



System i
Database
Database administration

Version 6 Release 1





System i
Database
Database administration

Version 6 Release 1

Note

Before using this information and the product it supports, read the information in "Notices," on page 11.

This edition applies to version 6, release 1, modification 0 of IBM i5/OS (product number 5761-SS1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

© Copyright International Business Machines Corporation 1998, 2008.

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

Contents

Database administration	1	Ensuring data integrity	5
What's new for V6R1	1	Importing and exporting data between systems	5
PDF file for Database administration	1	Working with multiple databases	5
Database administration	2	Working with triggers and constraints	6
Accessing data through client interfaces	2	Writing DB2 programs	7
Accessing data with Java	2	Database backup and recovery	7
Accessing data with Domino	2	Distributed database administration	7
Accessing data with ODBC	2	Queries and reports	7
Accessing data with i5/OS PASE	2	BINARY, VARBINARY, HEX, and LENGTH	
Accessing data with System i Access for		functions for Query	8
Windows OLE DB Provider	2	Security	10
Accessing data with Net.Data.	3	Appendix. Notices	11
Accessing data through a Linux partition	3	Programming interface information	12
Accessing data using Distributed Relational		Trademarks	13
Database Architecture (DRDA)	3	Terms and conditions	13
Altering and managing database objects	3		
Creating database objects	4		

Database administration

DB2[®] for i5/OS[®] provides database administration, backup and recovery, query, and security functions.

You can also explore other database information using the main navigation tree or Database information finder.

What's new for V6R1

Read about new or significantly changed information for the Database administration topic collection.

Accessing data through client interfaces

The “Accessing data through client interfaces” on page 2 topic has been moved from the SQL programming topic collection to this topic collection. This topic was originally entitled Using dynamic SQL through client interfaces.

How to see what's new or changed

To help you see where technical changes have been made, the information center uses:

- The  image to mark where new or changed information begins.
- The  image to mark where new or changed information ends.

In PDF files, you might see revision bars (|) in the left margin of new and changed information.

To find other information about what's new or changed this release, see the Memo to users.

PDF file for Database administration

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

To view or download the PDF version of this document, select Database administration (about 270 KB).

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 (www.adobe.com/products/acrobat/readstep.html)  .

Database administration

DB2 for i5/OS provides various methods for setting up and managing databases.

Related concepts

Journal management

Accessing data through client interfaces

You can access DB2 for i5/OS data through client interfaces on the server, such as the Java Database Connectivity (JDBC) driver, the Open Database Connectivity (ODBC) driver, i5/OS Portable Application Solutions Environment (i5/OS PASE), OLE DB Provider, Net.Data[®], or Distributed Relational Database Architecture[™] (DRDA[®]).

Accessing data with Java

You can access DB2 for i5/OS data in your Java[™] programs by using the Java Database Connectivity (JDBC) driver that is included with the IBM[®] Developer Kit for Java licensed program.

The driver allows you to perform the following tasks:

- Access database files.
- Access JDBC database functions with embedded Structured Query Language (SQL) for Java.
- Run SQL statements and process results.

Related concepts

Accessing your System i5 database with the IBM Developer Kit for Java JDBC driver

Accessing data with Domino

You can use IBM Lotus[®] Domino[®] for i5/OS to integrate data from DB2 for i5/OS databases and Domino databases in both directions.

To take advantage of this integration, you need to understand and manage how authorizations work between the two types of databases.

Related concepts

Lotus Domino for i5/OS

Accessing data with ODBC

You use the System i[®] Access for Windows[®] Open Database Connectivity (ODBC) driver to enable your ODBC client applications to effectively share data with each other and with the server.

Related concepts

ODBC administration

Accessing data with i5/OS PASE

IBM i5/OS Portable Application Solutions Environment (i5/OS PASE) is an integrated runtime environment for AIX[®], UNIX[®], or other applications that are running on the i5/OS operating system. i5/OS PASE supports the DB2 for i5/OS call level interface (CLI).

Related concepts

Database

Accessing data with System i Access for Windows OLE DB Provider

The System i Access for Windows OLE DB Provider, along with the Programmer's Toolkit, facilitates the System i client/server application development from the Microsoft[®] Windows client PC.

- | The System i Access for Windows OLE DB Provider gives programmers record-level access interfaces to
- | DB2 for i5/OS database files. In addition, it provides support for SQL, data queues, programs, and
- | commands.

Related reference

System i Access for Windows OLE DB Provider

Accessing data with Net.Data

Net.Data is an application that runs on a server. You can use Net.Data to easily create dynamic Web documents that are called Web macros. Web macros that are created for Net.Data have the simplicity of HTML with the functionality of CGI-BIN applications.

Net.Data makes it easy to add live data to static Web pages. Live data includes information that is stored in databases, files, applications, and system services.

Related concepts

Net.Data applications for the HTTP Server

Accessing data through a Linux partition

IBM and a variety of Linux[®] distributors have cooperated to integrate the Linux operating system with the reliability of the System i architecture.

Linux brings a new generation of Web-based applications to the System i product. IBM has changed the Linux PowerPC[®] kernel to run in a secondary logical partition and contributed the kernel back to the Linux community.

Related concepts

Linux in a logical partition

Accessing data using Distributed Relational Database Architecture (DRDA)

A *distributed relational database* consists of a set of SQL objects that are spread across interconnected computer systems. Each relational database has a relational database manager to manage the tables in its environment.

The database managers communicate and cooperate with each other in a way that allows a given database manager access to run SQL statements on a relational database on another system.

Related reference

Distributed relational database function and SQL

Altering and managing database objects

DB2 for i5/OS provides both Structured Query Language (SQL) and system methods for altering and managing database objects.

Several methods are available for working with database objects. You can use the System i Navigator interface, SQL methods, or the traditional system interface. The following table shows the available options for each task. Click the appropriate X for more information about performing the task.

The System i Navigator tasks are documented in the online help.

Task	System i Navigator	SQL	Traditional system interface
Adding a column to a table	X	X	X
Adding or altering an identity column	X	X	
Altering a sequence	X	X	
Changing a table (file) definition	X	X	X

Task	System i Navigator	SQL	Traditional system interface
Changing the schema (library) list	X		X
Copying a table (file)	X		X
Copying column definitions	X		
Displaying contents of tables and views (files)	X	X	X
Displaying locked rows (records)	X		X
Displaying table (file) attributes (catalog)	X	X	X
Dropping database objects	X	X	X
Editing table (file) data	X	X	X
Moving a table (file)	X		X
Reorganizing a table (physical file member)	X		X

Related concepts

System i Navigator database tasks

Related reference

Terminology: SQL versus traditional file access

Creating database objects

The first step in developing your database is to create the objects that hold your data. You can create tables, views, and indexes with SQL. You can also create physical and logical files using the traditional system interface.

You can create database objects using System i Navigator, SQL, or the traditional system interface. The following table shows the available options for each task. Click the appropriate **X** for more information about performing the task.

The System i Navigator tasks are documented in the online help.

Task	System i Navigator	SQL	Traditional system interface
Creating a map of database object relationships	X		
Creating an alias	X	X	X
Creating an index or access path	X	X	X
Creating an object using Run SQL Scripts	X		
Creating a package	X	X	X
Creating a schema (library)	X	X	
Creating a sequence	X	X	
Creating a stored procedure	X	X	
Creating a table (physical file)	X	X	X

Task	System i Navigator	SQL	Traditional system interface
Creating a user-defined function	X	X	
Creating a user-defined type	X	X	
Creating a view (logical file)	X	X	X
Generating SQL for database objects	X		

Related concepts

System i Navigator database tasks

Related reference

Terminology: SQL versus traditional file access

Ensuring data integrity

DB2 for i5/OS provides several integrity measures, such as constraints, trigger programs, and commitment control.

Constraints, triggers, and commitment control can protect your database against inadvertent insertions, deletions, and updates. Constraints basically govern how data values can change, while triggers are automatic actions that start, or *trigger*, an event, such as an update of a specific table.

Related concepts

Commitment control

“Working with triggers and constraints” on page 6

You can use triggers or constraints to manage data in your database tables.

Importing and exporting data between systems

Importing data is the process of retrieving data from external sources, while *exporting data* is the process of extracting data from DB2 for i5/OS and copying data to another system.

Importing data into DB2 for i5/OS can be a one-time event or it can be an ongoing task, like weekly updates for business reporting purposes. These types of data move operations are typically accomplished through import, export, or load functions.

Related concepts

Copying a file

Copying files

Copying source file data

Moving a file

Related tasks

Importing and exporting data

Loading and unloading data from systems other than System i

Working with multiple databases

The system provides a system database (identified as *SYSBAS*) and the ability to work with one or more user databases.

User databases are implemented through the use of independent disk pools, which are set up in the disk management function of System i Navigator. After an independent disk pool is set up, it appears as another database in the Databases folder of System i Navigator.

When you expand a system in System i Navigator and then expand **Databases**, a list of databases that you can work with is shown. To establish a connection to a database, expand the database that you want to work with.

Related concepts

Disk management

Working with triggers and constraints

You can use triggers or constraints to manage data in your database tables.

A *trigger* is a type of stored procedure program that is automatically called whenever a specified action is performed on a specific table. Triggers are useful for keeping audit trails, detecting exceptional conditions, maintaining relationships in the database, and running applications and operations that coincide with the change operation.

A *constraint* is a restriction or limitation that you place on your database. Constraints are implemented at the table level. You can use constraints to create referential integrity in your database.

You can work with triggers and constraints using System i Navigator, SQL, or the traditional system interface. The following table shows the available options for each task. Click the appropriate **X** for more information about performing the task.

The System i Navigator tasks are also documented in the online help.

Task	System i Navigator	SQL	Traditional system interface
Adding an external trigger	X		X
Adding an SQL trigger	X	X	
Creating a check constraint	X	X	X
Creating a key constraint	X	X	X
Creating a referential constraint	X	X	X
Enabling and disabling a referential constraint	X		X
Enabling and disabling a trigger	X		X
Managing check pending constraints	X		
Removing a check constraint	X	X	X
Removing a key constraint	X	X	X
Removing a referential constraint	X	X	X
Removing a trigger	X	X	X
Writing an external trigger program	X		X
Writing an SQL trigger program	X	X	

Related concepts

System i Navigator database tasks

Writing DB2 programs

DB2 for i5/OS provides various methods for writing applications that access or update data.

You can write embedded SQL programs, external functions, external procedures, DB2 for i5/OS CLI applications, and trigger programs.

Related concepts

Embedded SQL programming

Writing a DB2 for i5/OS CLI application

Related tasks

Creating trigger programs

Related reference

Defining an external procedure

Writing UDFs as external functions

Database backup and recovery

Saving your data can be time-consuming and requires discipline. However, it is crucial that you back up your data because you never know when you might need to recover it.

Related concepts

Backup and recovery

Journal management

Recovering and restoring your database

Distributed database administration

With DB2 for i5/OS, you can work with databases that are distributed across several systems.

Related concepts

Distributed database programming

Queries and reports

You can use SQL, the Open Query File (OPNQRYF) command, the Query (QQQQRY) API, Open Database Connectivity (ODBC), or the IBM Query for i5/OS licensed program to create and run queries.

One of the most common tasks that you perform with your database is to retrieve information. The system provides several methods to create and run queries and reports.

You can use an SQL statement to retrieve information. This SQL statement is called a *query*. The query searches the tables stored in your database to find the answer to the question that you posed with your SQL statement. The answer is expressed as a set of rows, which is referred to as the result set. After a query has been run, you can also create a report to display the data provided in your result set.

In addition to using SQL, you can use other functions and products to create and run queries and reports. See the following information for details.

- IBM DB2 Web Query for System i overview
- Query for i5/OS 

- Query Management Programming 
- Query Manager Use 

In addition, you can build SELECT, INSERT, UPDATE, and DELETE SQL statements in the SQL Assist window of System i Navigator.

Related concepts

SQL programming

Related tasks

Building SQL statements with SQL Assist

Related reference

Open Query File (OPNQRYF) command

Query (QQQQRY) API

BINARY, VARBINARY, HEX, and LENGTH functions for Query

This topic describes the BINARY, VARBINARY, HEX, and LENGTH built-in functions that are supported by the IBM Query for i5/OS licensed program.

Note: In addition to the BINARY, VARBINARY, HEX, and LENGTH functions, the Query for i5/OS licensed program provides the following support:

- Support for BINARY, VARBINARY, BLOB, CLOB, DBCLOB, and ROWID data types.
- Support for larger numeric and decimal numbers and for large numeric literals.
- The VARCHAR function supports conversion from CLOB to VARCHAR.
- The VARGRAPHIC function supports conversion from CLOB to DBCS graphic, CLOB to UCS2 graphic, DBCLOB to DBCS graphic, and DBCLOB to UCS2 graphic.

A binary constant is used for comparing a literal with a binary field (BINARY, VARBINARY, or BLOB). Binary constants are represented with an X followed by a sequence of characters that starts and ends with a string delimiter. The characters between the string delimiters must be an even number of hexadecimal digits. A hexadecimal digit is a digit or any of the letters A through F (uppercase and lowercase), as shown in the following example:

Field	Test	Value
binarycol	eq	X'12AF'

BINARY built-in function

The BINARY function returns a BINARY representation of a string of any type. The form is:

►► BINARY (*string-expression* [, *integer*]) ►►

The result of the function is a fixed-length binary string. If the first argument can be null, the result can be null; if the first argument is null, the result is the null value.

The first argument must be a string-expression whose value must be a built-in character string, graphic string, binary string, or row ID data type.

The second argument specifies the length attribute for the resulting binary string. The value must be between 1 and 32766. If the second argument is not specified, the following rules apply:

- If the string-expression is the empty string constant, the length attribute of the result is 1.

- Otherwise, the length attribute of the result is the same as the length attribute of the first argument, unless the argument is a graphic string. In this case, the length attribute of the result is twice the length attribute of the argument.

The actual length is the same as the length attribute of the result. If the length of the string-expression is less than the length of the result, the result is padded with hexadecimal zeros up to the length of the result. If the length of the string-expression is greater than the length attribute of the result, truncation is performed.

VARBINARY built-in function

The VARBINARY function returns a VARBINARY representation of a string of any type. The form is:

►►—VARBINARY—(—*string-expression*—, *integer*)—

The result of the function is VARBINARY. If the first argument can be null, the result can be null; if the first argument is null, the result is the null value.

The first argument is a string-expression whose value can be a character string, graphic string, binary string, or row ID.

The second argument specifies the length attribute for the resulting binary string. The value must be between 1 and 32740 (32739 if nullable). If the second argument is not specified, the following rules apply:

- If the string-expression is the empty string constant, the length attribute of the result is 1.
- Otherwise, the length attribute of the result is the same as the length attribute of the first argument, unless the argument is a graphic string. In this case, the length attribute of the result is twice the length attribute of the argument.

The actual length is the same as the length attribute of the result. If the length of the string-expression is less than the length of the result, the result is padded with hexadecimal zeros up to the length of the result. If the length of the string-expression is greater than the length attribute of the result, truncation is performed.

HEX built-in function

The HEX function returns a hexadecimal representation of a value. The form is:

►►—HEX—(—*expression*—)

The argument can be of any built-in data type. The result of the function is a character string. If the argument can be null, the result can be null; if the argument is null, the result is the null value.

The result is a string of hexadecimal digits. The first two digits represent the first byte of the argument, the next two digits represent the second byte of the argument, and so forth. If the argument is a datetime value, the result is the hexadecimal representation of the internal form of the argument.

The length attribute of the result is twice the storage length attribute of the argument. The length attribute of the result cannot be greater than 32766 for fixed-length results or greater than 32740 for varying-length results. If the argument is a varying-length string, the result is a varying-length string. Otherwise, the result is a fixed-length string.

The coded character set identifier (CCSID) of the string is the default single-byte character set (CCSID) on the current system.

LENGTH built-in function

The LENGTH function returns the length of a value. The form is:

►►—LENGTH—(*expression*)—◄◄

The argument must be an expression that returns a value of any built-in data type. The result of the function is a large integer. If the argument can be null, the result can be null; if the argument is null, the result is the null value.

The result is the length of the argument. The length of strings includes blanks. The length of a varying-length string is the actual length, not the length attribute. The length of a graphic string is the number of double-byte characters (the number of bytes divided by 2). The length of all other values is the number of bytes used to represent the value:

- 2 for small integer
- 4 for large integer
- 8 for big integer
- The integral part of $(p/2)+1$ for packed decimal numbers with precision p
- p for zoned decimal numbers with precision p
- 4 for single-precision float
- 8 for double-precision float
- The length of the string for strings
- 3 for time
- 4 for date
- 10 for timestamp
- 26 for row ID

Security

Authorizing users to data at the system and data levels allows you to control access to your database.

Securing your database requires you to establish ownership and public authority to objects and specific authority to your applications.

Related concepts

DRDA server access control exit programs

Granting file and data authority

Limiting access to specific fields in a database file

Security

Specifying public authority

Using database file capabilities to control I/O operations

Using logical files to secure data

Appendix. 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:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, 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,
- | IBM License Agreement for Machine Code, 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.

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.

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_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

| **Programming interface information**

This Database administration publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM i5/OS.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX
DB2
Distributed Relational Database Architecture
Domino
DRDA
i5/OS
IBM
IBM (logo)
Lotus
Net.Data
PowerPC
System i
System i5

- | 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.
- | 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.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

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.



Printed in USA