LATIN SMALL LETTER E CIRCUMFLEX	LATIN CAPITAL LETTER E CIRCUMFLEX
00EB	00CB	LATIN SMALL LETTER E DIAERESIS	LATIN CAPITAL LETTER E DIAERESIS
00EC	00CC	LATIN SMALL LETTER I GRAVE	LATIN CAPITAL LETTER I GRAVE

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
00ED	00CD	LATIN SMALL LETTER I ACUTE	LATIN CAPITAL LETTER I ACUTE
00EE	00CE	LATIN SMALL LETTER I CIRCUMFLEX	LATIN CAPITAL LETTER I CIRCUMFLEX
00EF	00CF	LATIN SMALL LETTER I DIAERESIS	LATIN CAPITAL LETTER I DIAERESIS
00F0	00D0	LATIN SMALL LETTER ETH	LATIN CAPITAL LETTER ETH
00F1	00D1	LATIN SMALL LETTER N TILDE	LATIN CAPITAL LETTER N TILDE
00F2	00D2	LATIN SMALL LETTER O GRAVE	LATIN CAPITAL LETTER O GRAVE
00F3	00D3	LATIN SMALL LETTER O ACUTE	LATIN CAPITAL LETTER O ACUTE
00F4	00D4	LATIN SMALL LETTER O CIRCUMFLEX	LATIN CAPITAL LETTER O CIRCUMFLEX
00F5	00D5	LATIN SMALL LETTER O TILDE	LATIN CAPITAL LETTER O TILDE
00F6	00D6	LATIN SMALL LETTER O DIAERESIS	LATIN CAPITAL LETTER O DIAERESIS
00F8	00D8	LATIN SMALL LETTER O SLASH	LATIN CAPITAL LETTER O SLASH
00F9	00D9	LATIN SMALL LETTER U GRAVE	LATIN CAPITAL LETTER U GRAVE
00FA	00DA	LATIN SMALL LETTER U ACUTE	LATIN CAPITAL LETTER U ACUTE
00FB	00DB	LATIN SMALL LETTER U CIRCUMFLEX	LATIN CAPITAL LETTER U CIRCUMFLEX
00FC	00DC	LATIN SMALL LETTER U DIAERESIS	LATIN CAPITAL LETTER U DIAERESIS
00FD	00DD	LATIN SMALL LETTER Y ACUTE	LATIN CAPITAL LETTER Y ACUTE
00FE	00DE	LATIN SMALL LETTER THORN	LATIN CAPITAL LETTER THORN
00FF	0178	LATIN SMALL LETTER Y DIAERESIS	LATIN CAPITAL LETTER Y WITH DIAERESIS
0101	0100	LATIN SMALL LETTER A WITH MACRON	LATIN CAPITAL LETTER A WITH MACRON
0103	0102	LATIN SMALL LETTER A WITH BREVE	LATIN CAPITAL LETTER A WITH BREVE
0105	0104	LATIN SMALL LETTER A WITH OGONEK	LATIN CAPITAL LETTER A WITH OGONEK
0107	0106	LATIN SMALL LETTER C WITH ACUTE	LATIN CAPITAL LETTER C WITH ACUTE
0109	0108	LATIN SMALL LETTER C WITH CIRCUMFLEX	LATIN CAPITAL LETTER C WITH CIRCUMFLEX
010B	010A	LATIN SMALL LETTER C WITH DOT ABOVE	LATIN CAPITAL LETTER C WITH DOT ABOVE
010D	010C	LATIN SMALL LETTER C WITH CARON	LATIN CAPITAL LETTER C WITH CARON
010F	010E	LATIN SMALL LETTER D WITH CARON	LATIN CAPITAL LETTER D WITH CARON
0111	0110	LATIN SMALL LETTER D WITH STROKE	LATIN CAPITAL LETTER D WITH STROKE
0113	0112	LATIN SMALL LETTER E WITH MACRON	LATIN CAPITAL LETTER E WITH MACRON
0115	0114	LATIN SMALL LETTER E WITH BREVE	LATIN CAPITAL LETTER E WITH BREVE
0117	0116	LATIN SMALL LETTER E WITH DOT ABOVE	LATIN CAPITAL LETTER E WITH DOT ABOVE
0119	0118	LATIN SMALL LETTER E WITH OGONEK	LATIN CAPITAL LETTER E WITH OGONEK
011B	011A	LATIN SMALL LETTER E WITH CARON	LATIN CAPITAL LETTER E WITH CARON

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
011D	011C	LATIN SMALL LETTER G WITH CIRCUMFLEX	LATIN CAPITAL LETTER G WITH CIRCUMFLEX
011F	011E	LATIN SMALL LETTER G WITH BREVE	LATIN CAPITAL LETTER G WITH BREVE
0121	0120	LATIN SMALL LETTER G WITH DOT ABOVE	LATIN CAPITAL LETTER G WITH DOT ABOVE
0123	0122	LATIN SMALL LETTER G WITH CEDILLA	LATIN CAPITAL LETTER G WITH CEDILLA
0125	0124	LATIN SMALL LETTER H WITH CIRCUMFLEX	LATIN CAPITAL LETTER H WITH CIRCUMFLEX
0127	0126	LATIN SMALL LETTER H WITH STROKE	LATIN CAPITAL LETTER H WITH STROKE
0129	0128	LATIN SMALL LETTER I WITH TILDE	LATIN CAPITAL LETTER I WITH TILDE
012B	012A	LATIN SMALL LETTER I WITH MACRON	LATIN CAPITAL LETTER I WITH MACRON
012D	012C	LATIN SMALL LETTER I WITH BREVE	LATIN CAPITAL LETTER I WITH BREVE
012F	012E	LATIN SMALL LETTER I WITH OGONEK	LATIN CAPITAL LETTER I WITH OGONEK
0131	0049	LATIN SMALL LETTER DOTLESS I	LATIN CAPITAL LETTER I
0133	0132	LATIN SMALL LIGATURE IJ	LATIN CAPITAL LIGATURE IJ
0135	0134	LATIN SMALL LETTER J WITH CIRCUMFLEX	LATIN CAPITAL LETTER J WITH CIRCUMFLEX
0137	0136	LATIN SMALL LETTER K WITH CEDILLA	LATIN CAPITAL LETTER K WITH CEDILLA
013A	0139	LATIN SMALL LETTER L WITH ACUTE	LATIN CAPITAL LETTER L WITH ACUTE
013C	013B	LATIN SMALL LETTER L WITH CEDILLA	LATIN CAPITAL LETTER L WITH CEDILLA
013E	013D	LATIN SMALL LETTER L WITH CARON	LATIN CAPITAL LETTER L WITH CARON
0140	013F	LATIN SMALL LETTER L WITH MIDDLE DOT	LATIN CAPITAL LETTER L WITH MIDDLE DOT
0142	0141	LATIN SMALL LETTER L WITH STROKE	LATIN CAPITAL LETTER L WITH STROKE
0144	0143	LATIN SMALL LETTER N WITH ACUTE	LATIN CAPITAL LETTER N WITH ACUTE
0146	0145	LATIN SMALL LETTER N WITH CEDILLA	LATIN CAPITAL LETTER N WITH CEDILLA
0148	0147	LATIN SMALL LETTER N WITH CARON	LATIN CAPITAL LETTER N WITH CARON
014B	014A	LATIN SMALL LETTER ENG (SAMI)	LATIN CAPITAL LETTER ENG (SAMI)
014D	014C	LATIN SMALL LETTER O WITH MACRON	LATIN CAPITAL LETTER O WITH MACRON
014F	014E	LATIN SMALL LETTER O WITH BREVE	LATIN CAPITAL LETTER O WITH BREVE
0151	0150	LATIN SMALL LETTER O WITH DOUBLE ACUTE	LATIN CAPITAL LETTER O WITH DOUBLE ACUTE
0153	0152	LATIN SMALL LIGATURE OE	LATIN CAPITAL LIGATURE OE
0155	0154	LATIN SMALL LETTER R WITH ACUTE	LATIN CAPITAL LETTER R WITH ACUTE
0157	0156	LATIN SMALL LETTER R WITH CEDILLA	LATIN CAPITAL LETTER R WITH CEDILLA

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
0159	0158	LATIN SMALL LETTER R WITH CARON	LATIN CAPITAL LETTER R WITH CARON
015B	015A	LATIN SMALL LETTER S WITH ACUTE	LATIN CAPITAL LETTER S WITH ACUTE
015D	015C	LATIN SMALL LETTER S WITH CIRCUMFLEX	LATIN CAPITAL LETTER S WITH CIRCUMFLEX
015F	015E	LATIN SMALL LETTER S WITH CEDILLA	LATIN CAPITAL LETTER S WITH CEDILLA
0161	0160	LATIN SMALL LETTER S WITH CARON	LATIN CAPITAL LETTER S WITH CARON
0163	0162	LATIN SMALL LETTER T WITH CEDILLA	LATIN CAPITAL LETTER T WITH CEDILLA
0165	0164	LATIN SMALL LETTER T WITH CARON	LATIN CAPITAL LETTER T WITH CARON
0167	0166	LATIN SMALL LETTER T WITH STROKE	LATIN CAPITAL LETTER T WITH STROKE
0169	0168	LATIN SMALL LETTER U WITH TILDE	LATIN CAPITAL LETTER U WITH TILDE
016B	016A	LATIN SMALL LETTER U WITH MACRON	LATIN CAPITAL LETTER U WITH MACRON
016D	016C	LATIN SMALL LETTER U WITH BREVE	LATIN CAPITAL LETTER U WITH BREVE
016F	016E	LATIN SMALL LETTER U WITH RING ABOVE	LATIN CAPITAL LETTER U WITH RING ABOVE
0171	0170	LATIN SMALL LETTER U WITH DOUBLE ACUTE	LATIN CAPITAL LETTER U WITH DOUBLE ACUTE
0173	0172	LATIN SMALL LETTER U WITH OGONEK	LATIN CAPITAL LETTER U WITH OGONEK
0175	0174	LATIN SMALL LETTER W WITH CIRCUMFLEX	LATIN CAPITAL LETTER W WITH CIRCUMFLEX
0177	0176	LATIN SMALL LETTER Y WITH CIRCUMFLEX	LATIN CAPITAL LETTER Y WITH CIRCUMFLEX
017A	0179	LATIN SMALL LETTER Z WITH ACUTE	LATIN CAPITAL LETTER Z WITH ACUTE
017C	017B	LATIN SMALL LETTER Z WITH DOT ABOVE	LATIN CAPITAL LETTER Z WITH DOT ABOVE
017E	017D	LATIN SMALL LETTER Z WITH CARON	LATIN CAPITAL LETTER Z WITH CARON
0183	0182	LATIN SMALL LETTER B WITH TOPBAR	LATIN CAPITAL LETTER B WITH TOPBAR
0185	0184	LATIN SMALL LETTER TONE SIX	LATIN CAPITAL LETTER TONE SIX
0188	0187	LATIN SMALL LETTER C WITH HOOK	LATIN CAPITAL LETTER C WITH HOOK
018C	018B	LATIN SMALL LETTER D WITH TOPBAR	LATIN CAPITAL LETTER D WITH TOPBAR
0192	0191	LATIN SMALL LETTER F WITH HOOK	LATIN CAPITAL LETTER F WITH HOOK
0199	0198	LATIN SMALL LETTER K WITH HOOK	LATIN CAPITAL LETTER K WITH HOOK
01A1	01A0	LATIN SMALL LETTER O WITH HORN	LATIN CAPITAL LETTER O WITH HORN
01A3	01A2	LATIN SMALL LETTER OI	LATIN CAPITAL LETTER OI
01A5	01A4	LATIN SMALL LETTER P WITH HOOK	LATIN CAPITAL LETTER P WITH HOOK
01A8	01A7	LATIN SMALL LETTER TONE TWO	LATIN CAPITAL LETTER TONE TWO

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
01AD	01AC	LATIN SMALL LETTER T WITH HOOK	LATIN CAPITAL LETTER T WITH HOOK
01B0	01AF	LATIN SMALL LETTER U WITH HORN	LATIN CAPITAL LETTER U WITH HORN
01B4	01B3	LATIN SMALL LETTER Y WITH HOOK	LATIN CAPITAL LETTER Y WITH HOOK
01B6	01B5	LATIN SMALL LETTER Z WITH STROKE	LATIN CAPITAL LETTER Z WITH STROKE
01B9	01B8	LATIN SMALL LETTER EZH REVERSED	LATIN CAPITAL LETTER EZH REVERSED
01BD	01BC	LATIN SMALL LETTER TONE FIVE	LATIN CAPITAL LETTER TONE FIVE
01C6	01C4	LATIN SMALL LETTER DZ WITH CARON	LATIN CAPITAL LETTER DZ WITH CARON
01C9	01C7	LATIN SMALL LETTER LJ	LATIN CAPITAL LETTER LJ
01CC	01CA	LATIN SMALL LETTER NJ	LATIN CAPITAL LETTER NJ
01CE	01CD	LATIN SMALL LETTER A WITH CARON	LATIN CAPITAL LETTER A WITH CARON
01D0	01CF	LATIN SMALL LETTER I WITH CARON	LATIN CAPITAL LETTER I WITH CARON
01D2	01D1	LATIN SMALL LETTER O WITH CARON	LATIN CAPITAL LETTER O WITH CARON
01D4	01D3	LATIN SMALL LETTER U WITH CARON	LATIN CAPITAL LETTER U WITH CARON
01D6	01D5	LATIN SMALL LETTER U WITH DIAERESIS AND MACRON	LATIN CAPITAL LETTER U WITH DIAERESIS AND MACRON
01D8	01D7	LATIN SMALL LETTER U WITH DIAERESIS AND ACUTE	LATIN CAPITAL LETTER U WITH DIAERESIS AND ACUTE
01DA	01D9	LATIN SMALL LETTER U WITH DIAERESIS AND CARON	LATIN CAPITAL LETTER U WITH DIAERESIS AND CARON
01DC	01DB	LATIN SMALL LETTER U WITH DIAERESIS AND GRAVE	LATIN CAPITAL LETTER U WITH DIAERESIS AND GRAVE
01DF	01DE	LATIN SMALL LETTER A WITH DIAERESIS AND MACRON	LATIN CAPITAL LETTER A WITH DIAERESIS AND MACRON
01E1	01E0	LATIN SMALL LETTER A WITH DOT ABOVE AND MACRON	LATIN CAPITAL LETTER A WITH DOT ABOVE AND MACRON
01E3	01E2	LATIN SMALL LIGATURE AE WITH MACRON	LATIN CAPITAL LIGATURE AE WITH MACRON
01E5	01E4	LATIN SMALL LETTER G WITH STROKE	LATIN CAPITAL LETTER G WITH STROKE
01E7	01E6	LATIN SMALL LETTER G WITH CARON	LATIN CAPITAL LETTER G WITH CARON
01E9	01E8	LATIN SMALL LETTER K WITH CARON	LATIN CAPITAL LETTER K WITH CARON
01EB	01EA	LATIN SMALL LETTER O WITH OGONEK	LATIN CAPITAL LETTER O WITH OGONEK
01ED	01EC	LATIN SMALL LETTER O WITH OGONEK AND MACRON	LATIN CAPITAL LETTER O WITH OGONEK AND MACRON
01EF	01EE	LATIN SMALL LETTER EZH WITH CARON	LATIN CAPITAL LETTER EZH WITH CARON
01F3	01F1	LATIN SMALL LETTER DZ	LATIN CAPITAL LETTER DZ
01F5	01F4	LATIN SMALL LETTER G WITH ACUTE	LATIN CAPITAL LETTER G WITH ACUTE
01FB	01FA	LATIN SMALL LETTER A WITH RING ABOVE AND ACUTE	LATIN CAPITAL LETTER A WITH RING ABOVE AND ACUTE
01FD	01FC	LATIN SMALL LIGATURE AE WITH ACUTE	LATIN CAPITAL LIGATURE AE WITH ACUTE

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
01FF	01FE	LATIN SMALL LETTER O WITH STROKE AND ACUTE	LATIN CAPITAL LETTER O WITH STROKE AND ACUTE
0201	0200	LATIN SMALL LETTER A WITH DOUBLE GRAVE	LATIN CAPITAL LETTER A WITH DOUBLE GRAVE
0203	0202	LATIN SMALL LETTER A WITH INVERTED BREVE	LATIN CAPITAL LETTER A WITH INVERTED BREVE
0205	0204	LATIN SMALL LETTER E WITH DOUBLE GRAVE	LATIN CAPITAL LETTER E WITH DOUBLE GRAVE
0207	0206	LATIN SMALL LETTER E WITH INVERTED BREVE	LATIN CAPITAL LETTER E WITH INVERTED BREVE
0209	0208	LATIN SMALL LETTER I WITH DOUBLE GRAVE	LATIN CAPITAL LETTER I WITH DOUBLE GRAVE
020B	020A	LATIN SMALL LETTER I WITH INVERTED BREVE	LATIN CAPITAL LETTER I WITH INVERTED BREVE
020D	020C	LATIN SMALL LETTER O WITH DOUBLE GRAVE	LATIN CAPITAL LETTER O WITH DOUBLE GRAVE
020F	020E	LATIN SMALL LETTER O WITH INVERTED BREVE	LATIN CAPITAL LETTER O WITH INVERTED BREVE
0211	0210	LATIN SMALL LETTER R WITH DOUBLE GRAVE	LATIN CAPITAL LETTER R WITH DOUBLE GRAVE
0213	0212	LATIN SMALL LETTER R WITH INVERTED BREVE	LATIN CAPITAL LETTER R WITH INVERTED BREVE
0215	0214	LATIN SMALL LETTER U WITH DOUBLE GRAVE	LATIN CAPITAL LETTER U WITH DOUBLE GRAVE
0217	0216	LATIN SMALL LETTER U WITH INVERTED BREVE	LATIN CAPITAL LETTER U WITH INVERTED BREVE
0253	0181	LATIN SMALL LETTER B WITH HOOK	LATIN CAPITAL LETTER B WITH HOOK
0254	0186	LATIN SMALL LETTER OPEN O	LATIN CAPITAL LETTER OPEN O
0257	018A	LATIN SMALL LETTER D WITH HOOK	LATIN CAPITAL LETTER D WITH HOOK
0258	018E	LATIN SMALL LETTER REVERSED E	LATIN CAPITAL LETTER REVERSED E
0259	018F	LATIN SMALL LETTER SCHWA	LATIN CAPITAL LETTER SCHWA
025B	0190	LATIN SMALL LETTER OPEN E	LATIN CAPITAL LETTER OPEN E
0260	0193	LATIN SMALL LETTER G WITH HOOK	LATIN CAPITAL LETTER G WITH HOOK
0263	0194	LATIN SMALL LETTER GAMMA	LATIN CAPITAL LETTER GAMMA
0268	0197	LATIN SMALL LETTER I WITH STROKE	LATIN CAPITAL LETTER I WITH STROKE
0269	0196	LATIN SMALL LETTER IOTA	LATIN CAPITAL LETTER IOTA
026F	019C	LATIN SMALL LETTER TURNED M	LATIN CAPITAL LETTER TURNED M
0272	019D	LATIN SMALL LETTER N WITH LEFT HOOK	LATIN CAPITAL LETTER N WITH LEFT HOOK
0275	019F	LATIN SMALL LETTER BARRED O	LATIN CAPITAL LETTER O WITH MIDDLE TILDE
0283	01A9	LATIN SMALL LETTER ESH	LATIN CAPITAL LETTER ESH
0288	01AE	LATIN SMALL LETTER T WITH RETROFLEX HOOK	LATIN CAPITAL LETTER T WITH RETROFLEX HOOK
028A	01B1	LATIN SMALL LETTER UPSILON	LATIN CAPITAL LETTER UPSILON
028B	01B2	LATIN SMALL LETTER V WITH HOOK	LATIN CAPITAL LETTER V WITH HOOK
0292	01B7	LATIN SMALL LETTER EZH	LATIN CAPITAL LETTER EZH

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
03AC	0386	GREEK SMALL LETTER ALPHA WITH TONOS	GREEK CAPITAL LETTER ALPHA WITH TONOS
03AD	0388	GREEK SMALL LETTER EPSILON WITH TONOS	GREEK CAPITAL LETTER EPSILON WITH TONOS
03AE	0389	GREEK SMALL LETTER ETA WITH TONOS	GREEK CAPITAL LETTER ETA WITH TONOS
03AF	038A	GREEK SMALL LETTER IOTA WITH TONOS	GREEK CAPITAL LETTER IOTA WITH TONOS
03B1	0391	GREEK SMALL LETTER ALPHA	GREEK CAPITAL LETTER ALPHA
03B2	0392	GREEK SMALL LETTER BETA	GREEK CAPITAL LETTER BETA
03B3	0393	GREEK SMALL LETTER GAMMA	GREEK CAPITAL LETTER GAMMA
03B4	0394	GREEK SMALL LETTER DELTA	GREEK CAPITAL LETTER DELTA
03B5	0395	GREEK SMALL LETTER EPSILON	GREEK CAPITAL LETTER EPSILON
03B6	0396	GREEK SMALL LETTER ZETA	GREEK CAPITAL LETTER ZETA
03B7	0397	GREEK SMALL LETTER ETA	GREEK CAPITAL LETTER ETA
03B8	0398	GREEK SMALL LETTER THETA	GREEK CAPITAL LETTER THETA
03B9	0399	GREEK SMALL LETTER IOTA	GREEK CAPITAL LETTER IOTA
03BA	039A	GREEK SMALL LETTER KAPPA	GREEK CAPITAL LETTER KAPPA
03BB	039B	GREEK SMALL LETTER LAMDA	GREEK CAPITAL LETTER LAMDA
03BC	039C	GREEK SMALL LETTER MU	GREEK CAPITAL LETTER MU
03BD	039D	GREEK SMALL LETTER NU	GREEK CAPITAL LETTER NU
03BE	039E	GREEK SMALL LETTER XI	GREEK CAPITAL LETTER XI
03BF	039F	GREEK SMALL LETTER OMICRON	GREEK CAPITAL LETTER OMICRON
03C0	03A0	GREEK SMALL LETTER PI	GREEK CAPITAL LETTER PI
03C1	03A1	GREEK SMALL LETTER RHO	GREEK CAPITAL LETTER RHO
03C3	03A3	GREEK SMALL LETTER SIGMA	GREEK CAPITAL LETTER SIGMA
03C4	03A4	GREEK SMALL LETTER TAU	GREEK CAPITAL LETTER TAU
03C5	03A5	GREEK SMALL LETTER UPSILON	GREEK CAPITAL LETTER UPSILON
03C6	03A6	GREEK SMALL LETTER PHI	GREEK CAPITAL LETTER PHI
03C7	03A7	GREEK SMALL LETTER CHI	GREEK CAPITAL LETTER CHI
03C8	03A8	GREEK SMALL LETTER PSI	GREEK CAPITAL LETTER PSI
03C9	03A9	GREEK SMALL LETTER OMEGA	GREEK CAPITAL LETTER OMEGA
03CA	03AA	GREEK SMALL LETTER IOTA WITH DIALYTIKA	GREEK CAPITAL LETTER IOTA WITH DIALYTIKA
03CB	03AB	GREEK SMALL LETTER UPSILON WITH DIALYTIKA	GREEK CAPITAL LETTER UPSILON WITH DIALYTIKA
03CC	038C	GREEK SMALL LETTER OMICRON WITH TONOS	GREEK CAPITAL LETTER OMICRON WITH TONOS
03CD	038E	GREEK SMALL LETTER UPSILON WITH TONOS	GREEK CAPITAL LETTER UPSILON WITH TONOS
03CE	038F	GREEK SMALL LETTER OMEGA WITH TONOS	GREEK CAPITAL LETTER OMEGA WITH TONOS
03E3	03E2	COPTIC SMALL LETTER SHEI	COPTIC CAPITAL LETTER SHEI
03E5	03E4	COPTIC SMALL LETTER FEI	COPTIC CAPITAL LETTER FEI
03E7	03E6	COPTIC SMALL LETTER KHEI	COPTIC CAPITAL LETTER KHEI
03E9	03E8	COPTIC SMALL LETTER HORI	COPTIC CAPITAL LETTER HORI
03EB	03EA	COPTIC SMALL LETTER GANGIA	COPTIC CAPITAL LETTER GANGIA
03ED	03EC	COPTIC SMALL LETTER SHIMA	COPTIC CAPITAL LETTER SHIMA



Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
03EF	03EE	COPTIC SMALL LETTER DEI	COPTIC CAPITAL LETTER DEI
0430	0410	CYRILLIC SMALL LETTER A	CYRILLIC CAPITAL LETTER A
0431	0411	CYRILLIC SMALL LETTER BE	CYRILLIC CAPITAL LETTER BE
0432	0412	CYRILLIC SMALL LETTER VE	CYRILLIC CAPITAL LETTER VE
0433	0413	CYRILLIC SMALL LETTER GHE	CYRILLIC CAPITAL LETTER GHE
0434	0414	CYRILLIC SMALL LETTER DE	CYRILLIC CAPITAL LETTER DE
0435	0415	CYRILLIC SMALL LETTER IE	CYRILLIC CAPITAL LETTER IE
0436	0416	CYRILLIC SMALL LETTER ZHE	CYRILLIC CAPITAL LETTER ZHE
0437	0417	CYRILLIC SMALL LETTER ZE	CYRILLIC CAPITAL LETTER ZE
0438	0418	CYRILLIC SMALL LETTER I	CYRILLIC CAPITAL LETTER I
0439	0419	CYRILLIC SMALL LETTER SHORT I	CYRILLIC CAPITAL LETTER SHORT I
043A	041A	CYRILLIC SMALL LETTER KA	CYRILLIC CAPITAL LETTER KA
043B	041B	CYRILLIC SMALL LETTER EL	CYRILLIC CAPITAL LETTER EL
043C	041C	CYRILLIC SMALL LETTER EM	CYRILLIC CAPITAL LETTER EM
043D	041D	CYRILLIC SMALL LETTER EN	CYRILLIC CAPITAL LETTER EN
043E	041E	CYRILLIC SMALL LETTER O	CYRILLIC CAPITAL LETTER O
043F	041F	CYRILLIC SMALL LETTER PE	CYRILLIC CAPITAL LETTER PE
0440	0420	CYRILLIC SMALL LETTER ER	CYRILLIC CAPITAL LETTER ER
0441	0421	CYRILLIC SMALL LETTER ES	CYRILLIC CAPITAL LETTER ES
0442	0422	CYRILLIC SMALL LETTER TE	CYRILLIC CAPITAL LETTER TE
0443	0423	CYRILLIC SMALL LETTER U	CYRILLIC CAPITAL LETTER U
0444	0424	CYRILLIC SMALL LETTER EF	CYRILLIC CAPITAL LETTER EF
0445	0425	CYRILLIC SMALL LETTER HA	CYRILLIC CAPITAL LETTER HA
0446	0426	CYRILLIC SMALL LETTER TSE	CYRILLIC CAPITAL LETTER TSE
0447	0427	CYRILLIC SMALL LETTER CHE	CYRILLIC CAPITAL LETTER CHE
0448	0428	CYRILLIC SMALL LETTER SHA	CYRILLIC CAPITAL LETTER SHA
0449	0429	CYRILLIC SMALL LETTER SHCHA	CYRILLIC CAPITAL LETTER SHCHA
044A	042A	CYRILLIC SMALL LETTER HARD SIGN	CYRILLIC CAPITAL LETTER HARD SIGN
044B	042B	CYRILLIC SMALL LETTER YERU	CYRILLIC CAPITAL LETTER YERU
044C	042C	CYRILLIC SMALL LETTER SOFT SIGN	CYRILLIC CAPITAL LETTER SOFT SIGN
044D	042D	CYRILLIC SMALL LETTER E	CYRILLIC CAPITAL LETTER E
044E	042E	CYRILLIC SMALL LETTER YU	CYRILLIC CAPITAL LETTER YU
044F	042F	CYRILLIC SMALL LETTER YA	CYRILLIC CAPITAL LETTER YA
0451	0401	CYRILLIC SMALL LETTER IO	CYRILLIC CAPITAL LETTER IO
0452	0402	CYRILLIC SMALL LETTER DJE (SERBOCROATIAN)	CYRILLIC CAPITAL LETTER DJE (SERBOCROATIAN)
0453	0403	CYRILLIC SMALL LETTER GJE	CYRILLIC CAPITAL LETTER GJE
0454	0404	CYRILLIC SMALL LETTER UKRAINIAN IE	CYRILLIC CAPITAL LETTER UKRAINIAN IE
0455	0405	CYRILLIC SMALL LETTER DZE	CYRILLIC CAPITAL LETTER DZE
0456	0406	CYRILLIC SMALL LETTER BYELORUSSIAN-UKRAINIAN I	CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I
0457	0407	CYRILLIC SMALL LETTER YI (UKRAINIAN)	CYRILLIC CAPITAL LETTER YI (UKRAINIAN)

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
0458	0408	CYRILLIC SMALL LETTER JE	CYRILLIC CAPITAL LETTER JE
0459	0409	CYRILLIC SMALL LETTER LJE	CYRILLIC CAPITAL LETTER LJE
045A	040A	CYRILLIC SMALL LETTER NJE	CYRILLIC CAPITAL LETTER NJE
045B	040B	CYRILLIC SMALL LETTER TSHE (SERBOCROATIAN)	CYRILLIC CAPITAL LETTER TSHE (SERBOCROATIAN)
045C	040C	CYRILLIC SMALL LETTER KJE	CYRILLIC CAPITAL LETTER KJE
045E	040E	CYRILLIC SMALL LETTER SHORT U (BYELORUSSIAN)	CYRILLIC CAPITAL LETTER SHORT U (BYELORUSSIAN)
045F	040F	CYRILLIC SMALL LETTER DZHE	CYRILLIC CAPITAL LETTER DZHE
0461	0460	CYRILLIC SMALL LETTER OMEGA	CYRILLIC CAPITAL LETTER OMEGA
0463	0462	CYRILLIC SMALL LETTER YAT	CYRILLIC CAPITAL LETTER YAT
0465	0464	CYRILLIC SMALL LETTER IOTIFIED E	CYRILLIC CAPITAL LETTER IOTIFIED E
0467	0466	CYRILLIC SMALL LETTER LITTLE YUS	CYRILLIC CAPITAL LETTER LITTLE YUS
0469	0468	CYRILLIC SMALL LETTER IOTIFIED LITTLE YUS	CYRILLIC CAPITAL LETTER IOTIFIED LITTLE YUS
046B	046A	CYRILLIC SMALL LETTER BIG YUS	CYRILLIC CAPITAL LETTER BIG YUS
046D	046C	CYRILLIC SMALL LETTER IOTIFIED BIG YUS	CYRILLIC CAPITAL LETTER IOTIFIED BIG YUS
046F	046E	CYRILLIC SMALL LETTER KSI	CYRILLIC CAPITAL LETTER KSI
0471	0470	CYRILLIC SMALL LETTER PSI	CYRILLIC CAPITAL LETTER PSI
0473	0472	CYRILLIC SMALL LETTER FITA	CYRILLIC CAPITAL LETTER FITA
0475	0474	CYRILLIC SMALL LETTER IZHITSA	CYRILLIC CAPITAL LETTER IZHITSA
0477	0476	CYRILLIC SMALL LETTER IZHITSA WITH DOUBLE GRAVE ACCENT	CYRILLIC CAPITAL LETTER IZHITSA WITH DOUBLE GRAVE ACCENT
0479	0478	CYRILLIC SMALL LETTER UK	CYRILLIC CAPITAL LETTER UK
047B	047A	CYRILLIC SMALL LETTER ROUND OMEGA	CYRILLIC CAPITAL LETTER ROUND OMEGA
047D	047C	CYRILLIC SMALL LETTER OMEGA WITH TITLO	CYRILLIC CAPITAL LETTER OMEGA WITH TITLO
047F	047E	CYRILLIC SMALL LETTER OT	CYRILLIC CAPITAL LETTER OT
0481	0480	CYRILLIC SMALL LETTER KOPPA	CYRILLIC CAPITAL LETTER KOPPA
0491	0490	CYRILLIC SMALL LETTER GHE WITH UPTURN	CYRILLIC CAPITAL LETTER GHE WITH UPTURN
0493	0492	CYRILLIC SMALL LETTER GHE WITH STROKE	CYRILLIC CAPITAL LETTER GHE WITH STROKE
0495	0494	CYRILLIC SMALL LETTER GHE WITH MIDDLE HOOK	CYRILLIC CAPITAL LETTER GHE WITH MIDDLE HOOK
0497	0496	CYRILLIC SMALL LETTER ZHE WITH DESCENDER	CYRILLIC CAPITAL LETTER ZHE WITH DESCENDER
0499	0498	CYRILLIC SMALL LETTER ZE WITH DESCENDER	CYRILLIC CAPITAL LETTER ZE WITH DESCENDER
049B	049A	CYRILLIC SMALL LETTER KA WITH DESCENDER	CYRILLIC CAPITAL LETTER KA WITH DESCENDER
049D	049C	CYRILLIC SMALL LETTER KA WITH VERTICAL STROKE	CYRILLIC CAPITAL LETTER KA WITH VERTICAL STROKE
049F	049E	CYRILLIC SMALL LETTER KA WITH STROKE	CYRILLIC CAPITAL LETTER KA WITH STROKE

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
04A1	04A0	CYRILLIC SMALL LETTER EASHKIR KA	CYRILLIC CAPITAL LETTER BASHKIR KA
04A3	04A2	CYRILLIC SMALL LETTER EN WITH DESCENDER	CYRILLIC CAPITAL LETTER EN WITH DESCENDER
04A5	04A4	CYRILLIC SMALL LIGATURE EN GHE	CYRILLIC CAPITAL LIGATURE EN GHF
04A7	04A6	CYRILLIC SMALL LETTER PE WITH MIDDLE HOOK (ABKHASIAN)	CYRILLIC CAPITAL LETTER PE WITH MIDDLE HOOK (ABKHASIAN)
04A9	04A8	CYRILLIC SMALL LETTER ABKHASIAN HA	CYRILLIC CAPITAL LETTER ABKHASIAN HA
04AB	04AA	CYRILLIC SMALL LETTER ES WITH DESCENDER	CYRILLIC CAPITAL LETTER ES WITH DESCENDER
04AD	04AC	CYRILLIC SMALL LETTER TE WITH DESCENDER	CYRILLIC CAPITAL LETTER TE WITH DESCENDER
04AF	04AE	CYRILLIC SMALL LETTER STRAIGHT U	CYRILLIC CAPITAL LETTER STRAIGHT U
04B1	04B0	CYRILLIC SMALL LETTER STRAIGHT U WITH STROKE	CYRILLIC CAPITAL LETTER STRAIGHT U WITH STROKE
04B3	04B2	CYRILLIC SMALL LETTER HA WITH DESCENDER	CYRILLIC CAPITAL LETTER HA WITH DESCENDER
04B5	04B4	CYRILLIC SMALL LIGATURE TE TSE (ABKHASIAN)	CYRILLIC CAPITAL LIGATURE TE TSE (ABKHASIAN)
04B7	04B6	CYRILLIC SMALL LETTER CHE WITH DESCENDER	CYRILLIC CAPITAL LETTER CHE WITH DESCENDER
04B9	04B8	CYRILLIC SMALL LETTER CHE WITH VERTICAL STROKE	CYRILLIC CAPITAL LETTER CHE WITH VERTICAL STROKE
04BB	04BA	CYRILLIC SMALL LETTER SHHA	CYRILLIC CAPITAL LETTER SHHA
04BD	04BC	CYRILLIC SMALL LETTER ABKHASIAN CHE	CYRILLIC CAPITAL LETTER ABKHASIAN CHE
04BF	04BE	CYRILLIC SMALL LETTER ABKHASIAN CHE WITH DESCENDER	CYRILLIC CAPITAL LETTER ABKHASIAN CHE WITH DESCENDER
04C2	04C1	CYRILLIC SMALL LETTER ZHE WITH BREVE	CYRILLIC CAPITAL LETTER ZHE WITH BREVE
04C4	04C3	CYRILLIC SMALL LETTER KA WITH HOOK	CYRILLIC CAPITAL LETTER KA WITH HOOK
04C8	04C7	CYRILLIC SMALL LETTER EN WITH HOOK	CYRILLIC CAPITAL LETTER EN WITH HOOK
04CC	04CB	CYRILLIC SMALL LETTER KHAKASSIAN CHE	CYRILLIC CAPITAL LETTER KHAKASSIAN CHE
04D1	04D0	CYRILLIC SMALL LETTER A WITH BREVE	CYRILLIC CAPITAL LETTER A WITH BREVE
04D3	04D2	CYRILLIC SMALL LETTER A WITH DIAERESIS	CYRILLIC CAPITAL LETTER A WITH DIAERESIS
04D5	04D4	CYRILLIC SMALL LIGATURE A IE	CYRILLIC CAPITAL LIGATURE A IE
04D7	04D6	CYRILLIC SMALL LETTER IE WITH BREVE	CYRILLIC CAPITAL LETTER IE WITH BREVE
04D9	04D8	CYRILLIC SMALL LETTER SCHWA	CYRILLIC CAPITAL LETTER SCHWA
04DB	04DA	CYRILLIC SMALL LETTER SCHWA WITH DIAERESIS	CYRILLIC CAPITAL LETTER SCHWA WITH DIAERESIS
04DD	04DC	CYRILLIC SMALL LETTER ZHE WITH DIAERESIS	CYRILLIC CAPITAL LETTER ZHE WITH DIAERESIS
04DF	04DE	CYRILLIC SMALL LETTER ZE WITH DIAERESIS	CYRILLIC CAPITAL LETTER ZE WITH DIAERESIS

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
04E1	04E0	CYRILLIC SMALL LETTER ABKHASIAN DZE	CYRILLIC CAPITAL LETTER ABKHASIAN DZE
04E3	04E2	CYRILLIC SMALL LETTER I WITH MACRON	CYRILLIC CAPITAL LETTER I WITH MACRON
04E5	04E4	CYRILLIC SMALL LETTER I WITH DIAERESIS	CYRILLIC CAPITAL LETTER I WITH DIAERESIS
04E7	04E6	CYRILLIC SMALL LETTER O WITH DIAERESIS	CYRILLIC CAPITAL LETTER O WITH DIAERESIS
04E9	04E8	CYRILLIC SMALL LETTER BARRED O	CYRILLIC CAPITAL LETTER BARRED O
04EB	04EA	CYRILLIC SMALL LETTER BARRED O WITH DIAERESIS	CYRILLIC CAPITAL LETTER BARRED O WITH DIAERESIS
04EF	04EE	CYRILLIC SMALL LETTER U WITH MACRON	CYRILLIC CAPITAL LETTER U WITH MACRON
04F1	04F0	CYRILLIC SMALL LETTER U WITH DIAERESIS	CYRILLIC CAPITAL LETTER U WITH DIAERESIS
04F3	04F2	CYRILLIC SMALL LETTER U WITH DOUBLE ACUTE	CYRILLIC CAPITAL LETTER U WITH DOUBLE ACUTE
04F5	04F4	CYRILLIC SMALL LETTER CHE WITH DIAERESIS	CYRILLIC CAPITAL LETTER CHE WITH DIAERESIS
04F9	04F8	CYRILLIC SMALL LETTER YERU WITH DIAERESIS	CYRILLIC CAPITAL LETTER YERU WITH DIAERESIS
0561	0531	ARMENIAN SMALL LETTER AYB	ARMENIAN CAPITAL LETTER AYB
0562	0532	ARMENIAN SMALL LETTER BEN	ARMENIAN CAPITAL LETTER BEN
0563	0533	ARMENIAN SMALL LETTER GIM	ARMENIAN CAPITAL LETTER GIM
0564	0534	ARMENIAN SMALL LETTER DA	ARMENIAN CAPITAL LETTER DA
0565	0535	ARMENIAN SMALL LETTER ECH	ARMENIAN CAPITAL LETTER ECH
0566	0536	ARMENIAN SMALL LETTER ZA	ARMENIAN CAPITAL LETTER ZA
0567	0537	ARMENIAN SMALL LETTER EH	ARMENIAN CAPITAL LETTER EH
0568	0538	ARMENIAN SMALL LETTER ET	ARMENIAN CAPITAL LETTER ET
0569	0539	ARMENIAN SMALL LETTER TO	ARMENIAN CAPITAL LETTER TO
056A	053A	ARMENIAN SMALL LETTER ZHE	ARMENIAN CAPITAL LETTER ZHE
056B	053B	ARMENIAN SMALL LETTER INI	ARMENIAN CAPITAL LETTER INI
056C	053C	ARMENIAN SMALL LETTER LIWN	ARMENIAN CAPITAL LETTER LIWN
056D	053D	ARMENIAN SMALL LETTER XEH	ARMENIAN CAPITAL LETTER XEH
056E	053E	ARMENIAN SMALL LETTER CA	ARMENIAN CAPITAL LETTER CA
056F	053F	ARMENIAN SMALL LETTER KEN	ARMENIAN CAPITAL LETTER KEN
0570	0540	ARMENIAN SMALL LETTER HO	ARMENIAN CAPITAL LETTER HO
0571	0541	ARMENIAN SMALL LETTER JA	ARMENIAN CAPITAL LETTER JA
0572	0542	ARMENIAN SMALL LETTER GHAD	ARMENIAN CAPITAL LETTER GHAD
0573	0543	ARMENIAN SMALL LETTER CHEH	ARMENIAN CAPITAL LETTER CHEH
0574	0544	ARMENIAN SMALL LETTER MEN	ARMENIAN CAPITAL LETTER MEN
0575	0545	ARMENIAN SMALL LETTER YI	ARMENIAN CAPITAL LETTER YI
0576	0546	ARMENIAN SMALL LETTER NOW	ARMENIAN CAPITAL LETTER NOW
0577	0547	ARMENIAN SMALL LETTER SNA	ARMENIAN CAPITAL LETTER SHA
0578	0548	ARMENIAN SMALL LETTER VO	ARMENIAN CAPITAL LETTER VO
0579	0549	ARMENIAN SMALL LETTER CHA	ARMENIAN CAPITAL LETTER CHA
057A	054A	ARMENIAN SMALL LETTER PEH	ARMENIAN CAPITAL LETTER PEH

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
057B	054B	ARMENIAN SMALL LETTER JHEH	ARMENIAN CAPITAL LETTER JHEH
057C	054C	ARMENIAN SMALL LETTER RA	ARMENIAN CAPITAL LETTER RA
057D	054D	ARMENIAN SMALL LETTER SEH	ARMENIAN CAPITAL LETTER SEH
057E	054E	ARMENIAN SMALL LETTER VEW	ARMENIAN CAPITAL LETTER VEW
057F	054F	ARMENIAN SMALL LETTER TIWN	ARMENIAN CAPITAL LETTER TIWN
0580	0550	ARMENIAN SMALL LETTER REH	ARMENIAN CAPITAL LETTER REH
0581	0551	ARMENIAN SMALL LETTER CO	ARMENIAN CAPITAL LETTER CO
0582	0552	ARMENIAN SMALL LETTER YIWN	ARMENIAN CAPITAL LETTER YIWN
0583	0553	ARMENIAN SMALL LETTER PIWP	ARMENIAN CAPITAL LETTER PIWR
0584	0554	ARMENIAN SMALL LETTER KEH	ARMENIAN CAPITAL LETTER KEH
0585	0555	ARMENIAN SMALL LETTER OH	ARMENIAN CAPITAL LETTER OH
0586	0556	ARMENIAN SMALL LETTER FEH	ARMENIAN CAPITAL LETTER FEH
10D0	10A0	GEORGIAN LETTER AN	GEORGIAN CAPITAL LETTER AN (KHUTSURI)
10D1	10A1	GEORGIAN LETTER BAN	GEORGIAN CAPITAL LETTER BAN (KHUTSURI)
10D2	10A2	GEORGIAN LETTER GAN	GEORGIAN CAPITAL LETTER GAN (KHUTSURI)
10D3	10A3	GEORGIAN LETTER DON	GEORGIAN CAPITAL LETTER DON (KHUTSURI)
10D4	10A4	GEORGIAN LETTER EN	GEORGIAN CAPITAL LETTER EN (KHUTSURI)
10D5	10A5	GEORGIAN LETTER VIN	GEORGIAN CAPITAL LETTER VIN (KHUTSURI)
10D6	10A6	GEORGIAN LETTER ZEN	GEORGIAN CAPITAL LETTER ZEN (KHUTSURI)
10D7	10A7	GEORGIAN LETTER TAN	GEORGIAN CAPITAL LETTER TAN (KHUTSURI)
10D8	10A8	GEORGIAN LETTER IN	GEORGIAN CAPITAL LETTER IN (KHUTSURI)
10D9	10A9	GEORGIAN LETTER KAN	GEORGIAN CAPITAL LETTER KAN (KHUTSURI)
10DA	10AA	GEORGIAN LETTER LAS	GEORGIAN CAPITAL LETTER LAS (KHUTSURI)
10DB	10AB	GEORGIAN LETTER MAN	GEORGIAN CAPITAL LETTER MAN (KHUTSURI)
10DC	10AC	GEORGIAN LETTER NAR	GEORGIAN CAPITAL LETTER NAR (KHUTSURI)
10DD	10AD	GEORGIAN LETTER ON	GEORGIAN CAPITAL LETTER ON (KHUTSURI)
10DE	10AE	GEORGIAN LETTER PAR	GEORGIAN CAPITAL LETTER PAR (KHUTSURI)
10DF	10AF	GEORGIAN LETTER ZHAR	GEORGIAN CAPITAL LETTER ZHAR (KHUTSURI)
10E0	10B0	GEORGIAN LETTER RAE	GEORGIAN CAPITAL LETTER RAE (KHUTSURI)
10E1	10B1	GEORGIAN LETTER SAN	GEORGIAN CAPITAL LETTER SAN (KHUTSURI)
10E2	10B2	GEORGIAN LETTER TAR	GEORGIAN CAPITAL LETTER TAR (KHUTSURI)
10E3	10B3	GEORGIAN LETTER UN	GEORGIAN CAPITAL LETTER UN (KHUTSURI)

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
10E4	10B4	GEORGIAN LETTER PHAR	GEORGIAN CAPITAL LETTER PHAR (KHUTSURI)
10E5	10B5	GEORGIAN LETTER KHAR	GEORGIAN CAPITAL LETTER KHAR (KHUTSURI)
10E6	10B6	GEORGIAN LETTER GHAN	GEORGIAN CAPITAL LETTER GHAN (KHUTSURI)
10E7	10B7	GEORGIAN LETTER QAR	GEORGIAN CAPITAL LETTER QAR (KHUTSURI)
10E8	10B8	GEORGIAN LETTER SHIN	GEORGIAN CAPITAL LETTER SHIN (KHUTSURI)
10E9	10B9	GEORGIAN LETTER CHIN	GEORGIAN CAPITAL LETTER CHIN (KHUTSURI)
10EA	10BA	GEORGIAN LETTER CAN	GEORGIAN CAPITAL LETTER CAN (KHUTSURI)
10EB	10BB	GEORGIAN LETTER JIL	GEORGIAN CAPITAL LETTER JIL (KHUTSURI)
10EC	10BC	GEORGIAN LETTER CIL	GEORGIAN CAPITAL LETTER CIL (KHUTSURI)
10ED	10BD	GEORGIAN LETTER CHAR	GEORGIAN CAPITAL LETTER CHAR (KHUTSURI)
10EE	10BE	GEORGIAN LETTER XAN	GEORGIAN CAPITAL LETTER XAN (KHUTSURI)
10EF	10BF	GEORGIAN LETTER JHAN	GEORGIAN CAPITAL LETTER JHAN (KHUTSURI)
10F0	10C0	GEORGIAN LETTER HAE	GEORGIAN CAPITAL LETTER HAE (KHUTSURI)
10F1	10C1	GEORGIAN LETTER HE	GEORGIAN CAPITAL LETTER HE (KHUTSURI)
10F2	10C2	GEORGIAN LETTER HIE	GEORGIAN CAPITAL LETTER HIE (KHUTSURI)
10F3	10C3	GEORGIAN LETTER WE	GEORGIAN CAPITAL LETTER WE (KHUTSURI)
10F4	10C4	GEORGIAN LETTER HAR	GEORGIAN CAPITAL LETTER HAR (KHUTSURI)
10F5	10C5	GEORGIAN LETTER HOE	GEORGIAN CAPITAL LETTER HOE (KHUTSURI)
1E01	1E00	LATIN SMALL LETTER A WITH RING BELOW	LATIN CAPITAL LETTER A WITH RING BELOW
1E03	1E02	LATIN SMALL LETTER B WITH DOT ABOVE	LATIN CAPITAL LETTER B WITH DOT ABOVE
1E05	1E04	LATIN SMALL LETTER B WITH DOT BELOW	LATIN CAPITAL LETTER B WITH DOT BELOW
1E07	1E06	LATIN SMALL LETTER B WITH LINE BELOW	LATIN CAPITAL LETTER B WITH LINE BELOW
1E09	1E08	LATIN SMALL LETTER C WITH CEDILLA AND ACUTE	LATIN CAPITAL LETTER C WITH CEDILLA AND ACUTE
1E0B	1E0A	LATIN SMALL LETTER D WITH DOT ABOVE	LATIN CAPITAL LETTER D WITH DOT ABOVE
1E0D	1E0C	LATIN SMALL LETTER D WITH DOT BELOW	LATIN CAPITAL LETTER D WITH DOT BELOW
1E0F	1E0E	LATIN SMALL LETTER D WITH LINE BELOW	LATIN CAPITAL LETTER D WITH LINE BELOW
1E11	1E10	LATIN SMALL LETTER D WITH CEDILLA	LATIN CAPITAL LETTER D WITH CEDILLA
1E13	1E12	LATIN SMALL LETTER D WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER D WITH CIRCUMFLEX BELOW

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1E15	1E14	LATIN SMALL LETTER E WITH MACRON AND GRAVE	LATIN CAPITAL LETTER E WITH MACRON AND GRAVE
1E17	1E16	LATIN SMALL LETTER E WITH MACRON AND ACUTE	LATIN CAPITAL LETTER E WITH MACRON AND ACUTE
1E19	1E18	LATIN SMALL LETTER E WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER E WITH CIRCUMFLEX BELOW
1E1B	1E1A	LATIN SMALL LETTER E WITH TILDE BELOW	LATIN CAPITAL LETTER E WITH TILDE BELOW
1E1D	1E1C	LATIN SMALL LETTER E WITH CEDILLA AND BREVE	LATIN CAPITAL LETTER E WITH CEDILLA AND BREVE
1E1F	1E1E	LATIN SMALL LETTER F WITH DOT ABOVE	LATIN CAPITAL LETTER F WITH DOT ABOVE
1E21	1E20	LATIN SMALL LETTER G WITH MACRON	LATIN CAPITAL LETTER G WITH MACRON
1E23	1E22	LATIN SMALL LETTER H WITH DOT ABOVE	LATIN CAPITAL LETTER H WITH DOT ABOVE
1E25	1E24	LATIN SMALL LETTER H WITH DOT BELOW	LATIN CAPITAL LETTER H WITH DOT BELOW
1E27	1E26	LATIN SMALL LETTER H WITH DIAERESIS	LATIN CAPITAL LETTER H WITH DIAERESIS
1E29	1E28	LATIN SMALL LETTER H WITH CEDILLA	LATIN CAPITAL LETTER H WITH CEDILLA
1E2B	1E2A	LATIN SMALL LETTER H WITH BREVE BELOW	LATIN CAPITAL LETTER H WITH BREVE BELOW
1E2D	1E2C	LATIN SMALL LETTER I WITH TILDE BELOW	LATIN CAPITAL LETTER I WITH TILDE BELOW
1E2F	1E2E	LATIN SMALL LETTER I WITH DIAERESIS AND ACUTE	LATIN CAPITAL LETTER I WITH DIAERESIS AND ACUTE
1E31	1E30	LATIN SMALL LETTER K WITH ACUTE	LATIN CAPITAL LETTER K WITH ACUTE
1E33	1E32	LATIN SMALL LETTER K WITH DOT BELOW	LATIN CAPITAL LETTER K WITH DOT BELOW
1E35	1E34	LATIN SMALL LETTER K WITH LINE BELOW	LATIN CAPITAL LETTER K WITH LINE BELOW
1E37	1E36	LATIN SMALL LETTER L WITH DOT BELOW	LATIN CAPITAL LETTER L WITH DOT BELOW
1E39	1E38	LATIN SMALL LETTER L WITH DOT BELOW AND MACRON	LATIN CAPITAL LETTER L WITH DOT BELOW AND MACRON
1E3B	1E3A	LATIN SMALL LETTER L WITH LINE BELOW	LATIN CAPITAL LETTER L WITH LINE BELOW
1E3D	1E3C	LATIN SMALL LETTER L WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER L WITH CIRCUMFLEX BELOW
1E3F	1E3E	LATIN SMALL LETTER M WITH ACUTE	LATIN CAPITAL LETTER M WITH ACUTE
1E41	1E40	LATIN SMALL LETTER M WITH DOT ABOVE	LATIN CAPITAL LETTER M WITH DOT ABOVE
1E43	1E42	LATIN SMALL LETTER M WITH DOT BELOW	LATIN CAPITAL LETTER M WITH DOT BELOW
1E45	1E44	LATIN SMALL LETTER N WITH DOT ABOVE	LATIN CAPITAL LETTER N WITH DOT ABOVE
1E47	1E46	LATIN SMALL LETTER N WITH DOT BELOW	LATIN CAPITAL LETTER N WITH DOT BELOW
1E49	1E48	LATIN SMALL LETTER N WITH LINE BELOW	LATIN CAPITAL LETTER N WITH LINE BELOW
1E4B	1E4A	LATIN SMALL LETTER N WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER N WITH CIRCUMFLEX BELOW

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1E4D	1E4C	LATIN SMALL LETTER O WITH TILDE AND ACUTE	LATIN CAPITAL LETTER O WITH TILDE AND ACUTE
1E4F	1E4E	LATIN SMALL LETTER O WITH TILDE AND DIAERESIS	LATIN CAPITAL LETTER O WITH TILDE AND DIAERESIS
1E51	1E50	LATIN SMALL LETTER O WITH MACRON AND GRAVE	LATIN CAPITAL LETTER O WITH MACRON AND GRAVE
1E53	1E52	LATIN SMALL LETTER O WITH MACRON AND ACUTE	LATIN CAPITAL LETTER O WITH MACRON AND ACUTE
1E55	1E54	LATIN SMALL LETTER P WITH ACUTE	LATIN CAPITAL LETTER P WITH ACUTE
1E57	1E56	LATIN SMALL LETTER P WITH DOT ABOVE	LATIN CAPITAL LETTER P WITH DOT ABOVE
1E59	1E58	LATIN SMALL LETTER R WITH DOT ABOVE	LATIN CAPITAL LETTER R WITH DOT ABOVE
1E5B	1E5A	LATIN SMALL LETTER R WITH DOT BELOW	LATIN CAPITAL LETTER R WITH DOT BELOW
1E5D	1E5C	LATIN SMALL LETTER R WITH DOT BELOW AND MACRON	LATIN CAPITAL LETTER R WITH DOT BELOW AND MACRON
1E5F	1E5E	LATIN SMALL LETTER R WITH LINE BELOW	LATIN CAPITAL LETTER R WITH LINE BELOW
1E61	1E60	LATIN SMALL LETTER S WITH DOT ABOVE	LATIN CAPITAL LETTER S WITH DOT ABOVE
1E63	1E62	LATIN SMALL LETTER S WITH DOT BELOW	LATIN CAPITAL LETTER S WITH DOT BELOW
1E65	1E64	LATIN SMALL LETTER S WITH ACUTE AND DOT ABOVE	LATIN CAPITAL LETTER S WITH ACUTE AND DOT ABOVE
1E67	1E66	LATIN SMALL LETTER S WITH CARON AND DOT ABOVE	LATIN CAPITAL LETTER S WITH CARON AND DOT ABOVE
1E69	1E68	LATIN SMALL LETTER S WITH DOT BELOW AND DOT ABOVE	LATIN CAPITAL LETTER S WITH DOT BELOW AND DOT ABOVE
1E6B	1E6A	LATIN SMALL LETTER T WITH DOT ABOVE	LATIN CAPITAL LETTER T WITH DOT ABOVE
1E6D	1E6C	LATIN SMALL LETTER T WITH DOT BELOW	LATIN CAPITAL LETTER T WITH DOT BELOW
1E6F	1E6E	LATIN SMALL LETTER T WITH LINE BELOW	LATIN CAPITAL LETTER T WITH LINE BELOW
1E71	1E70	LATIN SMALL LETTER T WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER T WITH CIRCUMFLEX BELOW
1E73	1E72	LATIN SMALL LETTER U WITH DIAERESIS BELOW	LATIN CAPITAL LETTER U WITH DIAERESIS BELOW
1E75	1E74	LATIN SMALL LETTER U WITH TILDE BELOW	LATIN CAPITAL LETTER U WITH TILDE BELOW
1E77	1E76	LATIN SMALL LETTER U WITH CIRCUMFLEX BELOW	LATIN CAPITAL LETTER U WITH CIRCUMFLEX BELOW
1E79	1E78	LATIN SMALL LETTER U WITH TILDE AND ACUTE	LATIN CAPITAL LETTER U WITH TILDE AND ACUTE
1E7B	1E7A	LATIN SMALL LETTER U WITH MACRON AND DIAERESIS	LATIN CAPITAL LETTER U WITH MACRON AND DIAERESIS
1E7D	1E7C	LATIN SMALL LETTER V WITH TILDE	LATIN CAPITAL LETTER V WITH TILDE
1E7F	1E7E	LATIN SMALL LETTER V WITH DOT BELOW	LATIN CAPITAL LETTER V WITH DOT BELOW
1E81	1E80	LATIN SMALL LETTER W WITH GRAVE	LATIN CAPITAL LETTER W WITH GRAVE
1E83	1E82	LATIN SMALL LETTER W WITH ACUTE	LATIN CAPITAL LETTER W WITH ACUTE



Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1E85	1E84	LATIN SMALL LETTER W WITH DIAERESIS	LATIN CAPITAL LETTER W WITH DIAERESIS
1E87	1E86	LATIN SMALL LETTER W WITH DOT ABOVE	LATIN CAPITAL LETTER W WITH DOT ABOVE
1E89	1E88	LATIN SMALL LETTER W WITH DOT BELOW	LATIN CAPITAL LETTER W WITH DOT BELOW
1E8B	1E8A	LATIN SMALL LETTER X WITH DOT ABOVE	LATIN CAPITAL LETTER X WITH DOT ABOVE
1E8D	1E8C	LATIN SMALL LETTER X WITH DIAERESIS	LATIN CAPITAL LETTER X5 WITH DIAERESIS
1E8F	1E8E	LATIN SMALL LETTER Y WITH DOT ABOVE	LATIN CAPITAL LETTER Y WITH DOT ABOVE
1E91	1E90	LATIN SMALL LETTER Z WITH CIRCUMFLEX	LATIN CAPITAL LETTER Z WITH CIRCUMFLEX
1E93	1E92	LATIN SMALL LETTER Z WITH DOT BELOW	LATIN CAPITAL LETTER Z WITH DOT BELOW
1E95	1E94	LATIN SMALL LETTER Z WITH LINE BELOW	LATIN CAPITAL LETTER Z WITH LINE BELOW
1EA1	1EA0	LATIN SMALL LETTER A WITH DOT BELOW	LATIN CAPITAL LETTER A WITH DOT BELOW
1EA3	1EA2	LATIN SMALL LETTER A WITH HOOK ABOVE	LATIN CAPITAL LETTER A WITH HOOK ABOVE
1EA5	1EA4	LATIN SMALL LETTER A WITH CIRCUMFLEX AND ACUTE	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND ACUTE
1EA7	1EA6	LATIN SMALL LETTER A WITH CIRCUMFLEX AND GRAVE	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND GRAVE
1EA9	1EA8	LATIN SMALL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND HOOK ABOVE
1EAB	1EAA	LATIN SMALL LETTER A WITH CIRCUMFLEX AND TILDE	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND TILDE
1EAD	1EAC	LATIN SMALL LETTER A WITH CIRCUMFLEX AND DOT BELOW	LATIN CAPITAL LETTER A WITH CIRCUMFLEX AND DOT BELOW
1EAF	1EAE	LATIN SMALL LETTER A WITH BREVE AND ACUTE	LATIN CAPITAL LETTER A WITH BREVE AND ACUTE
1EB1	1EB0	LATIN SMALL LETTER A WITH BREVE AND GRAVE	LATIN CAPITAL LETTER A WITH BREVE AND GRAVE
1EB3	1EB2	LATIN SMALL LETTER A WITH BREVE AND HOOK ABOVE	LATIN CAPITAL LETTER A WITH BREVE AND HOOK ABOVE
1EB5	1EB4	LATIN SMALL LETTER A WITH BREVE AND TILDE	LATIN CAPITAL LETTER A WITH BREVE AND TILDE
1EB7	1EB6	LATIN SMALL LETTER A WITH BREVE AND DOT BELOW	LATIN CAPITAL LETTER A WITH BREVE AND DOT BELOW
1EB9	1EB8	LATIN SMALL LETTER E WITH DOT BELOW	LATIN CAPITAL LETTER E WITH DOT BELOW
1EBB	1EBA	LATIN SMALL LETTER E WITH HOOK ABOVE	LATIN CAPITAL LETTER E WITH HOOK ABOVE
1EBD	1EBC	LATIN SMALL LETTER E WITH TILDE	LATIN CAPITAL LETTER E WITH TILDE
1EBF	1EBE	LATIN SMALL LETTER E WITH CIRCUMFLEX AND ACUTE	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND ACUTE
1EC1	1ECO	LATIN SMALL LETTER E WITH CIRCUMFLEX AND GRAVE	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND GRAVE
1EC3	1EC2	LATIN SMALL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND HOOK ABOVE
1EC5	1EC4	LATIN SMALL LETTER E WITH CIRCUMFLEX AND TILDE	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND TILDE

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1EC7	1EC6	LATIN SMALL LETTER E WITH CIRCUMFLEX AND DOT BELOW	LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND DOT BELOW
1EC9	1EC8	LATIN SMALL LETTER I WITH HOOK ABOVE	LATIN CAPITAL LETTER I WITH HOOK ABOVE
1ECB	1ECA	LATIN SMALL LETTER I WITH DOT BELOW	LATIN CAPITAL LETTER I WITH DOT BELOW
1ECD	1ECC	LATIN SMALL LETTER O WITH DOT BELOW	LATIN CAPITAL LETTER O WITH DOT BELOW
1ECF	1ECE	LATIN SMALL LETTER O WITH HOOK ABOVE	LATIN CAPITAL LETTER O WITH HOOK ABOVE
1ED1	1ED0	LATIN SMALL LETTER O WITH CIRCUMFLEX AND ACUTE	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND ACUTE
1ED3	1ED2	LATIN SMALL LETTER O WITH CIRCUMFLEX AND GRAVE	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND GRAVE
1ED5	1ED4	LATIN SMALL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND HOOK ABOVE
1ED7	1ED6	LATIN SMALL LETTER O WITH CIRCUMFLEX AND TILDE	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND TILDE
1ED9	1ED8	LATIN SMALL LETTER O WITH CIRCUMFLEX AND DOT BELOW	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND DOT BELOW
1EDB	1EDA	LATIN SMALL LETTER O WITH HORN AND ACUTE	LATIN CAPITAL LETTER O WITH HORN AND ACUTE
1EDD	1EDC	LATIN SMALL LETTER O WITH HORN AND GRAVE	LATIN CAPITAL LETTER O WITH HORN AND GRAVE
1EDF	1EDE	LATIN SMALL LETTER O WITH HORN AND HOOK ABOVE	LATIN CAPITAL LETTER O WITH HORN AND HOOK ABOVE
1EE1	1EE0	LATIN SMALL LETTER O WITH HORN AND TILDE	LATIN CAPITAL LETTER O WITH HORN AND TILDE
1EE3	1EE2	LATIN SMALL LETTER O WITH HORN AND DOT BELOW	LATIN CAPITAL LETTER O WITH HORN AND DOT BELOW
1EE5	1EE4	LATIN SMALL LETTER U WITH DOT BELOW	LATIN CAPITAL LETTER U WITH DOT BELOW
1EE7	1EE6	LATIN SMALL LETTER U WITH HOOK ABOVE	LATIN CAPITAL LETTER U WITH HOOK ABOVE
1EE9	1EE8	LATIN SMALL LETTER U WITH HORN AND ACUTE	LATIN CAPITAL LETTER U WITH HORN AND ACUTE
1EEB	1EEA	LATIN SMALL LETTER U WITH HORN AND GRAVE	LATIN CAPITAL LETTER U WITH HORN AND GRAVE
1EED	1EEC	LATIN SMALL LETTER U WITH HORN AND HOOK ABOVE	LATIN CAPITAL LETTER U WITH HORN AND HOOK ABOVE
1EEF	1EEE	LATIN SMALL LETTER U WITH HORN AND TILDE	LATIN CAPITAL LETTER U WITH HORN AND TILDE
1EF1	1EF0	LATIN SMALL LETTER U WITH HORN AND DOT BELOW	LATIN CAPITAL LETTER U WITH HORN AND DOT BELOW
1EF3	1EF2	LATIN SMALL LETTER Y WITH GRAVE	LATIN CAPITAL LETTER Y WITH GRAVE
1EF5	1EF4	LATIN SMALL LETTER Y WITH DOT BELOW	LATIN CAPITAL LETTER Y WITH DOT BELOW
1EF7	1EF6	LATIN SMALL LETTER Y WITH HOOK ABOVE	LATIN CAPITAL LETTER Y WITH HOOK ABOVE
1EF9	1EF8	LATIN SMALL LETTER Y WITH TILDE	LATIN CAPITAL LETTER Y WITH TILDE
1F00	1F08	GREEK SMALL LETTER ALPHA WITH PSILI	GREEK CAPITAL LETTER ALPHA WITH PSILI
1F01	1F09	GREEK SMALL LETTER ALPHA WITH DASIA	GREEK CAPITAL LETTER ALPHA WITH DASIA

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1F02	1F0A	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA
1F03	1F0B	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA
1F04	1F0C	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA
1F05	1F0D	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA
1F06	1F0E	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI
1F07	1F0F	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI
1F10	1F18	GREEK SMALL LETTER EPSILON WITH PSILI	GREEK CAPITAL LETTER EPSILON WITH PSILI
1F11	1F19	GREEK SMALL LETTER EPSILON WITH DASIA	GREEK CAPITAL LETTER EPSILON WITH DASIA
1F12	1F1A	GREEK SMALL LETTER EPSILON WITH PSILI AND VARIA	GREEK CAPITAL LETTER EPSILON WITH PSILI AND VARIA
1F13	1F1B	GREEK SMALL LETTER EPSILON WITH DASIA AND VARIA	GREEK CAPITAL LETTER EPSILON WITH DASIA AND VARIA
1F14	1F1C	GREEK SMALL LETTER EPSILON WITH PSILI AND OXIA	GREEK CAPITAL LETTER EPSILON WITH PSILI AND OXIA
1F15	1F1D	GREEK SMALL LETTER EPSILON WITH DASIA AND OXIA	GREEK CAPITAL LETTER EPSILON WITH DASIA AND OXIA
1F20	1F28	GREEK SMALL LETTER ETA WITH PSILI	GREEK CAPITAL LETTER ETA WITH PSILI
1F21	1F29	GREEK SMALL LETTER ETA WITH DASIA	GREEK CAPITAL LETTER ETA WITH DASIA
1F22	1F2A	GREEK SMALL LETTER ETA WITH PSILI AND VARIA	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA
1F23	1F2B	GREEK SMALL LETTER ETA WITH DASIA AND VARIA	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA
1F24	1F2C	GREEK SMALL LETTER ETA WITH PSILI AND OXIA	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA
1F25	1F2D	GREEK SMALL LETTER ETA WITH DASIA AND OXIA	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA
1F26	1F2E	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI
1F27	1F2F	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI
1F30	1F38	GREEK SMALL LETTER IOTA WITH PSILI	GREEK CAPITAL LETTER IOTA WITH PSILI
1F31	1F39	GREEK SMALL LETTER IOTA WITH DASIA	GREEK CAPITAL LETTER IOTA WITH DASIA
1F32	1F3A	GREEK SMALL LETTER IOTA WITH PSILI AND VARIA	GREEK CAPITAL LETTER IOTA WITH PSILI AND VARIA
1F33	1F3B	GREEK SMALL LETTER IOTA WITH DASIA AND VARIA	GREEK CAPITAL LETTER IOTA WITH DASIA AND VARIA
1F34	1F3C	GREEK SMALL LETTER IOTA WITH PSILI AND OXIA	GREEK CAPITAL LETTER IOTA WITH PSILI AND OXIA
1F35	1F3D	GREEK SMALL LETTER IOTA WITH DASIA AND OXIA	GREEK CAPITAL LETTER IOTA WITH DASIA AND OXIA
1F36	1F3E	GREEK SMALL LETTER IOTA WITH PSILI AND PERISPOMENI	GREEK CAPITAL LETTER IOTA WITH PSILI AND PERISPOMENI
1F37	1F3F	GREEK SMALL LETTER IOTA WITH DASIA AND PERISPOMENI	GREEK CAPITAL LETTER IOTA WITH DASIA AND PERISPOMENI

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1F40	1F48	GREEK SMALL LETTER OMICRON WITH PSILI	GREEK CAPITAL LETTER OMICRON WITH PSILI
1F41	1F49	GREEK SMALL LETTER OMICRON WITH DASIA	GREEK CAPITAL LETTER OMICRON WITH DASIA
1F42	1F4A	GREEK SMALL LETTER OMICRON WITH PSILI AND VARIA	GREEK CAPITAL LETTER OMICRON WITH PSILI AND VARIA
1F43	1F4B	GREEK SMALL LETTER OMICRON WITH DASIA AND VARIA	GREEK CAPITAL LETTER OMICRON WITH DASIA AND VARIA
1F44	1F4C	GREEK SMALL LETTER OMICRON WITH PSILI AND OXIA	GREEK CAPITAL LETTER OMICRON WITH PSILI AND OXIA
1F45	1F4D	GREEK SMALL LETTER OMICRON WITH DASIA AND OXIA	GREEK CAPITAL LETTER OMICRON WITH DASIA AND OXIA
1F51	1F59	GREEK SMALL LETTER UPSILON WITH DASIA	GREEK CAPITAL LETTER UPSILON WITH OASIS
1F53	1F5B	GREEK SMALL LETTER UPSILON WITH DASIA AND VARIA	GREEK CAPITAL LETTER UPSILON WITH DASIA AND VARIA
1F55	1F5D	GREEK SMALL LETTER UPSILON WITH DASIA AND OXIA	GREEK CAPITAL LETTER UPSILON WITH DASIA AND OXIA
1F57	1F5F	GREEK SMALL LETTER UPSILON WITH DASIA AND PERISPOMENI	GREEK CAPITAL LETTER UPSILON WITH DASIA AND PERISPOMENI
1F60	1F68	GREEK SMALL LETTER OMEGA WITH PSILI	GREEK CAPITAL LETTER OMEGA WITH PSILI
1F61	1F69	GREEK SMALL LETTER OMEGA WITH DASIA	GREEK CAPITAL LETTER OMEGA WITH DASIA
1F62	1F6A	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA
1F63	1F6B	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA
1F64	1F6C	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA
1F65	1F6D	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA
1F66	1F6E	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI
1F67	1F6F	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI
1F80	1F88	GREEK SMALL LETTER ALPHA WITH PSILI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PROSGEGRAMMENI
1F81	1F89	GREEK SMALL LETTER ALPHA WITH DASIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PROSGEGRAMMENI
1F82	1F8A	GREEK SMALL LETTER ALPHA WITH PSILI AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH PSILI AND VARIA AND PROSGEGRAMMENI
1F83	1F8B	GREEK SMALL LETTER ALPHA WITH DASIA AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH DASIA AND VARIA AND PROSGEGRAMMENI
1F84	1F8C	GREEK SMALL LETTER ALPHA WITH PSILI AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH PSILI AND OXIA AND PROSGEGRAMMENI
1F85	1F8D	GREEK SMALL LETTER ALPHA WITH DASIA AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH DASIA AND OXIA AND PROSGEGRAMMENI
1F86	1F8E	GREEK SMALL LETTER ALPHA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
1F87	1F8F	GREEK SMALL LETTER ALPHA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ALPHA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
1F90	1F98	GREEK SMALL LETTER ETA WITH PSILI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH PSILI AND PROSGEGRAMMENI
1F91	1F99	GREEK SMALL LETTER ETA WITH DASIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH DASIA AND PROSGEGRAMMENI
1F92	1F9A	GREEK SMALL LETTER ETA WITH PSILI AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH PSILI AND VARIA AND PROSGEGRAMMENI
1F93	1F9B	GREEK SMALL LETTER ETA WITH DASIA AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH DASIA AND VARIA AND PROSGEGRAMMENI
1F94	1F9C	GREEK SMALL LETTER ETA WITH PSILI AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH PSILI AND OXIA AND PROSGEGRAMMENI
1F95	1F9D	GREEK SMALL LETTER ETA WITH DASIA AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH DASIA AND OXIA AND PROSGEGRAMMENI
1F96	1F9E	GREEK SMALL LETTER ETA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
1F97	1F9F	GREEK SMALL LETTER ETA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER ETA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
1FA0	1FA8	GREEK SMALL LETTER OMEGA WITH PSILI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PROSGEGRAMMENI
1FA1	1FA9	GREEK SMALL LETTER OMEGA WITH DASIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PROSGEGRAMMENI
1FA2	1FAA	GREEK SMALL LETTER OMEGA WITH PSILI AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH PSILI AND VARIA AND PROSGEGRAMMENI
1FA3	1FAB	GREEK SMALL LETTER OMEGA WITH DASIA AND VARIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH DASIA AND VARIA AND PROSGEGRAMMENI
1FA4	1FAC	GREEK SMALL LETTER OMEGA WITH PSILI AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH PSILI AND OXIA AND PROSGEGRAMMENI
1FA5	1FAD	GREEK SMALL LETTER OMEGA WITH DASIA AND OXIA AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH DASIA AND OXIA AND PROSGEGRAMMENI
1FA6	1FAE	GREEK SMALL LETTER OMEGA WITH PSILI AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH PSILI AND PERISPOMENI AND PROSGEGRAMMENI
1FA7	1FAF	GREEK SMALL LETTER OMEGA WITH DASIA AND PERISPOMENI AND YPOGEGRAMMENI	GREEK CAPITAL LETTER OMEGA WITH DASIA AND PERISPOMENI AND PROSGEGRAMMENI
1FB0	1FB8	GREEK SMALL LETTER ALPHA WITH VRACHY	GREEK CAPITAL LETTER ALPHA WITH VRACHY
1FB1	1FB9	GREEK SMALL LETTER ALPHA WITH MACRON	GREEK CAPITAL LETTER ALPHA WITH MACRON
1FD0	1FD8	GREEK SMALL LETTER IOTA WITH VRACHY	GREEK CAPITAL LETTER IOTA WITH VRACHY
1FD1	1FD9	GREEK SMALL LETTER IOTA WITH MACRON	GREEK CAPITAL LETTER IOTA WITH MACRON
1FE0	1FE8	GREEK SMALL LETTER UPSILON WITH VRACHY	GREEK CAPITAL LETTER UPSILON WITH VRACHY
1FE1	1FE9	GREEK SMALL LETTER UPSILON WITH MACRON	GREEK CAPITAL LETTER UPSILON WITH MACRON

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
24D0	24B6	CIRCLED LATIN SMALL LETTER A	CIRCLED LATIN CAPITAL LETTER A
24D1	24B7	CIRCLED LATIN SMALL LETTER B	CIRCLED LATIN CAPITAL LETTER B
24D2	24B8	CIRCLED LATIN SMALL LETTER C	CIRCLED LATIN CAPITAL LETTER C
24D3	24B9	CIRCLED LATIN SMALL LETTER D	CIRCLED LATIN CAPITAL LETTER D
24D4	24BA	CIRCLED LATIN SMALL LETTER E	CIRCLED LATIN CAPITAL LETTER E
24D5	24BB	CIRCLED LATIN SMALL LETTER F	CIRCLED LATIN CAPITAL LETTER F
24D6	24BC	CIRCLED LATIN SMALL LETTER G	CIRCLED LATIN CAPITAL LETTER G
24D7	24BD	CIRCLED LATIN SMALL LETTER H	CIRCLED LATIN CAPITAL LETTER H
24D8	24BE	CIRCLED LATIN SMALL LETTER I	CIRCLED LATIN CAPITAL LETTER I
24D9	24BF	CIRCLED LATIN SMALL LETTER J	CIRCLED LATIN CAPITAL LETTER J
24DA	24C0	CIRCLED LATIN SMALL LETTER K	CIRCLED LATIN CAPITAL LETTER K
24DB	24C1	CIRCLED LATIN SMALL LETTER L	CIRCLED LATIN CAPITAL LETTER L
24DC	24C2	CIRCLED LATIN SMALL LETTER M	CIRCLED LATIN CAPITAL LETTER M
24DD	24C3	CIRCLED LATIN SMALL LETTER N	CIRCLED LATIN CAPITAL LETTER N
24DE	24C4	CIRCLED LATIN SMALL LETTER O	CIRCLED LATIN CAPITAL LETTER O
24DF	24C5	CIRCLED LATIN SMALL LETTER P	CIRCLED LATIN CAPITAL LETTER P
24E0	24C6	CIRCLED LATIN SMALL LETTER Q	CIRCLED LATIN CAPITAL LETTER Q
24E1	24C7	CIRCLED LATIN SMALL LETTER R	CIRCLED LATIN CAPITAL LETTER R
24E2	24C8	CIRCLED LATIN SMALL LETTER S	CIRCLED LATIN CAPITAL LETTER S
24E3	24C9	CIRCLED LATIN SMALL LETTER T	CIRCLED LATIN CAPITAL LETTER T
24E4	24CA	CIRCLED LATIN SMALL LETTER U	CIRCLED LATIN CAPITAL LETTER U
24E5	24CB	CIRCLED LATIN SMALL LETTER V	CIRCLED LATIN CAPITAL LETTER V
24E6	24CC	CIRCLED LATIN SMALL LETTER W	CIRCLED LATIN CAPITAL LETTER W
24E7	24CD	CIRCLED LATIN SMALL LETTER X	CIRCLED LATIN CAPITAL LETTER X
24E8	24CE	CIRCLED LATIN SMALL LETTER Y	CIRCLED LATIN CAPITAL LETTER Y
24E9	24CF	CIRCLED LATIN SMALL LETTER Z	CIRCLED LATIN CAPITAL LETTER Z
FF41	FF21	FULLWIDTH LATIN SMALL LETTER A	FULLWIDTH LATIN CAPITAL LETTER A
FF42	FF22	FULLWIDTH LATIN SMALL LETTER B	FULLWIDTH LATIN CAPITAL LETTER B
FF43	FF23	FULLWIDTH LATIN SMALL LETTER C	FULLWIDTH LATIN CAPITAL LETTER C
FF44	FF24	FULLWIDTH LATIN SMALL LETTER D	FULLWIDTH LATIN CAPITAL LETTER D
FF45	FF25	FULLWIDTH LATIN SMALL LETTER E	FULLWIDTH LATIN CAPITAL LETTER E
FF46	FF26	FULLWIDTH LATIN SMALL LETTER F	FULLWIDTH LATIN CAPITAL LETTER F
FF47	FF27	FULLWIDTH LATIN SMALL LETTER G	FULLWIDTH LATIN CAPITAL LETTER G
FF48	FF28	FULLWIDTH LATIN SMALL LETTER H	FULLWIDTH LATIN CAPITAL LETTER H
FF49	FF29	FULLWIDTH LATIN SMALL LETTER I	FULLWIDTH LATIN CAPITAL LETTER I
FF4A	FF2A	FULLWIDTH LATIN SMALL LETTER J	FULLWIDTH LATIN CAPITAL LETTER J
FF4B	FF2B	FULLWIDTH LATIN SMALL LETTER K	FULLWIDTH LATIN CAPITAL LETTER K

Lowercase code point	Uppercase code point	Lowercase character description	Uppercase character description
FF4C	FF2C	FULLWIDTH LATIN SMALL LETTER L	FULLWIDTH LATIN CAPITAL LETTER L
FF4D	FF2D	FULLWIDTH LATIN SMALL LETTER M	FULLWIDTH LATIN CAPITAL LETTER M
FF4E	FF2E	FULLWIDTH LATIN SMALL LETTER N	FULLWIDTH LATIN CAPITAL LETTER N
FF4F	FF2F	FULLWIDTH LATIN SMALL LETTER O	FULLWIDTH LATIN CAPITAL LETTER O
FF50	FF30	FULLWIDTH LATIN SMALL LETTER P	FULLWIDTH LATIN CAPITAL LETTER P
FF51	FF31	FULLWIDTH LATIN SMALL LETTER Q	FULLWIDTH LATIN CAPITAL LETTER Q
FF52	FF32	FULLWIDTH LATIN SMALL LETTER R	FULLWIDTH LATIN CAPITAL LETTER R
FF53	FF33	FULLWIDTH LATIN SMALL LETTER S	FULLWIDTH LATIN CAPITAL LETTER S
FF54	FF34	FULLWIDTH LATIN SMALL LETTER T	FULLWIDTH LATIN CAPITAL LETTER T
FF55	FF35	FULLWIDTH LATIN SMALL LETTER U	FULLWIDTH LATIN CAPITAL LETTER U
FF56	FF36	FULLWIDTH LATIN SMALL LETTER V	FULLWIDTH LATIN CAPITAL LETTER V
FF57	FF37	FULLWIDTH LATIN SMALL LETTER W	FULLWIDTH LATIN CAPITAL LETTER W
FF58	FF38	FULLWIDTH LATIN SMALL LETTER X	FULLWIDTH LATIN CAPITAL LETTER X
FF59	FF39	FULLWIDTH LATIN SMALL LETTER Y	FULLWIDTH LATIN CAPITAL LETTER Y
FF5A	FF3A	FULLWIDTH LATIN SMALL LETTER Z	FULLWIDTH LATIN CAPITAL LETTER Z

## Unicode Private Use Area mapping on IBM i

IBM i can allow a custom mapping between the EBCDIC user-defined character (UDC) range and the Unicode private use area (PUA) range for the mapping between Unicode and CCSID 1371.

The standard conversion support maps the first EBCDIC UDC to the first Unicode PUA, then the second EBCDIC UDC to the second Unicode PUA and so on.

### Background

In Ideographic languages, such as Chinese, a need to create custom ideographs to represent new words not in the current standards exists. This custom ideograph is supported by “user-defined characters” that extend an encoding standard. EBCDIC calls these custom ideographs *user-defined characters* (UDC) and the standard has an allocated range for them. Unicode calls these custom ideographs *private use area* (PUA) characters and also has a range that is defined for them. These definitions do not use the same hexadecimal ranges. In addition, these user-defined characters are not standardized and their usage is unique by user. Therefore, user-defined characters can cause issues when mapping between EBCDIC and Unicode.

### Support provided

Customers can indicate, on a system level, how they want the mapping between the UDC characters and the PUA characters to map. To simplify this text, further use of the abbreviation of “PUA” in this text is used to mean both EBCDIC UDC and Unicode PUA.

### Note:

- CCSID 1371 is the only CCSID for which this support is provided.

- Each system image can have only one version of each direction updated.
- This update does not change the base mapping, only the UCD mapping.
- An IPL is required to activate any change.
- This support does not support Unicode UCDs outside the BMP.
- For more information about the Unicode standard, see <http://www.unicode.org>.
- For more information about the EBCDIC standard, see <http://www.ibm.com/software/globalization/g11n-res.jsp>.

### **How to use the custom PUA mapping support**

Several steps are required to use the PUA mapping support.

1. Copy the provided sample source files to create a copy for you to work on.
2. Modify the provided sample source files (see appendix A) to reflect the mapping you want to happen.
3. Call the create table command to create the new \*TBL objects for the system to use.
4. End all the subsystems. Then, call QTQUSRCV to rebuild the systems CCSID index to reflect the changes.
5. IPL the system to activate the changes.

### **Copy the sample source**

Copy the provided sample source files to create a version for you to modify.

The default mappings for you to base your mappings on are provided. To copy the default mappings, see “Default mapping for EBCDIC UDC to Unicode PUA on IBM i” on page 437 and “Default mapping for Unicode PUA to EBCDIC UDC on IBM i” on page 512. The rest of this document makes the assumption that you copied these two samples to a file called **QTBLSRC** in a library that is called **MYLIB** and called them **QE2UMAP** and **QU2EMAP**.

### **Modify the sample source files**

Modify the provided sample source files to reflect the mapping you want to happen.

You can use any text editor to modify the mappings, for example use the following sample CL to use the EDTF editor. Replace **MYLIB** with the library you used to save your sample source file.

```
EDTF FILE(MYLIB/QTBLSRC) MBR(MYE2UMAP)
```

When in the editor, you see the source that is used to modify the mappings.

The layout of the source starts with one line that tells the source type. Either Q00005 for EBCDIC to Unicode or Q00006 for Unicode to EBCDIC, in the first six columns of the file. The source type is followed by a comment line to the end of the first record.

The rest of the records in the file define the mapping. The first four hexadecimal digits (in column 1 - 4) are the source code point and the next 4 (in column 6 - 9) are the target code point. An optional comment can start in column 10 to the end of the record.

### **Note:**

- The source needs to include the PUA mappings you want to override. Any mapping that is not defined uses the default mapping. You need to change only the PUA values that are of a concern to you.
- Two different source members are used, one for each direction; therefore, you can have different mappings to and from Unicode if you want.
- Provide mappings for only hexadecimal codes that are in the PUA ranges defined by the standard. Codes outside these ranges are ignored.
- If the source has two or more entries for the same from hexadecimal code, the last one in the file is used.
- You cannot define a PUA that uses the extended Unicode (surrogate) range. All Unicode PUA hexadecimal codes must be in the range of E000 to F8FF.



### Example EBCDIC to Unicode source

As an example, in the following segment of source, the default mapping for the PUA C242 is changed. By default it would map to E001 (since it is the second PUA) but this example source has it mapping to E040 and you would use this source.

```
Q00005  ! Created by Pat for Dept ABC. PUA override map layout. EBCDIC to Unicode
C241 E000
C242 E040 ! Comment that this Line changed
C243 E002
```

### Example Unicode to EBCDIC source

If you wanted to have the reverse mapping also, you can use this source.

```
Q00006  ! Created by Pat for Dept ABC PUA override map layout. Unicode to EBCDIC
E000 C241
E002 C243
E040 C242 ! Line changed
```

### Create \*TBL objects to modify PUA mappings

After you have your source edited the way that you want, you use the CRTTBL command to generate a \*TBL object that the system uses later to modify the PUA mappings.

You need to run the CRTTBL command twice, one time for the EBCDIC to Unicode mapping and another for the Unicode to EBCDIC mapping.

#### Note:

- You must specify a TBLTYPE of \*SRTSEQ. The table type is needed to allow the CCSID keyword; however, the PUA function has nothing to do with sort sequence support.
- The provided table name is not used. When used for the PUA function, the CRTTBL command creates a table object that is named Q1371E2U (for source type 5) or Q1371U2E (for source type 6) in the library you specify.
- You can specify any library; however, you can activate the support that uses a table in only the QSYS library.
- Only one copy of the table can exist in QSYS at one time. If you created a mapping and then decide you do not want to use that version, you must delete the table from QSYS before you can use the CRTTBL command to create a new one.
- Creating the tables does not activate this support. The tables can be created at any time. You can delete and re-create them as many times as you need to. No modifications to the existing conversion support are active until the next step is done.

### Example of CRTTBL command

The following CRTTBL commands are used to create the two table objects:

- **CRTTBL TBL(QSYS/Q1371E2U) SRCFILE(MYLIB/QTBLSRC) SRCMBR(MYE2UMAP) TBLTYPE(\*SRTSEQ) CCSID(1371)**
- **CRTTBL TBL(QSYS/Q1371U2E) SRCFILE(MYLIB/QTBLSRC) SRCMBR(MYU2EMAP) TBLTYPE(\*SRTSEQ) CCSID(1371)**

### Activate the system CCSID support changes

After you are satisfied with the two tables you created, you need to update the system's CCSID support.

1. Verify that the updated tables, Q1371E2U and Q1371U2E, exist in QSYS.
2. End all subsystems.
3. Call the program QTQUSRCV in QSYS to rebuild the systems CCSID support to reflect the changes.
  - You can ignore error messages since this service resets many system pointers.

4. Validate that the message CPC2619 was sent twice, one time for each table. These two messages are the indication that the updates were made.
5. IPL the system.

**Reset the system CCSID support to default state**

If you want to reset your IBM i to its default state, you need to follow these steps.

1. Verify that the updated tables, Q1371E2U and Q1371U2E, do not exist in QSYS.
2. End all subsystems.
3. Call QTQUSRCV to rebuild the systems CCSID information.
4. IPL the system.

**Release upgrade considerations for system CCSID support**

When you install the next release of IBM i, you will need to reactivate any system CCSID support changes.

1. Before the upgrade, save the two tables Q1371E2U and Q1371U2E in QSYS to your backup.
2. Upgrade to the new version of IBM i with your normal process.
3. Restore the two tables Q1371E2U and Q1371U2E in QSYS from backup.
4. Activate the changes by following the steps that are outlined in “Activate the system CCSID support changes” on page 405.

**REXX extension characters**

These tables show the REXX extension characters that are supported on the IBM i operating system.

**REXX/400 extension characters: Axxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with A.

GCGID	Description	Token type	Token flag
AA010000	Aleph (A/F/U) - isolated	NAME	
AA010002	Aleph (A/F/U) - final	NAME	
AA010006	Aleph (after Lam) (A/F/U) - final	NAME	
AA020000	Aleph Maksura (A) - isolated	NAME	
AA020002	Aleph Maksura (A) - final	NAME	
AA070009	Fathatan (A) - intrinsic	NAME	
AA210000	Aleph Madda (A), Aleph Maddey (F), Aleph Madd (U) - Isolated	NAME	
AA210002	Aleph Madda (A), Aleph Maddey (F) - final	NAME	
AA210006	Aleph Madda (after Lam) (A), Aleph Maddey (after Lam) (F) - final	NAME	
AA310000	Aleph Hamza (A), Aleph Hamzey (F) - isolated	NAME	
AA310002	Aleph Hamza (A), Aleph Hamzey (F) - final	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
AA310006	Aleph Hamza (after Lam) (A), Aleph Hamzey (after Lam) (F) - final	NAME	
AB010000	Beh (A/F/U) - isolated-final	NAME	
AB010003	Beh (A/F/U) - initial-middle	NAME	
AC210000	Tcheh (F/U) - isolated-final	NAME	
AC210003	Tcheh (F/U) - initial-middle	NAME	
AC470000	Ayn (A/F/U) - isolated	NAME	
AC470002	Ayn (A/F/U) - final	NAME	
AC470003	Ayn (A/F/U) - initial	NAME	
AC470004	Ayn (A/F/U) - middle	NAME	
AD010000	Dal (A/F/U) - isolated-final	NAME	
AD450000	Dud (1st part) (A) - isolated-final	NAME	
AD450003	Dud (A), Zad (F), Duad (U) - initial-middle	NAME	
AD450006	Dud (A), Zad (F), Duad (U) - isolated-final	NAME	
AD470000	Thal (A), Zal (F/U) - isolated-final	NAME	
AF010000	Feh (A/F/U) - isolated-final	NAME	
AF010003	Feh (A/F/U) - initial-middle	NAME	
AG010000	Gaf (F/U) - isolated-final	NAME	
AG010003	Gaf (F/U) - initial-middle	NAME	
AG230000	Jeem (A/F/U) - isolated-final	NAME	
AG230003	Jeem (A/F/U) - initial-middle	NAME	
AG310000	Ghayn (A/F/U) - isolated	NAME	
AG310002	Ghayn (A/F/U) - final	NAME	
AG310003	Ghayn (A/F/U) - initial	NAME	
AG310004	Ghayn (A/F/U) - middle	NAME	
AH010000	Heh (A/F) - isolated-final	NAME	
AH010003	Heh (A/F) - initial	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
AH010004	Heh (A/F) - middle	NAME	
AH210000	Heh Yey (F) - isolated-final	NAME	
AH450000	Hah (A), Hey (F), Heh (U) - Isolated-Final	NAME	
AH450003	Hah (A), Hey (F), Heh (U) - initial-middle	NAME	
AH470000	Khah (A), Khey (F), Kheh (U) - Isolated-Final	NAME	
AH470003	Khah (A), Khey (F), Kheh (U) - initial-middle	NAME	
AK010000	Caf (A) - isolated-final	NAME	
AK010003	Caf (A/F/U) - initial-middle	NAME	
AK010006	Caf (F/U) - isolated-final	NAME	
AL010000	Lam (A/F/U) - isolated-final	NAME	
AL010003	Lam (A/F) - initial-middle	NAME	
AL020000	Lamaleph (A/F) - isolated	NAME	
AL020003	Lamaleph (A/F) - final	NAME	
AL220000	Lamaleph Madda (A), Lamaleph Maddey (F) - Isolated	NAME	
AL220003	Lamaleph Madda (A), Lamaleph Maddey (F) - final	NAME	
AL320000	Lamaleph Hamza (A), Lamaleph Hamzey (F) - isolated	NAME	
AL320003	Lamaleph Hamza (A), Lamaleph Hamzey (F) - final	NAME	
AM010000	Meem (A/F/U) - isolated-final	NAME	
AM010003	Meem (A/F/U) - initial-middle	NAME	
AN010000	Noon (A/F/U) - isolated-final	NAME	
AN010003	Noon (A/F/U) - initial-middle	NAME	
AP010000	Peh (F/U) - isolated-final	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
AP010003	Peh (F/U) - initial-middle	NAME	
AQ010000	Qaf (A/F/U) - isolated-final	NAME	
AQ010003	Qaf (A/F/U) - initial-middle	NAME	
AR010000	Reh (A/F/U) - isolated-final	NAME	
AS010000	Seen (1st part) (A) - isolated-final	NAME	
AS010003	Seen (A/F/U) - initial-middle	NAME	
AS010006	Seen (A/F/U) - isolated-final	NAME	
AS230000	Sheen (1st part) (A) - isolated-final	NAME	
AS230003	Sheen (A/F/U) - initial-middle	NAME	
AS230006	Sheen (A/F/U) - isolated-final	NAME	
AS450000	Sad (1st part) (A) - isolated-final	NAME	
AS450003	Sad (A/F), Suad (U) - initial-middle	NAME	
AS450006	Sad (A/F), Suad (U) - isolated-final	NAME	
AT010000	Teh (A/F/U) - isolated-final	NAME	
AT010003	Teh (A/F/U) - initial-middle	NAME	
AT020000	Teh Marbuta (A), Teh Mudawara (U) - isolated-final	NAME	
AT450000	Tah (A/F), Toey (U) - isolated-final-initial-middle	NAME	
AT450001	Tah (A/F), Toey (U) - isolated-final	NAME	
AT450002	Tah (A/F), Toey (U) - initial-middle	NAME	
AT470000	Theh (A/F/U) - isolated-final	NAME	
AT470003	Theh (A/F/U) - initial-middle	NAME	

GCGID	Description	Token type	Token flag
AW010000	Waw (A), Vav (F), Waow (U) - isolated-final	NAME	
AW310000	Waw Hamza (A), Vav Hamzey (F), Waow Hamza (U) - isolated-final	NAME	
AX100000	Shadda (A/F), Shadd (U) - isolated	NAME	
AX100004	Shadda (A/F), Shadd (U) - middle	NAME	
AX300000	Hamza (A), Hamzey (F), Hamza (U) - isolated	NAME	
AY010000	Yeh (A) - isolated	NAME	
AY010002	Yeh (A) - final	NAME	
AY010003	Yeh (A) - initial-middle	NAME	
AY020000	Yey (F), Yeh Chotee (U) - isolated	NAME	
AY020002	Yey (F), Yeh Chotee (U) - final	NAME	
AY020003	Yey (F), Yeh Chotee (U) - initial-middle	NAME	
AY310000	Yeh Hamza (A) - initial-middle	NAME	
AY320003	Yey Hamzey (F), Yeh Chotee Hamza (U) - initial-middle	NAME	
AZ010000	Zayn (A), Zey (F), Zeh (U) - isolated-final	NAME	
AZ210000	Jey (F), Zzeh (U) - isolated-final	NAME	
AZ450000	Zah (F), Zoey (U) - isolated-final-initial-middle	NAME	
AZ450001	Zah (A/F), Zoey (U) - isolated-final	NAME	
AZ450002	Zah (A/F), Zoey (U) - initial-middle	NAME	

**REXX/400 extension characters: Bxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with B.

GCGID	Description	Token type	Token flag
BA100000	a - (upper vowel)	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
BA200000	a - (middle vowel)	NAME	
BA300000	a - (middle vowel)	NAME	
BA400000	am - (middle vowel)	NAME	
BA500000	ai - (middle vowel)	NAME	
BA600000	ai - (middle vowel)	NAME	
BA700000	a - (middle vowel)	NAME	
BB100000	Bo	NAME	
BC100000	Cho	NAME	
BD100000	Do	NAME	
BD200000	Do	NAME	
BE100000	e/a - (upper vowel)	NAME	
BE200000	e - (middle vowel)	NAME	
BE300000	e - (middle vowel)	NAME	
BE400000	Yamakkan	NAME	
BF100000	Fo	NAME	
BF200000	Fo	NAME	
BH100000	Ho	NAME	
BH200000	Ho	NAME	
BI100000	i - (upper vowel)	NAME	
BI200000	i - (upper vowel)	NAME	
BK100000	Ko	NAME	
BK200000	Kho	NAME	
BK300000	Kho	NAME	
BK400000	Kho	NAME	
BK500000	Kho	NAME	
BK600000	Kho	NAME	
BL100000	Lo	NAME	
BL200000	Lu	NAME	
BL300000	Lo	NAME	
BM100000	Mo	NAME	
BN100000	Ngo	NAME	
BN200000	No	NAME	
BN300000	No	NAME	
BN400000	a - (upper vowel)	NAME	
BO100000	o	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
BO200000	o - (middle vowel)	NAME	
BP100000	Po	NAME	
BP200000	Pho	NAME	
BP300000	Pho	NAME	
BP400000	Pho	NAME	
BQ100000	Thai repeat sign	NAME	
BQ200000	Thai Ellipsis	NAME	
BQ300000	a - (lower vowel)	NAME	
BQ400000	Fongmann	NAME	
BQ500000	Angkhankhu	NAME	
BQ600000	Komut	NAME	
BR100000	Ro	NAME	
BR200000	Ro	NAME	
BS100000	So	NAME	
BS200000	So	NAME	
BS300000	So	NAME	
BS400000	So	NAME	
BT100000	To	NAME	
BT200000	Tho	NAME	
BT300000	Tho	NAME	
BT400000	Tho	NAME	
BT500000	To	NAME	
BT600000	Tho	NAME	
BT700000	Tho	NAME	
BT800000	Tho	NAME	
BU100000	u - (upper vowel)	NAME	
BU200000	u - (upper vowel)	NAME	
BU300000	u - (lower vowel)	NAME	
BU400000	u - (lower vowel)	NAME	
BW100000	Wo	NAME	
BX100000	Xo	NAME	
BX200000	Xo	NAME	
BX300000	Xo	NAME	
BY100000	Jo	NAME	
BY200000	Yo	NAME	



<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
BZ100000	1st tone mark	NAME	
BZ100300	1st tone mark, low position	NAME	
BZ200000	2nd tone mark	NAME	
BZ200300	2nd tone mark, low position	NAME	
BZ300000	3rd tone mark	NAME	
BZ300300	3rd tone mark, low position	NAME	
BZ400000	4th tone mark	NAME	
BZ400300	4th tone mark, low position	NAME	
BZ500000	5th tone mark	NAME	
BZ500300	5th tone mark, low position	NAME	

**REXX/400 extension characters: Gxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with G.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
GA010000	Alpha small	NAME	
GA020000	Alpha capital	NAME	
GA110000	Alpha acute small	NAME	
GA120000	Alpha acute capital	NAME	
GB010000	Beta small	NAME	
GB020000	Beta capital	NAME	
GD010000	Delta small	NAME	
GD020000	Delta capital	NAME	
GE010000	Epsilon small	NAME	
GE020000	Epsilon capital	NAME	
GE110000	Epsilon acute small	NAME	
GE120000	Epsilon acute capital	NAME	
GE310000	Eta small	NAME	
GE320000	Eta capital	NAME	
GE710000	Eta acute small	NAME	
GE720000	Eta acute capital	NAME	
GF010000	Phi small	NAME	
GF020000	Phi capital	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
GG010000	Gamma small	NAME	
GG020000	Gamma capital	NAME	
GH010000	Chi small	NAME	
GH020000	Chi capital	NAME	
GI010000	Iota small	NAME	
GI020000	Iota capital	NAME	
GI110000	Iota acute small	NAME	
GI120000	Iota acute capital	NAME	
GI170000	Iota diaeresis small	NAME	
GI180000	Iota diaeresis capital	NAME	
GI730000	Iota acute and diaeresis small	NAME	
GK010000	Kappa small	NAME	
GK020000	Kappa capital	NAME	
GL010000	Lambda small	NAME	
GL020000	Lambda capital	NAME	
GM010000	Mu small	NAME	
GM020000	Mu capital	NAME	
GN010000	Nu small	NAME	
GN020000	Nu capital	NAME	
GO010000	Omicron small	NAME	
GO020000	Omicron capital	NAME	
GO110000	Omicron acute small	NAME	
GO120000	Omicron acute capital	NAME	
GO310000	Omega small	NAME	
GO320000	Omega capital	NAME	
GO710000	Omega acute small	NAME	
GO720000	Omega acute capital	NAME	
GP010000	Pi small	NAME	
GP020000	Pi capital	NAME	
GP610000	Psi small	NAME	
GP620000	Psi capital	NAME	
GR010000	Rho small	NAME	
GR020000	Rho capital	NAME	
GS010000	Sigma small	NAME	
GS020000	Sigma capital	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
GS610000	Sigma small (final form)	NAME	
GT010000	Tau small	NAME	
GT020000	Tau capital	NAME	
GT610000	Theta small	NAME	
GT620000	Theta capital	NAME	
GU010000	Upsilon small	NAME	
GU020000	Upsilon capital	NAME	
GU110000	Upsilon acute small	NAME	
GU120000	Upsilon acute capital	NAME	
GU170000	Upsilon diaeresis small	NAME	
GU180000	Upsilon diaeresis capital	NAME	
GU730000	Upsilon acute and diaeresis small	NAME	
GX010000	Xi small	NAME	
GX020000	Xi capital	NAME	
GZ010000	Zeta small	NAME	
GZ020000	Zeta capital	NAME	

**REXX/400 extension characters: Hxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with H.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
HB010000	Bet	NAME	
HD010000	Dalet	NAME	
HG010000	Gimel	NAME	
HH010000	He	NAME	
HH450000	Het	NAME	
HK010000	Kaf	NAME	
HK610000	Kaf (Final Form)	NAME	
HL010000	Lamed	NAME	
HM010000	Mem	NAME	
HM610000	Mem (final form)	NAME	
HN010000	Nun	NAME	
HN610000	Nun (final form)	NAME	
HP010000	Pe	NAME	
HP610000	Pe (final form)	NAME	
HQ010000	Qof	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
HR010000	Resh	NAME	
HS010000	Samech	NAME	
HS210000	Shin	NAME	
HS450000	Zadi	NAME	
HS610000	Zadi (Final Form)	NAME	
HT010000	Tav	NAME	
HT450000	Tet	NAME	
HW010000	Waw	NAME	
HX330000	Alef	NAME	
HX350000	Ayin	NAME	
HY010000	Yod	NAME	
HZ010000	Zayin	NAME	

**REXX/400 extension characters: Jxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with J.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
JA000000	A	NAME	
JA010000	a	NAME	
JE000000	E	NAME	
JE010000	e	NAME	
JH100000	HA	NAME	
JH200000	HI	NAME	
JH300000	HU or FU	NAME	
JH400000	HE	NAME	
JH500000	HO	NAME	
JI000000	I	NAME	
JI010000	i	NAME	
JK100000	KA	NAME	
JK200000	KI	NAME	
JK300000	KU	NAME	
JK400000	KE	NAME	
JK500000	KO	NAME	
JM100000	MA	NAME	
JM200000	MI	NAME	
JM300000	MU	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
JM400000	ME	NAME	
JM500000	MO	NAME	
JN000000	N	NAME	
JN100000	NA	NAME	
JN200000	NI	NAME	
JN300000	NU	NAME	
JN400000	NE	NAME	
JN500000	NO	NAME	
JO000000	O	NAME	
JO010000	o	NAME	
JQ700000	Katakana full stop	NAME	
JQ710000	Katakana left bracket	NAME	
JQ720000	Katakana right bracket	NAME	
JQ730000	Katakana comma	NAME	
JQ740000	Katakana conjunctive symbol	NAME	
JR100000	RA	NAME	
JR200000	RI	NAME	
JR300000	RU	NAME	
JR400000	RE	NAME	
JR500000	RO	NAME	
JS100000	SA	NAME	
JS200000	SI or SHI	NAME	
JS300000	SU	NAME	
JS400000	SE	NAME	
JS500000	SO	NAME	
JT100000	TA	NAME	
JT200000	TI or CHI	NAME	
JT300000	TU or TSU	NAME	
JT310000	tu or tsu	NAME	
JT400000	TE	NAME	
JT500000	TO	NAME	
JU000000	U	NAME	
JU010000	u	NAME	
JW100000	WA	NAME	
JW500000	WO, Katakana participle	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
JX700000	Prolonged sound symbol	NAME	
JX710000	Voiced sound symbol	NAME	
JX720000	Semi-voiced sound symbol	NAME	
JY100000	YA	NAME	
JY110000	ya	NAME	
JY300000	YU	NAME	
JY310000	yu	NAME	
JY500000	YO	NAME	
JY510000	yo	NAME	

**REXX/400 extension characters: Kxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with K.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
KA010000	a small	NAME	
KA020000	A capital	NAME	
KA150000	ya small	NAME	
KA160000	YA capital	NAME	
KB010000	b small	NAME	
KB020000	B capital	NAME	
KC010000	ts small	NAME	
KC020000	TS capital	NAME	
KC110000	c special small	NAME	
KC120000	C special capital	NAME	
KC210000	ch small	NAME	
KC220000	CH capital	NAME	
KD010000	d small	NAME	
KD020000	D capital	NAME	
KD610000	d special small	NAME	
KD620000	D special capital	NAME	
KE010000	e small	NAME	
KE020000	E capital	NAME	
KE130000	e special small	NAME	
KE140000	E special capital	NAME	
KE150000	ye small	NAME	
KE160000	YE capital	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
KE170000	e diaeresis small	NAME	
KE180000	E diaeresis capital	NAME	
KF010000	f small	NAME	
KF020000	F capital	NAME	
KG010000	g small	NAME	
KG020000	G capital	NAME	
KG110000	g special small	NAME	
KG120000	G special capital	NAME	
KG210000	dz special small	NAME	
KG220000	DZ special capital	NAME	
KH010000	kh small	NAME	
KH020000	KH capital	NAME	
KI010000	i small	NAME	
KI020000	I capital	NAME	
KI110000	i special small	NAME	
KI120000	I special capital	NAME	
KI170000	i diaeresis small	NAME	
KI180000	I diaeresis capital	NAME	
KJ010000	j small	NAME	
KJ020000	J capital	NAME	
KJ110000	j special small	NAME	
KJ120000	J special capital	NAME	
KK010000	k small	NAME	
KK020000	K capital	NAME	
KK110000	k special small	NAME	
KK120000	K special capital	NAME	
KL010000	l small	NAME	
KL020000	L capital	NAME	
KL410000	lj small	NAME	
KL420000	LJ capital	NAME	
KM010000	m small	NAME	
KM020000	M capital	NAME	
KN010000	n small	NAME	
KN020000	N capital	NAME	
KN110000	nj small	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
KN120000	NJ capital	NAME	
KO010000	o small	NAME	
KO020000	O capital	NAME	
KP010000	p small	NAME	
KP020000	P capital	NAME	
KR010000	r small	NAME	
KR020000	R capital	NAME	
KS010000	s small	NAME	
KS020000	S capital	NAME	
KS150000	shch small	NAME	
KS160000	SHCH capital	NAME	
KS210000	sh small	NAME	
KS220000	SH capital	NAME	
KT010000	t small	NAME	
KT020000	T capital	NAME	
KU010000	u small	NAME	
KU020000	U capital	NAME	
KU150000	yu small	NAME	
KU160000	YU capital	NAME	
KU210000	Hard sign small	NAME	
KU220000	Hard sign capital	NAME	
KU230000	u Breve small	NAME	
KU240000	U Breve capital	NAME	
KV010000	v small	NAME	
KV020000	V capital	NAME	
KX110000	Soft sign small	NAME	
KX120000	Soft sign capital	NAME	
KY010000	y small	NAME	
KY020000	Y capital	NAME	
KZ010000	z small	NAME	
KZ020000	Z capital	NAME	
KZ150000	s special small	NAME	
KZ160000	S special capital	NAME	
KZ210000	zh small	NAME	
KZ220000	zh capital	NAME	



**REXX/400 extension characters: Lxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with L.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LA010000	a small	NAME	
LA020000	A capital	NAME	
LA110000	a acute small	NAME	
LA120000	A acute capital	NAME	
LA130000	a grave small	NAME	
LA140000	A grave capital	NAME	
LA150000	a circumflex small	NAME	
LA160000	A circumflex capital	NAME	
LA170000	a diaeresis small	NAME	
LA180000	A diaeresis capital	NAME	
LA190000	a tilde Small	NAME	
LA200000	A tilde capital	NAME	
LA230000	a breve small	NAME	
LA240000	A breve capital	NAME	
LA270000	a overcircle small	NAME	
LA280000	A overcircle capital	NAME	
LA430000	a ogonek small	NAME	
LA440000	A ogonek capital	NAME	
LA510000	ae diphthong small	NAME	
LA520000	ae diphthong capital	NAME	
LB010000	b small	NAME	
LB020000	B capital	NAME	
LC010000	c small	NAME	
LC020000	C capital	NAME	
LC110000	c acute small	NAME	
LC120000	C acute capital	NAME	
LC150000	c circumflex small	NAME	
LC160000	C circumflex capital	NAME	
LC210000	c caron small	NAME	
LC220000	C caron capital	NAME	
LC290000	c overdot small	NAME	
LC300000	C overdot capital	NAME	
LC410000	c cedilla small	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LC420000	C cedilla capital	NAME	
LD010000	d small	NAME	
LD020000	D capital	NAME	
LD210000	d caron small	NAME	
LD220000	D caron capital	NAME	
LD610000	d stroke small	NAME	
LD620000	D stroke capital/Eth Icelandic capital	NAME	
LD630000	eth Icelandic small	NAME	
LE010000	e small	NAME	
LE020000	E capital	NAME	
LE110000	e acute small	NAME	
LE120000	E acute capital	NAME	
LE130000	e grave small	NAME	
LE140000	E grave capital	NAME	
LE150000	e circumflex small	NAME	
LE160000	E circumflex capital	NAME	
LE170000	e diaeresis small	NAME	
LE180000	E diaeresis capital	NAME	
LE210000	e caron small	NAME	
LE220000	E caron capital	NAME	
LE430000	e ogonek small	NAME	
LE440000	E ogonek capital	NAME	
LF010000	f small	NAME	
LF020000	F capital	NAME	
LG010000	g small	NAME	
LG020000	G capital	NAME	
LG150000	g circumflex small	NAME	
LG160000	G circumflex capital	NAME	
LG230000	g breve small	NAME	
LG240000	G breve capital	NAME	
LG290000	g overdot small	NAME	
LG300000	G overdot capital	NAME	
LH010000	h small	NAME	
LH020000	H capital	NAME	
LH150000	h circumflex small	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LH160000	H circumflex capital	NAME	
LH610000	h stroke small	NAME	
LH620000	H stroke capital	NAME	
LI010000	i small	NAME	
LI020000	I capital	NAME	
LI110000	i acute small	NAME	
LI120000	I acute capital	NAME	
LI130000	i grave small	NAME	
LI140000	I grave capital	NAME	
LI150000	i circumflex small	NAME	
LI160000	I circumflex capital	NAME	
LI170000	i diaeresis small	NAME	
LI180000	I diaeresis capital	NAME	
LI300000	I overdot capital	NAME	
LI610000	i dotless small	NAME	
LJ010000	j small	NAME	
LJ020000	J capital	NAME	
LJ150000	j circumflex small	NAME	
LJ160000	J circumflex capital	NAME	
LK010000	k small	NAME	
LK020000	K capital	NAME	
LL010000	l small	NAME	
LL020000	L capital	NAME	
LL110000	l acute small	NAME	
LL120000	L acute capital	NAME	
LL210000	l caron small	NAME	
LL220000	L caron capital	NAME	
LL610000	l stroke small	NAME	
LL620000	L stroke capital	NAME	
LM010000	m small	NAME	
LM020000	M capital	NAME	
LN010000	n small	NAME	
LN020000	N capital	NAME	
LN110000	n acute small	NAME	
LN120000	N acute capital	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LN190000	n tilde small	NAME	
LN200000	N tilde capital	NAME	
LN210000	n caron small	NAME	
LN220000	N caron capital	NAME	
LO010000	o small	NAME	
LO020000	O capital	NAME	
LO110000	o acute small	NAME	
LO120000	O acute capital	NAME	
LO130000	o grave small	NAME	
LO140000	O grave capital	NAME	
LO150000	o circumflex small	NAME	
LO160000	O circumflex capital	NAME	
LO170000	o diaeresis small	NAME	
LO180000	O diaeresis capital	NAME	
LO190000	o tilde small	NAME	
LO200000	O tilde capital	NAME	
LO250000	o double acute small	NAME	
LO260000	O double acute capital	NAME	
LO610000	o slash small	NAME	
LO620000	O slash capital	NAME	
LP010000	p small	NAME	
LP020000	P capital	NAME	
LQ010000	q small	NAME	
LQ020000	Q capital	NAME	
LR010000	r small	NAME	
LR020000	R capital	NAME	
LR110000	r acute small	NAME	
LR120000	R acute capital	NAME	
LR210000	r caron small	NAME	
LR220000	R caron capital	NAME	
LS010000	s small	NAME	
LS020000	S capital	NAME	
LS110000	s acute small	NAME	
LS120000	S acute capital	NAME	
LS150000	s circumflex small	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LS160000	S circumflex capital	NAME	
LS210000	s caron small	NAME	
LS220000	S caron capital	NAME	
LS410000	s cedilla small	NAME	
LS420000	S cedilla capital	NAME	
LS610000	Sharp s small	NAME	
LT010000	t small	NAME	
LT020000	T capital	NAME	
LT210000	t caron small	NAME	
LT220000	T caron capital	NAME	
LT410000	t cedilla small	NAME	
LT420000	T cedilla capital	NAME	
LT630000	Thorn Icelandic small	NAME	
LT640000	Thorn Icelandic capital	NAME	
LU010000	u small	NAME	
LU020000	U capital	NAME	
LU110000	u acute small	NAME	
LU120000	U acute capital	NAME	
LU130000	u grave small	NAME	
LU140000	U grave capital	NAME	
LU150000	u circumflex small	NAME	
LU160000	U circumflex capital	NAME	
LU170000	u diaeresis small	NAME	
LU180000	U diaeresis capital	NAME	
LU230000	u breve small	NAME	
LU240000	U breve capital	NAME	
LU250000	u double acute small	NAME	
LU260000	U double acute capital	NAME	
LU270000	u overcircle small	NAME	
LU280000	u overcircle capital	NAME	
LV010000	v small	NAME	
LV020000	V capital	NAME	
LW010000	w small	NAME	
LW020000	W capital	NAME	
LX010000	x small	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
LX020000	X capital	NAME	
LY010000	y small	NAME	
LY020000	Y capital	NAME	
LY110000	y acute small	NAME	
LY120000	Y acute capital	NAME	
LY170000	y diaeresis small	NAME	
LZ010000	z small	NAME	
LZ020000	Z capital	NAME	
LZ110000	z acute small	NAME	
LZ120000	Z acute capital	NAME	
LZ210000	z caron small	NAME	
LZ220000	Z caron capital	NAME	
LZ290000	z overdot small	NAME	
LZ300000	Z overdot capital	NAME	

**REXX/400 extension characters: Nxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with N.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
ND010000	One	NUMBER	
ND010001	One (Arabic, Farsi, Urdu)	INVALID	
ND010002	One, Thai	INVALID	
ND011000	One superscript	INVALID	
ND020000	Two	NUMBER	
ND020001	Two (Arabic, Farsi, Urdu)	INVALID	
ND020002	Two, Thai	INVALID	
ND021000	Two superscript	INVALID	
ND030000	Three	NUMBER	
ND030001	Three (Arabic, Farsi, Urdu)	INVALID	
ND030002	Three, Thai	INVALID	
ND031000	Three superscript	INVALID	
ND040000	Four	NUMBER	
ND040001	Four (Arabic)	INVALID	
ND040002	Four, Thai	INVALID	
ND040003	Four (Farsi)	INVALID	
ND050000	Five	NUMBER	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
ND050001	Five (Arabic)	INVALID	
ND050002	Five, Thai	INVALID	
ND050004	Five (Farsi, Urdu)	INVALID	
ND060000	Six	NUMBER	
ND060001	Six (Arabic, Urdu)	INVALID	
ND060002	Six, Thai	INVALID	
ND060003	Six (Farsi)	INVALID	
ND070000	Seven	NUMBER	
ND070001	Seven (Arabic, Farsi)	INVALID	
ND070002	Seven, Thai	INVALID	
ND080000	Eight	NUMBER	
ND080001	Eight (Arabic, Farsi, Urdu)	INVALID	
ND080002	Eight, Thai	INVALID	
ND090000	Nine	NUMBER	
ND090001	Nine (Arabic, Farsi, Urdu)	INVALID	
ND090002	Nine, Thai	INVALID	
ND100000	Zero	NUMBER	
ND100001	Zero (Arabic, Urdu)	INVALID	
ND100002	Zero, Thai	INVALID	
ND100003	Zero (Farsi)	INVALID	
NF010000	One half	INVALID	
NF040000	One quarter	INVALID	
NF050000	Three quarters	INVALID	

**REXX/400 extension characters: Oxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with O.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
OA000000	A (basic vowel)	NAME	
OA200000	AE (compound vowel)	NAME	
OB000000	B (basic consonant)	NAME	
OB100000	BB (compound consonant)	NAME	
OB200000	BS (compound consonant)	NAME	
OC200000	CH (basic consonant)	NAME	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
OD000000	D (basic consonant)	NAME	
OD100000	DD (compound consonant)	NAME	
OE000000	E (compound vowel)	NAME	
OE200000	EO (basic vowel)	NAME	
OE300000	EU (basic vowel)	NAME	
OE400000	EUI (compound vowel)	NAME	
OG000000	G (basic consonant)	NAME	
OG100000	GG (compound consonant)	NAME	
OG200000	GS (compound consonant)	NAME	
OH000000	H (basic consonant)	NAME	
OI000000	I (basic vowel)	NAME	
OJ000000	J (basic consonant)	NAME	
OJ100000	JJ (compound consonant)	NAME	
OK000000	K (basic consonant)	NAME	
OL000000	L (basic consonant)	NAME	
OL100000	LB (compound consonant)	NAME	
OL200000	LG (compound consonant)	NAME	
OL300000	LH (compound consonant)	NAME	
OL400000	:c 2 .LM (compound consonant)	NAME	
OL500000	LP (compound consonant)	NAME	
OL600000	LS (compound consonant)	NAME	
OL700000	LT (compound consonant)	NAME	
OM000000	M (basic consonant)	NAME	
ON000000	N (basic consonant)	NAME	
ON100000	NH (compound consonant)	NAME	
ON150000	NJ (compound consonant)	NAME	



<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
ON200000	NG or W (basic consonant)	NAME	
OO000000	O (basic vowel)	NAME	
OO100000	OA (compound vowel)	NAME	
OO200000	OAE (compound vowel)	NAME	
OO300000	OI (compound vowel)	NAME	
OP000000	P (basic consonant)	NAME	
OS000000	S (basic consonant)	NAME	
OS100000	SS (compound consonant)	NAME	
OT000000	T (basic consonant)	NAME	
OU000000	U (basic vowel)	NAME	
OU200000	UE (compound vowel)	NAME	
OU300000	UEO (compound vowel)	NAME	
OU400000	UI (compound vowel)	NAME	
OY200000	YA (basic vowel)	NAME	
OY250000	YAE (compound vowel)	NAME	
OY300000	YE (compound vowel)	NAME	
OY400000	YEO (basic vowel)	NAME	
OY500000	YO (basic vowel)	NAME	
OY600000	YU (basic vowel)	NAME	

#### **REXX/400 extension characters: Sxxxxxxx GCGIDs**

The IBM i operating system supports REXX extension characters. This table lists system-supported REXX extension characters whose GCGID begins with S.

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
SA010000	Plus sign	OPER	PLUS
SA020000	Plus or minus sign	INVALID	
SA030000	Less than sign/greater than sign (Arabic)	OPER	LESS_THAN
SA040000	Equal sign	OPER	EQUAL
SA050000	Greater than sign/less than sign (Arabic)	OPER	GREAT_THAN
SA060000	Divide sign	INVALID	
SA070000	Multiply sign	INVALID	
SC010000	International currency symbol	INVALID	
SC020000	Pound sterling sign	INVALID	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
SC030000	Dollar sign	INVALID	
SC040000	Cent sign	INVALID	
SC050000	Yen sign	INVALID	
SC060000	Peseta sign	INVALID	
SC070000	Florin sign	INVALID	
SC120000	Yuan sign	INVALID	
SC130000	Currency symbol, Thailand	INVALID	
SC140000	Won sign	INVALID	
SC160000	Rial sign, Iran	INVALID	
SD110000	Acute accent	INVALID	
SD130000	Grave accent	INVALID	
SD150000	Circumflex accent	INVALID	
SD170000	Diaeresis/Umlaut accent	INVALID	
SD190000	Tilde accent	INVALID	
SD210000	Caron accent	INVALID	
SD230000	Breve accent	INVALID	
SD250000	Double acute accent	INVALID	
SD290000	Overdot accent	INVALID	
SD410000	Cedilla or sedila accent	INVALID	
SD430000	Ogonek accent	INVALID	
SD630000	Middle dot	INVALID	
SD730000	Acute and diaeresis accent	INVALID	
SM000000	Numero sign	INVALID	
SM010000	Number sign	INVALID	
SM020000	Percent sign	OPER	PERCENT
SM020007	Percent sign (Arabic)	OPER	PERCENT
SM030000	Ampersand	OPER	AMPERSAND
SM040000	Asterisk	OPER	ASTERISK
SM040007	Asterisk (Arabic preference - 5 points)	OPER	ASTERISK
SM050000	At sign	INVALID	
SM060000	Left bracket	INVALID	
SM070000	Backslash	OPER	BACKSLASH
SM080000	Right bracket	INVALID	

<b>GCGID</b>	<b>Description</b>	<b>Token type</b>	<b>Token flag</b>
SM100000	Double underscore	INVALID	
SM110000	Left brace	INVALID	
SM120000	Long dash/throughscore	INVALID	
SM130000	Vertical line/logical OR	OPER	VERTI_BAR
SM140000	Right brace	INVALID	
SM150000	Overline	INVALID	
SM170000	Micro symbol	INVALID	
SM190000	Degree symbol	INVALID	
SM200000	Ordinal indicator, Masculine	INVALID	
SM210000	Ordinal indicator, Feminine	INVALID	
SM240000	Section symbol (USA)/ paragraph symbol (Europe)	INVALID	
SM250000	Paragraph symbol (USA)	INVALID	
SM520000	Copyright symbol	INVALID	
SM530000	Registered trademark symbol	INVALID	
SM570000	Bullet	INVALID	
SM650000	Vertical line, broken	INVALID	
SM660000	Logical NOT/End Of Line symbol	OPER	NOT_SIGN
SM860000	Tatweel (Connector)	INVALID	
SM870000	Kasseh (Tail)	INVALID	
SP010000	Space	WHITE	BLANK
SP020000	Exclamation point	NAME	
SP030000	Exclamation point, inverted	INVALID	
SP040000	Quotation marks	STRING	QUOTE
SP050000	Apostrophe	STRING	APOSTROPH
SP060000	Left parenthesis	PUNCT	L_PAREN
SP070000	Right parenthesis	PUNCT	R_PAREN
SP080000	Comma	PUNCT	COMMA
SP080007	Comma rotated (Arabic)	INVALID	
SP090000	Underline/continuous underscore	NAME	
SP100000	Hyphen/minus sign	OPER	MINUS

GCGID	Description	Token type	Token flag
SP110000	Period/full stop	NAME	PERIOD
SP120000	Slash	OPER	SLASH
SP130000	Colon	PUNCT	COLON
SP140000	Semicolon	PUNCT	SEMI_COLON
SP140007	Semicolon, rotated (Arabic)	INVALID	
SP150000	Question mark	NAME	QUESTION
SP150007	Question mark, reversed (Arabic)	INVALID	
SP160000	Question mark, inverted	INVALID	
SP170000	Left angle quotation marks	INVALID	
SP180000	Right angle quotation marks	INVALID	
SP190000	Left single quotation mark	INVALID	
SP200000	Right single quotation mark	INVALID	
SP300000	Required space	INVALID	
SP310000	Numeric space	INVALID	
SP320000	Syllable hyphen	INVALID	
SP490000	Korean fill (NULL) character	INVALID	

### Default character data conversion that can use substitution

The default CCSID conversions use substitution because the character sets within the CCSIDs are different. The table shows which CCSIDs (From CCSID column) can be substituted by other CCSIDs (To CCSID column).

<i>Table 2. Default conversion that can use substitution</i>	
From CCSID	To CCSID
00037	00290, 00833, 00836, 00838, 00930, 00933, 00835, 00939, 00948, 01027, 01043, 05026, 05035
00256	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00273	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00277	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 05026, 05035
00278	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035

Table 2. Default conversion that can use substitution (continued)

From CCSID	To CCSID
00280	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00284	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00285	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00290	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00897, 00933, 00935, 00937, 00942, 01041
00297	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00300	00301
00301	00300, 04396
00420	04960
00437	00870, 00871, 00880, 01025
00500	00290, 00367, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01010, 01011, 01012, 01013, 01014, 01015, 01016, 01017, 01018, 01019, 01027, 05026, 05035
00833	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00836, 00850, 00871, 00930, 00935, 00937, 00939, 00944, 01027, 01040, 05026, 05035
00836	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00833, 00871, 00903, 00930, 00933, 00937, 00939, 00946, 01027, 01042, 01115, 05026, 05035
00837	00928
00838	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00871, 00937
00850	00290, 00833, 00870, 00880, 00930, 00933, 00939, 01025, 01027, 01041, 05026, 05035
00870	00437, 00880, 00915, 01025, 04951
00871	00290, 00437, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 01027, 05026, 05035
00880	00437, 00850, 00870, 00912, 00915, 04948, 04951
00897	01027
00903	00836
00912	00880, 01025
00915	00870, 00880

Table 2. Default conversion that can use substitution (continued)

From CCSID	To CCSID
00930	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00933, 00935, 00937, 00942, 01041
00933	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00836, 00850, 00871, 00930, 00934, 00935, 00937, 00939, 00944, 01027, 01040, 05026, 05035
00934	00933
00935	00037, 00256, 00277, 00278, 00280, 00284, 00285, 00290, 00297, 00500, 00833, 00871, 00930, 00933, 00936, 00937, 00939, 00946, 01027, 01042, 05026, 05035
00936	00935
00937	00290, 00833, 00836, 00930, 00933, 00935, 00938, 00939, 00948, 01027, 01043, 05026, 05035
00938	00937
00939	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00933, 00935, 00937, 00942, 01041
00942	00290, 00930, 00939, 01027, 05026, 05035
00944	00833, 00933
00946	00836, 00935
00948	00037, 00937
01010	00500
01011	00500
01012	00500
01013	00500
01014	00500
01015	00500
01016	00500
01017	00500
01018	00500
01019	00500
01025	00437, 00850, 00870, 00912, 04948

Table 2. Default conversion that can use substitution (continued)

From CCSID	To CCSID
01027	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00933, 00935, 00937, 00942, 01041
01040	00833, 00933
01041	00290, 00850, 00930, 00939, 01027, 05026, 05035
01042	00836, 00935
01043	00037, 00937
01114	28709
01115	00836
04396	00301
04948	00880, 01025
04951	00870, 00880
04960	00420
05026	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00933, 00935, 00937, 00942, 01041
05035	00037, 00256, 00273, 00277, 00278, 00280, 00284, 00285, 00297, 00500, 00833, 00836, 00850, 00871, 00933, 00935, 00937, 00942, 01041
28709	00290, 00833, 00836, 00838, 00930, 00933, 00935, 00939, 00948, 01027, 01043, 05026, 05035

### Related concepts

#### Conversion of character data

The Character Data Representation Architecture (CDRA) system of tags ensures that you can convert character data in a predictable, repeatable way.

## Globalization checklists

The numerous checklists, presented within the IBM i globalization topic collection, are useful tools for ensuring that you have considered key items that are important in each particular aspect of your globalization planning and implementation.

The following table provides the link to each checklist and shows where you can go to read related topics.


Checklist	Where you can go to read related topics
<a href="#">“Checklist: Globalization planning” on page 29</a>	This comprehensive two-part checklist, which you can find in <a href="#">“Setting up IBM i with a national language version” on page 27</a> , provides a good summary of items that relate to hardware installation, software installation, and system configuration.

Checklist	Where you can go to read related topics
<a href="#">“Checklist: Application design” on page 52</a>	Use this checklist as you begin planning the development of a globalized application. You can find this checklist in <a href="#">“Designing globalized applications” on page 52.</a>
<a href="#">“Checklist: User interface design” on page 60</a>	Use this checklist to ensure that the user interfaces you create for your applications anticipate unique requirements from multiple linguistic and cultural environments. You can find this checklist in <a href="#">“User interfaces” on page 60.</a>
<a href="#">“Checklist: Bidirectional support guidelines” on page 167</a>	Languages, such as Arabic and Hebrew, are displayed in a right-to-left direction. Because of this, you must take care that your applications handle bidirectional data properly, and that your interfaces can accommodate this presentation of text and data. You can find this checklist in <a href="#">“Working with bidirectional data” on page 166.</a>
<a href="#">“Checklist: DBCS application design” on page 169</a>	You need to take numerous items into consideration when you develop applications that use double-byte character set (DBCS) support. You can find this checklist in <a href="#">“Working with DBCS data” on page 168.</a>

## Related information for IBM i globalization

Product manuals, Web sites, and information center topic collections contain information that relates to the IBM i globalization topic collection. You can view or print any of the PDFs.

### Manuals

The following manuals are not included in the V6R1 IBM i Information Center. However, these manuals might be a useful reference to you. Each of the manuals is available from the [IBM Publications Center](#)  as a printed hardcopy that you can order, in an online format that you can download at no charge, or both.

- *Character Data Representation Architecture: Reference & Registry*, SC09-2190

This manual provides a complete discussion of the Character Data Representation Architecture (CDRA) and a complete list of the CDRA CCSIDs.

- *Character Data Representation Architecture: Overview*, GC09-2207

This manual provides an overview of CDRA.

- *Communications Configuration*, SC41-5401

This manual describes the objects, commands, and parameters used to configure operating system communications. It includes a general discussion of the objects and methods used to configure communications and detailed descriptions of all parameters that can be specified for the commands used to create the configuration objects. You can find more information about configuration names in this book.

- *National Language Design Guide Volume 2*, SE09-8002

This manual provides information about linguistic and cultural conventions by country.

- *National Language Design Guide Volume 1*, SE09-8001

This manual provides a complete list of the guidelines of DBCS application design, as well as a full description of each guideline.

### Web sites

- [Globalize your On Demand Business](http://www.ibm.com/software/globalization)  ([www.ibm.com/software/globalization](http://www.ibm.com/software/globalization))
- [IBM i Globalization](http://www.ibm.com/systems/power/software/i/globalization/)  (<http://www.ibm.com/systems/power/software/i/globalization/>)



## Other information

- [Database file management](#)
- [Database programming](#)
- [DDS concepts](#)

## Default mapping for EBCDIC UDC to Unicode PUA on IBM i

---

The default mapping for EBCDIC UDC to Unicode PUA on IBM i.

C241	E000
C242	E001
C243	E002
C244	E003
C245	E004
C246	E005
C247	E006
C248	E007
C249	E008
C24A	E009
C24B	E00A
C24C	E00B
C24D	E00C
C24E	E00D
C24F	E00E
C250	E00F
C251	E010
C252	E011
C253	E012
C254	E013
C255	E014
C256	E015
C257	E016
C258	E017
C259	E018
C25A	E019
C25B	E01A
C25C	E01B
C25D	E01C
C25E	E01D
C25F	E01E
C260	E01F
C261	E020
C262	E021
C263	E022
C264	E023
C265	E024
C266	E025
C267	E026
C268	E027
C269	E028
C26A	E029
C26B	E02A
C26C	E02B
C26D	E02C
C26E	E02D
C26F	E02E
C270	E02F
C271	E030
C272	E031
C273	E032
C274	E033
C275	E034
C276	E035
C277	E036
C278	E037
C279	E038
C27A	E039
C27B	E03A
C27C	E03B
C27D	E03C
C27E	E03D
C27F	E03E
C281	E03F
C282	E040
C283	E041

C284 E042  
C285 E043  
C286 E044  
C287 E045  
C288 E046  
C289 E047  
C28A E048  
C28B E049  
C28C E04A  
C28D E04B  
C28E E04C  
C28F E04D  
C290 E04E  
C291 E04F  
C292 E050  
C293 E051  
C294 E052  
C295 E053  
C296 E054  
C297 E055  
C298 E056  
C299 E057  
C29A E058  
C29B E059  
C29C E05A  
C29D E05B  
C29E E05C  
C29F E05D  
C2A0 E05E  
C2A1 E05F  
C2A2 E060  
C2A3 E061  
C2A4 E062  
C2A5 E063  
C2A6 E064  
C2A7 E065  
C2A8 E066  
C2A9 E067  
C2AA E068  
C2AB E069  
C2AC E06A  
C2AD E06B  
C2AE E06C  
C2AF E06D  
C2B0 E06E  
C2B1 E06F  
C2B2 E070  
C2B3 E071  
C2B4 E072  
C2B5 E073  
C2B6 E074  
C2B7 E075  
C2B8 E076  
C2B9 E077  
C2BA E078  
C2BB E079  
C2BC E07A  
C2BD E07B  
C2BE E07C  
C2BF E07D  
C2C0 E07E  
C2C1 E07F  
C2C2 E080  
C2C3 E081  
C2C4 E082  
C2C5 E083  
C2C6 E084  
C2C7 E085  
C2C8 E086  
C2C9 E087  
C2CA E088  
C2CB E089  
C2CC E08A  
C2CD E08B  
C2CE E08C  
C2CF E08D  
C2D0 E08E  
C2D1 E08F  
C2D2 E090  
C2D3 E091  
C2D4 E092  
C2D5 E093

C2D6 E094  
C2D7 E095  
C2D8 E096  
C2D9 E097  
C2DA E098  
C2DB E099  
C2DC E09A  
C2DD E09B  
C2DE E09C  
C2DF E09D  
C2E0 E09E  
C2E1 E09F  
C2E2 E0A0  
C2E3 E0A1  
C2E4 E0A2  
C2E5 E0A3  
C2E6 E0A4  
C2E7 E0A5  
C2E8 E0A6  
C2E9 E0A7  
C2EA E0A8  
C2EB E0A9  
C2EC E0AA  
C2ED E0AB  
C2EE E0AC  
C2EF E0AD  
C2F0 E0AE  
C2F1 E0AF  
C2F2 E0B0  
C2F3 E0B1  
C2F4 E0B2  
C2F5 E0B3  
C2F6 E0B4  
C2F7 E0B5  
C2F8 E0B6  
C2F9 E0B7  
C2FA E0B8  
C2FB E0B9  
C2FC E0BA  
C2FD E0BB  
C341 E0BC  
C342 E0BD  
C343 E0BE  
C344 E0BF  
C345 E0C0  
C346 E0C1  
C347 E0C2  
C348 E0C3  
C349 E0C4  
C34A E0C5  
C34B E0C6  
C34C E0C7  
C34D E0C8  
C34E E0C9  
C34F E0CA  
C350 E0CB  
C351 E0CC  
C352 E0CD  
C353 E0CE  
C354 E0CF  
C355 E0D0  
C356 E0D1  
C357 E0D2  
C358 E0D3  
C359 E0D4  
C35A E0D5  
C35B E0D6  
C35C E0D7  
C35D E0D8  
C35E E0D9  
C35F E0DA  
C360 E0DB  
C361 E0DC  
C362 E0DD  
C363 E0DE  
C364 E0DF  
C365 E0E0  
C366 E0E1  
C367 E0E2  
C368 E0E3  
C369 E0E4  
C36A E0E5

C36B E0E6  
C36C E0E7  
C36D E0E8  
C36E E0E9  
C36F E0EA  
C370 E0EB  
C371 E0EC  
C372 E0ED  
C373 E0EE  
C374 E0EF  
C375 E0F0  
C376 E0F1  
C377 E0F2  
C378 E0F3  
C379 E0F4  
C37A E0F5  
C37B E0F6  
C37C E0F7  
C37D E0F8  
C37E E0F9  
C37F E0FA  
C381 E0FB  
C382 E0FC  
C383 E0FD  
C384 E0FE  
C385 E0FF  
C386 E100  
C387 E101  
C388 E102  
C389 E103  
C38A E104  
C38B E105  
C38C E106  
C38D E107  
C38E E108  
C38F E109  
C390 E10A  
C391 E10B  
C392 E10C  
C393 E10D  
C394 E10E  
C395 E10F  
C396 E110  
C397 E111  
C398 E112  
C399 E113  
C39A E114  
C39B E115  
C39C E116  
C39D E117  
C39E E118  
C39F E119  
C3A0 E11A  
C3A1 E11B  
C3A2 E11C  
C3A3 E11D  
C3A4 E11E  
C3A5 E11F  
C3A6 E120  
C3A7 E121  
C3A8 E122  
C3A9 E123  
C3AA E124  
C3AB E125  
C3AC E126  
C3AD E127  
C3AE E128  
C3AF E129  
C3B0 E12A  
C3B1 E12B  
C3B2 E12C  
C3B3 E12D  
C3B4 E12E  
C3B5 E12F  
C3B6 E130  
C3B7 E131  
C3B8 E132  
C3B9 E133  
C3BA E134  
C3BB E135  
C3BC E136  
C3BD E137

C3BE E138  
C3BF E139  
C3C0 E13A  
C3C1 E13B  
C3C2 E13C  
C3C3 E13D  
C3C4 E13E  
C3C5 E13F  
C3C6 E140  
C3C7 E141  
C3C8 E142  
C3C9 E143  
C3CA E144  
C3CB E145  
C3CC E146  
C3CD E147  
C3CE E148  
C3CF E149  
C3D0 E14A  
C3D1 E14B  
C3D2 E14C  
C3D3 E14D  
C3D4 E14E  
C3D5 E14F  
C3D6 E150  
C3D7 E151  
C3D8 E152  
C3D9 E153  
C3DA E154  
C3DB E155  
C3DC E156  
C3DD E157  
C3DE E158  
C3DF E159  
C3E0 E15A  
C3E1 E15B  
C3E2 E15C  
C3E3 E15D  
C3E4 E15E  
C3E5 E15F  
C3E6 E160  
C3E7 E161  
C3E8 E162  
C3E9 E163  
C3EA E164  
C3EB E165  
C3EC E166  
C3ED E167  
C3EE E168  
C3EF E169  
C3F0 E16A  
C3F1 E16B  
C3F2 E16C  
C3F3 E16D  
C3F4 E16E  
C3F5 E16F  
C3F6 E170  
C3F7 E171  
C3F8 E172  
C3F9 E173  
C3FA E174  
C3FB E175  
C3FC E176  
C3FD E177  
C441 E178  
C442 E179  
C443 E17A  
C444 E17B  
C445 E17C  
C446 E17D  
C447 E17E  
C448 E17F  
C449 E180  
C44A E181  
C44B E182  
C44C E183  
C44D E184  
C44E E185  
C44F E186  
C450 E187  
C451 E188  
C452 E189

C453 E18A  
C454 E18B  
C455 E18C  
C456 E18D  
C457 E18E  
C458 E18F  
C459 E190  
C45A E191  
C45B E192  
C45C E193  
C45D E194  
C45E E195  
C45F E196  
C460 E197  
C461 E198  
C462 E199  
C463 E19A  
C464 E19B  
C465 E19C  
C466 E19D  
C467 E19E  
C468 E19F  
C469 E1A0  
C46A E1A1  
C46B E1A2  
C46C E1A3  
C46D E1A4  
C46E E1A5  
C46F E1A6  
C470 E1A7  
C471 E1A8  
C472 E1A9  
C473 E1AA  
C474 E1AB  
C475 E1AC  
C476 E1AD  
C477 E1AE  
C478 E1AF  
C479 E1B0  
C47A E1B1  
C47B E1B2  
C47C E1B3  
C47D E1B4  
C47E E1B5  
C47F E1B6  
C481 E1B7  
C482 E1B8  
C483 E1B9  
C484 E1BA  
C485 E1BB  
C486 E1BC  
C487 E1BD  
C488 E1BE  
C489 E1BF  
C48A E1C0  
C48B E1C1  
C48C E1C2  
C48D E1C3  
C48E E1C4  
C48F E1C5  
C490 E1C6  
C491 E1C7  
C492 E1C8  
C493 E1C9  
C494 E1CA  
C495 E1CB  
C496 E1CC  
C497 E1CD  
C498 E1CE  
C499 E1CF  
C49A E1D0  
C49B E1D1  
C49C E1D2  
C49D E1D3  
C49E E1D4  
C49F E1D5  
C4A0 E1D6  
C4A1 E1D7  
C4A2 E1D8  
C4A3 E1D9  
C4A4 E1DA  
C4A5 E1DB

C4A6	E1DC
C4A7	E1DD
C4A8	E1DE
C4A9	E1DF
C4AA	E1E0
C4AB	E1E1
C4AC	E1E2
C4AD	E1E3
C4AE	E1E4
C4AF	E1E5
C4B0	E1E6
C4B1	E1E7
C4B2	E1E8
C4B3	E1E9
C4B4	E1EA
C4B5	E1EB
C4B6	E1EC
C4B7	E1ED
C4B8	E1EE
C4B9	E1EF
C4BA	E1F0
C4BB	E1F1
C4BC	E1F2
C4BD	E1F3
C4BE	E1F4
C4BF	E1F5
C4C0	E1F6
C4C1	E1F7
C4C2	E1F8
C4C3	E1F9
C4C4	E1FA
C4C5	E1FB
C4C6	E1FC
C4C7	E1FD
C4C8	E1FE
C4C9	E1FF
C4CA	E200
C4CB	E201
C4CC	E202
C4CD	E203
C4CE	E204
C4CF	E205
C4D0	E206
C4D1	E207
C4D2	E208
C4D3	E209
C4D4	E20A
C4D5	E20B
C4D6	E20C
C4D7	E20D
C4D8	E20E
C4D9	E20F
C4DA	E210
C4DB	E211
C4DC	E212
C4DD	E213
C4DE	E214
C4DF	E215
C4E0	E216
C4E1	E217
C4E2	E218
C4E3	E219
C4E4	E21A
C4E5	E21B
C4E6	E21C
C4E7	E21D
C4E8	E21E
C4E9	E21F
C4EA	E220
C4EB	E221
C4EC	E222
C4ED	E223
C4EE	E224
C4EF	E225
C4F0	E226
C4F1	E227
C4F2	E228
C4F3	E229
C4F4	E22A
C4F5	E22B
C4F6	E22C
C4F7	E22D

C4F8 E22E  
C4F9 E22F  
C4FA E230  
C4FB E231  
C4FC E232  
C4FD E233  
C541 E234  
C542 E235  
C543 E236  
C544 E237  
C545 E238  
C546 E239  
C547 E23A  
C548 E23B  
C549 E23C  
C54A E23D  
C54B E23E  
C54C E23F  
C54D E240  
C54E E241  
C54F E242  
C550 E243  
C551 E244  
C552 E245  
C553 E246  
C554 E247  
C555 E248  
C556 E249  
C557 E24A  
C558 E24B  
C559 E24C  
C55A E24D  
C55B E24E  
C55C E24F  
C55D E250  
C55E E251  
C55F E252  
C560 E253  
C561 E254  
C562 E255  
C563 E256  
C564 E257  
C565 E258  
C566 E259  
C567 E25A  
C568 E25B  
C569 E25C  
C56A E25D  
C56B E25E  
C56C E25F  
C56D E260  
C56E E261  
C56F E262  
C570 E263  
C571 E264  
C572 E265  
C573 E266  
C574 E267  
C575 E268  
C576 E269  
C577 E26A  
C578 E26B  
C579 E26C  
C57A E26D  
C57B E26E  
C57C E26F  
C57D E270  
C57E E271  
C57F E272  
C581 E273  
C582 E274  
C583 E275  
C584 E276  
C585 E277  
C586 E278  
C587 E279  
C588 E27A  
C589 E27B  
C58A E27C  
C58B E27D  
C58C E27E  
C58D E27F



C58E E280  
C58F E281  
C590 E282  
C591 E283  
C592 E284  
C593 E285  
C594 E286  
C595 E287  
C596 E288  
C597 E289  
C598 E28A  
C599 E28B  
C59A E28C  
C59B E28D  
C59C E28E  
C59D E28F  
C59E E290  
C59F E291  
C5A0 E292  
C5A1 E293  
C5A2 E294  
C5A3 E295  
C5A4 E296  
C5A5 E297  
C5A6 E298  
C5A7 E299  
C5A8 E29A  
C5A9 E29B  
C5AA E29C  
C5AB E29D  
C5AC E29E  
C5AD E29F  
C5AE E2A0  
C5AF E2A1  
C5B0 E2A2  
C5B1 E2A3  
C5B2 E2A4  
C5B3 E2A5  
C5B4 E2A6  
C5B5 E2A7  
C5B6 E2A8  
C5B7 E2A9  
C5B8 E2AA  
C5B9 E2AB  
C5BA E2AC  
C5BB E2AD  
C5BC E2AE  
C5BD E2AF  
C5BE E2B0  
C5BF E2B1  
C5C0 E2B2  
C5C1 E2B3  
C5C2 E2B4  
C5C3 E2B5  
C5C4 E2B6  
C5C5 E2B7  
C5C6 E2B8  
C5C7 E2B9  
C5C8 E2BA  
C5C9 E2BB  
C5CA E2BC  
C5CB E2BD  
C5CC E2BE  
C5CD E2BF  
C5CE E2C0  
C5CF E2C1  
C5D0 E2C2  
C5D1 E2C3  
C5D2 E2C4  
C5D3 E2C5  
C5D4 E2C6  
C5D5 E2C7  
C5D6 E2C8  
C5D7 E2C9  
C5D8 E2CA  
C5D9 E2CB  
C5DA E2CC  
C5DB E2CD  
C5DC E2CE  
C5DD E2CF  
C5DE E2D0  
C5DF E2D1

C5E0	E2D2
C5E1	E2D3
C5E2	E2D4
C5E3	E2D5
C5E4	E2D6
C5E5	E2D7
C5E6	E2D8
C5E7	E2D9
C5E8	E2DA
C5E9	E2DB
C5EA	E2DC
C5EB	E2DD
C5EC	E2DE
C5ED	E2DF
C5EE	E2E0
C5EF	E2E1
C5F0	E2E2
C5F1	E2E3
C5F2	E2E4
C5F3	E2E5
C5F4	E2E6
C5F5	E2E7
C5F6	E2E8
C5F7	E2E9
C5F8	E2EA
C5F9	E2EB
C5FA	E2EC
C5FB	E2ED
C5FC	E2EE
C5FD	E2EF
C641	E2F0
C642	E2F1
C643	E2F2
C644	E2F3
C645	E2F4
C646	E2F5
C647	E2F6
C648	E2F7
C649	E2F8
C64A	E2F9
C64B	E2FA
C64C	E2FB
C64D	E2FC
C64E	E2FD
C64F	E2FE
C650	E2FF
C651	E300
C652	E301
C653	E302
C654	E303
C655	E304
C656	E305
C657	E306
C658	E307
C659	E308
C65A	E309
C65B	E30A
C65C	E30B
C65D	E30C
C65E	E30D
C65F	E30E
C660	E30F
C661	E310
C662	E311
C663	E312
C664	E313
C665	E314
C666	E315
C667	E316
C668	E317
C669	E318
C66A	E319
C66B	E31A
C66C	E31B
C66D	E31C
C66E	E31D
C66F	E31E
C670	E31F
C671	E320
C672	E321
C673	E322
C674	E323

C675 E324  
C676 E325  
C677 E326  
C678 E327  
C679 E328  
C67A E329  
C67B E32A  
C67C E32B  
C67D E32C  
C67E E32D  
C67F E32E  
C681 E32F  
C682 E330  
C683 E331  
C684 E332  
C685 E333  
C686 E334  
C687 E335  
C688 E336  
C689 E337  
C68A E338  
C68B E339  
C68C E33A  
C68D E33B  
C68E E33C  
C68F E33D  
C690 E33E  
C691 E33F  
C692 E340  
C693 E341  
C694 E342  
C695 E343  
C696 E344  
C697 E345  
C698 E346  
C699 E347  
C69A E348  
C69B E349  
C69C E34A  
C69D E34B  
C69E E34C  
C69F E34D  
C6A0 E34E  
C6A1 E34F  
C6A2 E350  
C6A3 E351  
C6A4 E352  
C6A5 E353  
C6A6 E354  
C6A7 E355  
C6A8 E356  
C6A9 E357  
C6AA E358  
C6AB E359  
C6AC E35A  
C6AD E35B  
C6AE E35C  
C6AF E35D  
C6B0 E35E  
C6B1 E35F  
C6B2 E360  
C6B3 E361  
C6B4 E362  
C6B5 E363  
C6B6 E364  
C6B7 E365  
C6B8 E366  
C6B9 E367  
C6BA E368  
C6BB E369  
C6BC E36A  
C6BD E36B  
C6BE E36C  
C6BF E36D  
C6C0 E36E  
C6C1 E36F  
C6C2 E370  
C6C3 E371  
C6C4 E372  
C6C5 E373  
C6C6 E374  
C6C7 E375

C6C8 E376  
C6C9 E377  
C6CA E378  
C6CB E379  
C6CC E37A  
C6CD E37B  
C6CE E37C  
C6CF E37D  
C6D0 E37E  
C6D1 E37F  
C6D2 E380  
C6D3 E381  
C6D4 E382  
C6D5 E383  
C6D6 E384  
C6D7 E385  
C6D8 E386  
C6D9 E387  
C6DA E388  
C6DB E389  
C6DC E38A  
C6DD E38B  
C6DE E38C  
C6DF E38D  
C6E0 E38E  
C6E1 E38F  
C6E2 E390  
C6E3 E391  
C6E4 E392  
C6E5 E393  
C6E6 E394  
C6E7 E395  
C6E8 E396  
C6E9 E397  
C6EA E398  
C6EB E399  
C6EC E39A  
C6ED E39B  
C6EE E39C  
C6EF E39D  
C6F0 E39E  
C6F1 E39F  
C6F2 E3A0  
C6F3 E3A1  
C6F4 E3A2  
C6F5 E3A3  
C6F6 E3A4  
C6F7 E3A5  
C6F8 E3A6  
C6F9 E3A7  
C6FA E3A8  
C6FB E3A9  
C6FC E3AA  
C6FD E3AB  
C741 E3AC  
C742 E3AD  
C743 E3AE  
C744 E3AF  
C745 E3B0  
C746 E3B1  
C747 E3B2  
C748 E3B3  
C749 E3B4  
C74A E3B5  
C74B E3B6  
C74C E3B7  
C74D E3B8  
C74E E3B9  
C74F E3BA  
C750 E3BB  
C751 E3BC  
C752 E3BD  
C753 E3BE  
C754 E3BF  
C755 E3C0  
C756 E3C1  
C757 E3C2  
C758 E3C3  
C759 E3C4  
C75A E3C5  
C75B E3C6  
C75C E3C7

C75D E3C8  
C75E E3C9  
C75F E3CA  
C760 E3CB  
C761 E3CC  
C762 E3CD  
C763 E3CE  
C764 E3CF  
C765 E3D0  
C766 E3D1  
C767 E3D2  
C768 E3D3  
C769 E3D4  
C76A E3D5  
C76B E3D6  
C76C E3D7  
C76D E3D8  
C76E E3D9  
C76F E3DA  
C770 E3DB  
C771 E3DC  
C772 E3DD  
C773 E3DE  
C774 E3DF  
C775 E3E0  
C776 E3E1  
C777 E3E2  
C778 E3E3  
C779 E3E4  
C77A E3E5  
C77B E3E6  
C77C E3E7  
C77D E3E8  
C77E E3E9  
C77F E3EA  
C781 E3EB  
C782 E3EC  
C783 E3ED  
C784 E3EE  
C785 E3EF  
C786 E3F0  
C787 E3F1  
C788 E3F2  
C789 E3F3  
C78A E3F4  
C78B E3F5  
C78C E3F6  
C78D E3F7  
C78E E3F8  
C78F E3F9  
C790 E3FA  
C791 E3FB  
C792 E3FC  
C793 E3FD  
C794 E3FE  
C795 E3FF  
C796 E400  
C797 E401  
C798 E402  
C799 E403  
C79A E404  
C79B E405  
C79C E406  
C79D E407  
C79E E408  
C79F E409  
C7A0 E40A  
C7A1 E40B  
C7A2 E40C  
C7A3 E40D  
C7A4 E40E  
C7A5 E40F  
C7A6 E410  
C7A7 E411  
C7A8 E412  
C7A9 E413  
C7AA E414  
C7AB E415  
C7AC E416  
C7AD E417  
C7AE E418  
C7AF E419

C7B0	E41A
C7B1	E41B
C7B2	E41C
C7B3	E41D
C7B4	E41E
C7B5	E41F
C7B6	E420
C7B7	E421
C7B8	E422
C7B9	E423
C7BA	E424
C7BB	E425
C7BC	E426
C7BD	E427
C7BE	E428
C7BF	E429
C7C0	E42A
C7C1	E42B
C7C2	E42C
C7C3	E42D
C7C4	E42E
C7C5	E42F
C7C6	E430
C7C7	E431
C7C8	E432
C7C9	E433
C7CA	E434
C7CB	E435
C7CC	E436
C7CD	E437
C7CE	E438
C7CF	E439
C7D0	E43A
C7D1	E43B
C7D2	E43C
C7D3	E43D
C7D4	E43E
C7D5	E43F
C7D6	E440
C7D7	E441
C7D8	E442
C7D9	E443
C7DA	E444
C7DB	E445
C7DC	E446
C7DD	E447
C7DE	E448
C7DF	E449
C7E0	E44A
C7E1	E44B
C7E2	E44C
C7E3	E44D
C7E4	E44E
C7E5	E44F
C7E6	E450
C7E7	E451
C7E8	E452
C7E9	E453
C7EA	E454
C7EB	E455
C7EC	E456
C7ED	E457
C7EE	E458
C7EF	E459
C7F0	E45A
C7F1	E45B
C7F2	E45C
C7F3	E45D
C7F4	E45E
C7F5	E45F
C7F6	E460
C7F7	E461
C7F8	E462
C7F9	E463
C7FA	E464
C7FB	E465
C7FC	E466
C7FD	E467
C841	E468
C842	E469
C843	E46A
C844	E46B

C845 E46C  
C846 E46D  
C847 E46E  
C848 E46F  
C849 E470  
C84A E471  
C84B E472  
C84C E473  
C84D E474  
C84E E475  
C84F E476  
C850 E477  
C851 E478  
C852 E479  
C853 E47A  
C854 E47B  
C855 E47C  
C856 E47D  
C857 E47E  
C858 E47F  
C859 E480  
C85A E481  
C85B E482  
C85C E483  
C85D E484  
C85E E485  
C85F E486  
C860 E487  
C861 E488  
C862 E489  
C863 E48A  
C864 E48B  
C865 E48C  
C866 E48D  
C867 E48E  
C868 E48F  
C869 E490  
C86A E491  
C86B E492  
C86C E493  
C86D E494  
C86E E495  
C86F E496  
C870 E497  
C871 E498  
C872 E499  
C873 E49A  
C874 E49B  
C875 E49C  
C876 E49D  
C877 E49E  
C878 E49F  
C879 E4A0  
C87A E4A1  
C87B E4A2  
C87C E4A3  
C87D E4A4  
C87E E4A5  
C87F E4A6  
C881 E4A7  
C882 E4A8  
C883 E4A9  
C884 E4AA  
C885 E4AB  
C886 E4AC  
C887 E4AD  
C888 E4AE  
C889 E4AF  
C88A E4B0  
C88B E4B1  
C88C E4B2  
C88D E4B3  
C88E E4B4  
C88F E4B5  
C890 E4B6  
C891 E4B7  
C892 E4B8  
C893 E4B9  
C894 E4BA  
C895 E4BB  
C896 E4BC  
C897 E4BD

C898	E4BE
C899	E4BF
C89A	E4C0
C89B	E4C1
C89C	E4C2
C89D	E4C3
C89E	E4C4
C89F	E4C5
C8A0	E4C6
C8A1	E4C7
C8A2	E4C8
C8A3	E4C9
C8A4	E4CA
C8A5	E4CB
C8A6	E4CC
C8A7	E4CD
C8A8	E4CE
C8A9	E4CF
C8AA	E4D0
C8AB	E4D1
C8AC	E4D2
C8AD	E4D3
C8AE	E4D4
C8AF	E4D5
C8B0	E4D6
C8B1	E4D7
C8B2	E4D8
C8B3	E4D9
C8B4	E4DA
C8B5	E4DB
C8B6	E4DC
C8B7	E4DD
C8B8	E4DE
C8B9	E4DF
C8BA	E4E0
C8BB	E4E1
C8BC	E4E2
C8BD	E4E3
C8BE	E4E4
C8BF	E4E5
C8C0	E4E6
C8C1	E4E7
C8C2	E4E8
C8C3	E4E9
C8C4	E4EA
C8C5	E4EB
C8C6	E4EC
C8C7	E4ED
C8C8	E4EE
C8C9	E4EF
C8CA	E4F0
C8CB	E4F1
C8CC	E4F2
C8CD	E4F3
C8CE	E4F4
C8CF	E4F5
C8D0	E4F6
C8D1	E4F7
C8D2	E4F8
C8D3	E4F9
C8D4	E4FA
C8D5	E4FB
C8D6	E4FC
C8D7	E4FD
C8D8	E4FE
C8D9	E4FF
C8DA	E500
C8DB	E501
C8DC	E502
C8DD	E503
C8DE	E504
C8DF	E505
C8E0	E506
C8E1	E507
C8E2	E508
C8E3	E509
C8E4	E50A
C8E5	E50B
C8E6	E50C
C8E7	E50D
C8E8	E50E
C8E9	E50F



C8EA E510  
C8EB E511  
C8EC E512  
C8ED E513  
C8EE E514  
C8EF E515  
C8F0 E516  
C8F1 E517  
C8F2 E518  
C8F3 E519  
C8F4 E51A  
C8F5 E51B  
C8F6 E51C  
C8F7 E51D  
C8F8 E51E  
C8F9 E51F  
C8FA E520  
C8FB E521  
C8FC E522  
C8FD E523  
C941 E524  
C942 E525  
C943 E526  
C944 E527  
C945 E528  
C946 E529  
C947 E52A  
C948 E52B  
C949 E52C  
C94A E52D  
C94B E52E  
C94C E52F  
C94D E530  
C94E E531  
C94F E532  
C950 E533  
C951 E534  
C952 E535  
C953 E536  
C954 E537  
C955 E538  
C956 E539  
C957 E53A  
C958 E53B  
C959 E53C  
C95A E53D  
C95B E53E  
C95C E53F  
C95D E540  
C95E E541  
C95F E542  
C960 E543  
C961 E544  
C962 E545  
C963 E546  
C964 E547  
C965 E548  
C966 E549  
C967 E54A  
C968 E54B  
C969 E54C  
C96A E54D  
C96B E54E  
C96C E54F  
C96D E550  
C96E E551  
C96F E552  
C970 E553  
C971 E554  
C972 E555  
C973 E556  
C974 E557  
C975 E558  
C976 E559  
C977 E55A  
C978 E55B  
C979 E55C  
C97A E55D  
C97B E55E  
C97C E55F  
C97D E560  
C97E E561

C97F E562  
C981 E563  
C982 E564  
C983 E565  
C984 E566  
C985 E567  
C986 E568  
C987 E569  
C988 E56A  
C989 E56B  
C98A E56C  
C98B E56D  
C98C E56E  
C98D E56F  
C98E E570  
C98F E571  
C990 E572  
C991 E573  
C992 E574  
C993 E575  
C994 E576  
C995 E577  
C996 E578  
C997 E579  
C998 E57A  
C999 E57B  
C99A E57C  
C99B E57D  
C99C E57E  
C99D E57F  
C99E E580  
C99F E581  
C9A0 E582  
C9A1 E583  
C9A2 E584  
C9A3 E585  
C9A4 E586  
C9A5 E587  
C9A6 E588  
C9A7 E589  
C9A8 E58A  
C9A9 E58B  
C9AA E58C  
C9AB E58D  
C9AC E58E  
C9AD E58F  
C9AE E590  
C9AF E591  
C9B0 E592  
C9B1 E593  
C9B2 E594  
C9B3 E595  
C9B4 E596  
C9B5 E597  
C9B6 E598  
C9B7 E599  
C9B8 E59A  
C9B9 E59B  
C9BA E59C  
C9BB E59D  
C9BC E59E  
C9BD E59F  
C9BE E5A0  
C9BF E5A1  
C9C0 E5A2  
C9C1 E5A3  
C9C2 E5A4  
C9C3 E5A5  
C9C4 E5A6  
C9C5 E5A7  
C9C6 E5A8  
C9C7 E5A9  
C9C8 E5AA  
C9C9 E5AB  
C9CA E5AC  
C9CB E5AD  
C9CC E5AE  
C9CD E5AF  
C9CE E5B0  
C9CF E5B1  
C9D0 E5B2  
C9D1 E5B3

C9D2 E5B4  
C9D3 E5B5  
C9D4 E5B6  
C9D5 E5B7  
C9D6 E5B8  
C9D7 E5B9  
C9D8 E5BA  
C9D9 E5BB  
C9DA E5BC  
C9DB E5BD  
C9DC E5BE  
C9DD E5BF  
C9DE E5C0  
C9DF E5C1  
C9E0 E5C2  
C9E1 E5C3  
C9E2 E5C4  
C9E3 E5C5  
C9E4 E5C6  
C9E5 E5C7  
C9E6 E5C8  
C9E7 E5C9  
C9E8 E5CA  
C9E9 E5CB  
C9EA E5CC  
C9EB E5CD  
C9EC E5CE  
C9ED E5CF  
C9EE E5D0  
C9EF E5D1  
C9F0 E5D2  
C9F1 E5D3  
C9F2 E5D4  
C9F3 E5D5  
C9F4 E5D6  
C9F5 E5D7  
C9F6 E5D8  
C9F7 E5D9  
C9F8 E5DA  
C9F9 E5DB  
C9FA E5DC  
C9FB E5DD  
C9FC E5DE  
C9FD E5DF  
CA41 E5E0  
CA42 E5E1  
CA43 E5E2  
CA44 E5E3  
CA45 E5E4  
CA46 E5E5  
CA47 E5E6  
CA48 E5E7  
CA49 E5E8  
CA4A E5E9  
CA4B E5EA  
CA4C E5EB  
CA4D E5EC  
CA4E E5ED  
CA4F E5EE  
CA50 E5EF  
CA51 E5F0  
CA52 E5F1  
CA53 E5F2  
CA54 E5F3  
CA55 E5F4  
CA56 E5F5  
CA57 E5F6  
CA58 E5F7  
CA59 E5F8  
CA5A E5F9  
CA5B E5FA  
CA5C E5FB  
CA5D E5FC  
CA5E E5FD  
CA5F E5FE  
CA60 E5FF  
CA61 E600  
CA62 E601  
CA63 E602  
CA64 E603  
CA65 E604  
CA66 E605

CA67	E606
CA68	E607
CA69	E608
CA6A	E609
CA6B	E60A
CA6C	E60B
CA6D	E60C
CA6E	E60D
CA6F	E60E
CA70	E60F
CA71	E610
CA72	E611
CA73	E612
CA74	E613
CA75	E614
CA76	E615
CA77	E616
CA78	E617
CA79	E618
CA7A	E619
CA7B	E61A
CA7C	E61B
CA7D	E61C
CA7E	E61D
CA7F	E61E
CA81	E61F
CA82	E620
CA83	E621
CA84	E622
CA85	E623
CA86	E624
CA87	E625
CA88	E626
CA89	E627
CA8A	E628
CA8B	E629
CA8C	E62A
CA8D	E62B
CA8E	E62C
CA8F	E62D
CA90	E62E
CA91	E62F
CA92	E630
CA93	E631
CA94	E632
CA95	E633
CA96	E634
CA97	E635
CA98	E636
CA99	E637
CA9A	E638
CA9B	E639
CA9C	E63A
CA9D	E63B
CA9E	E63C
CA9F	E63D
CAA0	E63E
CAA1	E63F
CAA2	E640
CAA3	E641
CAA4	E642
CAA5	E643
CAA6	E644
CAA7	E645
CAA8	E646
CAA9	E647
CAA	E648
CAAB	E649
CAAC	E64A
CAAD	E64B
CAAE	E64C
CAAF	E64D
CAB0	E64E
CAB1	E64F
CAB2	E650
CAB3	E651
CAB4	E652
CAB5	E653
CAB6	E654
CAB7	E655
CAB8	E656
CAB9	E657

CABA E658  
CABB E659  
CABC E65A  
CABD E65B  
CABE E65C  
CABF E65D  
CAC0 E65E  
CAC1 E65F  
CAC2 E660  
CAC3 E661  
CAC4 E662  
CAC5 E663  
CAC6 E664  
CAC7 E665  
CAC8 E666  
CAC9 E667  
CACA E668  
CACB E669  
CACC E66A  
CACD E66B  
CACE E66C  
CACF E66D  
CAD0 E66E  
CAD1 E66F  
CAD2 E670  
CAD3 E671  
CAD4 E672  
CAD5 E673  
CAD6 E674  
CAD7 E675  
CAD8 E676  
CAD9 E677  
CADA E678  
CADB E679  
CADC E67A  
CADD E67B  
CADE E67C  
CADF E67D  
CAE0 E67E  
CAE1 E67F  
CAE2 E680  
CAE3 E681  
CAE4 E682  
CAE5 E683  
CAE6 E684  
CAE7 E685  
CAE8 E686  
CAE9 E687  
CAEA E688  
CAEB E689  
CAEC E68A  
CAED E68B  
CAEE E68C  
CAEF E68D  
CAF0 E68E  
CAF1 E68F  
CAF2 E690  
CAF3 E691  
CAF4 E692  
CAF5 E693  
CAF6 E694  
CAF7 E695  
CAF8 E696  
CAF9 E697  
CAFA E698  
CAFB E699  
CAFC E69A  
CAFD E69B  
CB41 E69C  
CB42 E69D  
CB43 E69E  
CB44 E69F  
CB45 E6A0  
CB46 E6A1  
CB47 E6A2  
CB48 E6A3  
CB49 E6A4  
CB4A E6A5  
CB4B E6A6  
CB4C E6A7  
CB4D E6A8  
CB4E E6A9

CB4F	E6AA
CB50	E6AB
CB51	E6AC
CB52	E6AD
CB53	E6AE
CB54	E6AF
CB55	E6B0
CB56	E6B1
CB57	E6B2
CB58	E6B3
CB59	E6B4
CB5A	E6B5
CB5B	E6B6
CB5C	E6B7
CB5D	E6B8
CB5E	E6B9
CB5F	E6BA
CB60	E6BB
CB61	E6BC
CB62	E6BD
CB63	E6BE
CB64	E6BF
CB65	E6C0
CB66	E6C1
CB67	E6C2
CB68	E6C3
CB69	E6C4
CB6A	E6C5
CB6B	E6C6
CB6C	E6C7
CB6D	E6C8
CB6E	E6C9
CB6F	E6CA
CB70	E6CB
CB71	E6CC
CB72	E6CD
CB73	E6CE
CB74	E6CF
CB75	E6D0
CB76	E6D1
CB77	E6D2
CB78	E6D3
CB79	E6D4
CB7A	E6D5
CB7B	E6D6
CB7C	E6D7
CB7D	E6D8
CB7E	E6D9
CB7F	E6DA
CB81	E6DB
CB82	E6DC
CB83	E6DD
CB84	E6DE
CB85	E6DF
CB86	E6E0
CB87	E6E1
CB88	E6E2
CB89	E6E3
CB8A	E6E4
CB8B	E6E5
CB8C	E6E6
CB8D	E6E7
CB8E	E6E8
CB8F	E6E9
CB90	E6EA
CB91	E6EB
CB92	E6EC
CB93	E6ED
CB94	E6EE
CB95	E6EF
CB96	E6F0
CB97	E6F1
CB98	E6F2
CB99	E6F3
CB9A	E6F4
CB9B	E6F5
CB9C	E6F6
CB9D	E6F7
CB9E	E6F8
CB9F	E6F9
CBA0	E6FA
CBA1	E6FB

CBA2 E6FC  
CBA3 E6FD  
CBA4 E6FE  
CBA5 E6FF  
CBA6 E700  
CBA7 E701  
CBA8 E702  
CBA9 E703  
CBAA E704  
CBAB E705  
CBAC E706  
CBAD E707  
CBAE E708  
CBAF E709  
CBB0 E70A  
CBB1 E70B  
CBB2 E70C  
CBB3 E70D  
CBB4 E70E  
CBB5 E70F  
CBB6 E710  
CBB7 E711  
CBB8 E712  
CBB9 E713  
CBBA E714  
CBBB E715  
CBBC E716  
CBBD E717  
CBBE E718  
CBBF E719  
CBC0 E71A  
CBC1 E71B  
CBC2 E71C  
CBC3 E71D  
CBC4 E71E  
CBC5 E71F  
CBC6 E720  
CBC7 E721  
CBC8 E722  
CBC9 E723  
CBCA E724  
CBCB E725  
CBCC E726  
CBCD E727  
CBCE E728  
CBCF E729  
CBD0 E72A  
CBD1 E72B  
CBD2 E72C  
CBD3 E72D  
CBD4 E72E  
CBD5 E72F  
CBD6 E730  
CBD7 E731  
CBD8 E732  
CBD9 E733  
CBDA E734  
CBDB E735  
CBDC E736  
CBDD E737  
CBDE E738  
CBDF E739  
CBE0 E73A  
CBE1 E73B  
CBE2 E73C  
CBE3 E73D  
CBE4 E73E  
CBE5 E73F  
CBE6 E740  
CBE7 E741  
CBE8 E742  
CBE9 E743  
CBEA E744  
CBEB E745  
CBEC E746  
CBED E747  
CBEE E748  
CBEF E749  
CBF0 E74A  
CBF1 E74B  
CBF2 E74C  
CBF3 E74D

CBF4 E74E  
CBF5 E74F  
CBF6 E750  
CBF7 E751  
CBF8 E752  
CBF9 E753  
CBFA E754  
CBFB E755  
CBFC E756  
CBFD E757  
CC41 E758  
CC42 E759  
CC43 E75A  
CC44 E75B  
CC45 E75C  
CC46 E75D  
CC47 E75E  
CC48 E75F  
CC49 E760  
CC4A E761  
CC4B E762  
CC4C E763  
CC4D E764  
CC4E E765  
CC4F E766  
CC50 E767  
CC51 E768  
CC52 E769  
CC53 E76A  
CC54 E76B  
CC55 E76C  
CC56 E76D  
CC57 E76E  
CC58 E76F  
CC59 E770  
CC5A E771  
CC5B E772  
CC5C E773  
CC5D E774  
CC5E E775  
CC5F E776  
CC60 E777  
CC61 E778  
CC62 E779  
CC63 E77A  
CC64 E77B  
CC65 E77C  
CC66 E77D  
CC67 E77E  
CC68 E77F  
CC69 E780  
CC6A E781  
CC6B E782  
CC6C E783  
CC6D E784  
CC6E E785  
CC6F E786  
CC70 E787  
CC71 E788  
CC72 E789  
CC73 E78A  
CC74 E78B  
CC75 E78C  
CC76 E78D  
CC77 E78E  
CC78 E78F  
CC79 E790  
CC7A E791  
CC7B E792  
CC7C E793  
CC7D E794  
CC7E E795  
CC7F E796  
CC81 E797  
CC82 E798  
CC83 E799  
CC84 E79A  
CC85 E79B  
CC86 E79C  
CC87 E79D  
CC88 E79E  
CC89 E79F



CC8A E7A0  
CC8B E7A1  
CC8C E7A2  
CC8D E7A3  
CC8E E7A4  
CC8F E7A5  
CC90 E7A6  
CC91 E7A7  
CC92 E7A8  
CC93 E7A9  
CC94 E7AA  
CC95 E7AB  
CC96 E7AC  
CC97 E7AD  
CC98 E7AE  
CC99 E7AF  
CC9A E7B0  
CC9B E7B1  
CC9C E7B2  
CC9D E7B3  
CC9E E7B4  
CC9F E7B5  
CCA0 E7B6  
CCA1 E7B7  
CCA2 E7B8  
CCA3 E7B9  
CCA4 E7BA  
CCA5 E7BB  
CCA6 E7BC  
CCA7 E7BD  
CCA8 E7BE  
CCA9 E7BF  
CCA E7C0  
CCAB E7C1  
CCAC E7C2  
CCAD E7C3  
CCAE E7C4  
CCAF E7C5  
CCB0 E7C6  
CCB1 E7C7  
CCB2 E7C8  
CCB3 E7C9  
CCB4 E7CA  
CCB5 E7CB  
CCB6 E7CC  
CCB7 E7CD  
CCB8 E7CE  
CCB9 E7CF  
CCBA E7D0  
CCBB E7D1  
CCBC E7D2  
CCBD E7D3  
CCBE E7D4  
CCBF E7D5  
CCC0 E7D6  
CCC1 E7D7  
CCC2 E7D8  
CCC3 E7D9  
CCC4 E7DA  
CCC5 E7DB  
CCC6 E7DC  
CCC7 E7DD  
CCC8 E7DE  
CCC9 E7DF  
CCCA E7E0  
CCCB E7E1  
CCCC E7E2  
CCCD E7E3  
CCCE E7E4  
CCCF E7E5  
CCD0 E7E6  
CCD1 E7E7  
CCD2 E7E8  
CCD3 E7E9  
CCD4 E7EA  
CCD5 E7EB  
CCD6 E7EC  
CCD7 E7ED  
CCD8 E7EE  
CCD9 E7EF  
CCDA E7F0  
CCDB E7F1

CCDC	E7F2
CCDD	E7F3
CCDE	E7F4
CCDF	E7F5
CCE0	E7F6
CCE1	E7F7
CCE2	E7F8
CCE3	E7F9
CCE4	E7FA
CCE5	E7FB
CCE6	E7FC
CCE7	E7FD
CCE8	E7FE
CCE9	E7FF
CCEA	E800
CCEB	E801
CCEC	E802
CCED	E803
CCEE	E804
CCEF	E805
CCF0	E806
CCF1	E807
CCF2	E808
CCF3	E809
CCF4	E80A
CCF5	E80B
CCF6	E80C
CCF7	E80D
CCF8	E80E
CCF9	E80F
CCFA	E810
CCFB	E811
CCFC	E812
CCFD	E813
CD41	E814
CD42	E815
CD43	E816
CD44	E817
CD45	E818
CD46	E819
CD47	E81A
CD48	E81B
CD49	E81C
CD4A	E81D
CD4B	E81E
CD4C	E81F
CD4D	E820
CD4E	E821
CD4F	E822
CD50	E823
CD51	E824
CD52	E825
CD53	E826
CD54	E827
CD55	E828
CD56	E829
CD57	E82A
CD58	E82B
CD59	E82C
CD5A	E82D
CD5B	E82E
CD5C	E82F
CD5D	E830
CD5E	E831
CD5F	E832
CD60	E833
CD61	E834
CD62	E835
CD63	E836
CD64	E837
CD65	E838
CD66	E839
CD67	E83A
CD68	E83B
CD69	E83C
CD6A	E83D
CD6B	E83E
CD6C	E83F
CD6D	E840
CD6E	E841
CD6F	E842
CD70	E843

CD71 E844  
CD72 E845  
CD73 E846  
CD74 E847  
CD75 E848  
CD76 E849  
CD77 E84A  
CD78 E84B  
CD79 E84C  
CD7A E84D  
CD7B E84E  
CD7C E84F  
CD7D E850  
CD7E E851  
CD7F E852  
CD81 E853  
CD82 E854  
CD83 E855  
CD84 E856  
CD85 E857  
CD86 E858  
CD87 E859  
CD88 E85A  
CD89 E85B  
CD8A E85C  
CD8B E85D  
CD8C E85E  
CD8D E85F  
CD8E E860  
CD8F E861  
CD90 E862  
CD91 E863  
CD92 E864  
CD93 E865  
CD94 E866  
CD95 E867  
CD96 E868  
CD97 E869  
CD98 E86A  
CD99 E86B  
CD9A E86C  
CD9B E86D  
CD9C E86E  
CD9D E86F  
CD9E E870  
CD9F E871  
CDA0 E872  
CDA1 E873  
CDA2 E874  
CDA3 E875  
CDA4 E876  
CDA5 E877  
CDA6 E878  
CDA7 E879  
CDA8 E87A  
CDA9 E87B  
CDAA E87C  
CDAB E87D  
CDAC E87E  
CDAD E87F  
CDAE E880  
CDAF E881  
CDB0 E882  
CDB1 E883  
CDB2 E884  
CDB3 E885  
CDB4 E886  
CDB5 E887  
CDB6 E888  
CDB7 E889  
CDB8 E88A  
CDB9 E88B  
CDBA E88C  
CDBB E88D  
CDBC E88E  
CDBD E88F  
CDBE E890  
CDBF E891  
CDC0 E892  
CDC1 E893  
CDC2 E894  
CDC3 E895

CDC4	E896
CDC5	E897
CDC6	E898
CDC7	E899
CDC8	E89A
CDC9	E89B
CDCA	E89C
CDCB	E89D
CDCC	E89E
CDCD	E89F
CDCE	E8A0
CDCF	E8A1
CDD0	E8A2
CDD1	E8A3
CDD2	E8A4
CDD3	E8A5
CDD4	E8A6
CDD5	E8A7
CDD6	E8A8
CDD7	E8A9
CDD8	E8AA
CDD9	E8AB
CDDA	E8AC
Cddb	E8AD
CDDC	E8AE
CDDD	E8AF
CDD E	E8B0
CDD F	E8B1
CDE0	E8B2
CDE1	E8B3
CDE2	E8B4
CDE3	E8B5
CDE4	E8B6
CDE5	E8B7
CDE6	E8B8
CDE7	E8B9
CDE8	E8BA
CDE9	E8BB
CDEA	E8BC
CDEB	E8BD
CDEC	E8BE
CDED	E8BF
CDEE	E8C0
CDEF	E8C1
CDF0	E8C2
CDF1	E8C3
CDF2	E8C4
CDF3	E8C5
CDF4	E8C6
CDF5	E8C7
CDF6	E8C8
CDF7	E8C9
CDF8	E8CA
CDF9	E8CB
CDFA	E8CC
CDFB	E8CD
CDFC	E8CE
CDFD	E8CF
CE41	E8D0
CE42	E8D1
CE43	E8D2
CE44	E8D3
CE45	E8D4
CE46	E8D5
CE47	E8D6
CE48	E8D7
CE49	E8D8
CE4A	E8D9
CE4B	E8DA
CE4C	E8DB
CE4D	E8DC
CE4E	E8DD
CE4F	E8DE
CE50	E8DF
CE51	E8E0
CE52	E8E1
CE53	E8E2
CE54	E8E3
CE55	E8E4
CE56	E8E5
CE57	E8E6
CE58	E8E7

CE59 E8E8  
CE5A E8E9  
CE5B E8EA  
CE5C E8EB  
CE5D E8EC  
CE5E E8ED  
CE5F E8EE  
CE60 E8EF  
CE61 E8F0  
CE62 E8F1  
CE63 E8F2  
CE64 E8F3  
CE65 E8F4  
CE66 E8F5  
CE67 E8F6  
CE68 E8F7  
CE69 E8F8  
CE6A E8F9  
CE6B E8FA  
CE6C E8FB  
CE6D E8FC  
CE6E E8FD  
CE6F E8FE  
CE70 E8FF  
CE71 E900  
CE72 E901  
CE73 E902  
CE74 E903  
CE75 E904  
CE76 E905  
CE77 E906  
CE78 E907  
CE79 E908  
CE7A E909  
CE7B E90A  
CE7C E90B  
CE7D E90C  
CE7E E90D  
CE7F E90E  
CE81 E90F  
CE82 E910  
CE83 E911  
CE84 E912  
CE85 E913  
CE86 E914  
CE87 E915  
CE88 E916  
CE89 E917  
CE8A E918  
CE8B E919  
CE8C E91A  
CE8D E91B  
CE8E E91C  
CE8F E91D  
CE90 E91E  
CE91 E91F  
CE92 E920  
CE93 E921  
CE94 E922  
CE95 E923  
CE96 E924  
CE97 E925  
CE98 E926  
CE99 E927  
CE9A E928  
CE9B E929  
CE9C E92A  
CE9D E92B  
CE9E E92C  
CE9F E92D  
CEA0 E92E  
CEA1 E92F  
CEA2 E930  
CEA3 E931  
CEA4 E932  
CEA5 E933  
CEA6 E934  
CEA7 E935  
CEA8 E936  
CEA9 E937  
CEAA E938  
CEAB E939

CEAC	E93A
CEAD	E93B
CEAE	E93C
CEAF	E93D
CEB0	E93E
CEB1	E93F
CEB2	E940
CEB3	E941
CEB4	E942
CEB5	E943
CEB6	E944
CEB7	E945
CEB8	E946
CEB9	E947
CEBA	E948
CEBB	E949
CEBC	E94A
CEBD	E94B
CEBE	E94C
CEBF	E94D
CEC0	E94E
CEC1	E94F
CEC2	E950
CEC3	E951
CEC4	E952
CEC5	E953
CEC6	E954
CEC7	E955
CEC8	E956
CEC9	E957
CECA	E958
CECB	E959
CECC	E95A
CECD	E95B
CECE	E95C
CECF	E95D
CED0	E95E
CED1	E95F
CED2	E960
CED3	E961
CED4	E962
CED5	E963
CED6	E964
CED7	E965
CED8	E966
CED9	E967
CEDA	E968
CEDB	E969
CEDC	E96A
CEDD	E96B
CEDE	E96C
CEDF	E96D
CEE0	E96E
CEE1	E96F
CEE2	E970
CEE3	E971
CEE4	E972
CEE5	E973
CEE6	E974
CEE7	E975
CEE8	E976
CEE9	E977
CEEA	E978
CEEB	E979
CEEC	E97A
CEED	E97B
CEEE	E97C
CEEF	E97D
CEF0	E97E
CEF1	E97F
CEF2	E980
CEF3	E981
CEF4	E982
CEF5	E983
CEF6	E984
CEF7	E985
CEF8	E986
CEF9	E987
CEFA	E988
CEFB	E989
CEFC	E98A
CEFD	E98B

CF41 E98C  
CF42 E98D  
CF43 E98E  
CF44 E98F  
CF45 E990  
CF46 E991  
CF47 E992  
CF48 E993  
CF49 E994  
CF4A E995  
CF4B E996  
CF4C E997  
CF4D E998  
CF4E E999  
CF4F E99A  
CF50 E99B  
CF51 E99C  
CF52 E99D  
CF53 E99E  
CF54 E99F  
CF55 E9A0  
CF56 E9A1  
CF57 E9A2  
CF58 E9A3  
CF59 E9A4  
CF5A E9A5  
CF5B E9A6  
CF5C E9A7  
CF5D E9A8  
CF5E E9A9  
CF5F E9AA  
CF60 E9AB  
CF61 E9AC  
CF62 E9AD  
CF63 E9AE  
CF64 E9AF  
CF65 E9B0  
CF66 E9B1  
CF67 E9B2  
CF68 E9B3  
CF69 E9B4  
CF6A E9B5  
CF6B E9B6  
CF6C E9B7  
CF6D E9B8  
CF6E E9B9  
CF6F E9BA  
CF70 E9BB  
CF71 E9BC  
CF72 E9BD  
CF73 E9BE  
CF74 E9BF  
CF75 E9C0  
CF76 E9C1  
CF77 E9C2  
CF78 E9C3  
CF79 E9C4  
CF7A E9C5  
CF7B E9C6  
CF7C E9C7  
CF7D E9C8  
CF7E E9C9  
CF7F E9CA  
CF81 E9CB  
CF82 E9CC  
CF83 E9CD  
CF84 E9CE  
CF85 E9CF  
CF86 E9D0  
CF87 E9D1  
CF88 E9D2  
CF89 E9D3  
CF8A E9D4  
CF8B E9D5  
CF8C E9D6  
CF8D E9D7  
CF8E E9D8  
CF8F E9D9  
CF90 E9DA  
CF91 E9DB  
CF92 E9DC  
CF93 E9DD

CF94	E9DE
CF95	E9DF
CF96	E9E0
CF97	E9E1
CF98	E9E2
CF99	E9E3
CF9A	E9E4
CF9B	E9E5
CF9C	E9E6
CF9D	E9E7
CF9E	E9E8
CF9F	E9E9
CFA0	E9EA
CFA1	E9EB
CFA2	E9EC
CFA3	E9ED
CFA4	E9EE
CFA5	E9EF
CFA6	E9F0
CFA7	E9F1
CFA8	E9F2
CFA9	E9F3
CFAA	E9F4
CFAB	E9F5
CFAC	E9F6
CFAD	E9F7
CFAE	E9F8
CFAF	E9F9
CFB0	E9FA
CFB1	E9FB
CFB2	E9FC
CFB3	E9FD
CFB4	E9FE
CFB5	E9FF
CFB6	EA00
CFB7	EA01
CFB8	EA02
CFB9	EA03
CFBA	EA04
CFBB	EA05
CFBC	EA06
CFBD	EA07
CFBE	EA08
CFBF	EA09
CFC0	EA0A
CFC1	EA0B
CFC2	EA0C
CFC3	EA0D
CFC4	EA0E
CFC5	EA0F
CFC6	EA10
CFC7	EA11
CFC8	EA12
CFC9	EA13
CFCA	EA14
CFCB	EA15
CFCC	EA16
CFCD	EA17
CFCE	EA18
CFCF	EA19
CFD0	EA1A
CFD1	EA1B
CFD2	EA1C
CFD3	EA1D
CFD4	EA1E
CFD5	EA1F
CFD6	EA20
CFD7	EA21
CFD8	EA22
CFD9	EA23
CFDA	EA24
CFDB	EA25
CFDC	EA26
CFDD	EA27
CFDE	EA28
CFDF	EA29
CFE0	EA2A
CFE1	EA2B
CFE2	EA2C
CFE3	EA2D
CFE4	EA2E
CFE5	EA2F



CFE6	EA30
CFE7	EA31
CFE8	EA32
CFE9	EA33
CFEA	EA34
CFEB	EA35
CFEC	EA36
CFED	EA37
CFEE	EA38
CFEF	EA39
CFF0	EA3A
CFF1	EA3B
CFF2	EA3C
CFF3	EA3D
CFF4	EA3E
CFF5	EA3F
CFF6	EA40
CFF7	EA41
CFF8	EA42
CFF9	EA43
CFFA	EA44
CFFB	EA45
CFFC	EA46
CFFD	EA47
D041	EA48
D042	EA49
D043	EA4A
D044	EA4B
D045	EA4C
D046	EA4D
D047	EA4E
D048	EA4F
D049	EA50
D04A	EA51
D04B	EA52
D04C	EA53
D04D	EA54
D04E	EA55
D04F	EA56
D050	EA57
D051	EA58
D052	EA59
D053	EA5A
D054	EA5B
D055	EA5C
D056	EA5D
D057	EA5E
D058	EA5F
D059	EA60
D05A	EA61
D05B	EA62
D05C	EA63
D05D	EA64
D05E	EA65
D05F	EA66
D060	EA67
D061	EA68
D062	EA69
D063	EA6A
D064	EA6B
D065	EA6C
D066	EA6D
D067	EA6E
D068	EA6F
D069	EA70
D06A	EA71
D06B	EA72
D06C	EA73
D06D	EA74
D06E	EA75
D06F	EA76
D070	EA77
D071	EA78
D072	EA79
D073	EA7A
D074	EA7B
D075	EA7C
D076	EA7D
D077	EA7E
D078	EA7F
D079	EA80
D07A	EA81

D07B EA82  
D07C EA83  
D07D EA84  
D07E EA85  
D07F EA86  
D081 EA87  
D082 EA88  
D083 EA89  
D084 EA8A  
D085 EA8B  
D086 EA8C  
D087 EA8D  
D088 EA8E  
D089 EA8F  
D08A EA90  
D08B EA91  
D08C EA92  
D08D EA93  
D08E EA94  
D08F EA95  
D090 EA96  
D091 EA97  
D092 EA98  
D093 EA99  
D094 EA9A  
D095 EA9B  
D096 EA9C  
D097 EA9D  
D098 EA9E  
D099 EA9F  
D09A EAA0  
D09B EAA1  
D09C EAA2  
D09D EAA3  
D09E EAA4  
D09F EAA5  
D0A0 EAA6  
D0A1 EAA7  
D0A2 EAA8  
D0A3 EAA9  
D0A4 EAAA  
D0A5 EAAB  
D0A6 EAAC  
D0A7 EAAD  
D0A8 EAAE  
D0A9 EAAF  
D0AA EAB0  
D0AB EAB1  
D0AC EAB2  
D0AD EAB3  
D0AE EAB4  
D0AF EAB5  
D0B0 EAB6  
D0B1 EAB7  
D0B2 EAB8  
D0B3 EAB9  
D0B4 EABA  
D0B5 EABB  
D0B6 EABC  
D0B7 EABD  
D0B8 EABE  
D0B9 EABF  
D0BA EAC0  
D0BB EAC1  
D0BC EAC2  
D0BD EAC3  
D0BE EAC4  
D0BF EAC5  
D0C0 EAC6  
D0C1 EAC7  
D0C2 EAC8  
D0C3 EAC9  
D0C4 EACA  
D0C5 EACB  
D0C6 EACC  
D0C7 EACD  
D0C8 EACE  
D0C9 EACF  
D0CA EAD0  
D0CB EAD1  
D0CC EAD2  
D0CD EAD3

D0CE	EAD4
D0CF	EAD5
D0D0	EAD6
D0D1	EAD7
D0D2	EAD8
D0D3	EAD9
D0D4	EADA
D0D5	EADB
D0D6	EADC
D0D7	EADD
D0D8	EADE
D0D9	EADF
D0DA	EAE0
D0DB	EAE1
D0DC	EAE2
D0DD	EAE3
D0DE	EAE4
D0DF	EAE5
D0E0	EAE6
D0E1	EAE7
D0E2	EAE8
D0E3	EAE9
D0E4	EAEA
D0E5	EAEB
D0E6	EAEC
D0E7	EAED
D0E8	EAEF
D0E9	EAEF
D0EA	EAFA
D0EB	EAFA
D0EC	EAFA
D0ED	EAFA
D0EE	EAFA
D0EF	EAFA
D0F0	EAFA
D0F1	EAFA
D0F2	EAFA
D0F3	EAFA
D0F4	EAFA
D0F5	EAFA
D0F6	EAFA
D0F7	EAFA
D0F8	EAFA
D0F9	EAFA
D0FA	EB00
D0FB	EB01
D0FC	EB02
D0FD	EB03
D141	EB04
D142	EB05
D143	EB06
D144	EB07
D145	EB08
D146	EB09
D147	EB0A
D148	EB0B
D149	EB0C
D14A	EB0D
D14B	EB0E
D14C	EB0F
D14D	EB10
D14E	EB11
D14F	EB12
D150	EB13
D151	EB14
D152	EB15
D153	EB16
D154	EB17
D155	EB18
D156	EB19
D157	EB1A
D158	EB1B
D159	EB1C
D15A	EB1D
D15B	EB1E
D15C	EB1F
D15D	EB20
D15E	EB21
D15F	EB22
D160	EB23
D161	EB24
D162	EB25

D163	EB26
D164	EB27
D165	EB28
D166	EB29
D167	EB2A
D168	EB2B
D169	EB2C
D16A	EB2D
D16B	EB2E
D16C	EB2F
D16D	EB30
D16E	EB31
D16F	EB32
D170	EB33
D171	EB34
D172	EB35
D173	EB36
D174	EB37
D175	EB38
D176	EB39
D177	EB3A
D178	EB3B
D179	EB3C
D17A	EB3D
D17B	EB3E
D17C	EB3F
D17D	EB40
D17E	EB41
D17F	EB42
D181	EB43
D182	EB44
D183	EB45
D184	EB46
D185	EB47
D186	EB48
D187	EB49
D188	EB4A
D189	EB4B
D18A	EB4C
D18B	EB4D
D18C	EB4E
D18D	EB4F
D18E	EB50
D18F	EB51
D190	EB52
D191	EB53
D192	EB54
D193	EB55
D194	EB56
D195	EB57
D196	EB58
D197	EB59
D198	EB5A
D199	EB5B
D19A	EB5C
D19B	EB5D
D19C	EB5E
D19D	EB5F
D19E	EB60
D19F	EB61
D1A0	EB62
D1A1	EB63
D1A2	EB64
D1A3	EB65
D1A4	EB66
D1A5	EB67
D1A6	EB68
D1A7	EB69
D1A8	EB6A
D1A9	EB6B
D1AA	EB6C
D1AB	EB6D
D1AC	EB6E
D1AD	EB6F
D1AE	EB70
D1AF	EB71
D1B0	EB72
D1B1	EB73
D1B2	EB74
D1B3	EB75
D1B4	EB76
D1B5	EB77

D1B6	EB78
D1B7	EB79
D1B8	EB7A
D1B9	EB7B
D1BA	EB7C
D1BB	EB7D
D1BC	EB7E
D1BD	EB7F
D1BE	EB80
D1BF	EB81
D1C0	EB82
D1C1	EB83
D1C2	EB84
D1C3	EB85
D1C4	EB86
D1C5	EB87
D1C6	EB88
D1C7	EB89
D1C8	EB8A
D1C9	EB8B
D1CA	EB8C
D1CB	EB8D
D1CC	EB8E
D1CD	EB8F
D1CE	EB90
D1CF	EB91
D1D0	EB92
D1D1	EB93
D1D2	EB94
D1D3	EB95
D1D4	EB96
D1D5	EB97
D1D6	EB98
D1D7	EB99
D1D8	EB9A
D1D9	EB9B
D1DA	EB9C
D1DB	EB9D
D1DC	EB9E
D1DD	EB9F
D1DE	EBA0
D1DF	EBA1
D1E0	EBA2
D1E1	EBA3
D1E2	EBA4
D1E3	EBA5
D1E4	EBA6
D1E5	EBA7
D1E6	EBA8
D1E7	EBA9
D1E8	EBAA
D1E9	EBAB
D1EA	EBAC
D1EB	EBAD
D1EC	EBAE
D1ED	EBAF
D1EE	EBB0
D1EF	EBB1
D1F0	EBB2
D1F1	EBB3
D1F2	EBB4
D1F3	EBB5
D1F4	EBB6
D1F5	EBB7
D1F6	EBB8
D1F7	EBB9
D1F8	EBBA
D1F9	EBBB
D1FA	EBBC
D1FB	EBBD
D1FC	EBBE
D1FD	EBBF
D241	EBC0
D242	EBC1
D243	EBC2
D244	EBC3
D245	EBC4
D246	EBC5
D247	EBC6
D248	EBC7
D249	EBC8
D24A	EBC9

D24B EBCA  
D24C EBCB  
D24D EBCC  
D24E EBCD  
D24F EBCE  
D250 EBCF  
D251 EBD0  
D252 EBD1  
D253 EBD2  
D254 EBD3  
D255 EBD4  
D256 EBD5  
D257 EBD6  
D258 EBD7  
D259 EBD8  
D25A EBD9  
D25B EBDA  
D25C EBDB  
D25D EBDC  
D25E EBDD  
D25F EBDE  
D260 EBD F  
D261 EBE0  
D262 EBE1  
D263 EBE2  
D264 EBE3  
D265 EBE4  
D266 EBE5  
D267 EBE6  
D268 EBE7  
D269 EBE8  
D26A EBE9  
D26B EB EA  
D26C EB EB  
D26D EB EC  
D26E EB ED  
D26F EB EE  
D270 EB EF  
D271 EB F0  
D272 EB F1  
D273 EB F2  
D274 EB F3  
D275 EB F4  
D276 EB F5  
D277 EB F6  
D278 EB F7  
D279 EB F8  
D27A EB F9  
D27B EB FA  
D27C EB FB  
D27D EB FC  
D27E EB FD  
D27F EB FE  
D281 EB FF  
D282 EC00  
D283 EC01  
D284 EC02  
D285 EC03  
D286 EC04  
D287 EC05  
D288 EC06  
D289 EC07  
D28A EC08  
D28B EC09  
D28C EC0A  
D28D EC0B  
D28E EC0C  
D28F EC0D  
D290 EC0E  
D291 EC0F  
D292 EC10  
D293 EC11  
D294 EC12  
D295 EC13  
D296 EC14  
D297 EC15  
D298 EC16  
D299 EC17  
D29A EC18  
D29B EC19  
D29C EC1A  
D29D EC1B

D29E EC1C  
D29F EC1D  
D2A0 EC1E  
D2A1 EC1F  
D2A2 EC20  
D2A3 EC21  
D2A4 EC22  
D2A5 EC23  
D2A6 EC24  
D2A7 EC25  
D2A8 EC26  
D2A9 EC27  
D2AA EC28  
D2AB EC29  
D2AC EC2A  
D2AD EC2B  
D2AE EC2C  
D2AF EC2D  
D2B0 EC2E  
D2B1 EC2F  
D2B2 EC30  
D2B3 EC31  
D2B4 EC32  
D2B5 EC33  
D2B6 EC34  
D2B7 EC35  
D2B8 EC36  
D2B9 EC37  
D2BA EC38  
D2BB EC39  
D2BC EC3A  
D2BD EC3B  
D2BE EC3C  
D2BF EC3D  
D2C0 EC3E  
D2C1 EC3F  
D2C2 EC40  
D2C3 EC41  
D2C4 EC42  
D2C5 EC43  
D2C6 EC44  
D2C7 EC45  
D2C8 EC46  
D2C9 EC47  
D2CA EC48  
D2CB EC49  
D2CC EC4A  
D2CD EC4B  
D2CE EC4C  
D2CF EC4D  
D2D0 EC4E  
D2D1 EC4F  
D2D2 EC50  
D2D3 EC51  
D2D4 EC52  
D2D5 EC53  
D2D6 EC54  
D2D7 EC55  
D2D8 EC56  
D2D9 EC57  
D2DA EC58  
D2DB EC59  
D2DC EC5A  
D2DD EC5B  
D2DE EC5C  
D2DF EC5D  
D2E0 EC5E  
D2E1 EC5F  
D2E2 EC60  
D2E3 EC61  
D2E4 EC62  
D2E5 EC63  
D2E6 EC64  
D2E7 EC65  
D2E8 EC66  
D2E9 EC67  
D2EA EC68  
D2EB EC69  
D2EC EC6A  
D2ED EC6B  
D2EE EC6C  
D2EF EC6D

D2F0 EC6E  
D2F1 EC6F  
D2F2 EC70  
D2F3 EC71  
D2F4 EC72  
D2F5 EC73  
D2F6 EC74  
D2F7 EC75  
D2F8 EC76  
D2F9 EC77  
D2FA EC78  
D2FB EC79  
D2FC EC7A  
D2FD EC7B  
D341 EC7C  
D342 EC7D  
D343 EC7E  
D344 EC7F  
D345 EC80  
D346 EC81  
D347 EC82  
D348 EC83  
D349 EC84  
D34A EC85  
D34B EC86  
D34C EC87  
D34D EC88  
D34E EC89  
D34F EC8A  
D350 EC8B  
D351 EC8C  
D352 EC8D  
D353 EC8E  
D354 EC8F  
D355 EC90  
D356 EC91  
D357 EC92  
D358 EC93  
D359 EC94  
D35A EC95  
D35B EC96  
D35C EC97  
D35D EC98  
D35E EC99  
D35F EC9A  
D360 EC9B  
D361 EC9C  
D362 EC9D  
D363 EC9E  
D364 EC9F  
D365 ECA0  
D366 ECA1  
D367 ECA2  
D368 ECA3  
D369 ECA4  
D36A ECA5  
D36B ECA6  
D36C ECA7  
D36D ECA8  
D36E ECA9  
D36F ECAA  
D370 ECAB  
D371 ECAC  
D372 ECAD  
D373 ECAE  
D374 ECAF  
D375 ECB0  
D376 ECB1  
D377 ECB2  
D378 ECB3  
D379 ECB4  
D37A ECB5  
D37B ECB6  
D37C ECB7  
D37D ECB8  
D37E ECB9  
D37F ECBA  
D381 ECBB  
D382 ECBC  
D383 ECBD  
D384 ECBE  
D385 ECBF



D386 ECC0  
D387 ECC1  
D388 ECC2  
D389 ECC3  
D38A ECC4  
D38B ECC5  
D38C ECC6  
D38D ECC7  
D38E ECC8  
D38F ECC9  
D390 ECCA  
D391 ECCB  
D392 ECCC  
D393 ECCD  
D394 ECCE  
D395 ECCF  
D396 ECD0  
D397 ECD1  
D398 ECD2  
D399 ECD3  
D39A ECD4  
D39B ECD5  
D39C ECD6  
D39D ECD7  
D39E ECD8  
D39F ECD9  
D3A0 ECDA  
D3A1 ECDB  
D3A2 ECDC  
D3A3 ECDD  
D3A4 ECDE  
D3A5 ECDF  
D3A6 ECE0  
D3A7 ECE1  
D3A8 ECE2  
D3A9 ECE3  
D3AA ECE4  
D3AB ECE5  
D3AC ECE6  
D3AD ECE7  
D3AE ECE8  
D3AF ECE9  
D3B0 ECEA  
D3B1 ECEB  
D3B2 ECEC  
D3B3 ECED  
D3B4 ECEE  
D3B5 ECEF  
D3B6 ECF0  
D3B7 ECF1  
D3B8 ECF2  
D3B9 ECF3  
D3BA ECF4  
D3BB ECF5  
D3BC ECF6  
D3BD ECF7  
D3BE ECF8  
D3BF ECF9  
D3C0 ECFA  
D3C1 ECFB  
D3C2 ECFC  
D3C3 ECFD  
D3C4 ECFE  
D3C5 ECFF  
D3C6 ED00  
D3C7 ED01  
D3C8 ED02  
D3C9 ED03  
D3CA ED04  
D3CB ED05  
D3CC ED06  
D3CD ED07  
D3CE ED08  
D3CF ED09  
D3D0 ED0A  
D3D1 ED0B  
D3D2 ED0C  
D3D3 ED0D  
D3D4 ED0E  
D3D5 ED0F  
D3D6 ED10  
D3D7 ED11

D3D8	ED12
D3D9	ED13
D3DA	ED14
D3DB	ED15
D3DC	ED16
D3DD	ED17
D3DE	ED18
D3DF	ED19
D3E0	ED1A
D3E1	ED1B
D3E2	ED1C
D3E3	ED1D
D3E4	ED1E
D3E5	ED1F
D3E6	ED20
D3E7	ED21
D3E8	ED22
D3E9	ED23
D3EA	ED24
D3EB	ED25
D3EC	ED26
D3ED	ED27
D3EE	ED28
D3EF	ED29
D3F0	ED2A
D3F1	ED2B
D3F2	ED2C
D3F3	ED2D
D3F4	ED2E
D3F5	ED2F
D3F6	ED30
D3F7	ED31
D3F8	ED32
D3F9	ED33
D3FA	ED34
D3FB	ED35
D3FC	ED36
D3FD	ED37
D441	ED38
D442	ED39
D443	ED3A
D444	ED3B
D445	ED3C
D446	ED3D
D447	ED3E
D448	ED3F
D449	ED40
D44A	ED41
D44B	ED42
D44C	ED43
D44D	ED44
D44E	ED45
D44F	ED46
D450	ED47
D451	ED48
D452	ED49
D453	ED4A
D454	ED4B
D455	ED4C
D456	ED4D
D457	ED4E
D458	ED4F
D459	ED50
D45A	ED51
D45B	ED52
D45C	ED53
D45D	ED54
D45E	ED55
D45F	ED56
D460	ED57
D461	ED58
D462	ED59
D463	ED5A
D464	ED5B
D465	ED5C
D466	ED5D
D467	ED5E
D468	ED5F
D469	ED60
D46A	ED61
D46B	ED62
D46C	ED63

D46D ED64  
D46E ED65  
D46F ED66  
D470 ED67  
D471 ED68  
D472 ED69  
D473 ED6A  
D474 ED6B  
D475 ED6C  
D476 ED6D  
D477 ED6E  
D478 ED6F  
D479 ED70  
D47A ED71  
D47B ED72  
D47C ED73  
D47D ED74  
D47E ED75  
D47F ED76  
D481 ED77  
D482 ED78  
D483 ED79  
D484 ED7A  
D485 ED7B  
D486 ED7C  
D487 ED7D  
D488 ED7E  
D489 ED7F  
D48A ED80  
D48B ED81  
D48C ED82  
D48D ED83  
D48E ED84  
D48F ED85  
D490 ED86  
D491 ED87  
D492 ED88  
D493 ED89  
D494 ED8A  
D495 ED8B  
D496 ED8C  
D497 ED8D  
D498 ED8E  
D499 ED8F  
D49A ED90  
D49B ED91  
D49C ED92  
D49D ED93  
D49E ED94  
D49F ED95  
D4A0 ED96  
D4A1 ED97  
D4A2 ED98  
D4A3 ED99  
D4A4 ED9A  
D4A5 ED9B  
D4A6 ED9C  
D4A7 ED9D  
D4A8 ED9E  
D4A9 ED9F  
D4AA EDA0  
D4AB EDA1  
D4AC EDA2  
D4AD EDA3  
D4AE EDA4  
D4AF EDA5  
D4B0 EDA6  
D4B1 EDA7  
D4B2 EDA8  
D4B3 EDA9  
D4B4 EDAA  
D4B5 EDAB  
D4B6 EDAC  
D4B7 EDAD  
D4B8 EDAE  
D4B9 EDAF  
D4BA EDB0  
D4BB EDB1  
D4BC EDB2  
D4BD EDB3  
D4BE EDB4  
D4BF EDB5

D4C0	EDB6
D4C1	EDB7
D4C2	EDB8
D4C3	EDB9
D4C4	EDBA
D4C5	EDBB
D4C6	EDBC
D4C7	EDBD
D4C8	EDBE
D4C9	EDBF
D4CA	EDC0
D4CB	EDC1
D4CC	EDC2
D4CD	EDC3
D4CE	EDC4
D4CF	EDC5
D4D0	EDC6
D4D1	EDC7
D4D2	EDC8
D4D3	EDC9
D4D4	EDCA
D4D5	EDCB
D4D6	EDCC
D4D7	EDCD
D4D8	EDCE
D4D9	EDCF
D4DA	EDD0
D4DB	EDD1
D4DC	EDD2
D4DD	EDD3
D4DE	EDD4
D4DF	EDD5
D4E0	EDD6
D4E1	EDD7
D4E2	EDD8
D4E3	EDD9
D4E4	EDDA
D4E5	EDDB
D4E6	EDDC
D4E7	EDDD
D4E8	EDDE
D4E9	EDDF
D4EA	EDE0
D4EB	EDE1
D4EC	EDE2
D4ED	EDE3
D4EE	EDE4
D4EF	EDE5
D4F0	EDE6
D4F1	EDE7
D4F2	EDE8
D4F3	EDE9
D4F4	EDEA
D4F5	EDEB
D4F6	EDEC
D4F7	EDED
D4F8	EDEE
D4F9	EDEF
D4FA	EDF0
D4FB	EDF1
D4FC	EDF2
D4FD	EDF3
D541	EDF4
D542	EDF5
D543	EDF6
D544	EDF7
D545	EDF8
D546	EDF9
D547	EDFA
D548	EDFB
D549	EDFC
D54A	EDFD
D54B	EDFE
D54C	EDFF
D54D	EE00
D54E	EE01
D54F	EE02
D550	EE03
D551	EE04
D552	EE05
D553	EE06
D554	EE07

D555 EE08  
D556 EE09  
D557 EE0A  
D558 EE0B  
D559 EE0C  
D55A EE0D  
D55B EE0E  
D55C EE0F  
D55D EE10  
D55E EE11  
D55F EE12  
D560 EE13  
D561 EE14  
D562 EE15  
D563 EE16  
D564 EE17  
D565 EE18  
D566 EE19  
D567 EE1A  
D568 EE1B  
D569 EE1C  
D56A EE1D  
D56B EE1E  
D56C EE1F  
D56D EE20  
D56E EE21  
D56F EE22  
D570 EE23  
D571 EE24  
D572 EE25  
D573 EE26  
D574 EE27  
D575 EE28  
D576 EE29  
D577 EE2A  
D578 EE2B  
D579 EE2C  
D57A EE2D  
D57B EE2E  
D57C EE2F  
D57D EE30  
D57E EE31  
D57F EE32  
D581 EE33  
D582 EE34  
D583 EE35  
D584 EE36  
D585 EE37  
D586 EE38  
D587 EE39  
D588 EE3A  
D589 EE3B  
D58A EE3C  
D58B EE3D  
D58C EE3E  
D58D EE3F  
D58E EE40  
D58F EE41  
D590 EE42  
D591 EE43  
D592 EE44  
D593 EE45  
D594 EE46  
D595 EE47  
D596 EE48  
D597 EE49  
D598 EE4A  
D599 EE4B  
D59A EE4C  
D59B EE4D  
D59C EE4E  
D59D EE4F  
D59E EE50  
D59F EE51  
D5A0 EE52  
D5A1 EE53  
D5A2 EE54  
D5A3 EE55  
D5A4 EE56  
D5A5 EE57  
D5A6 EE58  
D5A7 EE59

D5A8	EE5A
D5A9	EE5B
D5AA	EE5C
D5AB	EE5D
D5AC	EE5E
D5AD	EE5F
D5AE	EE60
D5AF	EE61
D5B0	EE62
D5B1	EE63
D5B2	EE64
D5B3	EE65
D5B4	EE66
D5B5	EE67
D5B6	EE68
D5B7	EE69
D5B8	EE6A
D5B9	EE6B
D5BA	EE6C
D5BB	EE6D
D5BC	EE6E
D5BD	EE6F
D5BE	EE70
D5BF	EE71
D5C0	EE72
D5C1	EE73
D5C2	EE74
D5C3	EE75
D5C4	EE76
D5C5	EE77
D5C6	EE78
D5C7	EE79
D5C8	EE7A
D5C9	EE7B
D5CA	EE7C
D5CB	EE7D
D5CC	EE7E
D5CD	EE7F
D5CE	EE80
D5CF	EE81
D5D0	EE82
D5D1	EE83
D5D2	EE84
D5D3	EE85
D5D4	EE86
D5D5	EE87
D5D6	EE88
D5D7	EE89
D5D8	EE8A
D5D9	EE8B
D5DA	EE8C
D5DB	EE8D
D5DC	EE8E
D5DD	EE8F
D5DE	EE90
D5DF	EE91
D5E0	EE92
D5E1	EE93
D5E2	EE94
D5E3	EE95
D5E4	EE96
D5E5	EE97
D5E6	EE98
D5E7	EE99
D5E8	EE9A
D5E9	EE9B
D5EA	EE9C
D5EB	EE9D
D5EC	EE9E
D5ED	EE9F
D5EE	EEA0
D5EF	EEA1
D5F0	EEA2
D5F1	EEA3
D5F2	EEA4
D5F3	EEA5
D5F4	EEA6
D5F5	EEA7
D5F6	EEA8
D5F7	EEA9
D5F8	EEAA
D5F9	EEAB

D5FA EEAC  
D5FB EEAD  
D5FC EEAE  
D5FD EEAF  
D641 EEB0  
D642 EEB1  
D643 EEB2  
D644 EEB3  
D645 EEB4  
D646 EEB5  
D647 EEB6  
D648 EEB7  
D649 EEB8  
D64A EEB9  
D64B EEBA  
D64C EEBC  
D64D EEBC  
D64E EEBD  
D64F EEBE  
D650 EEBF  
D651 EEC0  
D652 EEC1  
D653 EEC2  
D654 EEC3  
D655 EEC4  
D656 EEC5  
D657 EEC6  
D658 EEC7  
D659 EEC8  
D65A EEC9  
D65B EECA  
D65C EECB  
D65D EECC  
D65E EECD  
D65F EECE  
D660 EECF  
D661 EED0  
D662 EED1  
D663 EED2  
D664 EED3  
D665 EED4  
D666 EED5  
D667 EED6  
D668 EED7  
D669 EED8  
D66A EED9  
D66B EEDA  
D66C EEDB  
D66D EEDC  
D66E EEDD  
D66F EEDE  
D670 EEDF  
D671 EEE0  
D672 EEE1  
D673 EEE2  
D674 EEE3  
D675 EEE4  
D676 EEE5  
D677 EEE6  
D678 EEE7  
D679 EEE8  
D67A EEE9  
D67B EEEA  
D67C EEEB  
D67D EEEC  
D67E EEEF  
D67F EEEF  
D681 EEEF  
D682 EEF0  
D683 EEF1  
D684 EEF2  
D685 EEF3  
D686 EEF4  
D687 EEF5  
D688 EEF6  
D689 EEF7  
D68A EEF8  
D68B EEF9  
D68C EEFA  
D68D EEFB  
D68E EEFC  
D68F EEFD

D690	EEFE
D691	EEFF
D692	EF00
D693	EF01
D694	EF02
D695	EF03
D696	EF04
D697	EF05
D698	EF06
D699	EF07
D69A	EF08
D69B	EF09
D69C	EF0A
D69D	EF0B
D69E	EF0C
D69F	EF0D
D6A0	EF0E
D6A1	EF0F
D6A2	EF10
D6A3	EF11
D6A4	EF12
D6A5	EF13
D6A6	EF14
D6A7	EF15
D6A8	EF16
D6A9	EF17
D6AA	EF18
D6AB	EF19
D6AC	EF1A
D6AD	EF1B
D6AE	EF1C
D6AF	EF1D
D6B0	EF1E
D6B1	EF1F
D6B2	EF20
D6B3	EF21
D6B4	EF22
D6B5	EF23
D6B6	EF24
D6B7	EF25
D6B8	EF26
D6B9	EF27
D6BA	EF28
D6BB	EF29
D6BC	EF2A
D6BD	EF2B
D6BE	EF2C
D6BF	EF2D
D6C0	EF2E
D6C1	EF2F
D6C2	EF30
D6C3	EF31
D6C4	EF32
D6C5	EF33
D6C6	EF34
D6C7	EF35
D6C8	EF36
D6C9	EF37
D6CA	EF38
D6CB	EF39
D6CC	EF3A
D6CD	EF3B
D6CE	EF3C
D6CF	EF3D
D6D0	EF3E
D6D1	EF3F
D6D2	EF40
D6D3	EF41
D6D4	EF42
D6D5	EF43
D6D6	EF44
D6D7	EF45
D6D8	EF46
D6D9	EF47
D6DA	EF48
D6DB	EF49
D6DC	EF4A
D6DD	EF4B
D6DE	EF4C
D6DF	EF4D
D6E0	EF4E
D6E1	EF4F



D6E2 EF50  
D6E3 EF51  
D6E4 EF52  
D6E5 EF53  
D6E6 EF54  
D6E7 EF55  
D6E8 EF56  
D6E9 EF57  
D6EA EF58  
D6EB EF59  
D6EC EF5A  
D6ED EF5B  
D6EE EF5C  
D6EF EF5D  
D6F0 EF5E  
D6F1 EF5F  
D6F2 EF60  
D6F3 EF61  
D6F4 EF62  
D6F5 EF63  
D6F6 EF64  
D6F7 EF65  
D6F8 EF66  
D6F9 EF67  
D6FA EF68  
D6FB EF69  
D6FC EF6A  
D6FD EF6B  
D741 EF6C  
D742 EF6D  
D743 EF6E  
D744 EF6F  
D745 EF70  
D746 EF71  
D747 EF72  
D748 EF73  
D749 EF74  
D74A EF75  
D74B EF76  
D74C EF77  
D74D EF78  
D74E EF79  
D74F EF7A  
D750 EF7B  
D751 EF7C  
D752 EF7D  
D753 EF7E  
D754 EF7F  
D755 EF80  
D756 EF81  
D757 EF82  
D758 EF83  
D759 EF84  
D75A EF85  
D75B EF86  
D75C EF87  
D75D EF88  
D75E EF89  
D75F EF8A  
D760 EF8B  
D761 EF8C  
D762 EF8D  
D763 EF8E  
D764 EF8F  
D765 EF90  
D766 EF91  
D767 EF92  
D768 EF93  
D769 EF94  
D76A EF95  
D76B EF96  
D76C EF97  
D76D EF98  
D76E EF99  
D76F EF9A  
D770 EF9B  
D771 EF9C  
D772 EF9D  
D773 EF9E  
D774 EF9F  
D775 EFA0  
D776 EFA1

D777 EFA2  
D778 EFA3  
D779 EFA4  
D77A EFA5  
D77B EFA6  
D77C EFA7  
D77D EFA8  
D77E EFA9  
D77F EFAA  
D781 EFAB  
D782 EFAC  
D783 EFAD  
D784 EFAE  
D785 EFAF  
D786 EFB0  
D787 EFB1  
D788 EFB2  
D789 EFB3  
D78A EFB4  
D78B EFB5  
D78C EFB6  
D78D EFB7  
D78E EFB8  
D78F EFB9  
D790 EFBA  
D791 EFBB  
D792 EFBC  
D793 EFBD  
D794 EFBE  
D795 EFBF  
D796 EFC0  
D797 EFC1  
D798 EFC2  
D799 EFC3  
D79A EFC4  
D79B EFC5  
D79C EFC6  
D79D EFC7  
D79E EFC8  
D79F EFC9  
D7A0 EFCA  
D7A1 EFCA  
D7A2 EFCC  
D7A3 EFCD  
D7A4 EFCE  
D7A5 EFCF  
D7A6 EFD0  
D7A7 EFD1  
D7A8 EFD2  
D7A9 EFD3  
D7AA EFD4  
D7AB EFD5  
D7AC EFD6  
D7AD EFD7  
D7AE EFD8  
D7AF EFD9  
D7B0 EFDA  
D7B1 EFDB  
D7B2 EFDC  
D7B3 EFDD  
D7B4 EFDE  
D7B5 EFDF  
D7B6 EFE0  
D7B7 EFE1  
D7B8 EFE2  
D7B9 EFE3  
D7BA EFE4  
D7BB EFE5  
D7BC EFE6  
D7BD EFE7  
D7BE EFE8  
D7BF EFE9  
D7C0 EFEA  
D7C1 EFEB  
D7C2 EFEC  
D7C3 EFED  
D7C4 EFEE  
D7C5 EFEF  
D7C6 EFF0  
D7C7 EFF1  
D7C8 EFF2  
D7C9 EFF3

D7CA	EFF4
D7CB	EFF5
D7CC	EFF6
D7CD	EFF7
D7CE	EFF8
D7CF	EFF9
D7D0	EFFA
D7D1	EFFB
D7D2	EFFC
D7D3	EFFD
D7D4	EFFE
D7D5	EFFF
D7D6	F000
D7D7	F001
D7D8	F002
D7D9	F003
D7DA	F004
D7DB	F005
D7DC	F006
D7DD	F007
D7DE	F008
D7DF	F009
D7E0	F00A
D7E1	F00B
D7E2	F00C
D7E3	F00D
D7E4	F00E
D7E5	F00F
D7E6	F010
D7E7	F011
D7E8	F012
D7E9	F013
D7EA	F014
D7EB	F015
D7EC	F016
D7ED	F017
D7EE	F018
D7EF	F019
D7F0	F01A
D7F1	F01B
D7F2	F01C
D7F3	F01D
D7F4	F01E
D7F5	F01F
D7F6	F020
D7F7	F021
D7F8	F022
D7F9	F023
D7FA	F024
D7FB	F025
D7FC	F026
D7FD	F027
D841	F028
D842	F029
D843	F02A
D844	F02B
D845	F02C
D846	F02D
D847	F02E
D848	F02F
D849	F030
D84A	F031
D84B	F032
D84C	F033
D84D	F034
D84E	F035
D84F	F036
D850	F037
D851	F038
D852	F039
D853	F03A
D854	F03B
D855	F03C
D856	F03D
D857	F03E
D858	F03F
D859	F040
D85A	F041
D85B	F042
D85C	F043
D85D	F044
D85E	F045

D85F F046  
D860 F047  
D861 F048  
D862 F049  
D863 F04A  
D864 F04B  
D865 F04C  
D866 F04D  
D867 F04E  
D868 F04F  
D869 F050  
D86A F051  
D86B F052  
D86C F053  
D86D F054  
D86E F055  
D86F F056  
D870 F057  
D871 F058  
D872 F059  
D873 F05A  
D874 F05B  
D875 F05C  
D876 F05D  
D877 F05E  
D878 F05F  
D879 F060  
D87A F061  
D87B F062  
D87C F063  
D87D F064  
D87E F065  
D87F F066  
D881 F067  
D882 F068  
D883 F069  
D884 F06A  
D885 F06B  
D886 F06C  
D887 F06D  
D888 F06E  
D889 F06F  
D88A F070  
D88B F071  
D88C F072  
D88D F073  
D88E F074  
D88F F075  
D890 F076  
D891 F077  
D892 F078  
D893 F079  
D894 F07A  
D895 F07B  
D896 F07C  
D897 F07D  
D898 F07E  
D899 F07F  
D89A F080  
D89B F081  
D89C F082  
D89D F083  
D89E F084  
D89F F085  
D8A0 F086  
D8A1 F087  
D8A2 F088  
D8A3 F089  
D8A4 F08A  
D8A5 F08B  
D8A6 F08C  
D8A7 F08D  
D8A8 F08E  
D8A9 F08F  
D8AA F090  
D8AB F091  
D8AC F092  
D8AD F093  
D8AE F094  
D8AF F095  
D8B0 F096  
D8B1 F097

D8B2 F098  
D8B3 F099  
D8B4 F09A  
D8B5 F09B  
D8B6 F09C  
D8B7 F09D  
D8B8 F09E  
D8B9 F09F  
D8BA F0A0  
D8BB F0A1  
D8BC F0A2  
D8BD F0A3  
D8BE F0A4  
D8BF F0A5  
D8C0 F0A6  
D8C1 F0A7  
D8C2 F0A8  
D8C3 F0A9  
D8C4 F0AA  
D8C5 F0AB  
D8C6 F0AC  
D8C7 F0AD  
D8C8 F0AE  
D8C9 F0AF  
D8CA F0B0  
D8CB F0B1  
D8CC F0B2  
D8CD F0B3  
D8CE F0B4  
D8CF F0B5  
D8D0 F0B6  
D8D1 F0B7  
D8D2 F0B8  
D8D3 F0B9  
D8D4 F0BA  
D8D5 F0BB  
D8D6 F0BC  
D8D7 F0BD  
D8D8 F0BE  
D8D9 F0BF  
D8DA F0C0  
D8DB F0C1  
D8DC F0C2  
D8DD F0C3  
D8DE F0C4  
D8DF F0C5  
D8E0 F0C6  
D8E1 F0C7  
D8E2 F0C8  
D8E3 F0C9  
D8E4 F0CA  
D8E5 F0CB  
D8E6 F0CC  
D8E7 F0CD  
D8E8 F0CE  
D8E9 F0CF  
D8EA F0D0  
D8EB F0D1  
D8EC F0D2  
D8ED F0D3  
D8EE F0D4  
D8EF F0D5  
D8F0 F0D6  
D8F1 F0D7  
D8F2 F0D8  
D8F3 F0D9  
D8F4 F0DA  
D8F5 F0DB  
D8F6 F0DC  
D8F7 F0DD  
D8F8 F0DE  
D8F9 F0DF  
D8FA F0E0  
D8FB F0E1  
D8FC F0E2  
D8FD F0E3  
D941 F0E4  
D942 F0E5  
D943 F0E6  
D944 F0E7  
D945 F0E8  
D946 F0E9

D947 F0EA  
D948 F0EB  
D949 F0EC  
D94A F0ED  
D94B F0EE  
D94C F0EF  
D94D F0F0  
D94E F0F1  
D94F F0F2  
D950 F0F3  
D951 F0F4  
D952 F0F5  
D953 F0F6  
D954 F0F7  
D955 F0F8  
D956 F0F9  
D957 F0FA  
D958 F0FB  
D959 F0FC  
D95A F0FD  
D95B F0FE  
D95C F0FF  
D95D F100  
D95E F101  
D95F F102  
D960 F103  
D961 F104  
D962 F105  
D963 F106  
D964 F107  
D965 F108  
D966 F109  
D967 F10A  
D968 F10B  
D969 F10C  
D96A F10D  
D96B F10E  
D96C F10F  
D96D F110  
D96E F111  
D96F F112  
D970 F113  
D971 F114  
D972 F115  
D973 F116  
D974 F117  
D975 F118  
D976 F119  
D977 F11A  
D978 F11B  
D979 F11C  
D97A F11D  
D97B F11E  
D97C F11F  
D97D F120  
D97E F121  
D97F F122  
D981 F123  
D982 F124  
D983 F125  
D984 F126  
D985 F127  
D986 F128  
D987 F129  
D988 F12A  
D989 F12B  
D98A F12C  
D98B F12D  
D98C F12E  
D98D F12F  
D98E F130  
D98F F131  
D990 F132  
D991 F133  
D992 F134  
D993 F135  
D994 F136  
D995 F137  
D996 F138  
D997 F139  
D998 F13A  
D999 F13B

D99A F13C  
D99B F13D  
D99C F13E  
D99D F13F  
D99E F140  
D99F F141  
D9A0 F142  
D9A1 F143  
D9A2 F144  
D9A3 F145  
D9A4 F146  
D9A5 F147  
D9A6 F148  
D9A7 F149  
D9A8 F14A  
D9A9 F14B  
D9AA F14C  
D9AB F14D  
D9AC F14E  
D9AD F14F  
D9AE F150  
D9AF F151  
D9B0 F152  
D9B1 F153  
D9B2 F154  
D9B3 F155  
D9B4 F156  
D9B5 F157  
D9B6 F158  
D9B7 F159  
D9B8 F15A  
D9B9 F15B  
D9BA F15C  
D9BB F15D  
D9BC F15E  
D9BD F15F  
D9BE F160  
D9BF F161  
D9C0 F162  
D9C1 F163  
D9C2 F164  
D9C3 F165  
D9C4 F166  
D9C5 F167  
D9C6 F168  
D9C7 F169  
D9C8 F16A  
D9C9 F16B  
D9CA F16C  
D9CB F16D  
D9CC F16E  
D9CD F16F  
D9CE F170  
D9CF F171  
D9D0 F172  
D9D1 F173  
D9D2 F174  
D9D3 F175  
D9D4 F176  
D9D5 F177  
D9D6 F178  
D9D7 F179  
D9D8 F17A  
D9D9 F17B  
D9DA F17C  
D9DB F17D  
D9DC F17E  
D9DD F17F  
D9DE F180  
D9DF F181  
D9E0 F182  
D9E1 F183  
D9E2 F184  
D9E3 F185  
D9E4 F186  
D9E5 F187  
D9E6 F188  
D9E7 F189  
D9E8 F18A  
D9E9 F18B  
D9EA F18C  
D9EB F18D

D9EC F18E  
D9ED F18F  
D9EE F190  
D9EF F191  
D9F0 F192  
D9F1 F193  
D9F2 F194  
D9F3 F195  
D9F4 F196  
D9F5 F197  
D9F6 F198  
D9F7 F199  
D9F8 F19A  
D9F9 F19B  
D9FA F19C  
D9FB F19D  
D9FC F19E  
D9FD F19F  
DA41 F1A0  
DA42 F1A1  
DA43 F1A2  
DA44 F1A3  
DA45 F1A4  
DA46 F1A5  
DA47 F1A6  
DA48 F1A7  
DA49 F1A8  
DA4A F1A9  
DA4B F1AA  
DA4C F1AB  
DA4D F1AC  
DA4E F1AD  
DA4F F1AE  
DA50 F1AF  
DA51 F1B0  
DA52 F1B1  
DA53 F1B2  
DA54 F1B3  
DA55 F1B4  
DA56 F1B5  
DA57 F1B6  
DA58 F1B7  
DA59 F1B8  
DA5A F1B9  
DA5B F1BA  
DA5C F1BB  
DA5D F1BC  
DA5E F1BD  
DA5F F1BE  
DA60 F1BF  
DA61 F1C0  
DA62 F1C1  
DA63 F1C2  
DA64 F1C3  
DA65 F1C4  
DA66 F1C5  
DA67 F1C6  
DA68 F1C7  
DA69 F1C8  
DA6A F1C9  
DA6B F1CA  
DA6C F1CB  
DA6D F1CC  
DA6E F1CD  
DA6F F1CE  
DA70 F1CF  
DA71 F1D0  
DA72 F1D1  
DA73 F1D2  
DA74 F1D3  
DA75 F1D4  
DA76 F1D5  
DA77 F1D6  
DA78 F1D7  
DA79 F1D8  
DA7A F1D9  
DA7B F1DA  
DA7C F1DB  
DA7D F1DC  
DA7E F1DD  
DA7F F1DE  
DA81 F1DF



DA82 F1E0  
DA83 F1E1  
DA84 F1E2  
DA85 F1E3  
DA86 F1E4  
DA87 F1E5  
DA88 F1E6  
DA89 F1E7  
DA8A F1E8  
DA8B F1E9  
DA8C F1EA  
DA8D F1EB  
DA8E F1EC  
DA8F F1ED  
DA90 F1EE  
DA91 F1EF  
DA92 F1F0  
DA93 F1F1  
DA94 F1F2  
DA95 F1F3  
DA96 F1F4  
DA97 F1F5  
DA98 F1F6  
DA99 F1F7  
DA9A F1F8  
DA9B F1F9  
DA9C F1FA  
DA9D F1FB  
DA9E F1FC  
DA9F F1FD  
DAA0 F1FE  
DAA1 F1FF  
DAA2 F200  
DAA3 F201  
DAA4 F202  
DAA5 F203  
DAA6 F204  
DAA7 F205  
DAA8 F206  
DAA9 F207  
DAAA F208  
DAAB F209  
DAAC F20A  
DAAD F20B  
DAAE F20C  
DAAF F20D  
DAB0 F20E  
DAB1 F20F  
DAB2 F210  
DAB3 F211  
DAB4 F212  
DAB5 F213  
DAB6 F214  
DAB7 F215  
DAB8 F216  
DAB9 F217  
DABA F218  
DABB F219  
DABC F21A  
DABD F21B  
DABE F21C  
DABF F21D  
DAC0 F21E  
DAC1 F21F  
DAC2 F220  
DAC3 F221  
DAC4 F222  
DAC5 F223  
DAC6 F224  
DAC7 F225  
DAC8 F226  
DAC9 F227  
DACA F228  
DACB F229  
DACC F22A  
DACD F22B  
DACE F22C  
DACF F22D  
DAD0 F22E  
DAD1 F22F  
DAD2 F230  
DAD3 F231

DAD4	F232
DAD5	F233
DAD6	F234
DAD7	F235
DAD8	F236
DAD9	F237
DADA	F238
DADB	F239
DADC	F23A
DADD	F23B
DADE	F23C
DADF	F23D
DAE0	F23E
DAE1	F23F
DAE2	F240
DAE3	F241
DAE4	F242
DAE5	F243
DAE6	F244
DAE7	F245
DAE8	F246
DAE9	F247
DAEA	F248
DAEB	F249
DAEC	F24A
DAED	F24B
DAEE	F24C
DAEF	F24D
DAF0	F24E
DAF1	F24F
DAF2	F250
DAF3	F251
DAF4	F252
DAF5	F253
DAF6	F254
DAF7	F255
DAF8	F256
DAF9	F257
DAFA	F258
DAFB	F259
DAFC	F25A
DAFD	F25B
DB41	F25C
DB42	F25D
DB43	F25E
DB44	F25F
DB45	F260
DB46	F261
DB47	F262
DB48	F263
DB49	F264
DB4A	F265
DB4B	F266
DB4C	F267
DB4D	F268
DB4E	F269
DB4F	F26A
DB50	F26B
DB51	F26C
DB52	F26D
DB53	F26E
DB54	F26F
DB55	F270
DB56	F271
DB57	F272
DB58	F273
DB59	F274
DB5A	F275
DB5B	F276
DB5C	F277
DB5D	F278
DB5E	F279
DB5F	F27A
DB60	F27B
DB61	F27C
DB62	F27D
DB63	F27E
DB64	F27F
DB65	F280
DB66	F281
DB67	F282
DB68	F283

DB69 F284  
DB6A F285  
DB6B F286  
DB6C F287  
DB6D F288  
DB6E F289  
DB6F F28A  
DB70 F28B  
DB71 F28C  
DB72 F28D  
DB73 F28E  
DB74 F28F  
DB75 F290  
DB76 F291  
DB77 F292  
DB78 F293  
DB79 F294  
DB7A F295  
DB7B F296  
DB7C F297  
DB7D F298  
DB7E F299  
DB7F F29A  
DB81 F29B  
DB82 F29C  
DB83 F29D  
DB84 F29E  
DB85 F29F  
DB86 F2A0  
DB87 F2A1  
DB88 F2A2  
DB89 F2A3  
DB8A F2A4  
DB8B F2A5  
DB8C F2A6  
DB8D F2A7  
DB8E F2A8  
DB8F F2A9  
DB90 F2AA  
DB91 F2AB  
DB92 F2AC  
DB93 F2AD  
DB94 F2AE  
DB95 F2AF  
DB96 F2B0  
DB97 F2B1  
DB98 F2B2  
DB99 F2B3  
DB9A F2B4  
DB9B F2B5  
DB9C F2B6  
DB9D F2B7  
DB9E F2B8  
DB9F F2B9  
DBA0 F2BA  
DBA1 F2BB  
DBA2 F2BC  
DBA3 F2BD  
DBA4 F2BE  
DBA5 F2BF  
DBA6 F2C0  
DBA7 F2C1  
DBA8 F2C2  
DBA9 F2C3  
DBAA F2C4  
DBAB F2C5  
DBAC F2C6  
DBAD F2C7  
DBAE F2C8  
DBAF F2C9  
DBB0 F2CA  
DBB1 F2CB  
DBB2 F2CC  
DBB3 F2CD  
DBB4 F2CE  
DBB5 F2CF  
DBB6 F2D0  
DBB7 F2D1  
DBB8 F2D2  
DBB9 F2D3  
DBBA F2D4  
DBBB F2D5

DBBC	F2D6
DBBD	F2D7
DBBE	F2D8
DBBF	F2D9
DBC0	F2DA
DBC1	F2DB
DBC2	F2DC
DBC3	F2DD
DBC4	F2DE
DBC5	F2DF
DBC6	F2E0
DBC7	F2E1
DBC8	F2E2
DBC9	F2E3
DBCA	F2E4
DBCB	F2E5
DBCC	F2E6
DBCD	F2E7
DBCE	F2E8
DBCF	F2E9
DBD0	F2EA
DBD1	F2EB
DBD2	F2EC
DBD3	F2ED
DBD4	F2EE
DBD5	F2EF
DBD6	F2F0
DBD7	F2F1
DBD8	F2F2
DBD9	F2F3
DBDA	F2F4
DBDB	F2F5
DBDC	F2F6
DBDD	F2F7
DBDE	F2F8
DBDF	F2F9
DBE0	F2FA
DBE1	F2FB
DBE2	F2FC
DBE3	F2FD
DBE4	F2FE
DBE5	F2FF
DBE6	F300
DBE7	F301
DBE8	F302
DBE9	F303
DBEA	F304
DBEB	F305
DBEC	F306
DBED	F307
DBEE	F308
DBEF	F309
DBF0	F30A
DBF1	F30B
DBF2	F30C
DBF3	F30D
DBF4	F30E
DBF5	F30F
DBF6	F310
DBF7	F311
DBF8	F312
DBF9	F313
DBFA	F314
DBFB	F315
DBFC	F316
DBFD	F317
DC41	F318
DC42	F319
DC43	F31A
DC44	F31B
DC45	F31C
DC46	F31D
DC47	F31E
DC48	F31F
DC49	F320
DC4A	F321
DC4B	F322
DC4C	F323
DC4D	F324
DC4E	F325
DC4F	F326
DC50	F327

DC51 F328  
DC52 F329  
DC53 F32A  
DC54 F32B  
DC55 F32C  
DC56 F32D  
DC57 F32E  
DC58 F32F  
DC59 F330  
DC5A F331  
DC5B F332  
DC5C F333  
DC5D F334  
DC5E F335  
DC5F F336  
DC60 F337  
DC61 F338  
DC62 F339  
DC63 F33A  
DC64 F33B  
DC65 F33C  
DC66 F33D  
DC67 F33E  
DC68 F33F  
DC69 F340  
DC6A F341  
DC6B F342  
DC6C F343  
DC6D F344  
DC6E F345  
DC6F F346  
DC70 F347  
DC71 F348  
DC72 F349  
DC73 F34A  
DC74 F34B  
DC75 F34C  
DC76 F34D  
DC77 F34E  
DC78 F34F  
DC79 F350  
DC7A F351  
DC7B F352  
DC7C F353  
DC7D F354  
DC7E F355  
DC7F F356  
DC81 F357  
DC82 F358  
DC83 F359  
DC84 F35A  
DC85 F35B  
DC86 F35C  
DC87 F35D  
DC88 F35E  
DC89 F35F  
DC8A F360  
DC8B F361  
DC8C F362  
DC8D F363  
DC8E F364  
DC8F F365  
DC90 F366  
DC91 F367  
DC92 F368  
DC93 F369  
DC94 F36A  
DC95 F36B  
DC96 F36C  
DC97 F36D  
DC98 F36E  
DC99 F36F  
DC9A F370  
DC9B F371  
DC9C F372  
DC9D F373  
DC9E F374  
DC9F F375  
DCA0 F376  
DCA1 F377  
DCA2 F378  
DCA3 F379

DCA4 F37A  
DCA5 F37B  
DCA6 F37C  
DCA7 F37D  
DCA8 F37E  
DCA9 F37F  
DCAA F380  
DCAB F381  
DCAC F382  
DCAD F383  
DCAE F384  
DCAF F385  
DCB0 F386  
DCB1 F387  
DCB2 F388  
DCB3 F389  
DCB4 F38A  
DCB5 F38B  
DCB6 F38C  
DCB7 F38D  
DCB8 F38E  
DCB9 F38F  
DCBA F390  
DCBB F391  
DCBC F392  
DCBD F393  
DCBE F394  
DCBF F395  
DCC0 F396  
DCC1 F397  
DCC2 F398  
DCC3 F399  
DCC4 F39A  
DCC5 F39B  
DCC6 F39C  
DCC7 F39D  
DCC8 F39E  
DCC9 F39F  
DCCA F3A0  
DCCB F3A1  
DCCC F3A2  
DCCD F3A3  
DCCE F3A4  
DCCF F3A5  
DCD0 F3A6  
DCD1 F3A7  
DCD2 F3A8  
DCD3 F3A9  
DCD4 F3AA  
DCD5 F3AB  
DCD6 F3AC  
DCD7 F3AD  
DCD8 F3AE  
DCD9 F3AF  
DCDA F3B0  
DCDB F3B1  
DCDC F3B2  
DCDD F3B3  
DCDE F3B4  
DCDF F3B5  
DCE0 F3B6  
DCE1 F3B7  
DCE2 F3B8  
DCE3 F3B9  
DCE4 F3BA  
DCE5 F3BB  
DCE6 F3BC  
DCE7 F3BD  
DCE8 F3BE  
DCE9 F3BF  
DCEA F3C0  
DCEB F3C1  
DCEC F3C2  
DCED F3C3  
DCEE F3C4  
DCEF F3C5  
DCF0 F3C6  
DCF1 F3C7  
DCF2 F3C8  
DCF3 F3C9  
DCF4 F3CA  
DCF5 F3CB

DCF6 F3CC  
DCF7 F3CD  
DCF8 F3CE  
DCF9 F3CF  
DCFA F3D0  
DCFB F3D1  
DCFC F3D2  
DCFD F3D3  
DD41 F3D4  
DD42 F3D5  
DD43 F3D6  
DD44 F3D7  
DD45 F3D8  
DD46 F3D9  
DD47 F3DA  
DD48 F3DB  
DD49 F3DC  
DD4A F3DD  
DD4B F3DE  
DD4C F3DF  
DD4D F3E0  
DD4E F3E1  
DD4F F3E2  
DD50 F3E3  
DD51 F3E4  
DD52 F3E5  
DD53 F3E6  
DD54 F3E7  
DD55 F3E8  
DD56 F3E9  
DD57 F3EA  
DD58 F3EB  
DD59 F3EC  
DD5A F3ED  
DD5B F3EE  
DD5C F3EF  
DD5D F3F0  
DD5E F3F1  
DD5F F3F2  
DD60 F3F3  
DD61 F3F4  
DD62 F3F5  
DD63 F3F6  
DD64 F3F7  
DD65 F3F8  
DD66 F3F9  
DD67 F3FA  
DD68 F3FB  
DD69 F3FC  
DD6A F3FD  
DD6B F3FE  
DD6C F3FF  
DD6D F400  
DD6E F401  
DD6F F402  
DD70 F403  
DD71 F404  
DD72 F405  
DD73 F406  
DD74 F407  
DD75 F408  
DD76 F409  
DD77 F40A  
DD78 F40B  
DD79 F40C  
DD7A F40D  
DD7B F40E  
DD7C F40F  
DD7D F410  
DD7E F411  
DD7F F412  
DD81 F413  
DD82 F414  
DD83 F415  
DD84 F416  
DD85 F417  
DD86 F418  
DD87 F419  
DD88 F41A  
DD89 F41B  
DD8A F41C  
DD8B F41D

DD8C F41E  
DD8D F41F  
DD8E F420  
DD8F F421  
DD90 F422  
DD91 F423  
DD92 F424  
DD93 F425  
DD94 F426  
DD95 F427  
DD96 F428  
DD97 F429  
DD98 F42A  
DD99 F42B  
DD9A F42C  
DD9B F42D  
DD9C F42E  
DD9D F42F  
DD9E F430  
DD9F F431  
DDA0 F432  
DDA1 F433  
DDA2 F434  
DDA3 F435  
DDA4 F436  
DDA5 F437  
DDA6 F438  
DDA7 F439  
DDA8 F43A  
DDA9 F43B  
DDAA F43C  
DDAB F43D  
DDAC F43E  
DDAD F43F  
DDAE F440  
DDAF F441  
DDB0 F442  
DDB1 F443  
DDB2 F444  
DDB3 F445  
DDB4 F446  
DDB5 F447  
DDB6 F448  
DDB7 F449  
DDB8 F44A  
DDB9 F44B  
DDBA F44C  
DDBB F44D  
DDBC F44E  
DDBD F44F  
DDBE F450  
DDBF F451  
DDC0 F452  
DDC1 F453  
DDC2 F454  
DDC3 F455  
DDC4 F456  
DDC5 F457  
DDC6 F458  
DDC7 F459  
DDC8 F45A  
DDC9 F45B  
DDCA F45C  
DDCB F45D  
DDCC F45E  
DDCD F45F  
DDCE F460  
DDCF F461  
DDD0 F462  
DDD1 F463  
DDD2 F464  
DDD3 F465  
DDD4 F466  
DDD5 F467  
DDD6 F468  
DDD7 F469  
DDD8 F46A  
DDD9 F46B  
DDDA F46C  
DDDB F46D  
DDDC F46E  
DDDD F46F



DDDE F470  
DDDF F471  
DDE0 F472  
DDE1 F473  
DDE2 F474  
DDE3 F475  
DDE4 F476  
DDE5 F477  
DDE6 F478  
DDE7 F479  
DDE8 F47A  
DDE9 F47B  
DDEA F47C  
DDEB F47D  
DDEC F47E  
DDED F47F  
DDEE F480  
DDEF F481  
DDF0 F482  
DDF1 F483  
DDF2 F484  
DDF3 F485  
DDF4 F486  
DDF5 F487  
DDF6 F488  
DDF7 F489  
DDF8 F48A  
DDF9 F48B  
DDFA F48C  
DDFB F48D  
DDFC F48E  
DDFD F48F  
DE41 F490  
DE42 F491  
DE43 F492  
DE44 F493  
DE45 F494  
DE46 F495  
DE47 F496  
DE48 F497  
DE49 F498  
DE4A F499  
DE4B F49A  
DE4C F49B  
DE4D F49C  
DE4E F49D  
DE4F F49E  
DE50 F49F  
DE51 F4A0  
DE52 F4A1  
DE53 F4A2  
DE54 F4A3  
DE55 F4A4  
DE56 F4A5  
DE57 F4A6  
DE58 F4A7  
DE59 F4A8  
DE5A F4A9  
DE5B F4AA  
DE5C F4AB  
DE5D F4AC  
DE5E F4AD  
DE5F F4AE  
DE60 F4AF  
DE61 F4B0  
DE62 F4B1  
DE63 F4B2  
DE64 F4B3  
DE65 F4B4  
DE66 F4B5  
DE67 F4B6  
DE68 F4B7  
DE69 F4B8  
DE6A F4B9  
DE6B F4BA  
DE6C F4BB  
DE6D F4BC  
DE6E F4BD  
DE6F F4BE  
DE70 F4BF  
DE71 F4C0  
DE72 F4C1

DE73	F4C2
DE74	F4C3
DE75	F4C4
DE76	F4C5
DE77	F4C6
DE78	F4C7
DE79	F4C8
DE7A	F4C9
DE7B	F4CA
DE7C	F4CB
DE7D	F4CC
DE7E	F4CD
DE7F	F4CE
DE81	F4CF
DE82	F4D0
DE83	F4D1
DE84	F4D2
DE85	F4D3
DE86	F4D4
DE87	F4D5
DE88	F4D6
DE89	F4D7
DE8A	F4D8
DE8B	F4D9
DE8C	F4DA
DE8D	F4DB
DE8E	F4DC
DE8F	F4DD
DE90	F4DE
DE91	F4DF
DE92	F4E0
DE93	F4E1
DE94	F4E2
DE95	F4E3
DE96	F4E4
DE97	F4E5
DE98	F4E6
DE99	F4E7
DE9A	F4E8
DE9B	F4E9
DE9C	F4EA
DE9D	F4EB
DE9E	F4EC
DE9F	F4ED
DEA0	F4EE
DEA1	F4EF
DEA2	F4F0
DEA3	F4F1
DEA4	F4F2
DEA5	F4F3
DEA6	F4F4
DEA7	F4F5
DEA8	F4F6
DEA9	F4F7
DEAA	F4F8
DEAB	F4F9
DEAC	F4FA
DEAD	F4FB
DEAE	F4FC
DEAF	F4FD
DEB0	F4FE
DEB1	F4FF
DEB2	F500
DEB3	F501
DEB4	F502
DEB5	F503
DEB6	F504
DEB7	F505
DEB8	F506
DEB9	F507
DEBA	F508
DEBB	F509
DEBC	F50A
DEBD	F50B
DEBE	F50C
DEBF	F50D
DECA	F518
DECB	F519
DECC	F51A
DECD	F51B
DECE	F51C
DECF	F51D

DED0 F51E  
DED1 F51F  
DED2 F520  
DED3 F521  
DED4 F522  
DED5 F523  
DED6 F524  
DED7 F525  
DED8 F526  
DED9 F527  
DEDA F528  
DEDB F529  
DEDC F52A  
DEDD F52B  
DEDE F52C  
DEDF F52D  
DEE0 F52E  
DEE1 F52F  
DEE2 F530  
DEE3 F531  
DEE4 F532  
DEE5 F533  
DEE6 F534  
DEE7 F535  
DEE8 F536  
DEE9 F537  
DEEA F538  
DEEB F539  
DEEC F53A  
DEED F53B  
DEEE F53C  
DEEF F53D  
DEF0 F53E  
DEF1 F53F  
DEF2 F540  
DEF3 F541  
DEF4 F542  
DEF5 F543  
DEF6 F544  
DEF7 F545  
DEF8 F546  
DEF9 F547  
DEFA F548  
DEFB F549  
DEFC F54A  
DEFD F54B  
DF41 F54C  
DF42 F54D  
DF43 F54E  
DF44 F54F  
DF45 F550  
DF46 F551  
DF47 F552  
DF48 F553  
DF49 F554  
DF4A F555  
DF4B F556  
DF4C F557  
DF4D F558  
DF4E F559  
DF4F F55A  
DF50 F55B  
DF51 F55C  
DF52 F55D  
DF53 F55E  
DF54 F55F  
DF55 F560  
DF56 F561  
DF57 F562  
DF58 F563  
DF59 F564  
DF5A F565  
DF5B F566  
DF5C F567  
DF5D F568  
DF5E F569  
DF5F F56A  
DF60 F56B  
DF61 F56C  
DF62 F56D  
DF63 F56E  
DF64 F56F

DF65	F570
DF66	F571
DF67	F572
DF68	F573
DF69	F574
DF6A	F575
DF6B	F576
DF6C	F577
DF6D	F578
DF6E	F579
DF6F	F57A
DF70	F57B
DF71	F57C
DF72	F57D
DF73	F57E
DF74	F57F
DF75	F580
DF76	F581
DF77	F582
DF78	F583
DF79	F584
DF7A	F585
DF7B	F586
DF7C	F587
DF7D	F588
DF7E	F589
DF7F	F58A
DF81	F58B
DF82	F58C
DF83	F58D
DF84	F58E
DF85	F58F
DF86	F590
DF87	F591
DF88	F592
DF89	F593
DF8A	F594
DF8B	F595
DF8C	F596
DF8D	F597
DF8E	F598
DF8F	F599
DF90	F59A
DF91	F59B
DF92	F59C
DF93	F59D
DF94	F59E
DF95	F59F
DF96	F5A0
DF97	F5A1
DF98	F5A2
DF99	F5A3
DF9A	F5A4
DF9B	F5A5
DF9C	F5A6
DF9D	F5A7
DF9E	F5A8
DF9F	F5A9
DFA0	F5AA
DFA1	F5AB
DFA2	F5AC
DFA3	F5AD
DFA4	F5AE
DFA5	F5AF
DFA6	F5B0
DFA7	F5B1
DFA8	F5B2
DFA9	F5B3
DFAA	F5B4
DFAB	F5B5
DFAC	F5B6
DFAD	F5B7
DFAE	F5B8
DFAF	F5B9
DFB0	F5BA
DFB1	F5BB
DFB2	F5BC
DFB3	F5BD
DFB4	F5BE
DFB5	F5BF
DFB6	F5C0
DFB7	F5C1

DFB8	F5C2
DFB9	F5C3
DFBA	F5C4
DFBB	F5C5
DFBC	F5C6
DFBD	F5C7
DFBE	F5C8
DFBF	F5C9
DFC0	F5CA
DFC1	F5CB
DFC2	F5CC
DFC3	F5CD
DFC4	F5CE
DFC5	F5CF
DFC6	F5D0
DFC7	F5D1
DFC8	F5D2
DFC9	F5D3
DFCA	F5D4
DFCB	F5D5
DFCC	F5D6
DFCD	F5D7
DFCE	F5D8
DFCF	F5D9
DFD0	F5DA
DFD1	F5DB
DFD2	F5DC
DFD3	F5DD
DFD4	F5DE
DFD5	F5DF
DFD6	F5E0
DFD7	F5E1
DFD8	F5E2
DFD9	F5E3
DFDA	F5E4
DFDB	F5E5
DFDC	F5E6
DFDD	F5E7
DFDE	F5E8
DFDF	F5E9
DFE0	F5EA
DFE1	F5EB
DFE2	F5EC
DFE3	F5ED
DFE4	F5EE
DFE5	F5EF
DFE6	F5F0
DFE7	F5F1
DFE8	F5F2
DFE9	F5F3
DFEA	F5F4
DFEB	F5F5
DFEC	F5F6
DFED	F5F7
DFEE	F5F8
DFEF	F5F9
DFF0	F5FA
DFF1	F5FB
DFF2	F5FC
DFF3	F5FD
DFF4	F5FE
DFF5	F5FF
DFF6	F600
DFF7	F601
DFF8	F602
DFF9	F603
DFFA	F604
DFFB	F605
DFFC	F606
DFFD	F607
E041	F608
E042	F609
E043	F60A
E044	F60B
E045	F60C
E046	F60D
E047	F60E
E048	F60F
E049	F610
E04A	F611
E04B	F612
E04C	F613

E04D	F614
E04E	F615
E04F	F616
E050	F617
E051	F618
E052	F619
E053	F61A
E054	F61B
E055	F61C
E056	F61D
E057	F61E
E058	F61F
E059	F620
E05A	F621
E05B	F622
E05C	F623
E05D	F624
E05E	F625
E05F	F626
E060	F627
E061	F628
E062	F629
E063	F62A
E064	F62B
E065	F62C
E066	F62D
E067	F62E
E068	F62F
E069	F630
E06A	F631
E06B	F632
E06C	F633
E06D	F634
E06E	F635
E06F	F636
E070	F637
E071	F638
E072	F639
E073	F63A
E074	F63B
E075	F63C
E076	F63D
E077	F63E
E078	F63F
E079	F640
E07A	F641
E07B	F642
E07C	F643
E07D	F644
E07E	F645
E07F	F646
E081	F647
E082	F648
E083	F649
E084	F64A
E085	F64B
E086	F64C
E087	F64D
E088	F64E
E089	F64F
E08A	F650
E08B	F651
E08C	F652
E08D	F653
E08E	F654
E08F	F655
E090	F656
E091	F657
E092	F658
E093	F659
E094	F65A
E095	F65B
E096	F65C
E097	F65D
E098	F65E
E099	F65F
E09A	F660
E09B	F661
E09C	F662
E09D	F663
E09E	F664
E09F	F665

E0A0	F666
E0A1	F667
E0A2	F668
E0A3	F669
E0A4	F66A
E0A5	F66B
E0A6	F66C
E0A7	F66D
E0A8	F66E
E0A9	F66F
E0AA	F670
E0AB	F671
E0AC	F672
E0AD	F673
E0AE	F674
E0AF	F675
E0B0	F676
E0B1	F677
E0B2	F678
E0B3	F679
E0B4	F67A
E0B5	F67B
E0B6	F67C
E0B7	F67D
E0B8	F67E
E0B9	F67F
E0BA	F680
E0BB	F681
E0BC	F682
E0BD	F683
E0BE	F684
E0BF	F685
E0C0	F686
E0C1	F687
E0C2	F688
E0C3	F689
E0C4	F68A
E0C5	F68B
E0C6	F68C
E0C7	F68D
E0C8	F68E
E0C9	F68F
E0CA	F690
E0CB	F691
E0CC	F692
E0CD	F693
E0CE	F694
E0CF	F695
E0D0	F696
E0D1	F697
E0D2	F698
E0D3	F699
E0D4	F69A
E0D5	F69B
E0D6	F69C
E0D7	F69D
E0D8	F69E
E0D9	F69F
E0DA	F6A0
E0DB	F6A1
E0DC	F6A2
E0DD	F6A3
E0DE	F6A4
E0DF	F6A5
E0E0	F6A6
E0E1	F6A7
E0E2	F6A8
E0E3	F6A9
E0E4	F6AA
E0E5	F6AB
E0E6	F6AC
E0E7	F6AD
E0E8	F6AE
E0E9	F6AF
E0EA	F6B0
E0EB	F6B1
E0EC	F6B2
E0ED	F6B3
E0EE	F6B4
E0EF	F6B5
E0F0	F6B6
E0F1	F6B7

E0F2	F6B8
E0F3	F6B9
E0F4	F6BA
E0F5	F6BB
E0F6	F6BC
E0F7	F6BD
E0F8	F6BE
E0F9	F6BF
E0FA	F6C0
E0FB	F6C1
E0FC	F6C2
E0FD	F6C3
E141	F6C4
E142	F6C5
E143	F6C6
E144	F6C7
E145	F6C8
E146	F6C9
E147	F6CA
E148	F6CB
E149	F6CC
E14A	F6CD
E14B	F6CE
E14C	F6CF
E14D	F6D0
E14E	F6D1
E14F	F6D2
E150	F6D3
E151	F6D4
E152	F6D5
E153	F6D6
E154	F6D7
E155	F6D8
E156	F6D9
E157	F6DA
E158	F6DB
E159	F6DC
E15A	F6DD
E15B	F6DE
E15C	F6DF
E15D	F6E0
E15E	F6E1
E15F	F6E2
E160	F6E3
E161	F6E4
E162	F6E5
E163	F6E6
E164	F6E7
E165	F6E8
E166	F6E9
E167	F6EA
E168	F6EB
E169	F6EC
E16A	F6ED
E16B	F6EE
E16C	F6EF
E16D	F6F0
E16E	F6F1
E16F	F6F2
E170	F6F3
E171	F6F4
E172	F6F5
E173	F6F6
E174	F6F7
E175	F6F8
E176	F6F9
E177	F6FA
E178	F6FB
E179	F6FC
E17A	F6FD
E17B	F6FE
E17C	F6FF
E17D	F700
E17E	F701
E17F	F702
E181	F703
E182	F704
E183	F705
E184	F706
E185	F707
E186	F708
E187	F709



E188 F70A  
E189 F70B  
E18A F70C  
E18B F70D  
E18C F70E  
E18D F70F  
E18E F710  
E18F F711  
E190 F712  
E191 F713  
E192 F714  
E193 F715  
E194 F716  
E195 F717  
E196 F718  
E197 F719  
E198 F71A  
E199 F71B  
E19A F71C  
E19B F71D  
E19C F71E  
E19D F71F  
E19E F720  
E19F F721  
E1A0 F722  
E1A1 F723  
E1A2 F724  
E1A3 F725  
E1A4 F726  
E1A5 F727  
E1A6 F728  
E1A7 F729  
E1A8 F72A  
E1A9 F72B  
E1AA F72C  
E1AB F72D  
E1AC F72E  
E1AD F72F  
E1AE F730  
E1AF F731  
E1B0 F732  
E1B1 F733  
E1B2 F734  
E1B3 F735  
E1B4 F736  
E1B5 F737  
E1B6 F738  
E1B7 F739  
E1B8 F73A  
E1B9 F73B  
E1BA F73C  
E1BB F73D  
E1BC F73E  
E1BD F73F  
E1BE F740  
E1BF F741  
E1C0 F742  
E1C1 F743  
E1C2 F744  
E1C3 F745  
E1C4 F746  
E1C5 F747  
E1C6 F748  
E1C7 F749  
E1C8 F74A  
E1C9 F74B  
E1CA F74C  
E1CB F74D  
E1CC F74E  
E1CD F74F  
E1CE F750  
E1CF F751  
E1D0 F752  
E1D1 F753  
E1D2 F754  
E1D3 F755  
E1D4 F756  
E1D5 F757  
E1D6 F758  
E1D7 F759  
E1D8 F75A  
E1D9 F75B

E1DA	F75C
E1DB	F75D
E1DC	F75E
E1DD	F75F
E1DE	F760
E1DF	F761
E1E0	F762
E1E1	F763
E1E2	F764
E1E3	F765
E1E4	F766
E1E5	F767
E1E6	F768
E1E7	F769
E1E8	F76A
E1E9	F76B
E1EA	F76C
E1EB	F76D
E1EC	F76E
E1ED	F76F
E1EE	F770
E1EF	F771
E1F0	F772
E1F1	F773
E1F2	F774
E1F3	F775
E1F4	F776
E1F5	F777
E1F6	F778
E1F7	F779
E1F8	F77A
E1F9	F77B
E1FA	F77C
E1FB	F77D
E1FC	F77E
E1FD	F77F
E241	F780
E242	F781
E243	F782
E244	F783
E245	F784
E246	F785
E247	F786
E248	F787
E249	F788
E24A	F789
E24B	F78A
E24C	F78B
E24D	F78C
E24E	F78D
E24F	F78E
E250	F78F
E251	F790
E252	F791
E253	F792
E254	F793
E255	F794
E256	F795
E257	F796
E258	F797
E259	F798
E25A	F799
E25B	F79A
E25C	F79B
E25D	F79C
E25E	F79D
E25F	F79E
E260	F79F
E261	F7A0
E262	F7A1
E263	F7A2
E264	F7A3
E265	F7A4
E266	F7A5
E267	F7A6
E268	F7A7
E269	F7A8
E26A	F7A9
E26B	F7AA
E26C	F7AB
E26D	F7AC
E26E	F7AD

E26F F7AE  
E270 F7AF  
E271 F7B0  
E272 F7B1  
E273 F7B2  
E274 F7B3  
E275 F7B4  
E276 F7B5  
E277 F7B6  
E278 F7B7  
E279 F7B8  
E27A F7B9  
E27B F7BA  
E27C F7BB  
E27D F7BC  
E27E F7BD  
E27F F7BE  
E281 F7BF  
E282 F7C0  
E283 F7C1  
E284 F7C2  
E285 F7C3  
E286 F7C4  
E287 F7C5  
E288 F7C6  
E289 F7C7  
E28A F7C8  
E28B F7C9  
E28C F7CA  
E28D F7CB  
E28E F7CC  
E28F F7CD  
E290 F7CE  
E291 F7CF  
E292 F7D0  
E293 F7D1  
E294 F7D2  
E295 F7D3  
E296 F7D4  
E297 F7D5  
E298 F7D6  
E299 F7D7  
E29A F7D8  
E29B F7D9  
E29C F7DA  
E29D F7DB  
E29E F7DC  
E29F F7DD  
E2A0 F7DE  
E2A1 F7DF  
E2A2 F7E0  
E2A3 F7E1  
E2A4 F7E2  
E2A5 F7E3  
E2A6 F7E4  
E2A7 F7E5  
E2A8 F7E6  
E2A9 F7E7  
E2AA F7E8  
E2AB F7E9  
E2AC F7EA  
E2AD F7EB  
E2AE F7EC  
E2AF F7ED  
E2B0 F7EE  
E2B1 F7EF  
E2B2 F7F0  
E2B3 F7F1  
E2B4 F7F2  
E2B5 F7F3  
E2B6 F7F4  
E2B7 F7F5  
E2B8 F7F6  
E2B9 F7F7  
E2BA F7F8  
E2BB F7F9  
E2BC F7FA  
E2BD F7FB  
E2BE F7FC  
E2BF F7FD  
E2C0 F7FE  
E2C1 F7FF

E2C2	F800
E2C3	F801
E2C4	F802
E2C5	F803
E2C6	F804
E2C7	F805
E2C8	F806
E2C9	F807
E2CA	F808
E2CB	F809
E2CC	F80A
E2CD	F80B
E2CE	F80C
E2CF	F80D
E2D0	F80E
E2D1	F80F
E2D2	F810
E2D3	F811
E2D4	F812
E2D5	F813
E2D6	F814
E2D7	F815
E2D8	F816
E2D9	F817
E2DA	F818
E2DB	F819
E2DC	F81A
E2DD	F81B
E2DE	F81C
E2DF	F81D
E2E0	F81E
E2E1	F81F
E2E2	F820
E2E3	F821
E2E4	F822
E2E5	F823
E2E6	F824
E2E7	F825
E2E8	F826
E2E9	F827
E2EA	F828
E2EB	F829
E2EC	F82A
E2ED	F82B
E2EE	F82C
E2EF	F82D
E2F0	F82E
E2F1	F82F
E2F2	F830
E2F3	F831
E2F4	F832
E2F5	F833
E2F6	F834
E2F7	F835
E2F8	F836
E2F9	F837
E2FA	F838
E2FB	F839
E2FC	F83A
E2FD	F83B

## Default mapping for Unicode PUA to EBCDIC UDC on IBM i

The default mapping for Unicode PUA to EBCDIC UDC on IBM i.

E000	C241
E001	C242
E002	C243
E003	C244
E004	C245
E005	C246
E006	C247
E007	C248
E008	C249
E009	C24A
E00A	C24B
E00B	C24C
E00C	C24D

E00D C24E  
E00E C24F  
E00F C250  
E010 C251  
E011 C252  
E012 C253  
E013 C254  
E014 C255  
E015 C256  
E016 C257  
E017 C258  
E018 C259  
E019 C25A  
E01A C25B  
E01B C25C  
E01C C25D  
E01D C25E  
E01E C25F  
E01F C260  
E020 C261  
E021 C262  
E022 C263  
E023 C264  
E024 C265  
E025 C266  
E026 C267  
E027 C268  
E028 C269  
E029 C26A  
E02A C26B  
E02B C26C  
E02C C26D  
E02D C26E  
E02E C26F  
E02F C270  
E030 C271  
E031 C272  
E032 C273  
E033 C274  
E034 C275  
E035 C276  
E036 C277  
E037 C278  
E038 C279  
E039 C27A  
E03A C27B  
E03B C27C  
E03C C27D  
E03D C27E  
E03E C27F  
E03F C281  
E040 C282  
E041 C283  
E042 C284  
E043 C285  
E044 C286  
E045 C287  
E046 C288  
E047 C289  
E048 C28A  
E049 C28B  
E04A C28C  
E04B C28D  
E04C C28E  
E04D C28F  
E04E C290  
E04F C291  
E050 C292  
E051 C293  
E052 C294  
E053 C295  
E054 C296  
E055 C297  
E056 C298  
E057 C299  
E058 C29A  
E059 C29B  
E05A C29C  
E05B C29D  
E05C C29E  
E05D C29F  
E05E C2A0

E05F C2A1  
E060 C2A2  
E061 C2A3  
E062 C2A4  
E063 C2A5  
E064 C2A6  
E065 C2A7  
E066 C2A8  
E067 C2A9  
E068 C2AA  
E069 C2AB  
E06A C2AC  
E06B C2AD  
E06C C2AE  
E06D C2AF  
E06E C2B0  
E06F C2B1  
E070 C2B2  
E071 C2B3  
E072 C2B4  
E073 C2B5  
E074 C2B6  
E075 C2B7  
E076 C2B8  
E077 C2B9  
E078 C2BA  
E079 C2BB  
E07A C2BC  
E07B C2BD  
E07C C2BE  
E07D C2BF  
E07E C2C0  
E07F C2C1  
E080 C2C2  
E081 C2C3  
E082 C2C4  
E083 C2C5  
E084 C2C6  
E085 C2C7  
E086 C2C8  
E087 C2C9  
E088 C2CA  
E089 C2CB  
E08A C2CC  
E08B C2CD  
E08C C2CE  
E08D C2CF  
E08E C2D0  
E08F C2D1  
E090 C2D2  
E091 C2D3  
E092 C2D4  
E093 C2D5  
E094 C2D6  
E095 C2D7  
E096 C2D8  
E097 C2D9  
E098 C2DA  
E099 C2DB  
E09A C2DC  
E09B C2DD  
E09C C2DE  
E09D C2DF  
E09E C2E0  
E09F C2E1  
E0A0 C2E2  
E0A1 C2E3  
E0A2 C2E4  
E0A3 C2E5  
E0A4 C2E6  
E0A5 C2E7  
E0A6 C2E8  
E0A7 C2E9  
E0A8 C2EA  
E0A9 C2EB  
E0AA C2EC  
E0AB C2ED  
E0AC C2EE  
E0AD C2EF  
E0AE C2F0  
E0AF C2F1  
E0B0 C2F2

E0B1	C2F3
E0B2	C2F4
E0B3	C2F5
E0B4	C2F6
E0B5	C2F7
E0B6	C2F8
E0B7	C2F9
E0B8	C2FA
E0B9	C2FB
E0BA	C2FC
E0BB	C2FD
E0BC	C341
E0BD	C342
E0BE	C343
E0BF	C344
E0C0	C345
E0C1	C346
E0C2	C347
E0C3	C348
E0C4	C349
E0C5	C34A
E0C6	C34B
E0C7	C34C
E0C8	C34D
E0C9	C34E
E0CA	C34F
E0CB	C350
E0CC	C351
E0CD	C352
E0CE	C353
E0CF	C354
E0D0	C355
E0D1	C356
E0D2	C357
E0D3	C358
E0D4	C359
E0D5	C35A
E0D6	C35B
E0D7	C35C
E0D8	C35D
E0D9	C35E
E0DA	C35F
E0DB	C360
E0DC	C361
E0DD	C362
E0DE	C363
E0DF	C364
E0E0	C365
E0E1	C366
E0E2	C367
E0E3	C368
E0E4	C369
E0E5	C36A
E0E6	C36B
E0E7	C36C
E0E8	C36D
E0E9	C36E
E0EA	C36F
E0EB	C370
E0EC	C371
E0ED	C372
E0EE	C373
E0EF	C374
E0F0	C375
E0F1	C376
E0F2	C377
E0F3	C378
E0F4	C379
E0F5	C37A
E0F6	C37B
E0F7	C37C
E0F8	C37D
E0F9	C37E
E0FA	C37F
E0FB	C381
E0FC	C382
E0FD	C383
E0FE	C384
E0FF	C385
E100	C386
E101	C387
E102	C388

E103	C389
E104	C38A
E105	C38B
E106	C38C
E107	C38D
E108	C38E
E109	C38F
E10A	C390
E10B	C391
E10C	C392
E10D	C393
E10E	C394
E10F	C395
E110	C396
E111	C397
E112	C398
E113	C399
E114	C39A
E115	C39B
E116	C39C
E117	C39D
E118	C39E
E119	C39F
E11A	C3A0
E11B	C3A1
E11C	C3A2
E11D	C3A3
E11E	C3A4
E11F	C3A5
E120	C3A6
E121	C3A7
E122	C3A8
E123	C3A9
E124	C3AA
E125	C3AB
E126	C3AC
E127	C3AD
E128	C3AE
E129	C3AF
E12A	C3B0
E12B	C3B1
E12C	C3B2
E12D	C3B3
E12E	C3B4
E12F	C3B5
E130	C3B6
E131	C3B7
E132	C3B8
E133	C3B9
E134	C3BA
E135	C3BB
E136	C3BC
E137	C3BD
E138	C3BE
E139	C3BF
E13A	C3C0
E13B	C3C1
E13C	C3C2
E13D	C3C3
E13E	C3C4
E13F	C3C5
E140	C3C6
E141	C3C7
E142	C3C8
E143	C3C9
E144	C3CA
E145	C3CB
E146	C3CC
E147	C3CD
E148	C3CE
E149	C3CF
E14A	C3D0
E14B	C3D1
E14C	C3D2
E14D	C3D3
E14E	C3D4
E14F	C3D5
E150	C3D6
E151	C3D7
E152	C3D8
E153	C3D9
E154	C3DA



E155 C3DB  
E156 C3DC  
E157 C3DD  
E158 C3DE  
E159 C3DF  
E15A C3E0  
E15B C3E1  
E15C C3E2  
E15D C3E3  
E15E C3E4  
E15F C3E5  
E160 C3E6  
E161 C3E7  
E162 C3E8  
E163 C3E9  
E164 C3EA  
E165 C3EB  
E166 C3EC  
E167 C3ED  
E168 C3EE  
E169 C3EF  
E16A C3F0  
E16B C3F1  
E16C C3F2  
E16D C3F3  
E16E C3F4  
E16F C3F5  
E170 C3F6  
E171 C3F7  
E172 C3F8  
E173 C3F9  
E174 C3FA  
E175 C3FB  
E176 C3FC  
E177 C3FD  
E178 C441  
E179 C442  
E17A C443  
E17B C444  
E17C C445  
E17D C446  
E17E C447  
E17F C448  
E180 C449  
E181 C44A  
E182 C44B  
E183 C44C  
E184 C44D  
E185 C44E  
E186 C44F  
E187 C450  
E188 C451  
E189 C452  
E18A C453  
E18B C454  
E18C C455  
E18D C456  
E18E C457  
E18F C458  
E190 C459  
E191 C45A  
E192 C45B  
E193 C45C  
E194 C45D  
E195 C45E  
E196 C45F  
E197 C460  
E198 C461  
E199 C462  
E19A C463  
E19B C464  
E19C C465  
E19D C466  
E19E C467  
E19F C468  
E1A0 C469  
E1A1 C46A  
E1A2 C46B  
E1A3 C46C  
E1A4 C46D  
E1A5 C46E  
E1A6 C46F

E1A7 C470  
E1A8 C471  
E1A9 C472  
E1AA C473  
E1AB C474  
E1AC C475  
E1AD C476  
E1AE C477  
E1AF C478  
E1B0 C479  
E1B1 C47A  
E1B2 C47B  
E1B3 C47C  
E1B4 C47D  
E1B5 C47E  
E1B6 C47F  
E1B7 C481  
E1B8 C482  
E1B9 C483  
E1BA C484  
E1BB C485  
E1BC C486  
E1BD C487  
E1BE C488  
E1BF C489  
E1C0 C48A  
E1C1 C48B  
E1C2 C48C  
E1C3 C48D  
E1C4 C48E  
E1C5 C48F  
E1C6 C490  
E1C7 C491  
E1C8 C492  
E1C9 C493  
E1CA C494  
E1CB C495  
E1CC C496  
E1CD C497  
E1CE C498  
E1CF C499  
E1D0 C49A  
E1D1 C49B  
E1D2 C49C  
E1D3 C49D  
E1D4 C49E  
E1D5 C49F  
E1D6 C4A0  
E1D7 C4A1  
E1D8 C4A2  
E1D9 C4A3  
E1DA C4A4  
E1DB C4A5  
E1DC C4A6  
E1DD C4A7  
E1DE C4A8  
E1DF C4A9  
E1E0 C4AA  
E1E1 C4AB  
E1E2 C4AC  
E1E3 C4AD  
E1E4 C4AE  
E1E5 C4AF  
E1E6 C4B0  
E1E7 C4B1  
E1E8 C4B2  
E1E9 C4B3  
E1EA C4B4  
E1EB C4B5  
E1EC C4B6  
E1ED C4B7  
E1EE C4B8  
E1EF C4B9  
E1F0 C4BA  
E1F1 C4BB  
E1F2 C4BC  
E1F3 C4BD  
E1F4 C4BE  
E1F5 C4BF  
E1F6 C4C0  
E1F7 C4C1  
E1F8 C4C2

E1F9	C4C3
E1FA	C4C4
E1FB	C4C5
E1FC	C4C6
E1FD	C4C7
E1FE	C4C8
E1FF	C4C9
E200	C4CA
E201	C4CB
E202	C4CC
E203	C4CD
E204	C4CE
E205	C4CF
E206	C4D0
E207	C4D1
E208	C4D2
E209	C4D3
E20A	C4D4
E20B	C4D5
E20C	C4D6
E20D	C4D7
E20E	C4D8
E20F	C4D9
E210	C4DA
E211	C4DB
E212	C4DC
E213	C4DD
E214	C4DE
E215	C4DF
E216	C4E0
E217	C4E1
E218	C4E2
E219	C4E3
E21A	C4E4
E21B	C4E5
E21C	C4E6
E21D	C4E7
E21E	C4E8
E21F	C4E9
E220	C4EA
E221	C4EB
E222	C4EC
E223	C4ED
E224	C4EE
E225	C4EF
E226	C4F0
E227	C4F1
E228	C4F2
E229	C4F3
E22A	C4F4
E22B	C4F5
E22C	C4F6
E22D	C4F7
E22E	C4F8
E22F	C4F9
E230	C4FA
E231	C4FB
E232	C4FC
E233	C4FD
E234	C541
E235	C542
E236	C543
E237	C544
E238	C545
E239	C546
E23A	C547
E23B	C548
E23C	C549
E23D	C54A
E23E	C54B
E23F	C54C
E240	C54D
E241	C54E
E242	C54F
E243	C550
E244	C551
E245	C552
E246	C553
E247	C554
E248	C555
E249	C556
E24A	C557

E24B	C558
E24C	C559
E24D	C55A
E24E	C55B
E24F	C55C
E250	C55D
E251	C55E
E252	C55F
E253	C560
E254	C561
E255	C562
E256	C563
E257	C564
E258	C565
E259	C566
E25A	C567
E25B	C568
E25C	C569
E25D	C56A
E25E	C56B
E25F	C56C
E260	C56D
E261	C56E
E262	C56F
E263	C570
E264	C571
E265	C572
E266	C573
E267	C574
E268	C575
E269	C576
E26A	C577
E26B	C578
E26C	C579
E26D	C57A
E26E	C57B
E26F	C57C
E270	C57D
E271	C57E
E272	C57F
E273	C581
E274	C582
E275	C583
E276	C584
E277	C585
E278	C586
E279	C587
E27A	C588
E27B	C589
E27C	C58A
E27D	C58B
E27E	C58C
E27F	C58D
E280	C58E
E281	C58F
E282	C590
E283	C591
E284	C592
E285	C593
E286	C594
E287	C595
E288	C596
E289	C597
E28A	C598
E28B	C599
E28C	C59A
E28D	C59B
E28E	C59C
E28F	C59D
E290	C59E
E291	C59F
E292	C5A0
E293	C5A1
E294	C5A2
E295	C5A3
E296	C5A4
E297	C5A5
E298	C5A6
E299	C5A7
E29A	C5A8
E29B	C5A9
E29C	C5AA

E29D C5AB  
E29E C5AC  
E29F C5AD  
E2A0 C5AE  
E2A1 C5AF  
E2A2 C5B0  
E2A3 C5B1  
E2A4 C5B2  
E2A5 C5B3  
E2A6 C5B4  
E2A7 C5B5  
E2A8 C5B6  
E2A9 C5B7  
E2AA C5B8  
E2AB C5B9  
E2AC C5BA  
E2AD C5BB  
E2AE C5BC  
E2AF C5BD  
E2B0 C5BE  
E2B1 C5BF  
E2B2 C5C0  
E2B3 C5C1  
E2B4 C5C2  
E2B5 C5C3  
E2B6 C5C4  
E2B7 C5C5  
E2B8 C5C6  
E2B9 C5C7  
E2BA C5C8  
E2BB C5C9  
E2BC C5CA  
E2BD C5CB  
E2BE C5CC  
E2BF C5CD  
E2C0 C5CE  
E2C1 C5CF  
E2C2 C5D0  
E2C3 C5D1  
E2C4 C5D2  
E2C5 C5D3  
E2C6 C5D4  
E2C7 C5D5  
E2C8 C5D6  
E2C9 C5D7  
E2CA C5D8  
E2CB C5D9  
E2CC C5DA  
E2CD C5DB  
E2CE C5DC  
E2CF C5DD  
E2D0 C5DE  
E2D1 C5DF  
E2D2 C5E0  
E2D3 C5E1  
E2D4 C5E2  
E2D5 C5E3  
E2D6 C5E4  
E2D7 C5E5  
E2D8 C5E6  
E2D9 C5E7  
E2DA C5E8  
E2DB C5E9  
E2DC C5EA  
E2DD C5EB  
E2DE C5EC  
E2DF C5ED  
E2E0 C5EE  
E2E1 C5EF  
E2E2 C5F0  
E2E3 C5F1  
E2E4 C5F2  
E2E5 C5F3  
E2E6 C5F4  
E2E7 C5F5  
E2E8 C5F6  
E2E9 C5F7  
E2EA C5F8  
E2EB C5F9  
E2EC C5FA  
E2ED C5FB  
E2EE C5FC

E2EF	C5FD
E2F0	C641
E2F1	C642
E2F2	C643
E2F3	C644
E2F4	C645
E2F5	C646
E2F6	C647
E2F7	C648
E2F8	C649
E2F9	C64A
E2FA	C64B
E2FB	C64C
E2FC	C64D
E2FD	C64E
E2FE	C64F
E2FF	C650
E300	C651
E301	C652
E302	C653
E303	C654
E304	C655
E305	C656
E306	C657
E307	C658
E308	C659
E309	C65A
E30A	C65B
E30B	C65C
E30C	C65D
E30D	C65E
E30E	C65F
E30F	C660
E310	C661
E311	C662
E312	C663
E313	C664
E314	C665
E315	C666
E316	C667
E317	C668
E318	C669
E319	C66A
E31A	C66B
E31B	C66C
E31C	C66D
E31D	C66E
E31E	C66F
E31F	C670
E320	C671
E321	C672
E322	C673
E323	C674
E324	C675
E325	C676
E326	C677
E327	C678
E328	C679
E329	C67A
E32A	C67B
E32B	C67C
E32C	C67D
E32D	C67E
E32E	C67F
E32F	C681
E330	C682
E331	C683
E332	C684
E333	C685
E334	C686
E335	C687
E336	C688
E337	C689
E338	C68A
E339	C68B
E33A	C68C
E33B	C68D
E33C	C68E
E33D	C68F
E33E	C690
E33F	C691
E340	C692

E341	C693
E342	C694
E343	C695
E344	C696
E345	C697
E346	C698
E347	C699
E348	C69A
E349	C69B
E34A	C69C
E34B	C69D
E34C	C69E
E34D	C69F
E34E	C6A0
E34F	C6A1
E350	C6A2
E351	C6A3
E352	C6A4
E353	C6A5
E354	C6A6
E355	C6A7
E356	C6A8
E357	C6A9
E358	C6AA
E359	C6AB
E35A	C6AC
E35B	C6AD
E35C	C6AE
E35D	C6AF
E35E	C6B0
E35F	C6B1
E360	C6B2
E361	C6B3
E362	C6B4
E363	C6B5
E364	C6B6
E365	C6B7
E366	C6B8
E367	C6B9
E368	C6BA
E369	C6BB
E36A	C6BC
E36B	C6BD
E36C	C6BE
E36D	C6BF
E36E	C6C0
E36F	C6C1
E370	C6C2
E371	C6C3
E372	C6C4
E373	C6C5
E374	C6C6
E375	C6C7
E376	C6C8
E377	C6C9
E378	C6CA
E379	C6CB
E37A	C6CC
E37B	C6CD
E37C	C6CE
E37D	C6CF
E37E	C6D0
E37F	C6D1
E380	C6D2
E381	C6D3
E382	C6D4
E383	C6D5
E384	C6D6
E385	C6D7
E386	C6D8
E387	C6D9
E388	C6DA
E389	C6DB
E38A	C6DC
E38B	C6DD
E38C	C6DE
E38D	C6DF
E38E	C6E0
E38F	C6E1
E390	C6E2
E391	C6E3
E392	C6E4

E393	C6E5
E394	C6E6
E395	C6E7
E396	C6E8
E397	C6E9
E398	C6EA
E399	C6EB
E39A	C6EC
E39B	C6ED
E39C	C6EE
E39D	C6EF
E39E	C6F0
E39F	C6F1
E3A0	C6F2
E3A1	C6F3
E3A2	C6F4
E3A3	C6F5
E3A4	C6F6
E3A5	C6F7
E3A6	C6F8
E3A7	C6F9
E3A8	C6FA
E3A9	C6FB
E3AA	C6FC
E3AB	C6FD
E3AC	C741
E3AD	C742
E3AE	C743
E3AF	C744
E3B0	C745
E3B1	C746
E3B2	C747
E3B3	C748
E3B4	C749
E3B5	C74A
E3B6	C74B
E3B7	C74C
E3B8	C74D
E3B9	C74E
E3BA	C74F
E3BB	C750
E3BC	C751
E3BD	C752
E3BE	C753
E3BF	C754
E3C0	C755
E3C1	C756
E3C2	C757
E3C3	C758
E3C4	C759
E3C5	C75A
E3C6	C75B
E3C7	C75C
E3C8	C75D
E3C9	C75E
E3CA	C75F
E3CB	C760
E3CC	C761
E3CD	C762
E3CE	C763
E3CF	C764
E3D0	C765
E3D1	C766
E3D2	C767
E3D3	C768
E3D4	C769
E3D5	C76A
E3D6	C76B
E3D7	C76C
E3D8	C76D
E3D9	C76E
E3DA	C76F
E3DB	C770
E3DC	C771
E3DD	C772
E3DE	C773
E3DF	C774
E3E0	C775
E3E1	C776
E3E2	C777
E3E3	C778
E3E4	C779



E3E5 C77A  
E3E6 C77B  
E3E7 C77C  
E3E8 C77D  
E3E9 C77E  
E3EA C77F  
E3EB C781  
E3EC C782  
E3ED C783  
E3EE C784  
E3EF C785  
E3F0 C786  
E3F1 C787  
E3F2 C788  
E3F3 C789  
E3F4 C78A  
E3F5 C78B  
E3F6 C78C  
E3F7 C78D  
E3F8 C78E  
E3F9 C78F  
E3FA C790  
E3FB C791  
E3FC C792  
E3FD C793  
E3FE C794  
E3FF C795  
E400 C796  
E401 C797  
E402 C798  
E403 C799  
E404 C79A  
E405 C79B  
E406 C79C  
E407 C79D  
E408 C79E  
E409 C79F  
E40A C7A0  
E40B C7A1  
E40C C7A2  
E40D C7A3  
E40E C7A4  
E40F C7A5  
E410 C7A6  
E411 C7A7  
E412 C7A8  
E413 C7A9  
E414 C7AA  
E415 C7AB  
E416 C7AC  
E417 C7AD  
E418 C7AE  
E419 C7AF  
E41A C7B0  
E41B C7B1  
E41C C7B2  
E41D C7B3  
E41E C7B4  
E41F C7B5  
E420 C7B6  
E421 C7B7  
E422 C7B8  
E423 C7B9  
E424 C7BA  
E425 C7BB  
E426 C7BC  
E427 C7BD  
E428 C7BE  
E429 C7BF  
E42A C7C0  
E42B C7C1  
E42C C7C2  
E42D C7C3  
E42E C7C4  
E42F C7C5  
E430 C7C6  
E431 C7C7  
E432 C7C8  
E433 C7C9  
E434 C7CA  
E435 C7CB  
E436 C7CC

E437	C7CD
E438	C7CE
E439	C7CF
E43A	C7D0
E43B	C7D1
E43C	C7D2
E43D	C7D3
E43E	C7D4
E43F	C7D5
E440	C7D6
E441	C7D7
E442	C7D8
E443	C7D9
E444	C7DA
E445	C7DB
E446	C7DC
E447	C7DD
E448	C7DE
E449	C7DF
E44A	C7E0
E44B	C7E1
E44C	C7E2
E44D	C7E3
E44E	C7E4
E44F	C7E5
E450	C7E6
E451	C7E7
E452	C7E8
E453	C7E9
E454	C7EA
E455	C7EB
E456	C7EC
E457	C7ED
E458	C7EE
E459	C7EF
E45A	C7F0
E45B	C7F1
E45C	C7F2
E45D	C7F3
E45E	C7F4
E45F	C7F5
E460	C7F6
E461	C7F7
E462	C7F8
E463	C7F9
E464	C7FA
E465	C7FB
E466	C7FC
E467	C7FD
E468	C841
E469	C842
E46A	C843
E46B	C844
E46C	C845
E46D	C846
E46E	C847
E46F	C848
E470	C849
E471	C84A
E472	C84B
E473	C84C
E474	C84D
E475	C84E
E476	C84F
E477	C850
E478	C851
E479	C852
E47A	C853
E47B	C854
E47C	C855
E47D	C856
E47E	C857
E47F	C858
E480	C859
E481	C85A
E482	C85B
E483	C85C
E484	C85D
E485	C85E
E486	C85F
E487	C860
E488	C861

E489 C862  
E48A C863  
E48B C864  
E48C C865  
E48D C866  
E48E C867  
E48F C868  
E490 C869  
E491 C86A  
E492 C86B  
E493 C86C  
E494 C86D  
E495 C86E  
E496 C86F  
E497 C870  
E498 C871  
E499 C872  
E49A C873  
E49B C874  
E49C C875  
E49D C876  
E49E C877  
E49F C878  
E4A0 C879  
E4A1 C87A  
E4A2 C87B  
E4A3 C87C  
E4A4 C87D  
E4A5 C87E  
E4A6 C87F  
E4A7 C881  
E4A8 C882  
E4A9 C883  
E4AA C884  
E4AB C885  
E4AC C886  
E4AD C887  
E4AE C888  
E4AF C889  
E4B0 C88A  
E4B1 C88B  
E4B2 C88C  
E4B3 C88D  
E4B4 C88E  
E4B5 C88F  
E4B6 C890  
E4B7 C891  
E4B8 C892  
E4B9 C893  
E4BA C894  
E4BB C895  
E4BC C896  
E4BD C897  
E4BE C898  
E4BF C899  
E4C0 C89A  
E4C1 C89B  
E4C2 C89C  
E4C3 C89D  
E4C4 C89E  
E4C5 C89F  
E4C6 C8A0  
E4C7 C8A1  
E4C8 C8A2  
E4C9 C8A3  
E4CA C8A4  
E4CB C8A5  
E4CC C8A6  
E4CD C8A7  
E4CE C8A8  
E4CF C8A9  
E4D0 C8AA  
E4D1 C8AB  
E4D2 C8AC  
E4D3 C8AD  
E4D4 C8AE  
E4D5 C8AF  
E4D6 C8B0  
E4D7 C8B1  
E4D8 C8B2  
E4D9 C8B3  
E4DA C8B4

E4DB	C8B5
E4DC	C8B6
E4DD	C8B7
E4DE	C8B8
E4DF	C8B9
E4E0	C8BA
E4E1	C8BB
E4E2	C8BC
E4E3	C8BD
E4E4	C8BE
E4E5	C8BF
E4E6	C8C0
E4E7	C8C1
E4E8	C8C2
E4E9	C8C3
E4EA	C8C4
E4EB	C8C5
E4EC	C8C6
E4ED	C8C7
E4EE	C8C8
E4EF	C8C9
E4F0	C8CA
E4F1	C8CB
E4F2	C8CC
E4F3	C8CD
E4F4	C8CE
E4F5	C8CF
E4F6	C8D0
E4F7	C8D1
E4F8	C8D2
E4F9	C8D3
E4FA	C8D4
E4FB	C8D5
E4FC	C8D6
E4FD	C8D7
E4FE	C8D8
E4FF	C8D9
E500	C8DA
E501	C8DB
E502	C8DC
E503	C8DD
E504	C8DE
E505	C8DF
E506	C8E0
E507	C8E1
E508	C8E2
E509	C8E3
E50A	C8E4
E50B	C8E5
E50C	C8E6
E50D	C8E7
E50E	C8E8
E50F	C8E9
E510	C8EA
E511	C8EB
E512	C8EC
E513	C8ED
E514	C8EE
E515	C8EF
E516	C8F0
E517	C8F1
E518	C8F2
E519	C8F3
E51A	C8F4
E51B	C8F5
E51C	C8F6
E51D	C8F7
E51E	C8F8
E51F	C8F9
E520	C8FA
E521	C8FB
E522	C8FC
E523	C8FD
E524	C941
E525	C942
E526	C943
E527	C944
E528	C945
E529	C946
E52A	C947
E52B	C948
E52C	C949

E52D C94A  
E52E C94B  
E52F C94C  
E530 C94D  
E531 C94E  
E532 C94F  
E533 C950  
E534 C951  
E535 C952  
E536 C953  
E537 C954  
E538 C955  
E539 C956  
E53A C957  
E53B C958  
E53C C959  
E53D C95A  
E53E C95B  
E53F C95C  
E540 C95D  
E541 C95E  
E542 C95F  
E543 C960  
E544 C961  
E545 C962  
E546 C963  
E547 C964  
E548 C965  
E549 C966  
E54A C967  
E54B C968  
E54C C969  
E54D C96A  
E54E C96B  
E54F C96C  
E550 C96D  
E551 C96E  
E552 C96F  
E553 C970  
E554 C971  
E555 C972  
E556 C973  
E557 C974  
E558 C975  
E559 C976  
E55A C977  
E55B C978  
E55C C979  
E55D C97A  
E55E C97B  
E55F C97C  
E560 C97D  
E561 C97E  
E562 C97F  
E563 C981  
E564 C982  
E565 C983  
E566 C984  
E567 C985  
E568 C986  
E569 C987  
E56A C988  
E56B C989  
E56C C98A  
E56D C98B  
E56E C98C  
E56F C98D  
E570 C98E  
E571 C98F  
E572 C990  
E573 C991  
E574 C992  
E575 C993  
E576 C994  
E577 C995  
E578 C996  
E579 C997  
E57A C998  
E57B C999  
E57C C99A  
E57D C99B  
E57E C99C

E57F C99D  
E580 C99E  
E581 C99F  
E582 C9A0  
E583 C9A1  
E584 C9A2  
E585 C9A3  
E586 C9A4  
E587 C9A5  
E588 C9A6  
E589 C9A7  
E58A C9A8  
E58B C9A9  
E58C C9AA  
E58D C9AB  
E58E C9AC  
E58F C9AD  
E590 C9AE  
E591 C9AF  
E592 C9B0  
E593 C9B1  
E594 C9B2  
E595 C9B3  
E596 C9B4  
E597 C9B5  
E598 C9B6  
E599 C9B7  
E59A C9B8  
E59B C9B9  
E59C C9BA  
E59D C9BB  
E59E C9BC  
E59F C9BD  
E5A0 C9BE  
E5A1 C9BF  
E5A2 C9C0  
E5A3 C9C1  
E5A4 C9C2  
E5A5 C9C3  
E5A6 C9C4  
E5A7 C9C5  
E5A8 C9C6  
E5A9 C9C7  
E5AA C9C8  
E5AB C9C9  
E5AC C9CA  
E5AD C9CB  
E5AE C9CC  
E5AF C9CD  
E5B0 C9CE  
E5B1 C9CF  
E5B2 C9D0  
E5B3 C9D1  
E5B4 C9D2  
E5B5 C9D3  
E5B6 C9D4  
E5B7 C9D5  
E5B8 C9D6  
E5B9 C9D7  
E5BA C9D8  
E5BB C9D9  
E5BC C9DA  
E5BD C9DB  
E5BE C9DC  
E5BF C9DD  
E5C0 C9DE  
E5C1 C9DF  
E5C2 C9E0  
E5C3 C9E1  
E5C4 C9E2  
E5C5 C9E3  
E5C6 C9E4  
E5C7 C9E5  
E5C8 C9E6  
E5C9 C9E7  
E5CA C9E8  
E5CB C9E9  
E5CC C9EA  
E5CD C9EB  
E5CE C9EC  
E5CF C9ED  
E5D0 C9EE

E5D1 C9EF  
E5D2 C9F0  
E5D3 C9F1  
E5D4 C9F2  
E5D5 C9F3  
E5D6 C9F4  
E5D7 C9F5  
E5D8 C9F6  
E5D9 C9F7  
E5DA C9F8  
E5DB C9F9  
E5DC C9FA  
E5DD C9FB  
E5DE C9FC  
E5DF C9FD  
E5E0 CA41  
E5E1 CA42  
E5E2 CA43  
E5E3 CA44  
E5E4 CA45  
E5E5 CA46  
E5E6 CA47  
E5E7 CA48  
E5E8 CA49  
E5E9 CA4A  
E5EA CA4B  
E5EB CA4C  
E5EC CA4D  
E5ED CA4E  
E5EE CA4F  
E5EF CA50  
E5F0 CA51  
E5F1 CA52  
E5F2 CA53  
E5F3 CA54  
E5F4 CA55  
E5F5 CA56  
E5F6 CA57  
E5F7 CA58  
E5F8 CA59  
E5F9 CA5A  
E5FA CA5B  
E5FB CA5C  
E5FC CA5D  
E5FD CA5E  
E5FE CA5F  
E5FF CA60  
E600 CA61  
E601 CA62  
E602 CA63  
E603 CA64  
E604 CA65  
E605 CA66  
E606 CA67  
E607 CA68  
E608 CA69  
E609 CA6A  
E60A CA6B  
E60B CA6C  
E60C CA6D  
E60D CA6E  
E60E CA6F  
E60F CA70  
E610 CA71  
E611 CA72  
E612 CA73  
E613 CA74  
E614 CA75  
E615 CA76  
E616 CA77  
E617 CA78  
E618 CA79  
E619 CA7A  
E61A CA7B  
E61B CA7C  
E61C CA7D  
E61D CA7E  
E61E CA7F  
E61F CA81  
E620 CA82  
E621 CA83  
E622 CA84

E623 CA85  
E624 CA86  
E625 CA87  
E626 CA88  
E627 CA89  
E628 CA8A  
E629 CA8B  
E62A CA8C  
E62B CA8D  
E62C CA8E  
E62D CA8F  
E62E CA90  
E62F CA91  
E630 CA92  
E631 CA93  
E632 CA94  
E633 CA95  
E634 CA96  
E635 CA97  
E636 CA98  
E637 CA99  
E638 CA9A  
E639 CA9B  
E63A CA9C  
E63B CA9D  
E63C CA9E  
E63D CA9F  
E63E CAA0  
E63F CAA1  
E640 CAA2  
E641 CAA3  
E642 CAA4  
E643 CAA5  
E644 CAA6  
E645 CAA7  
E646 CAA8  
E647 CAA9  
E648 CAAA  
E649 CAAB  
E64A CAAC  
E64B CAAD  
E64C CAAE  
E64D CAAF  
E64E CAB0  
E64F CAB1  
E650 CAB2  
E651 CAB3  
E652 CAB4  
E653 CAB5  
E654 CAB6  
E655 CAB7  
E656 CAB8  
E657 CAB9  
E658 CABA  
E659 CABB  
E65A CABC  
E65B CABD  
E65C CABE  
E65D CABF  
E65E CAC0  
E65F CAC1  
E660 CAC2  
E661 CAC3  
E662 CAC4  
E663 CAC5  
E664 CAC6  
E665 CAC7  
E666 CAC8  
E667 CAC9  
E668 CACA  
E669 CACB  
E66A CACC  
E66B CACD  
E66C CACE  
E66D CACF  
E66E CAD0  
E66F CAD1  
E670 CAD2  
E671 CAD3  
E672 CAD4  
E673 CAD5  
E674 CAD6



E675 CAD7  
E676 CAD8  
E677 CAD9  
E678 CADA  
E679 CADB  
E67A CADC  
E67B CADD  
E67C CADE  
E67D CADF  
E67E CAE0  
E67F CAE1  
E680 CAE2  
E681 CAE3  
E682 CAE4  
E683 CAE5  
E684 CAE6  
E685 CAE7  
E686 CAE8  
E687 CAE9  
E688 CAEA  
E689 CAEB  
E68A CAEC  
E68B CAED  
E68C CAEE  
E68D CAEF  
E68E CAF0  
E68F CAF1  
E690 CAF2  
E691 CAF3  
E692 CAF4  
E693 CAF5  
E694 CAF6  
E695 CAF7  
E696 CAF8  
E697 CAF9  
E698 CAFA  
E699 CAFB  
E69A CAFC  
E69B CAFD  
E69C CB41  
E69D CB42  
E69E CB43  
E69F CB44  
E6A0 CB45  
E6A1 CB46  
E6A2 CB47  
E6A3 CB48  
E6A4 CB49  
E6A5 CB4A  
E6A6 CB4B  
E6A7 CB4C  
E6A8 CB4D  
E6A9 CB4E  
E6AA CB4F  
E6AB CB50  
E6AC CB51  
E6AD CB52  
E6AE CB53  
E6AF CB54  
E6B0 CB55  
E6B1 CB56  
E6B2 CB57  
E6B3 CB58  
E6B4 CB59  
E6B5 CB5A  
E6B6 CB5B  
E6B7 CB5C  
E6B8 CB5D  
E6B9 CB5E  
E6BA CB5F  
E6BB CB60  
E6BC CB61  
E6BD CB62  
E6BE CB63  
E6BF CB64  
E6C0 CB65  
E6C1 CB66  
E6C2 CB67  
E6C3 CB68  
E6C4 CB69  
E6C5 CB6A  
E6C6 CB6B

E6C7	CB6C
E6C8	CB6D
E6C9	CB6E
E6CA	CB6F
E6CB	CB70
E6CC	CB71
E6CD	CB72
E6CE	CB73
E6CF	CB74
E6D0	CB75
E6D1	CB76
E6D2	CB77
E6D3	CB78
E6D4	CB79
E6D5	CB7A
E6D6	CB7B
E6D7	CB7C
E6D8	CB7D
E6D9	CB7E
E6DA	CB7F
E6DB	CB81
E6DC	CB82
E6DD	CB83
E6DE	CB84
E6DF	CB85
E6E0	CB86
E6E1	CB87
E6E2	CB88
E6E3	CB89
E6E4	CB8A
E6E5	CB8B
E6E6	CB8C
E6E7	CB8D
E6E8	CB8E
E6E9	CB8F
E6EA	CB90
E6EB	CB91
E6EC	CB92
E6ED	CB93
E6EE	CB94
E6EF	CB95
E6F0	CB96
E6F1	CB97
E6F2	CB98
E6F3	CB99
E6F4	CB9A
E6F5	CB9B
E6F6	CB9C
E6F7	CB9D
E6F8	CB9E
E6F9	CB9F
E6FA	CBA0
E6FB	CBA1
E6FC	CBA2
E6FD	CBA3
E6FE	CBA4
E6FF	CBA5
E700	CBA6
E701	CBA7
E702	CBA8
E703	CBA9
E704	CBAA
E705	CBAB
E706	CBAC
E707	CBAD
E708	CBAE
E709	CBAF
E70A	CBB0
E70B	CBB1
E70C	CBB2
E70D	CBB3
E70E	CBB4
E70F	CBB5
E710	CBB6
E711	CBB7
E712	CBB8
E713	CBB9
E714	CBBA
E715	CBBB
E716	CBBC
E717	CBBD
E718	CBBE

E719 CBBF  
E71A CBC0  
E71B CBC1  
E71C CBC2  
E71D CBC3  
E71E CBC4  
E71F CBC5  
E720 CBC6  
E721 CBC7  
E722 CBC8  
E723 CBC9  
E724 CBCA  
E725 CBCB  
E726 CBCC  
E727 CBCD  
E728 CBCE  
E729 CBCF  
E72A CBD0  
E72B CBD1  
E72C CBD2  
E72D CBD3  
E72E CBD4  
E72F CBD5  
E730 CBD6  
E731 CBD7  
E732 CBD8  
E733 CBD9  
E734 CBDA  
E735 CBDB  
E736 CBDC  
E737 CBDD  
E738 CBDE  
E739 CBDF  
E73A CBE0  
E73B CBE1  
E73C CBE2  
E73D CBE3  
E73E CBE4  
E73F CBE5  
E740 CBE6  
E741 CBE7  
E742 CBE8  
E743 CBE9  
E744 CBEA  
E745 CBEB  
E746 CBEC  
E747 CBED  
E748 CBEE  
E749 CBEF  
E74A CBF0  
E74B CBF1  
E74C CBF2  
E74D CBF3  
E74E CBF4  
E74F CBF5  
E750 CBF6  
E751 CBF7  
E752 CBF8  
E753 CBF9  
E754 CBFA  
E755 CBFB  
E756 CBFC  
E757 CBFD  
E758 CC41  
E759 CC42  
E75A CC43  
E75B CC44  
E75C CC45  
E75D CC46  
E75E CC47  
E75F CC48  
E760 CC49  
E761 CC4A  
E762 CC4B  
E763 CC4C  
E764 CC4D  
E765 CC4E  
E766 CC4F  
E767 CC50  
E768 CC51  
E769 CC52  
E76A CC53

E76B CC54  
E76C CC55  
E76D CC56  
E76E CC57  
E76F CC58  
E770 CC59  
E771 CC5A  
E772 CC5B  
E773 CC5C  
E774 CC5D  
E775 CC5E  
E776 CC5F  
E777 CC60  
E778 CC61  
E779 CC62  
E77A CC63  
E77B CC64  
E77C CC65  
E77D CC66  
E77E CC67  
E77F CC68  
E780 CC69  
E781 CC6A  
E782 CC6B  
E783 CC6C  
E784 CC6D  
E785 CC6E  
E786 CC6F  
E787 CC70  
E788 CC71  
E789 CC72  
E78A CC73  
E78B CC74  
E78C CC75  
E78D CC76  
E78E CC77  
E78F CC78  
E790 CC79  
E791 CC7A  
E792 CC7B  
E793 CC7C  
E794 CC7D  
E795 CC7E  
E796 CC7F  
E797 CC81  
E798 CC82  
E799 CC83  
E79A CC84  
E79B CC85  
E79C CC86  
E79D CC87  
E79E CC88  
E79F CC89  
E7A0 CC8A  
E7A1 CC8B  
E7A2 CC8C  
E7A3 CC8D  
E7A4 CC8E  
E7A5 CC8F  
E7A6 CC90  
E7A7 CC91  
E7A8 CC92  
E7A9 CC93  
E7AA CC94  
E7AB CC95  
E7AC CC96  
E7AD CC97  
E7AE CC98  
E7AF CC99  
E7B0 CC9A  
E7B1 CC9B  
E7B2 CC9C  
E7B3 CC9D  
E7B4 CC9E  
E7B5 CC9F  
E7B6 CCA0  
E7B7 CCA1  
E7B8 CCA2  
E7B9 CCA3  
E7BA CCA4  
E7BB CCA5  
E7BC CCA6

E7BD CCA7  
E7BE CCA8  
E7BF CCA9  
E7C0 CCAA  
E7C1 CCAB  
E7C2 CCAC  
E7C3 CCAD  
E7C4 CCAE  
E7C5 CCAF  
E7C6 CCB0  
E7C7 CCB1  
E7C8 CCB2  
E7C9 CCB3  
E7CA CCB4  
E7CB CCB5  
E7CC CCB6  
E7CD CCB7  
E7CE CCB8  
E7CF CCB9  
E7D0 CCBA  
E7D1 CCBB  
E7D2 CCBC  
E7D3 CCBD  
E7D4 CCBE  
E7D5 CCBF  
E7D6 CCC0  
E7D7 CCC1  
E7D8 CCC2  
E7D9 CCC3  
E7DA CCC4  
E7DB CCC5  
E7DC CCC6  
E7DD CCC7  
E7DE CCC8  
E7DF CCC9  
E7E0 CCCA  
E7E1 CCCB  
E7E2 CCCC  
E7E3 CCCD  
E7E4 CCCE  
E7E5 CCCF  
E7E6 CCD0  
E7E7 CCD1  
E7E8 CCD2  
E7E9 CCD3  
E7EA CCD4  
E7EB CCD5  
E7EC CCD6  
E7ED CCD7  
E7EE CCD8  
E7EF CCD9  
E7F0 CCDA  
E7F1 CCDB  
E7F2 CCDC  
E7F3 CCDD  
E7F4 CCDE  
E7F5 CCDF  
E7F6 CCE0  
E7F7 CCE1  
E7F8 CCE2  
E7F9 CCE3  
E7FA CCE4  
E7FB CCE5  
E7FC CCE6  
E7FD CCE7  
E7FE CCE8  
E7FF CCE9  
E800 CCEA  
E801 CCEB  
E802 CCEC  
E803 CCED  
E804 CCEE  
E805 CCEF  
E806 CCF0  
E807 CCF1  
E808 CCF2  
E809 CCF3  
E80A CCF4  
E80B CCF5  
E80C CCF6  
E80D CCF7  
E80E CCF8

E80F	CCF9
E810	CCFA
E811	CCFB
E812	CCFC
E813	CCFD
E814	CD41
E815	CD42
E816	CD43
E817	CD44
E818	CD45
E819	CD46
E81A	CD47
E81B	CD48
E81C	CD49
E81D	CD4A
E81E	CD4B
E81F	CD4C
E820	CD4D
E821	CD4E
E822	CD4F
E823	CD50
E824	CD51
E825	CD52
E826	CD53
E827	CD54
E828	CD55
E829	CD56
E82A	CD57
E82B	CD58
E82C	CD59
E82D	CD5A
E82E	CD5B
E82F	CD5C
E830	CD5D
E831	CD5E
E832	CD5F
E833	CD60
E834	CD61
E835	CD62
E836	CD63
E837	CD64
E838	CD65
E839	CD66
E83A	CD67
E83B	CD68
E83C	CD69
E83D	CD6A
E83E	CD6B
E83F	CD6C
E840	CD6D
E841	CD6E
E842	CD6F
E843	CD70
E844	CD71
E845	CD72
E846	CD73
E847	CD74
E848	CD75
E849	CD76
E84A	CD77
E84B	CD78
E84C	CD79
E84D	CD7A
E84E	CD7B
E84F	CD7C
E850	CD7D
E851	CD7E
E852	CD7F
E853	CD81
E854	CD82
E855	CD83
E856	CD84
E857	CD85
E858	CD86
E859	CD87
E85A	CD88
E85B	CD89
E85C	CD8A
E85D	CD8B
E85E	CD8C
E85F	CD8D
E860	CD8E

E861 CD8F  
E862 CD90  
E863 CD91  
E864 CD92  
E865 CD93  
E866 CD94  
E867 CD95  
E868 CD96  
E869 CD97  
E86A CD98  
E86B CD99  
E86C CD9A  
E86D CD9B  
E86E CD9C  
E86F CD9D  
E870 CD9E  
E871 CD9F  
E872 CDA0  
E873 CDA1  
E874 CDA2  
E875 CDA3  
E876 CDA4  
E877 CDA5  
E878 CDA6  
E879 CDA7  
E87A CDA8  
E87B CDA9  
E87C CDAA  
E87D CDAB  
E87E CDAC  
E87F CDAD  
E880 CDAE  
E881 CDAF  
E882 CDB0  
E883 CDB1  
E884 CDB2  
E885 CDB3  
E886 CDB4  
E887 CDB5  
E888 CDB6  
E889 CDB7  
E88A CDB8  
E88B CDB9  
E88C CDBA  
E88D CDBB  
E88E CDBC  
E88F CDBD  
E890 CDBE  
E891 CDBF  
E892 CDC0  
E893 CDC1  
E894 CDC2  
E895 CDC3  
E896 CDC4  
E897 CDC5  
E898 CDC6  
E899 CDC7  
E89A CDC8  
E89B CDC9  
E89C CDCA  
E89D CDCB  
E89E CDCC  
E89F CDCD  
E8A0 CDCE  
E8A1 CDCF  
E8A2 CDD0  
E8A3 CDD1  
E8A4 CDD2  
E8A5 CDD3  
E8A6 CDD4  
E8A7 CDD5  
E8A8 CDD6  
E8A9 CDD7  
E8AA CDD8  
E8AB CDD9  
E8AC CDDA  
E8AD CDDB  
E8AE CDDC  
E8AF CDDD  
E8B0 CDDE  
E8B1 CDDF  
E8B2 CDE0

E8B3 CDE1  
E8B4 CDE2  
E8B5 CDE3  
E8B6 CDE4  
E8B7 CDE5  
E8B8 CDE6  
E8B9 CDE7  
E8BA CDE8  
E8BB CDE9  
E8BC CDEA  
E8BD CDEB  
E8BE CDEC  
E8BF CDED  
E8C0 CDEE  
E8C1 CDEF  
E8C2 CDF0  
E8C3 CDF1  
E8C4 CDF2  
E8C5 CDF3  
E8C6 CDF4  
E8C7 CDF5  
E8C8 CDF6  
E8C9 CDF7  
E8CA CDF8  
E8CB CDF9  
E8CC CDFA  
E8CD CDFB  
E8CE CDFC  
E8CF CDFD  
E8D0 CE41  
E8D1 CE42  
E8D2 CE43  
E8D3 CE44  
E8D4 CE45  
E8D5 CE46  
E8D6 CE47  
E8D7 CE48  
E8D8 CE49  
E8D9 CE4A  
E8DA CE4B  
E8DB CE4C  
E8DC CE4D  
E8DD CE4E  
E8DE CE4F  
E8DF CE50  
E8E0 CE51  
E8E1 CE52  
E8E2 CE53  
E8E3 CE54  
E8E4 CE55  
E8E5 CE56  
E8E6 CE57  
E8E7 CE58  
E8E8 CE59  
E8E9 CE5A  
E8EA CE5B  
E8EB CE5C  
E8EC CE5D  
E8ED CE5E  
E8EE CE5F  
E8EF CE60  
E8F0 CE61  
E8F1 CE62  
E8F2 CE63  
E8F3 CE64  
E8F4 CE65  
E8F5 CE66  
E8F6 CE67  
E8F7 CE68  
E8F8 CE69  
E8F9 CE6A  
E8FA CE6B  
E8FB CE6C  
E8FC CE6D  
E8FD CE6E  
E8FE CE6F  
E8FF CE70  
E900 CE71  
E901 CE72  
E902 CE73  
E903 CE74  
E904 CE75



E905 CE76  
E906 CE77  
E907 CE78  
E908 CE79  
E909 CE7A  
E90A CE7B  
E90B CE7C  
E90C CE7D  
E90D CE7E  
E90E CE7F  
E90F CE81  
E910 CE82  
E911 CE83  
E912 CE84  
E913 CE85  
E914 CE86  
E915 CE87  
E916 CE88  
E917 CE89  
E918 CE8A  
E919 CE8B  
E91A CE8C  
E91B CE8D  
E91C CE8E  
E91D CE8F  
E91E CE90  
E91F CE91  
E920 CE92  
E921 CE93  
E922 CE94  
E923 CE95  
E924 CE96  
E925 CE97  
E926 CE98  
E927 CE99  
E928 CE9A  
E929 CE9B  
E92A CE9C  
E92B CE9D  
E92C CE9E  
E92D CE9F  
E92E CEA0  
E92F CEA1  
E930 CEA2  
E931 CEA3  
E932 CEA4  
E933 CEA5  
E934 CEA6  
E935 CEA7  
E936 CEA8  
E937 CEA9  
E938 CEAA  
E939 CEAB  
E93A CEAC  
E93B CEAD  
E93C CEAE  
E93D CEAF  
E93E CEB0  
E93F CEB1  
E940 CEB2  
E941 CEB3  
E942 CEB4  
E943 CEB5  
E944 CEB6  
E945 CEB7  
E946 CEB8  
E947 CEB9  
E948 CEBA  
E949 CEBB  
E94A CEBC  
E94B CEBD  
E94C CEBE  
E94D CEBF  
E94E CEC0  
E94F CEC1  
E950 CEC2  
E951 CEC3  
E952 CEC4  
E953 CEC5  
E954 CEC6  
E955 CEC7  
E956 CEC8

E957 CEC9  
E958 CECA  
E959 CECB  
E95A CECC  
E95B CECD  
E95C CECE  
E95D CECF  
E95E CED0  
E95F CED1  
E960 CED2  
E961 CED3  
E962 CED4  
E963 CED5  
E964 CED6  
E965 CED7  
E966 CED8  
E967 CED9  
E968 CEDA  
E969 CEDB  
E96A CEDC  
E96B CEDD  
E96C CEDE  
E96D CEDF  
E96E CEE0  
E96F CEE1  
E970 CEE2  
E971 CEE3  
E972 CEE4  
E973 CEE5  
E974 CEE6  
E975 CEE7  
E976 CEE8  
E977 CEE9  
E978 CEEA  
E979 CEEB  
E97A CEEC  
E97B CEED  
E97C CEEE  
E97D CEEF  
E97E CEF0  
E97F CEF1  
E980 CEF2  
E981 CEF3  
E982 CEF4  
E983 CEF5  
E984 CEF6  
E985 CEF7  
E986 CEF8  
E987 CEF9  
E988 CEFA  
E989 CEFB  
E98A CEFC  
E98B CEFD  
E98C CF41  
E98D CF42  
E98E CF43  
E98F CF44  
E990 CF45  
E991 CF46  
E992 CF47  
E993 CF48  
E994 CF49  
E995 CF4A  
E996 CF4B  
E997 CF4C  
E998 CF4D  
E999 CF4E  
E99A CF4F  
E99B CF50  
E99C CF51  
E99D CF52  
E99E CF53  
E99F CF54  
E9A0 CF55  
E9A1 CF56  
E9A2 CF57  
E9A3 CF58  
E9A4 CF59  
E9A5 CF5A  
E9A6 CF5B  
E9A7 CF5C  
E9A8 CF5D

E9A9	CF5E
E9AA	CF5F
E9AB	CF60
E9AC	CF61
E9AD	CF62
E9AE	CF63
E9AF	CF64
E9B0	CF65
E9B1	CF66
E9B2	CF67
E9B3	CF68
E9B4	CF69
E9B5	CF6A
E9B6	CF6B
E9B7	CF6C
E9B8	CF6D
E9B9	CF6E
E9BA	CF6F
E9BB	CF70
E9BC	CF71
E9BD	CF72
E9BE	CF73
E9BF	CF74
E9C0	CF75
E9C1	CF76
E9C2	CF77
E9C3	CF78
E9C4	CF79
E9C5	CF7A
E9C6	CF7B
E9C7	CF7C
E9C8	CF7D
E9C9	CF7E
E9CA	CF7F
E9CB	CF81
E9CC	CF82
E9CD	CF83
E9CE	CF84
E9CF	CF85
E9D0	CF86
E9D1	CF87
E9D2	CF88
E9D3	CF89
E9D4	CF8A
E9D5	CF8B
E9D6	CF8C
E9D7	CF8D
E9D8	CF8E
E9D9	CF8F
E9DA	CF90
E9DB	CF91
E9DC	CF92
E9DD	CF93
E9DE	CF94
E9DF	CF95
E9E0	CF96
E9E1	CF97
E9E2	CF98
E9E3	CF99
E9E4	CF9A
E9E5	CF9B
E9E6	CF9C
E9E7	CF9D
E9E8	CF9E
E9E9	CF9F
E9EA	CFA0
E9EB	CFA1
E9EC	CFA2
E9ED	CFA3
E9EE	CFA4
E9EF	CFA5
E9F0	CFA6
E9F1	CFA7
E9F2	CFA8
E9F3	CFA9
E9F4	CFAA
E9F5	CFAB
E9F6	CFAC
E9F7	CFAD
E9F8	CFAE
E9F9	CFAF
E9FA	CFB0

E9FB	CFB1
E9FC	CFB2
E9FD	CFB3
E9FE	CFB4
E9FF	CFB5
EA00	CFB6
EA01	CFB7
EA02	CFB8
EA03	CFB9
EA04	CFBA
EA05	CFBB
EA06	CFBC
EA07	CFBD
EA08	CFBE
EA09	CFBF
EA0A	CFC0
EA0B	CFC1
EA0C	CFC2
EA0D	CFC3
EA0E	CFC4
EA0F	CFC5
EA10	CFC6
EA11	CFC7
EA12	CFC8
EA13	CFC9
EA14	CFCA
EA15	CFCB
EA16	CFCC
EA17	CFCD
EA18	CFCE
EA19	CFCF
EA1A	CFD0
EA1B	CFD1
EA1C	CFD2
EA1D	CFD3
EA1E	CFD4
EA1F	CFD5
EA20	CFD6
EA21	CFD7
EA22	CFD8
EA23	CFD9
EA24	CFDA
EA25	CFDB
EA26	CFDC
EA27	CFDD
EA28	CFDE
EA29	CFDF
EA2A	CFE0
EA2B	CFE1
EA2C	CFE2
EA2D	CFE3
EA2E	CFE4
EA2F	CFE5
EA30	CFE6
EA31	CFE7
EA32	CFE8
EA33	CFE9
EA34	CFEA
EA35	CFEB
EA36	CFEC
EA37	CFED
EA38	CFEE
EA39	CFEF
EA3A	CFF0
EA3B	CFF1
EA3C	CFF2
EA3D	CFF3
EA3E	CFF4
EA3F	CFF5
EA40	CFF6
EA41	CFF7
EA42	CFF8
EA43	CFF9
EA44	CFFA
EA45	CFFB
EA46	CFFC
EA47	CFFD
EA48	D041
EA49	D042
EA4A	D043
EA4B	D044
EA4C	D045

EA4D D046  
EA4E D047  
EA4F D048  
EA50 D049  
EA51 D04A  
EA52 D04B  
EA53 D04C  
EA54 D04D  
EA55 D04E  
EA56 D04F  
EA57 D050  
EA58 D051  
EA59 D052  
EA5A D053  
EA5B D054  
EA5C D055  
EA5D D056  
EA5E D057  
EA5F D058  
EA60 D059  
EA61 D05A  
EA62 D05B  
EA63 D05C  
EA64 D05D  
EA65 D05E  
EA66 D05F  
EA67 D060  
EA68 D061  
EA69 D062  
EA6A D063  
EA6B D064  
EA6C D065  
EA6D D066  
EA6E D067  
EA6F D068  
EA70 D069  
EA71 D06A  
EA72 D06B  
EA73 D06C  
EA74 D06D  
EA75 D06E  
EA76 D06F  
EA77 D070  
EA78 D071  
EA79 D072  
EA7A D073  
EA7B D074  
EA7C D075  
EA7D D076  
EA7E D077  
EA7F D078  
EA80 D079  
EA81 D07A  
EA82 D07B  
EA83 D07C  
EA84 D07D  
EA85 D07E  
EA86 D07F  
EA87 D081  
EA88 D082  
EA89 D083  
EA8A D084  
EA8B D085  
EA8C D086  
EA8D D087  
EA8E D088  
EA8F D089  
EA90 D08A  
EA91 D08B  
EA92 D08C  
EA93 D08D  
EA94 D08E  
EA95 D08F  
EA96 D090  
EA97 D091  
EA98 D092  
EA99 D093  
EA9A D094  
EA9B D095  
EA9C D096  
EA9D D097  
EA9E D098

EA9F	D099
EAA0	D09A
EAA1	D09B
EAA2	D09C
EAA3	D09D
EAA4	D09E
EAA5	D09F
EAA6	D0A0
EAA7	D0A1
EAA8	D0A2
EAA9	D0A3
EAAA	D0A4
EAAB	D0A5
EAAC	D0A6
EAAD	D0A7
EAAE	D0A8
EAAF	D0A9
EAB0	D0AA
EAB1	D0AB
EAB2	D0AC
EAB3	D0AD
EAB4	D0AE
EAB5	D0AF
EAB6	D0B0
EAB7	D0B1
EAB8	D0B2
EAB9	D0B3
EABA	D0B4
EABB	D0B5
EABC	D0B6
EABD	D0B7
EABE	D0B8
EABF	D0B9
EAC0	D0BA
EAC1	D0BB
EAC2	D0BC
EAC3	D0BD
EAC4	D0BE
EAC5	D0BF
EAC6	D0C0
EAC7	D0C1
EAC8	D0C2
EAC9	D0C3
EACA	D0C4
EACB	D0C5
EACC	D0C6
EACD	D0C7
EACE	D0C8
EACF	D0C9
EAD0	D0CA
EAD1	D0CB
EAD2	D0CC
EAD3	D0CD
EAD4	D0CE
EAD5	D0CF
EAD6	D0D0
EAD7	D0D1
EAD8	D0D2
EAD9	D0D3
EADA	D0D4
EADB	D0D5
EADC	D0D6
EADD	D0D7
EADE	D0D8
EADF	D0D9
EAE0	D0DA
EAE1	D0DB
EAE2	D0DC
EAE3	D0DD
EAE4	D0DE
EAE5	D0DF
EAE6	D0E0
EAE7	D0E1
EAE8	D0E2
EAE9	D0E3
EAEA	D0E4
EAEB	D0E5
EAEC	D0E6
EAED	D0E7
EAEF	D0E8
EAEF	D0E9
EAF0	D0EA

EAF1 D0EB  
EAF2 D0EC  
EAF3 D0ED  
EAF4 D0EE  
EAF5 D0EF  
EAF6 D0F0  
EAF7 D0F1  
EAF8 D0F2  
EAF9 D0F3  
EAF A D0F4  
EAFB D0F5  
E AFC D0F6  
E AFD D0F7  
E AFE D0F8  
E AFF D0F9  
EB00 D0FA  
EB01 D0FB  
EB02 D0FC  
EB03 D0FD  
EB04 D141  
EB05 D142  
EB06 D143  
EB07 D144  
EB08 D145  
EB09 D146  
EB0A D147  
EB0B D148  
EB0C D149  
EB0D D14A  
EB0E D14B  
EB0F D14C  
EB10 D14D  
EB11 D14E  
EB12 D14F  
EB13 D150  
EB14 D151  
EB15 D152  
EB16 D153  
EB17 D154  
EB18 D155  
EB19 D156  
EB1A D157  
EB1B D158  
EB1C D159  
EB1D D15A  
EB1E D15B  
EB1F D15C  
EB20 D15D  
EB21 D15E  
EB22 D15F  
EB23 D160  
EB24 D161  
EB25 D162  
EB26 D163  
EB27 D164  
EB28 D165  
EB29 D166  
EB2A D167  
EB2B D168  
EB2C D169  
EB2D D16A  
EB2E D16B  
EB2F D16C  
EB30 D16D  
EB31 D16E  
EB32 D16F  
EB33 D170  
EB34 D171  
EB35 D172  
EB36 D173  
EB37 D174  
EB38 D175  
EB39 D176  
EB3A D177  
EB3B D178  
EB3C D179  
EB3D D17A  
EB3E D17B  
EB3F D17C  
EB40 D17D  
EB41 D17E  
EB42 D17F

EB43 D181  
EB44 D182  
EB45 D183  
EB46 D184  
EB47 D185  
EB48 D186  
EB49 D187  
EB4A D188  
EB4B D189  
EB4C D18A  
EB4D D18B  
EB4E D18C  
EB4F D18D  
EB50 D18E  
EB51 D18F  
EB52 D190  
EB53 D191  
EB54 D192  
EB55 D193  
EB56 D194  
EB57 D195  
EB58 D196  
EB59 D197  
EB5A D198  
EB5B D199  
EB5C D19A  
EB5D D19B  
EB5E D19C  
EB5F D19D  
EB60 D19E  
EB61 D19F  
EB62 D1A0  
EB63 D1A1  
EB64 D1A2  
EB65 D1A3  
EB66 D1A4  
EB67 D1A5  
EB68 D1A6  
EB69 D1A7  
EB6A D1A8  
EB6B D1A9  
EB6C D1AA  
EB6D D1AB  
EB6E D1AC  
EB6F D1AD  
EB70 D1AE  
EB71 D1AF  
EB72 D1B0  
EB73 D1B1  
EB74 D1B2  
EB75 D1B3  
EB76 D1B4  
EB77 D1B5  
EB78 D1B6  
EB79 D1B7  
EB7A D1B8  
EB7B D1B9  
EB7C D1BA  
EB7D D1BB  
EB7E D1BC  
EB7F D1BD  
EB80 D1BE  
EB81 D1BF  
EB82 D1C0  
EB83 D1C1  
EB84 D1C2  
EB85 D1C3  
EB86 D1C4  
EB87 D1C5  
EB88 D1C6  
EB89 D1C7  
EB8A D1C8  
EB8B D1C9  
EB8C D1CA  
EB8D D1CB  
EB8E D1CC  
EB8F D1CD  
EB90 D1CE  
EB91 D1CF  
EB92 D1D0  
EB93 D1D1  
EB94 D1D2



EB95 D1D3  
EB96 D1D4  
EB97 D1D5  
EB98 D1D6  
EB99 D1D7  
EB9A D1D8  
EB9B D1D9  
EB9C D1DA  
EB9D D1DB  
EB9E D1DC  
EB9F D1DD  
EBA0 D1DE  
EBA1 D1DF  
EBA2 D1E0  
EBA3 D1E1  
EBA4 D1E2  
EBA5 D1E3  
EBA6 D1E4  
EBA7 D1E5  
EBA8 D1E6  
EBA9 D1E7  
EBAA D1E8  
EBAB D1E9  
EBAC D1EA  
EBAD D1EB  
EBAE D1EC  
EBAF D1ED  
EBB0 D1EE  
EBB1 D1EF  
EBB2 D1F0  
EBB3 D1F1  
EBB4 D1F2  
EBB5 D1F3  
EBB6 D1F4  
EBB7 D1F5  
EBB8 D1F6  
EBB9 D1F7  
EBBA D1F8  
EBBB D1F9  
EBBC D1FA  
EBBD D1FB  
EBBE D1FC  
EBBF D1FD  
EBC0 D241  
EBC1 D242  
EBC2 D243  
EBC3 D244  
EBC4 D245  
EBC5 D246  
EBC6 D247  
EBC7 D248  
EBC8 D249  
EBC9 D24A  
EBCA D24B  
EBCB D24C  
EBCC D24D  
EBCD D24E  
EBCE D24F  
EBCF D250  
EBD0 D251  
EBD1 D252  
EBD2 D253  
EBD3 D254  
EBD4 D255  
EBD5 D256  
EBD6 D257  
EBD7 D258  
EBD8 D259  
EBD9 D25A  
EBDA D25B  
EBDB D25C  
EBDC D25D  
EBDD D25E  
EBDE D25F  
EBDF D260  
EBE0 D261  
EBE1 D262  
EBE2 D263  
EBE3 D264  
EBE4 D265  
EBE5 D266  
EBE6 D267

EBE7	D268
EBE8	D269
EBE9	D26A
EBEA	D26B
EBEB	D26C
EBEC	D26D
EBED	D26E
EBEE	D26F
EBEF	D270
EBF0	D271
EBF1	D272
EBF2	D273
EBF3	D274
EBF4	D275
EBF5	D276
EBF6	D277
EBF7	D278
EBF8	D279
EBF9	D27A
EBFA	D27B
EBFB	D27C
EBFC	D27D
EBFD	D27E
EBFE	D27F
EBFF	D281
EC00	D282
EC01	D283
EC02	D284
EC03	D285
EC04	D286
EC05	D287
EC06	D288
EC07	D289
EC08	D28A
EC09	D28B
EC0A	D28C
EC0B	D28D
EC0C	D28E
EC0D	D28F
EC0E	D290
EC0F	D291
EC10	D292
EC11	D293
EC12	D294
EC13	D295
EC14	D296
EC15	D297
EC16	D298
EC17	D299
EC18	D29A
EC19	D29B
EC1A	D29C
EC1B	D29D
EC1C	D29E
EC1D	D29F
EC1E	D2A0
EC1F	D2A1
EC20	D2A2
EC21	D2A3
EC22	D2A4
EC23	D2A5
EC24	D2A6
EC25	D2A7
EC26	D2A8
EC27	D2A9
EC28	D2AA
EC29	D2AB
EC2A	D2AC
EC2B	D2AD
EC2C	D2AE
EC2D	D2AF
EC2E	D2B0
EC2F	D2B1
EC30	D2B2
EC31	D2B3
EC32	D2B4
EC33	D2B5
EC34	D2B6
EC35	D2B7
EC36	D2B8
EC37	D2B9
EC38	D2BA

EC39 D2BB  
EC3A D2BC  
EC3B D2BD  
EC3C D2BE  
EC3D D2BF  
EC3E D2C0  
EC3F D2C1  
EC40 D2C2  
EC41 D2C3  
EC42 D2C4  
EC43 D2C5  
EC44 D2C6  
EC45 D2C7  
EC46 D2C8  
EC47 D2C9  
EC48 D2CA  
EC49 D2CB  
EC4A D2CC  
EC4B D2CD  
EC4C D2CE  
EC4D D2CF  
EC4E D2D0  
EC4F D2D1  
EC50 D2D2  
EC51 D2D3  
EC52 D2D4  
EC53 D2D5  
EC54 D2D6  
EC55 D2D7  
EC56 D2D8  
EC57 D2D9  
EC58 D2DA  
EC59 D2DB  
EC5A D2DC  
EC5B D2DD  
EC5C D2DE  
EC5D D2DF  
EC5E D2E0  
EC5F D2E1  
EC60 D2E2  
EC61 D2E3  
EC62 D2E4  
EC63 D2E5  
EC64 D2E6  
EC65 D2E7  
EC66 D2E8  
EC67 D2E9  
EC68 D2EA  
EC69 D2EB  
EC6A D2EC  
EC6B D2ED  
EC6C D2EE  
EC6D D2EF  
EC6E D2F0  
EC6F D2F1  
EC70 D2F2  
EC71 D2F3  
EC72 D2F4  
EC73 D2F5  
EC74 D2F6  
EC75 D2F7  
EC76 D2F8  
EC77 D2F9  
EC78 D2FA  
EC79 D2FB  
EC7A D2FC  
EC7B D2FD  
EC7C D341  
EC7D D342  
EC7E D343  
EC7F D344  
EC80 D345  
EC81 D346  
EC82 D347  
EC83 D348  
EC84 D349  
EC85 D34A  
EC86 D34B  
EC87 D34C  
EC88 D34D  
EC89 D34E  
EC8A D34F

EC8B D350  
EC8C D351  
EC8D D352  
EC8E D353  
EC8F D354  
EC90 D355  
EC91 D356  
EC92 D357  
EC93 D358  
EC94 D359  
EC95 D35A  
EC96 D35B  
EC97 D35C  
EC98 D35D  
EC99 D35E  
EC9A D35F  
EC9B D360  
EC9C D361  
EC9D D362  
EC9E D363  
EC9F D364  
ECA0 D365  
ECA1 D366  
ECA2 D367  
ECA3 D368  
ECA4 D369  
ECA5 D36A  
ECA6 D36B  
ECA7 D36C  
ECA8 D36D  
ECA9 D36E  
ECAA D36F  
ECAB D370  
ECAC D371  
ECAD D372  
ECAE D373  
ECAF D374  
ECB0 D375  
ECB1 D376  
ECB2 D377  
ECB3 D378  
ECB4 D379  
ECB5 D37A  
ECB6 D37B  
ECB7 D37C  
ECB8 D37D  
ECB9 D37E  
ECBA D37F  
ECBB D381  
ECBC D382  
ECBD D383  
ECBE D384  
ECBF D385  
ECC0 D386  
ECC1 D387  
ECC2 D388  
ECC3 D389  
ECC4 D38A  
ECC5 D38B  
ECC6 D38C  
ECC7 D38D  
ECC8 D38E  
ECC9 D38F  
ECCA D390  
ECCB D391  
ECCC D392  
ECCD D393  
ECCE D394  
ECCF D395  
ECD0 D396  
ECD1 D397  
ECD2 D398  
ECD3 D399  
ECD4 D39A  
ECD5 D39B  
ECD6 D39C  
ECD7 D39D  
ECD8 D39E  
ECD9 D39F  
ECDA D3A0  
ECDB D3A1  
ECDC D3A2

ECDD D3A3  
ECDE D3A4  
ECDF D3A5  
ECE0 D3A6  
ECE1 D3A7  
ECE2 D3A8  
ECE3 D3A9  
ECE4 D3AA  
ECE5 D3AB  
ECE6 D3AC  
ECE7 D3AD  
ECE8 D3AE  
ECE9 D3AF  
ECEA D3B0  
ECEB D3B1  
ECEC D3B2  
ECED D3B3  
ECEE D3B4  
ECEF D3B5  
ECF0 D3B6  
ECF1 D3B7  
ECF2 D3B8  
ECF3 D3B9  
ECF4 D3BA  
ECF5 D3BB  
ECF6 D3BC  
ECF7 D3BD  
ECF8 D3BE  
ECF9 D3BF  
ECFA D3C0  
ECFB D3C1  
ECFC D3C2  
ECFD D3C3  
ECFE D3C4  
ECFF D3C5  
ED00 D3C6  
ED01 D3C7  
ED02 D3C8  
ED03 D3C9  
ED04 D3CA  
ED05 D3CB  
ED06 D3CC  
ED07 D3CD  
ED08 D3CE  
ED09 D3CF  
ED0A D3D0  
ED0B D3D1  
ED0C D3D2  
ED0D D3D3  
ED0E D3D4  
ED0F D3D5  
ED10 D3D6  
ED11 D3D7  
ED12 D3D8  
ED13 D3D9  
ED14 D3DA  
ED15 D3DB  
ED16 D3DC  
ED17 D3DD  
ED18 D3DE  
ED19 D3DF  
ED1A D3E0  
ED1B D3E1  
ED1C D3E2  
ED1D D3E3  
ED1E D3E4  
ED1F D3E5  
ED20 D3E6  
ED21 D3E7  
ED22 D3E8  
ED23 D3E9  
ED24 D3EA  
ED25 D3EB  
ED26 D3EC  
ED27 D3ED  
ED28 D3EE  
ED29 D3EF  
ED2A D3F0  
ED2B D3F1  
ED2C D3F2  
ED2D D3F3  
ED2E D3F4

ED2F D3F5  
ED30 D3F6  
ED31 D3F7  
ED32 D3F8  
ED33 D3F9  
ED34 D3FA  
ED35 D3FB  
ED36 D3FC  
ED37 D3FD  
ED38 D441  
ED39 D442  
ED3A D443  
ED3B D444  
ED3C D445  
ED3D D446  
ED3E D447  
ED3F D448  
ED40 D449  
ED41 D44A  
ED42 D44B  
ED43 D44C  
ED44 D44D  
ED45 D44E  
ED46 D44F  
ED47 D450  
ED48 D451  
ED49 D452  
ED4A D453  
ED4B D454  
ED4C D455  
ED4D D456  
ED4E D457  
ED4F D458  
ED50 D459  
ED51 D45A  
ED52 D45B  
ED53 D45C  
ED54 D45D  
ED55 D45E  
ED56 D45F  
ED57 D460  
ED58 D461  
ED59 D462  
ED5A D463  
ED5B D464  
ED5C D465  
ED5D D466  
ED5E D467  
ED5F D468  
ED60 D469  
ED61 D46A  
ED62 D46B  
ED63 D46C  
ED64 D46D  
ED65 D46E  
ED66 D46F  
ED67 D470  
ED68 D471  
ED69 D472  
ED6A D473  
ED6B D474  
ED6C D475  
ED6D D476  
ED6E D477  
ED6F D478  
ED70 D479  
ED71 D47A  
ED72 D47B  
ED73 D47C  
ED74 D47D  
ED75 D47E  
ED76 D47F  
ED77 D481  
ED78 D482  
ED79 D483  
ED7A D484  
ED7B D485  
ED7C D486  
ED7D D487  
ED7E D488  
ED7F D489  
ED80 D48A

ED81	D48B
ED82	D48C
ED83	D48D
ED84	D48E
ED85	D48F
ED86	D490
ED87	D491
ED88	D492
ED89	D493
ED8A	D494
ED8B	D495
ED8C	D496
ED8D	D497
ED8E	D498
ED8F	D499
ED90	D49A
ED91	D49B
ED92	D49C
ED93	D49D
ED94	D49E
ED95	D49F
ED96	D4A0
ED97	D4A1
ED98	D4A2
ED99	D4A3
ED9A	D4A4
ED9B	D4A5
ED9C	D4A6
ED9D	D4A7
ED9E	D4A8
ED9F	D4A9
EDA0	D4AA
EDA1	D4AB
EDA2	D4AC
EDA3	D4AD
EDA4	D4AE
EDA5	D4AF
EDA6	D4B0
EDA7	D4B1
EDA8	D4B2
EDA9	D4B3
EDAA	D4B4
EDAB	D4B5
EDAC	D4B6
EDAD	D4B7
EDAE	D4B8
EDAF	D4B9
EDB0	D4BA
EDB1	D4BB
EDB2	D4BC
EDB3	D4BD
EDB4	D4BE
EDB5	D4BF
EDB6	D4C0
EDB7	D4C1
EDB8	D4C2
EDB9	D4C3
EDBA	D4C4
EDBB	D4C5
EDBC	D4C6
EDBD	D4C7
EDBE	D4C8
EDBF	D4C9
EDC0	D4CA
EDC1	D4CB
EDC2	D4CC
EDC3	D4CD
EDC4	D4CE
EDC5	D4CF
EDC6	D4D0
EDC7	D4D1
EDC8	D4D2
EDC9	D4D3
EDCA	D4D4
EDCB	D4D5
EDCC	D4D6
EDCD	D4D7
EDCE	D4D8
EDCF	D4D9
EDD0	D4DA
EDD1	D4DB
EDD2	D4DC

EDD3	D4DD
EDD4	D4DE
EDD5	D4DF
EDD6	D4E0
EDD7	D4E1
EDD8	D4E2
EDD9	D4E3
EDDA	D4E4
EDDB	D4E5
EDDC	D4E6
EDDD	D4E7
EDDE	D4E8
EDDF	D4E9
EDE0	D4EA
EDE1	D4EB
EDE2	D4EC
EDE3	D4ED
EDE4	D4EE
EDE5	D4EF
EDE6	D4F0
EDE7	D4F1
EDE8	D4F2
EDE9	D4F3
EDEA	D4F4
EDEB	D4F5
EDEC	D4F6
EDED	D4F7
EDEE	D4F8
EDEF	D4F9
EDF0	D4FA
EDF1	D4FB
EDF2	D4FC
EDF3	D4FD
EDF4	D541
EDF5	D542
EDF6	D543
EDF7	D544
EDF8	D545
EDF9	D546
EDFA	D547
EDFB	D548
EDFC	D549
EDFD	D54A
EDFE	D54B
EDFF	D54C
EE00	D54D
EE01	D54E
EE02	D54F
EE03	D550
EE04	D551
EE05	D552
EE06	D553
EE07	D554
EE08	D555
EE09	D556
EE0A	D557
EE0B	D558
EE0C	D559
EE0D	D55A
EE0E	D55B
EE0F	D55C
EE10	D55D
EE11	D55E
EE12	D55F
EE13	D560
EE14	D561
EE15	D562
EE16	D563
EE17	D564
EE18	D565
EE19	D566
EE1A	D567
EE1B	D568
EE1C	D569
EE1D	D56A
EE1E	D56B
EE1F	D56C
EE20	D56D
EE21	D56E
EE22	D56F
EE23	D570
EE24	D571



EE25 D572  
EE26 D573  
EE27 D574  
EE28 D575  
EE29 D576  
EE2A D577  
EE2B D578  
EE2C D579  
EE2D D57A  
EE2E D57B  
EE2F D57C  
EE30 D57D  
EE31 D57E  
EE32 D57F  
EE33 D581  
EE34 D582  
EE35 D583  
EE36 D584  
EE37 D585  
EE38 D586  
EE39 D587  
EE3A D588  
EE3B D589  
EE3C D58A  
EE3D D58B  
EE3E D58C  
EE3F D58D  
EE40 D58E  
EE41 D58F  
EE42 D590  
EE43 D591  
EE44 D592  
EE45 D593  
EE46 D594  
EE47 D595  
EE48 D596  
EE49 D597  
EE4A D598  
EE4B D599  
EE4C D59A  
EE4D D59B  
EE4E D59C  
EE4F D59D  
EE50 D59E  
EE51 D59F  
EE52 D5A0  
EE53 D5A1  
EE54 D5A2  
EE55 D5A3  
EE56 D5A4  
EE57 D5A5  
EE58 D5A6  
EE59 D5A7  
EE5A D5A8  
EE5B D5A9  
EE5C D5AA  
EE5D D5AB  
EE5E D5AC  
EE5F D5AD  
EE60 D5AE  
EE61 D5AF  
EE62 D5B0  
EE63 D5B1  
EE64 D5B2  
EE65 D5B3  
EE66 D5B4  
EE67 D5B5  
EE68 D5B6  
EE69 D5B7  
EE6A D5B8  
EE6B D5B9  
EE6C D5BA  
EE6D D5BB  
EE6E D5BC  
EE6F D5BD  
EE70 D5BE  
EE71 D5BF  
EE72 D5C0  
EE73 D5C1  
EE74 D5C2  
EE75 D5C3  
EE76 D5C4

EE77 D5C5  
EE78 D5C6  
EE79 D5C7  
EE7A D5C8  
EE7B D5C9  
EE7C D5CA  
EE7D D5CB  
EE7E D5CC  
EE7F D5CD  
EE80 D5CE  
EE81 D5CF  
EE82 D5D0  
EE83 D5D1  
EE84 D5D2  
EE85 D5D3  
EE86 D5D4  
EE87 D5D5  
EE88 D5D6  
EE89 D5D7  
EE8A D5D8  
EE8B D5D9  
EE8C D5DA  
EE8D D5DB  
EE8E D5DC  
EE8F D5DD  
EE90 D5DE  
EE91 D5DF  
EE92 D5E0  
EE93 D5E1  
EE94 D5E2  
EE95 D5E3  
EE96 D5E4  
EE97 D5E5  
EE98 D5E6  
EE99 D5E7  
EE9A D5E8  
EE9B D5E9  
EE9C D5EA  
EE9D D5EB  
EE9E D5EC  
EE9F D5ED  
EEA0 D5EE  
EEA1 D5EF  
EEA2 D5F0  
EEA3 D5F1  
EEA4 D5F2  
EEA5 D5F3  
EEA6 D5F4  
EEA7 D5F5  
EEA8 D5F6  
EEA9 D5F7  
EEAA D5F8  
EEAB D5F9  
EEAC D5FA  
EEAD D5FB  
EEAE D5FC  
EEAF D5FD  
EEB0 D641  
EEB1 D642  
EEB2 D643  
EEB3 D644  
EEB4 D645  
EEB5 D646  
EEB6 D647  
EEB7 D648  
EEB8 D649  
EEB9 D64A  
EEBA D64B  
EEBB D64C  
EEBC D64D  
EEBD D64E  
EEBE D64F  
EEBF D650  
EEC0 D651  
EEC1 D652  
EEC2 D653  
EEC3 D654  
EEC4 D655  
EEC5 D656  
EEC6 D657  
EEC7 D658  
EEC8 D659

EEC9 D65A  
EECA D65B  
EECB D65C  
EECC D65D  
EECD D65E  
EECE D65F  
EECF D660  
EED0 D661  
EED1 D662  
EED2 D663  
EED3 D664  
EED4 D665  
EED5 D666  
EED6 D667  
EED7 D668  
EED8 D669  
EED9 D66A  
EEDA D66B  
EEDB D66C  
EEDC D66D  
EEDD D66E  
EEDF D66F  
EEF0 D670  
EEF1 D671  
EEF2 D672  
EEF3 D673  
EEF4 D674  
EEF5 D675  
EEF6 D676  
EEF7 D677  
EEF8 D678  
EEF9 D679  
EEFA D67A  
EEFB D67B  
EEFC D67C  
EEDC D67D  
EEDD D67E  
EEDF D67F  
EEF0 D681  
EEF1 D682  
EEF2 D683  
EEF3 D684  
EEF4 D685  
EEF5 D686  
EEF6 D687  
EEF7 D688  
EEF8 D689  
EEF9 D68A  
EEFA D68B  
EEFB D68C  
EEFC D68D  
EEDC D68E  
EEDD D68F  
EEDF D690  
EEF0 D691  
EF00 D692  
EF01 D693  
EF02 D694  
EF03 D695  
EF04 D696  
EF05 D697  
EF06 D698  
EF07 D699  
EF08 D69A  
EF09 D69B  
EF0A D69C  
EF0B D69D  
EF0C D69E  
EF0D D69F  
EF0E D6A0  
EF0F D6A1  
EF10 D6A2  
EF11 D6A3  
EF12 D6A4  
EF13 D6A5  
EF14 D6A6  
EF15 D6A7  
EF16 D6A8  
EF17 D6A9  
EF18 D6AA  
EF19 D6AB  
EF1A D6AC

EF1B	D6AD
EF1C	D6AE
EF1D	D6AF
EF1E	D6B0
EF1F	D6B1
EF20	D6B2
EF21	D6B3
EF22	D6B4
EF23	D6B5
EF24	D6B6
EF25	D6B7
EF26	D6B8
EF27	D6B9
EF28	D6BA
EF29	D6BB
EF2A	D6BC
EF2B	D6BD
EF2C	D6BE
EF2D	D6BF
EF2E	D6C0
EF2F	D6C1
EF30	D6C2
EF31	D6C3
EF32	D6C4
EF33	D6C5
EF34	D6C6
EF35	D6C7
EF36	D6C8
EF37	D6C9
EF38	D6CA
EF39	D6CB
EF3A	D6CC
EF3B	D6CD
EF3C	D6CE
EF3D	D6CF
EF3E	D6D0
EF3F	D6D1
EF40	D6D2
EF41	D6D3
EF42	D6D4
EF43	D6D5
EF44	D6D6
EF45	D6D7
EF46	D6D8
EF47	D6D9
EF48	D6DA
EF49	D6DB
EF4A	D6DC
EF4B	D6DD
EF4C	D6DE
EF4D	D6DF
EF4E	D6E0
EF4F	D6E1
EF50	D6E2
EF51	D6E3
EF52	D6E4
EF53	D6E5
EF54	D6E6
EF55	D6E7
EF56	D6E8
EF57	D6E9
EF58	D6EA
EF59	D6EB
EF5A	D6EC
EF5B	D6ED
EF5C	D6EE
EF5D	D6EF
EF5E	D6F0
EF5F	D6F1
EF60	D6F2
EF61	D6F3
EF62	D6F4
EF63	D6F5
EF64	D6F6
EF65	D6F7
EF66	D6F8
EF67	D6F9
EF68	D6FA
EF69	D6FB
EF6A	D6FC
EF6B	D6FD
EF6C	D741

EF6D D742  
EF6E D743  
EF6F D744  
EF70 D745  
EF71 D746  
EF72 D747  
EF73 D748  
EF74 D749  
EF75 D74A  
EF76 D74B  
EF77 D74C  
EF78 D74D  
EF79 D74E  
EF7A D74F  
EF7B D750  
EF7C D751  
EF7D D752  
EF7E D753  
EF7F D754  
EF80 D755  
EF81 D756  
EF82 D757  
EF83 D758  
EF84 D759  
EF85 D75A  
EF86 D75B  
EF87 D75C  
EF88 D75D  
EF89 D75E  
EF8A D75F  
EF8B D760  
EF8C D761  
EF8D D762  
EF8E D763  
EF8F D764  
EF90 D765  
EF91 D766  
EF92 D767  
EF93 D768  
EF94 D769  
EF95 D76A  
EF96 D76B  
EF97 D76C  
EF98 D76D  
EF99 D76E  
EF9A D76F  
EF9B D770  
EF9C D771  
EF9D D772  
EF9E D773  
EF9F D774  
EFA0 D775  
EFA1 D776  
EFA2 D777  
EFA3 D778  
EFA4 D779  
EFA5 D77A  
EFA6 D77B  
EFA7 D77C  
EFA8 D77D  
EFA9 D77E  
EFAA D77F  
EFAB D781  
EFAC D782  
EFAD D783  
EFAE D784  
EFAF D785  
EFB0 D786  
EFB1 D787  
EFB2 D788  
EFB3 D789  
EFB4 D78A  
EFB5 D78B  
EFB6 D78C  
EFB7 D78D  
EFB8 D78E  
EFB9 D78F  
EFBA D790  
EFBB D791  
EFBC D792  
EFBD D793  
EFBE D794

EFBF	D795
EFC0	D796
EFC1	D797
EFC2	D798
EFC3	D799
EFC4	D79A
EFC5	D79B
EFC6	D79C
EFC7	D79D
EFC8	D79E
EFC9	D79F
ECCA	D7A0
EFCB	D7A1
ECCC	D7A2
ECCD	D7A3
ECCF	D7A4
ECCF	D7A5
EFD0	D7A6
EFD1	D7A7
EFD2	D7A8
EFD3	D7A9
EFD4	D7AA
EFD5	D7AB
EFD6	D7AC
EFD7	D7AD
EFD8	D7AE
EFD9	D7AF
EFDA	D7B0
EFDB	D7B1
EFDC	D7B2
EFDD	D7B3
EFDE	D7B4
EFDF	D7B5
EFE0	D7B6
EFE1	D7B7
EFE2	D7B8
EFE3	D7B9
EFE4	D7BA
EFE5	D7BB
EFE6	D7BC
EFE7	D7BD
EFE8	D7BE
EFE9	D7BF
EFEA	D7C0
EFEB	D7C1
EFEC	D7C2
EFED	D7C3
EFEE	D7C4
EFEF	D7C5
EFF0	D7C6
EFF1	D7C7
EFF2	D7C8
EFF3	D7C9
EFF4	D7CA
EFF5	D7CB
EFF6	D7CC
EFF7	D7CD
EFF8	D7CE
EFF9	D7CF
EFFA	D7D0
EFFB	D7D1
EFFC	D7D2
EFFD	D7D3
EFEE	D7D4
EFFE	D7D5
F000	D7D6
F001	D7D7
F002	D7D8
F003	D7D9
F004	D7DA
F005	D7DB
F006	D7DC
F007	D7DD
F008	D7DE
F009	D7DF
F00A	D7E0
F00B	D7E1
F00C	D7E2
F00D	D7E3
F00E	D7E4
F00F	D7E5
F010	D7E6

F011 D7E7  
F012 D7E8  
F013 D7E9  
F014 D7EA  
F015 D7EB  
F016 D7EC  
F017 D7ED  
F018 D7EE  
F019 D7EF  
F01A D7F0  
F01B D7F1  
F01C D7F2  
F01D D7F3  
F01E D7F4  
F01F D7F5  
F020 D7F6  
F021 D7F7  
F022 D7F8  
F023 D7F9  
F024 D7FA  
F025 D7FB  
F026 D7FC  
F027 D7FD  
F028 D841  
F029 D842  
F02A D843  
F02B D844  
F02C D845  
F02D D846  
F02E D847  
F02F D848  
F030 D849  
F031 D84A  
F032 D84B  
F033 D84C  
F034 D84D  
F035 D84E  
F036 D84F  
F037 D850  
F038 D851  
F039 D852  
F03A D853  
F03B D854  
F03C D855  
F03D D856  
F03E D857  
F03F D858  
F040 D859  
F041 D85A  
F042 D85B  
F043 D85C  
F044 D85D  
F045 D85E  
F046 D85F  
F047 D860  
F048 D861  
F049 D862  
F04A D863  
F04B D864  
F04C D865  
F04D D866  
F04E D867  
F04F D868  
F050 D869  
F051 D86A  
F052 D86B  
F053 D86C  
F054 D86D  
F055 D86E  
F056 D86F  
F057 D870  
F058 D871  
F059 D872  
F05A D873  
F05B D874  
F05C D875  
F05D D876  
F05E D877  
F05F D878  
F060 D879  
F061 D87A  
F062 D87B

F063 D87C  
F064 D87D  
F065 D87E  
F066 D87F  
F067 D881  
F068 D882  
F069 D883  
F06A D884  
F06B D885  
F06C D886  
F06D D887  
F06E D888  
F06F D889  
F070 D88A  
F071 D88B  
F072 D88C  
F073 D88D  
F074 D88E  
F075 D88F  
F076 D890  
F077 D891  
F078 D892  
F079 D893  
F07A D894  
F07B D895  
F07C D896  
F07D D897  
F07E D898  
F07F D899  
F080 D89A  
F081 D89B  
F082 D89C  
F083 D89D  
F084 D89E  
F085 D89F  
F086 D8A0  
F087 D8A1  
F088 D8A2  
F089 D8A3  
F08A D8A4  
F08B D8A5  
F08C D8A6  
F08D D8A7  
F08E D8A8  
F08F D8A9  
F090 D8AA  
F091 D8AB  
F092 D8AC  
F093 D8AD  
F094 D8AE  
F095 D8AF  
F096 D8B0  
F097 D8B1  
F098 D8B2  
F099 D8B3  
F09A D8B4  
F09B D8B5  
F09C D8B6  
F09D D8B7  
F09E D8B8  
F09F D8B9  
F0A0 D8BA  
F0A1 D8BB  
F0A2 D8BC  
F0A3 D8BD  
F0A4 D8BE  
F0A5 D8BF  
F0A6 D8C0  
F0A7 D8C1  
F0A8 D8C2  
F0A9 D8C3  
F0AA D8C4  
F0AB D8C5  
F0AC D8C6  
F0AD D8C7  
F0AE D8C8  
F0AF D8C9  
F0B0 D8CA  
F0B1 D8CB  
F0B2 D8CC  
F0B3 D8CD  
F0B4 D8CE



F0B5	D8CF
F0B6	D8D0
F0B7	D8D1
F0B8	D8D2
F0B9	D8D3
F0BA	D8D4
F0BB	D8D5
F0BC	D8D6
F0BD	D8D7
F0BE	D8D8
F0BF	D8D9
F0C0	D8DA
F0C1	D8DB
F0C2	D8DC
F0C3	D8DD
F0C4	D8DE
F0C5	D8DF
F0C6	D8E0
F0C7	D8E1
F0C8	D8E2
F0C9	D8E3
F0CA	D8E4
F0CB	D8E5
F0CC	D8E6
F0CD	D8E7
F0CE	D8E8
F0CF	D8E9
F0D0	D8EA
F0D1	D8EB
F0D2	D8EC
F0D3	D8ED
F0D4	D8EE
F0D5	D8EF
F0D6	D8F0
F0D7	D8F1
F0D8	D8F2
F0D9	D8F3
F0DA	D8F4
F0DB	D8F5
F0DC	D8F6
F0DD	D8F7
F0DE	D8F8
F0DF	D8F9
F0E0	D8FA
F0E1	D8FB
F0E2	D8FC
F0E3	D8FD
F0E4	D941
F0E5	D942
F0E6	D943
F0E7	D944
F0E8	D945
F0E9	D946
F0EA	D947
F0EB	D948
F0EC	D949
F0ED	D94A
F0EE	D94B
F0EF	D94C
F0F0	D94D
F0F1	D94E
F0F2	D94F
F0F3	D950
F0F4	D951
F0F5	D952
F0F6	D953
F0F7	D954
F0F8	D955
F0F9	D956
F0FA	D957
F0FB	D958
F0FC	D959
F0FD	D95A
F0FE	D95B
F0FF	D95C
F100	D95D
F101	D95E
F102	D95F
F103	D960
F104	D961
F105	D962
F106	D963

F107 D964  
F108 D965  
F109 D966  
F10A D967  
F10B D968  
F10C D969  
F10D D96A  
F10E D96B  
F10F D96C  
F110 D96D  
F111 D96E  
F112 D96F  
F113 D970  
F114 D971  
F115 D972  
F116 D973  
F117 D974  
F118 D975  
F119 D976  
F11A D977  
F11B D978  
F11C D979  
F11D D97A  
F11E D97B  
F11F D97C  
F120 D97D  
F121 D97E  
F122 D97F  
F123 D981  
F124 D982  
F125 D983  
F126 D984  
F127 D985  
F128 D986  
F129 D987  
F12A D988  
F12B D989  
F12C D98A  
F12D D98B  
F12E D98C  
F12F D98D  
F130 D98E  
F131 D98F  
F132 D990  
F133 D991  
F134 D992  
F135 D993  
F136 D994  
F137 D995  
F138 D996  
F139 D997  
F13A D998  
F13B D999  
F13C D99A  
F13D D99B  
F13E D99C  
F13F D99D  
F140 D99E  
F141 D99F  
F142 D9A0  
F143 D9A1  
F144 D9A2  
F145 D9A3  
F146 D9A4  
F147 D9A5  
F148 D9A6  
F149 D9A7  
F14A D9A8  
F14B D9A9  
F14C D9AA  
F14D D9AB  
F14E D9AC  
F14F D9AD  
F150 D9AE  
F151 D9AF  
F152 D9B0  
F153 D9B1  
F154 D9B2  
F155 D9B3  
F156 D9B4  
F157 D9B5  
F158 D9B6

F159 D9B7  
F15A D9B8  
F15B D9B9  
F15C D9BA  
F15D D9BB  
F15E D9BC  
F15F D9BD  
F160 D9BE  
F161 D9BF  
F162 D9C0  
F163 D9C1  
F164 D9C2  
F165 D9C3  
F166 D9C4  
F167 D9C5  
F168 D9C6  
F169 D9C7  
F16A D9C8  
F16B D9C9  
F16C D9CA  
F16D D9CB  
F16E D9CC  
F16F D9CD  
F170 D9CE  
F171 D9CF  
F172 D9D0  
F173 D9D1  
F174 D9D2  
F175 D9D3  
F176 D9D4  
F177 D9D5  
F178 D9D6  
F179 D9D7  
F17A D9D8  
F17B D9D9  
F17C D9DA  
F17D D9DB  
F17E D9DC  
F17F D9DD  
F180 D9DE  
F181 D9DF  
F182 D9E0  
F183 D9E1  
F184 D9E2  
F185 D9E3  
F186 D9E4  
F187 D9E5  
F188 D9E6  
F189 D9E7  
F18A D9E8  
F18B D9E9  
F18C D9EA  
F18D D9EB  
F18E D9EC  
F18F D9ED  
F190 D9EE  
F191 D9EF  
F192 D9F0  
F193 D9F1  
F194 D9F2  
F195 D9F3  
F196 D9F4  
F197 D9F5  
F198 D9F6  
F199 D9F7  
F19A D9F8  
F19B D9F9  
F19C D9FA  
F19D D9FB  
F19E D9FC  
F19F D9FD  
F1A0 DA41  
F1A1 DA42  
F1A2 DA43  
F1A3 DA44  
F1A4 DA45  
F1A5 DA46  
F1A6 DA47  
F1A7 DA48  
F1A8 DA49  
F1A9 DA4A  
F1AA DA4B

F1AB	DA4C
F1AC	DA4D
F1AD	DA4E
F1AE	DA4F
F1AF	DA50
F1B0	DA51
F1B1	DA52
F1B2	DA53
F1B3	DA54
F1B4	DA55
F1B5	DA56
F1B6	DA57
F1B7	DA58
F1B8	DA59
F1B9	DA5A
F1BA	DA5B
F1BB	DA5C
F1BC	DA5D
F1BD	DA5E
F1BE	DA5F
F1BF	DA60
F1C0	DA61
F1C1	DA62
F1C2	DA63
F1C3	DA64
F1C4	DA65
F1C5	DA66
F1C6	DA67
F1C7	DA68
F1C8	DA69
F1C9	DA6A
F1CA	DA6B
F1CB	DA6C
F1CC	DA6D
F1CD	DA6E
F1CE	DA6F
F1CF	DA70
F1D0	DA71
F1D1	DA72
F1D2	DA73
F1D3	DA74
F1D4	DA75
F1D5	DA76
F1D6	DA77
F1D7	DA78
F1D8	DA79
F1D9	DA7A
F1DA	DA7B
F1DB	DA7C
F1DC	DA7D
F1DD	DA7E
F1DE	DA7F
F1DF	DA81
F1E0	DA82
F1E1	DA83
F1E2	DA84
F1E3	DA85
F1E4	DA86
F1E5	DA87
F1E6	DA88
F1E7	DA89
F1E8	DA8A
F1E9	DA8B
F1EA	DA8C
F1EB	DA8D
F1EC	DA8E
F1ED	DA8F
F1EE	DA90
F1EF	DA91
F1F0	DA92
F1F1	DA93
F1F2	DA94
F1F3	DA95
F1F4	DA96
F1F5	DA97
F1F6	DA98
F1F7	DA99
F1F8	DA9A
F1F9	DA9B
F1FA	DA9C
F1FB	DA9D
F1FC	DA9E

F1FD DA9F  
F1FE DAA0  
F1FF DAA1  
F200 DAA2  
F201 DAA3  
F202 DAA4  
F203 DAA5  
F204 DAA6  
F205 DAA7  
F206 DAA8  
F207 DAA9  
F208 DAAA  
F209 DAAB  
F20A DAAC  
F20B DAAD  
F20C DAAE  
F20D DAAF  
F20E DAB0  
F20F DAB1  
F210 DAB2  
F211 DAB3  
F212 DAB4  
F213 DAB5  
F214 DAB6  
F215 DAB7  
F216 DAB8  
F217 DAB9  
F218 DABA  
F219 DABB  
F21A DABC  
F21B DABD  
F21C DABE  
F21D DABF  
F21E DAC0  
F21F DAC1  
F220 DAC2  
F221 DAC3  
F222 DAC4  
F223 DAC5  
F224 DAC6  
F225 DAC7  
F226 DAC8  
F227 DAC9  
F228 DACA  
F229 DACB  
F22A DACC  
F22B DACD  
F22C DACE  
F22D DACF  
F22E DAD0  
F22F DAD1  
F230 DAD2  
F231 DAD3  
F232 DAD4  
F233 DAD5  
F234 DAD6  
F235 DAD7  
F236 DAD8  
F237 DAD9  
F238 DADA  
F239 DADB  
F23A DADC  
F23B DADD  
F23C DADE  
F23D DADF  
F23E DAE0  
F23F DAE1  
F240 DAE2  
F241 DAE3  
F242 DAE4  
F243 DAE5  
F244 DAE6  
F245 DAE7  
F246 DAE8  
F247 DAE9  
F248 DAEA  
F249 DAEB  
F24A DAEC  
F24B DAED  
F24C DAEF  
F24D DAEF  
F24E DAF0

F24F DAF1  
F250 DAF2  
F251 DAF3  
F252 DAF4  
F253 DAF5  
F254 DAF6  
F255 DAF7  
F256 DAF8  
F257 DAF9  
F258 DAFA  
F259 DAFB  
F25A DAFC  
F25B DAFD  
F25C DB41  
F25D DB42  
F25E DB43  
F25F DB44  
F260 DB45  
F261 DB46  
F262 DB47  
F263 DB48  
F264 DB49  
F265 DB4A  
F266 DB4B  
F267 DB4C  
F268 DB4D  
F269 DB4E  
F26A DB4F  
F26B DB50  
F26C DB51  
F26D DB52  
F26E DB53  
F26F DB54  
F270 DB55  
F271 DB56  
F272 DB57  
F273 DB58  
F274 DB59  
F275 DB5A  
F276 DB5B  
F277 DB5C  
F278 DB5D  
F279 DB5E  
F27A DB5F  
F27B DB60  
F27C DB61  
F27D DB62  
F27E DB63  
F27F DB64  
F280 DB65  
F281 DB66  
F282 DB67  
F283 DB68  
F284 DB69  
F285 DB6A  
F286 DB6B  
F287 DB6C  
F288 DB6D  
F289 DB6E  
F28A DB6F  
F28B DB70  
F28C DB71  
F28D DB72  
F28E DB73  
F28F DB74  
F290 DB75  
F291 DB76  
F292 DB77  
F293 DB78  
F294 DB79  
F295 DB7A  
F296 DB7B  
F297 DB7C  
F298 DB7D  
F299 DB7E  
F29A DB7F  
F29B DB81  
F29C DB82  
F29D DB83  
F29E DB84  
F29F DB85  
F2A0 DB86

F2A1	DB87
F2A2	DB88
F2A3	DB89
F2A4	DB8A
F2A5	DB8B
F2A6	DB8C
F2A7	DB8D
F2A8	DB8E
F2A9	DB8F
F2AA	DB90
F2AB	DB91
F2AC	DB92
F2AD	DB93
F2AE	DB94
F2AF	DB95
F2B0	DB96
F2B1	DB97
F2B2	DB98
F2B3	DB99
F2B4	DB9A
F2B5	DB9B
F2B6	DB9C
F2B7	DB9D
F2B8	DB9E
F2B9	DB9F
F2BA	DBA0
F2BB	DBA1
F2BC	DBA2
F2BD	DBA3
F2BE	DBA4
F2BF	DBA5
F2C0	DBA6
F2C1	DBA7
F2C2	DBA8
F2C3	DBA9
F2C4	DBAA
F2C5	DBAB
F2C6	DBAC
F2C7	DBAD
F2C8	DBAE
F2C9	DBAF
F2CA	DBB0
F2CB	DBB1
F2CC	DBB2
F2CD	DBB3
F2CE	DBB4
F2CF	DBB5
F2D0	DBB6
F2D1	DBB7
F2D2	DBB8
F2D3	DBB9
F2D4	DBBA
F2D5	DBBB
F2D6	DBBC
F2D7	DBBD
F2D8	DBBE
F2D9	DBBF
F2DA	DBC0
F2DB	DBC1
F2DC	DBC2
F2DD	DBC3
F2DE	DBC4
F2DF	DBC5
F2E0	DBC6
F2E1	DBC7
F2E2	DBC8
F2E3	DBC9
F2E4	DBCA
F2E5	DBCB
F2E6	DBCC
F2E7	DBCD
F2E8	DBCE
F2E9	DBCF
F2EA	DBD0
F2EB	DBD1
F2EC	DBD2
F2ED	DBD3
F2EE	DBD4
F2EF	DBD5
F2F0	DBD6
F2F1	DBD7
F2F2	DBD8

F2F3 DBD9  
F2F4 DBDA  
F2F5 DBDB  
F2F6 DBDC  
F2F7 DBDD  
F2F8 DBDE  
F2F9 DBDF  
F2FA DBE0  
F2FB DBE1  
F2FC DBE2  
F2FD DBE3  
F2FE DBE4  
F2FF DBE5  
F300 DBE6  
F301 DBE7  
F302 DBE8  
F303 DBE9  
F304 DBEA  
F305 DBEB  
F306 DBEC  
F307 DBED  
F308 DBEE  
F309 DBEF  
F30A DBF0  
F30B DBF1  
F30C DBF2  
F30D DBF3  
F30E DBF4  
F30F DBF5  
F310 DBF6  
F311 DBF7  
F312 DBF8  
F313 DBF9  
F314 DBFA  
F315 DBFB  
F316 DBFC  
F317 DBFD  
F318 DC41  
F319 DC42  
F31A DC43  
F31B DC44  
F31C DC45  
F31D DC46  
F31E DC47  
F31F DC48  
F320 DC49  
F321 DC4A  
F322 DC4B  
F323 DC4C  
F324 DC4D  
F325 DC4E  
F326 DC4F  
F327 DC50  
F328 DC51  
F329 DC52  
F32A DC53  
F32B DC54  
F32C DC55  
F32D DC56  
F32E DC57  
F32F DC58  
F330 DC59  
F331 DC5A  
F332 DC5B  
F333 DC5C  
F334 DC5D  
F335 DC5E  
F336 DC5F  
F337 DC60  
F338 DC61  
F339 DC62  
F33A DC63  
F33B DC64  
F33C DC65  
F33D DC66  
F33E DC67  
F33F DC68  
F340 DC69  
F341 DC6A  
F342 DC6B  
F343 DC6C  
F344 DC6D



F345 DC6E  
F346 DC6F  
F347 DC70  
F348 DC71  
F349 DC72  
F34A DC73  
F34B DC74  
F34C DC75  
F34D DC76  
F34E DC77  
F34F DC78  
F350 DC79  
F351 DC7A  
F352 DC7B  
F353 DC7C  
F354 DC7D  
F355 DC7E  
F356 DC7F  
F357 DC81  
F358 DC82  
F359 DC83  
F35A DC84  
F35B DC85  
F35C DC86  
F35D DC87  
F35E DC88  
F35F DC89  
F360 DC8A  
F361 DC8B  
F362 DC8C  
F363 DC8D  
F364 DC8E  
F365 DC8F  
F366 DC90  
F367 DC91  
F368 DC92  
F369 DC93  
F36A DC94  
F36B DC95  
F36C DC96  
F36D DC97  
F36E DC98  
F36F DC99  
F370 DC9A  
F371 DC9B  
F372 DC9C  
F373 DC9D  
F374 DC9E  
F375 DC9F  
F376 DCA0  
F377 DCA1  
F378 DCA2  
F379 DCA3  
F37A DCA4  
F37B DCA5  
F37C DCA6  
F37D DCA7  
F37E DCA8  
F37F DCA9  
F380 DCAA  
F381 DCAB  
F382 DCAC  
F383 DCAD  
F384 DCAE  
F385 DCAF  
F386 DCB0  
F387 DCB1  
F388 DCB2  
F389 DCB3  
F38A DCB4  
F38B DCB5  
F38C DCB6  
F38D DCB7  
F38E DCB8  
F38F DCB9  
F390 DCBA  
F391 DCBB  
F392 DCBC  
F393 DCBD  
F394 DCBE  
F395 DCBF  
F396 DCC0

F397 DCC1  
F398 DCC2  
F399 DCC3  
F39A DCC4  
F39B DCC5  
F39C DCC6  
F39D DCC7  
F39E DCC8  
F39F DCC9  
F3A0 DCCA  
F3A1 DCCB  
F3A2 DCCC  
F3A3 DCCD  
F3A4 DCCE  
F3A5 DCCF  
F3A6 DCD0  
F3A7 DCD1  
F3A8 DCD2  
F3A9 DCD3  
F3AA DCD4  
F3AB DCD5  
F3AC DCD6  
F3AD DCD7  
F3AE DCD8  
F3AF DCD9  
F3B0 DCDA  
F3B1 DCDB  
F3B2 DCDC  
F3B3 DCDD  
F3B4 DCDE  
F3B5 DCDF  
F3B6 DCE0  
F3B7 DCE1  
F3B8 DCE2  
F3B9 DCE3  
F3BA DCE4  
F3BB DCE5  
F3BC DCE6  
F3BD DCE7  
F3BE DCE8  
F3BF DCE9  
F3C0 DCEA  
F3C1 DCEB  
F3C2 DCEC  
F3C3 DCED  
F3C4 DCEE  
F3C5 DCEF  
F3C6 DCF0  
F3C7 DCF1  
F3C8 DCF2  
F3C9 DCF3  
F3CA DCF4  
F3CB DCF5  
F3CC DCF6  
F3CD DCF7  
F3CE DCF8  
F3CF DCF9  
F3D0 DCFA  
F3D1 DCFB  
F3D2 DCFC  
F3D3 DCFD  
F3D4 DD41  
F3D5 DD42  
F3D6 DD43  
F3D7 DD44  
F3D8 DD45  
F3D9 DD46  
F3DA DD47  
F3DB DD48  
F3DC DD49  
F3DD DD4A  
F3DE DD4B  
F3DF DD4C  
F3E0 DD4D  
F3E1 DD4E  
F3E2 DD4F  
F3E3 DD50  
F3E4 DD51  
F3E5 DD52  
F3E6 DD53  
F3E7 DD54  
F3E8 DD55

F3E9 DD56  
F3EA DD57  
F3EB DD58  
F3EC DD59  
F3ED DD5A  
F3EE DD5B  
F3EF DD5C  
F3F0 DD5D  
F3F1 DD5E  
F3F2 DD5F  
F3F3 DD60  
F3F4 DD61  
F3F5 DD62  
F3F6 DD63  
F3F7 DD64  
F3F8 DD65  
F3F9 DD66  
F3FA DD67  
F3FB DD68  
F3FC DD69  
F3FD DD6A  
F3FE DD6B  
F3FF DD6C  
F400 DD6D  
F401 DD6E  
F402 DD6F  
F403 DD70  
F404 DD71  
F405 DD72  
F406 DD73  
F407 DD74  
F408 DD75  
F409 DD76  
F40A DD77  
F40B DD78  
F40C DD79  
F40D DD7A  
F40E DD7B  
F40F DD7C  
F410 DD7D  
F411 DD7E  
F412 DD7F  
F413 DD81  
F414 DD82  
F415 DD83  
F416 DD84  
F417 DD85  
F418 DD86  
F419 DD87  
F41A DD88  
F41B DD89  
F41C DD8A  
F41D DD8B  
F41E DD8C  
F41F DD8D  
F420 DD8E  
F421 DD8F  
F422 DD90  
F423 DD91  
F424 DD92  
F425 DD93  
F426 DD94  
F427 DD95  
F428 DD96  
F429 DD97  
F42A DD98  
F42B DD99  
F42C DD9A  
F42D DD9B  
F42E DD9C  
F42F DD9D  
F430 DD9E  
F431 DD9F  
F432 DDA0  
F433 DDA1  
F434 DDA2  
F435 DDA3  
F436 DDA4  
F437 DDA5  
F438 DDA6  
F439 DDA7  
F43A DDA8

F43B DDA9  
F43C DDAA  
F43D DDAB  
F43E DDAC  
F43F DDAD  
F440 DDAE  
F441 DDAF  
F442 DDB0  
F443 DDB1  
F444 DDB2  
F445 DDB3  
F446 DDB4  
F447 DDB5  
F448 DDB6  
F449 DDB7  
F44A DDB8  
F44B DDB9  
F44C DDBA  
F44D DDBB  
F44E DDBC  
F44F DDBD  
F450 DDBE  
F451 DDBF  
F452 DDC0  
F453 DDC1  
F454 DDC2  
F455 DDC3  
F456 DDC4  
F457 DDC5  
F458 DDC6  
F459 DDC7  
F45A DDC8  
F45B DDC9  
F45C DDCA  
F45D DDCB  
F45E DDCC  
F45F DDCD  
F460 DDCE  
F461 DDCF  
F462 DDD0  
F463 DDD1  
F464 DDD2  
F465 DDD3  
F466 DDD4  
F467 DDD5  
F468 DDD6  
F469 DDD7  
F46A DDD8  
F46B DDD9  
F46C DDDA  
F46D DDDB  
F46E DDDC  
F46F DDDD  
F470 DDDE  
F471 DDDF  
F472 DDE0  
F473 DDE1  
F474 DDE2  
F475 DDE3  
F476 DDE4  
F477 DDE5  
F478 DDE6  
F479 DDE7  
F47A DDE8  
F47B DDE9  
F47C DDEA  
F47D DDEB  
F47E DDEC  
F47F DDED  
F480 DDEE  
F481 DDEF  
F482 DDF0  
F483 DDF1  
F484 DDF2  
F485 DDF3  
F486 DDF4  
F487 DDF5  
F488 DDF6  
F489 DDF7  
F48A DDF8  
F48B DDF9  
F48C DDFA

F48D DDFB  
F48E DDFC  
F48F DDFD  
F490 DE41  
F491 DE42  
F492 DE43  
F493 DE44  
F494 DE45  
F495 DE46  
F496 DE47  
F497 DE48  
F498 DE49  
F499 DE4A  
F49A DE4B  
F49B DE4C  
F49C DE4D  
F49D DE4E  
F49E DE4F  
F49F DE50  
F4A0 DE51  
F4A1 DE52  
F4A2 DE53  
F4A3 DE54  
F4A4 DE55  
F4A5 DE56  
F4A6 DE57  
F4A7 DE58  
F4A8 DE59  
F4A9 DE5A  
F4AA DE5B  
F4AB DE5C  
F4AC DE5D  
F4AD DE5E  
F4AE DE5F  
F4AF DE60  
F4B0 DE61  
F4B1 DE62  
F4B2 DE63  
F4B3 DE64  
F4B4 DE65  
F4B5 DE66  
F4B6 DE67  
F4B7 DE68  
F4B8 DE69  
F4B9 DE6A  
F4BA DE6B  
F4BB DE6C  
F4BC DE6D  
F4BD DE6E  
F4BE DE6F  
F4BF DE70  
F4C0 DE71  
F4C1 DE72  
F4C2 DE73  
F4C3 DE74  
F4C4 DE75  
F4C5 DE76  
F4C6 DE77  
F4C7 DE78  
F4C8 DE79  
F4C9 DE7A  
F4CA DE7B  
F4CB DE7C  
F4CC DE7D  
F4CD DE7E  
F4CE DE7F  
F4CF DE81  
F4D0 DE82  
F4D1 DE83  
F4D2 DE84  
F4D3 DE85  
F4D4 DE86  
F4D5 DE87  
F4D6 DE88  
F4D7 DE89  
F4D8 DE8A  
F4D9 DE8B  
F4DA DE8C  
F4DB DE8D  
F4DC DE8E  
F4DD DE8F  
F4DE DE90

F4DF DE91  
F4E0 DE92  
F4E1 DE93  
F4E2 DE94  
F4E3 DE95  
F4E4 DE96  
F4E5 DE97  
F4E6 DE98  
F4E7 DE99  
F4E8 DE9A  
F4E9 DE9B  
F4EA DE9C  
F4EB DE9D  
F4EC DE9E  
F4ED DE9F  
F4EE DEA0  
F4EF DEA1  
F4F0 DEA2  
F4F1 DEA3  
F4F2 DEA4  
F4F3 DEA5  
F4F4 DEA6  
F4F5 DEA7  
F4F6 DEA8  
F4F7 DEA9  
F4F8 DEAA  
F4F9 DEAB  
F4FA DEAC  
F4FB DEAD  
F4FC DEAE  
F4FD DEAF  
F4FE DEB0  
F4FF DEB1  
F500 DEB2  
F501 DEB3  
F502 DEB4  
F503 DEB5  
F504 DEB6  
F505 DEB7  
F506 DEB8  
F507 DEB9  
F508 DEBA  
F509 DEBB  
F50A DEBC  
F50B DEBD  
F50C DEBE  
F50D DEBF  
F50E DEC0  
F50F DEC1  
F510 DEC2  
F511 DEC3  
F512 DEC4  
F513 DEC5  
F514 DEC6  
F515 DEC7  
F516 DEC8  
F517 DEC9  
F518 DECA  
F519 DECB  
F51A DECC  
F51B DECD  
F51C DECE  
F51D DECF  
F51E DED0  
F51F DED1  
F520 DED2  
F521 DED3  
F522 DED4  
F523 DED5  
F524 DED6  
F525 DED7  
F526 DED8  
F527 DED9  
F528 DEDA  
F529 DEDB  
F52A DEDC  
F52B DEDD  
F52C DEDE  
F52D DEDF  
F52E DEE0  
F52F DEE1  
F530 DEE2

F531 DEE3  
F532 DEE4  
F533 DEE5  
F534 DEE6  
F535 DEE7  
F536 DEE8  
F537 DEE9  
F538 DEEA  
F539 DEEB  
F53A DEEC  
F53B DEED  
F53C DEEE  
F53D DEEF  
F53E DEF0  
F53F DEF1  
F540 DEF2  
F541 DEF3  
F542 DEF4  
F543 DEF5  
F544 DEF6  
F545 DEF7  
F546 DEF8  
F547 DEF9  
F548 DEFA  
F549 DEFB  
F54A DEFC  
F54B DEFD  
F54C DF41  
F54D DF42  
F54E DF43  
F54F DF44  
F550 DF45  
F551 DF46  
F552 DF47  
F553 DF48  
F554 DF49  
F555 DF4A  
F556 DF4B  
F557 DF4C  
F558 DF4D  
F559 DF4E  
F55A DF4F  
F55B DF50  
F55C DF51  
F55D DF52  
F55E DF53  
F55F DF54  
F560 DF55  
F561 DF56  
F562 DF57  
F563 DF58  
F564 DF59  
F565 DF5A  
F566 DF5B  
F567 DF5C  
F568 DF5D  
F569 DF5E  
F56A DF5F  
F56B DF60  
F56C DF61  
F56D DF62  
F56E DF63  
F56F DF64  
F570 DF65  
F571 DF66  
F572 DF67  
F573 DF68  
F574 DF69  
F575 DF6A  
F576 DF6B  
F577 DF6C  
F578 DF6D  
F579 DF6E  
F57A DF6F  
F57B DF70  
F57C DF71  
F57D DF72  
F57E DF73  
F57F DF74  
F580 DF75  
F581 DF76  
F582 DF77

F583 DF78  
F584 DF79  
F585 DF7A  
F586 DF7B  
F587 DF7C  
F588 DF7D  
F589 DF7E  
F58A DF7F  
F58B DF81  
F58C DF82  
F58D DF83  
F58E DF84  
F58F DF85  
F590 DF86  
F591 DF87  
F592 DF88  
F593 DF89  
F594 DF8A  
F595 DF8B  
F596 DF8C  
F597 DF8D  
F598 DF8E  
F599 DF8F  
F59A DF90  
F59B DF91  
F59C DF92  
F59D DF93  
F59E DF94  
F59F DF95  
F5A0 DF96  
F5A1 DF97  
F5A2 DF98  
F5A3 DF99  
F5A4 DF9A  
F5A5 DF9B  
F5A6 DF9C  
F5A7 DF9D  
F5A8 DF9E  
F5A9 DF9F  
F5AA DFA0  
F5AB DFA1  
F5AC DFA2  
F5AD DFA3  
F5AE DFA4  
F5AF DFA5  
F5B0 DFA6  
F5B1 DFA7  
F5B2 DFA8  
F5B3 DFA9  
F5B4 DFAA  
F5B5 DFAB  
F5B6 DFAC  
F5B7 DFAD  
F5B8 DFAE  
F5B9 DFAF  
F5BA DFB0  
F5BB DFB1  
F5BC DFB2  
F5BD DFB3  
F5BE DFB4  
F5BF DFB5  
F5C0 DFB6  
F5C1 DFB7  
F5C2 DFB8  
F5C3 DFB9  
F5C4 DFBA  
F5C5 DFBB  
F5C6 DFBC  
F5C7 DFBD  
F5C8 DFBE  
F5C9 DFBF  
F5CA DFC0  
F5CB DFC1  
F5CC DFC2  
F5CD DFC3  
F5CE DFC4  
F5CF DFC5  
F5D0 DFC6  
F5D1 DFC7  
F5D2 DFC8  
F5D3 DFC9  
F5D4 DFCA



F5D5 DFCE  
F5D6 DFCC  
F5D7 DFCD  
F5D8 DFCE  
F5D9 DFCE  
F5DA DFD0  
F5DB DFD1  
F5DC DFD2  
F5DD DFD3  
F5DE DFD4  
F5DF DFD5  
F5E0 DFD6  
F5E1 DFD7  
F5E2 DFD8  
F5E3 DFD9  
F5E4 DFDA  
F5E5 DFDB  
F5E6 DFDC  
F5E7 DFDD  
F5E8 DFDE  
F5E9 DFDF  
F5EA DFE0  
F5EB DFE1  
F5EC DFE2  
F5ED DFE3  
F5EE DFE4  
F5EF DFE5  
F5F0 DFE6  
F5F1 DFE7  
F5F2 DFE8  
F5F3 DFE9  
F5F4 DFEA  
F5F5 DFEB  
F5F6 DFEC  
F5F7 DFED  
F5F8 DFEE  
F5F9 DFEE  
F5FA DFF0  
F5FB DFF1  
F5FC DFF2  
F5FD DFF3  
F5FE DFF4  
F5FF DFF5  
F600 DFF6  
F601 DFF7  
F602 DFF8  
F603 DFF9  
F604 DFFA  
F605 DFFB  
F606 DFFC  
F607 DFFD  
F608 E041  
F609 E042  
F60A E043  
F60B E044  
F60C E045  
F60D E046  
F60E E047  
F60F E048  
F610 E049  
F611 E04A  
F612 E04B  
F613 E04C  
F614 E04D  
F615 E04E  
F616 E04F  
F617 E050  
F618 E051  
F619 E052  
F61A E053  
F61B E054  
F61C E055  
F61D E056  
F61E E057  
F61F E058  
F620 E059  
F621 E05A  
F622 E05B  
F623 E05C  
F624 E05D  
F625 E05E  
F626 E05F

F627 E060  
F628 E061  
F629 E062  
F62A E063  
F62B E064  
F62C E065  
F62D E066  
F62E E067  
F62F E068  
F630 E069  
F631 E06A  
F632 E06B  
F633 E06C  
F634 E06D  
F635 E06E  
F636 E06F  
F637 E070  
F638 E071  
F639 E072  
F63A E073  
F63B E074  
F63C E075  
F63D E076  
F63E E077  
F63F E078  
F640 E079  
F641 E07A  
F642 E07B  
F643 E07C  
F644 E07D  
F645 E07E  
F646 E07F  
F647 E081  
F648 E082  
F649 E083  
F64A E084  
F64B E085  
F64C E086  
F64D E087  
F64E E088  
F64F E089  
F650 E08A  
F651 E08B  
F652 E08C  
F653 E08D  
F654 E08E  
F655 E08F  
F656 E090  
F657 E091  
F658 E092  
F659 E093  
F65A E094  
F65B E095  
F65C E096  
F65D E097  
F65E E098  
F65F E099  
F660 E09A  
F661 E09B  
F662 E09C  
F663 E09D  
F664 E09E  
F665 E09F  
F666 E0A0  
F667 E0A1  
F668 E0A2  
F669 E0A3  
F66A E0A4  
F66B E0A5  
F66C E0A6  
F66D E0A7  
F66E E0A8  
F66F E0A9  
F670 E0AA  
F671 E0AB  
F672 E0AC  
F673 E0AD  
F674 E0AE  
F675 E0AF  
F676 E0B0  
F677 E0B1  
F678 E0B2

F679 E0B3  
F67A E0B4  
F67B E0B5  
F67C E0B6  
F67D E0B7  
F67E E0B8  
F67F E0B9  
F680 E0BA  
F681 E0BB  
F682 E0BC  
F683 E0BD  
F684 E0BE  
F685 E0BF  
F686 E0C0  
F687 E0C1  
F688 E0C2  
F689 E0C3  
F68A E0C4  
F68B E0C5  
F68C E0C6  
F68D E0C7  
F68E E0C8  
F68F E0C9  
F690 E0CA  
F691 E0CB  
F692 E0CC  
F693 E0CD  
F694 E0CE  
F695 E0CF  
F696 E0D0  
F697 E0D1  
F698 E0D2  
F699 E0D3  
F69A E0D4  
F69B E0D5  
F69C E0D6  
F69D E0D7  
F69E E0D8  
F69F E0D9  
F6A0 E0DA  
F6A1 E0DB  
F6A2 E0DC  
F6A3 E0DD  
F6A4 E0DE  
F6A5 E0DF  
F6A6 E0E0  
F6A7 E0E1  
F6A8 E0E2  
F6A9 E0E3  
F6AA E0E4  
F6AB E0E5  
F6AC E0E6  
F6AD E0E7  
F6AE E0E8  
F6AF E0E9  
F6B0 E0EA  
F6B1 E0EB  
F6B2 E0EC  
F6B3 E0ED  
F6B4 E0EE  
F6B5 E0EF  
F6B6 E0F0  
F6B7 E0F1  
F6B8 E0F2  
F6B9 E0F3  
F6BA E0F4  
F6BB E0F5  
F6BC E0F6  
F6BD E0F7  
F6BE E0F8  
F6BF E0F9  
F6C0 E0FA  
F6C1 E0FB  
F6C2 E0FC  
F6C3 E0FD  
F6C4 E141  
F6C5 E142  
F6C6 E143  
F6C7 E144  
F6C8 E145  
F6C9 E146  
F6CA E147

F6CB	E148
F6CC	E149
F6CD	E14A
F6CE	E14B
F6CF	E14C
F6D0	E14D
F6D1	E14E
F6D2	E14F
F6D3	E150
F6D4	E151
F6D5	E152
F6D6	E153
F6D7	E154
F6D8	E155
F6D9	E156
F6DA	E157
F6DB	E158
F6DC	E159
F6DD	E15A
F6DE	E15B
F6DF	E15C
F6E0	E15D
F6E1	E15E
F6E2	E15F
F6E3	E160
F6E4	E161
F6E5	E162
F6E6	E163
F6E7	E164
F6E8	E165
F6E9	E166
F6EA	E167
F6EB	E168
F6EC	E169
F6ED	E16A
F6EE	E16B
F6EF	E16C
F6F0	E16D
F6F1	E16E
F6F2	E16F
F6F3	E170
F6F4	E171
F6F5	E172
F6F6	E173
F6F7	E174
F6F8	E175
F6F9	E176
F6FA	E177
F6FB	E178
F6FC	E179
F6FD	E17A
F6FE	E17B
F6FF	E17C
F700	E17D
F701	E17E
F702	E17F
F703	E181
F704	E182
F705	E183
F706	E184
F707	E185
F708	E186
F709	E187
F70A	E188
F70B	E189
F70C	E18A
F70D	E18B
F70E	E18C
F70F	E18D
F710	E18E
F711	E18F
F712	E190
F713	E191
F714	E192
F715	E193
F716	E194
F717	E195
F718	E196
F719	E197
F71A	E198
F71B	E199
F71C	E19A

F71D E19B  
F71E E19C  
F71F E19D  
F720 E19E  
F721 E19F  
F722 E1A0  
F723 E1A1  
F724 E1A2  
F725 E1A3  
F726 E1A4  
F727 E1A5  
F728 E1A6  
F729 E1A7  
F72A E1A8  
F72B E1A9  
F72C E1AA  
F72D E1AB  
F72E E1AC  
F72F E1AD  
F730 E1AE  
F731 E1AF  
F732 E1B0  
F733 E1B1  
F734 E1B2  
F735 E1B3  
F736 E1B4  
F737 E1B5  
F738 E1B6  
F739 E1B7  
F73A E1B8  
F73B E1B9  
F73C E1BA  
F73D E1BB  
F73E E1BC  
F73F E1BD  
F740 E1BE  
F741 E1BF  
F742 E1C0  
F743 E1C1  
F744 E1C2  
F745 E1C3  
F746 E1C4  
F747 E1C5  
F748 E1C6  
F749 E1C7  
F74A E1C8  
F74B E1C9  
F74C E1CA  
F74D E1CB  
F74E E1CC  
F74F E1CD  
F750 E1CE  
F751 E1CF  
F752 E1D0  
F753 E1D1  
F754 E1D2  
F755 E1D3  
F756 E1D4  
F757 E1D5  
F758 E1D6  
F759 E1D7  
F75A E1D8  
F75B E1D9  
F75C E1DA  
F75D E1DB  
F75E E1DC  
F75F E1DD  
F760 E1DE  
F761 E1DF  
F762 E1E0  
F763 E1E1  
F764 E1E2  
F765 E1E3  
F766 E1E4  
F767 E1E5  
F768 E1E6  
F769 E1E7  
F76A E1E8  
F76B E1E9  
F76C E1EA  
F76D E1EB  
F76E E1EC

F76F	E1ED
F770	E1EE
F771	E1EF
F772	E1F0
F773	E1F1
F774	E1F2
F775	E1F3
F776	E1F4
F777	E1F5
F778	E1F6
F779	E1F7
F77A	E1F8
F77B	E1F9
F77C	E1FA
F77D	E1FB
F77E	E1FC
F77F	E1FD
F780	E241
F781	E242
F782	E243
F783	E244
F784	E245
F785	E246
F786	E247
F787	E248
F788	E249
F789	E24A
F78A	E24B
F78B	E24C
F78C	E24D
F78D	E24E
F78E	E24F
F78F	E250
F790	E251
F791	E252
F792	E253
F793	E254
F794	E255
F795	E256
F796	E257
F797	E258
F798	E259
F799	E25A
F79A	E25B
F79B	E25C
F79C	E25D
F79D	E25E
F79E	E25F
F79F	E260
F7A0	E261
F7A1	E262
F7A2	E263
F7A3	E264
F7A4	E265
F7A5	E266
F7A6	E267
F7A7	E268
F7A8	E269
F7A9	E26A
F7AA	E26B
F7AB	E26C
F7AC	E26D
F7AD	E26E
F7AE	E26F
F7AF	E270
F7B0	E271
F7B1	E272
F7B2	E273
F7B3	E274
F7B4	E275
F7B5	E276
F7B6	E277
F7B7	E278
F7B8	E279
F7B9	E27A
F7BA	E27B
F7BB	E27C
F7BC	E27D
F7BD	E27E
F7BE	E27F
F7BF	E281
F7C0	E282

F7C1	E283
F7C2	E284
F7C3	E285
F7C4	E286
F7C5	E287
F7C6	E288
F7C7	E289
F7C8	E28A
F7C9	E28B
F7CA	E28C
F7CB	E28D
F7CC	E28E
F7CD	E28F
F7CE	E290
F7CF	E291
F7D0	E292
F7D1	E293
F7D2	E294
F7D3	E295
F7D4	E296
F7D5	E297
F7D6	E298
F7D7	E299
F7D8	E29A
F7D9	E29B
F7DA	E29C
F7DB	E29D
F7DC	E29E
F7DD	E29F
F7DE	E2A0
F7DF	E2A1
F7E0	E2A2
F7E1	E2A3
F7E2	E2A4
F7E3	E2A5
F7E4	E2A6
F7E5	E2A7
F7E6	E2A8
F7E7	E2A9
F7E8	E2AA
F7E9	E2AB
F7EA	E2AC
F7EB	E2AD
F7EC	E2AE
F7ED	E2AF
F7EE	E2B0
F7EF	E2B1
F7F0	E2B2
F7F1	E2B3
F7F2	E2B4
F7F3	E2B5
F7F4	E2B6
F7F5	E2B7
F7F6	E2B8
F7F7	E2B9
F7F8	E2BA
F7F9	E2BB
F7FA	E2BC
F7FB	E2BD
F7FC	E2BE
F7FD	E2BF
F7FE	E2C0
F7FF	E2C1
F800	E2C2
F801	E2C3
F802	E2C4
F803	E2C5
F804	E2C6
F805	E2C7
F806	E2C8
F807	E2C9
F808	E2CA
F809	E2CB
F80A	E2CC
F80B	E2CD
F80C	E2CE
F80D	E2CF
F80E	E2D0
F80F	E2D1
F810	E2D2
F811	E2D3
F812	E2D4

F813	E2D5
F814	E2D6
F815	E2D7
F816	E2D8
F817	E2D9
F818	E2DA
F819	E2DB
F81A	E2DC
F81B	E2DD
F81C	E2DE
F81D	E2DF
F81E	E2E0
F81F	E2E1
F820	E2E2
F821	E2E3
F822	E2E4
F823	E2E5
F824	E2E6
F825	E2E7
F826	E2E8
F827	E2E9
F828	E2EA
F829	E2EB
F82A	E2EC
F82B	E2ED
F82C	E2EE
F82D	E2EF
F82E	E2F0
F82F	E2F1
F830	E2F2
F831	E2F3
F832	E2F4
F833	E2F5
F834	E2F6
F835	E2F7
F836	E2F8
F837	E2F9
F838	E2FA
F839	E2FB
F83A	E2FC
F83B	E2FD
F83E	445C
F83F	46A8
F840	46A9
F841	46AA
F842	46AB

## Code license and disclaimer information

---

IBM grants you a nonexclusive copyright license to use all programming code examples from which you can generate similar function tailored to your own specific needs.

SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM, ITS PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

1. LOSS OF, OR DAMAGE TO, DATA;
2. DIRECT, SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC CONSEQUENTIAL DAMAGES; OR
3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF DIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.



## Notices

---

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
Software Interoperability Coordinator, Department YBWA  
3605 Highway 52 N  
Rochester, MN 55901  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. \_enter the year or years\_.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

## Programming interface information

---

This Installing, upgrading, or deleting IBM i and related software publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM i.

## Trademarks

---

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks and logos are trademarks of Oracle, Inc. in the United States, other countries, or both.

Other product and service names might be trademarks of IBM or other companies.

## Terms and conditions

---

Permissions for the use of these publications is granted subject to the following terms and conditions.

**Personal Use:** You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

**Commercial Use:** You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.



## Terms and conditions

---

Permissions for the use of these publications is granted subject to the following terms and conditions.

**Personal Use:** You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

**Commercial Use:** You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.





