DBMaker

JDBA Tool User's Guide

CASEMaker®

CASEMaker Inc./Corporate Headquarters

1680 Civic Center Drive

Santa Clara, CA 95050, U.S.A.

www.casemaker.com

www.casemaker.com/support

©Copyright 1995-2026 by CASEMaker Inc.

Document No. 645049-244048/DBM547-M09202026-JDBA

Publication Date: 2026-09-20

All rights reserved. No part of this manual may be reproduced, stored in a retrieval system, or transmitted in any form, without the prior written permission of the manufacturer.

For a description of updated functions that do not appear in this manual, read the file named README.TXT after installing the CASEMaker DBMaker software.

Trademarks

CASEMaker, the CASEMaker logo, and DBMaker are registered trademarks of CASEMaker Inc. Microsoft, MS-DOS, Windows, and Windows NT are registered trademarks of Microsoft Corp. UNIX is a registered trademark of The Open Group. ANSI is a registered trademark of American National Standards Institute, Inc.

Other product names mentioned herein may be trademarks of their respective holders and are mentioned only form information purposes. SQL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

Notices

The software described in this manual is covered by the license agreement supplied with the software.

Contact your dealer for warranty details. Your dealer makes no representations or warranties with respect to the merchantability or fitness of this computer product for any particular purpose. Your dealer is not responsible for any damage caused to this computer product by external forces including sudden shock, excess heat, cold, or humidity, nor for any loss or damage caused by incorrect voltage or incompatible hardware and/or software.

Information in this manual has been carefully checked for reliability; however, no responsibility is assumed for inaccuracies. This manual is subject to change without notice.

Contents

1 Introduction 1-1

1.1 Additional Resources 1-2

1.2 Technical Support 1-3

1.3 Document Conventions 1-4

2 Getting to know JDBA Tool 2-1

2.1 Summary of Features 2-2

2.2 Explanation of Privilege 2-4

2.3 About the JDBA Tool Workspace 2-5

Tool Bar 2-6

Menu Bar 2-7

The Database in Tree View 2-8

Main Console 2-8

2.4 Connecting to a Database 2-9

2.5 Disconnecting from a Database 2-12

3 Working with Tablespaces 3-1

3.1 Browsing Tablespaces 3-2

3.2 Creating a Tablespace 3-4

3.3 Altering a Tablespace 3-8

©Copyright 1995-2026 CASEMaker Inc.

Adding Files to a Tablespace....3-8
Shrinking and Compressing a Tablespace....3-10
Modifying a File in a Tablespace....3-12
Changing Tablespace Modes....3-13
Changing the Read Only Mode....3-14
Updating Statistics in a Tablespace....3-14
3.4 Altering multiple tables to another tablespace....3-16
3.5 Altering multiple indexes to another tablespace....3-18
3.6 Dropping a Datafile from a Tablespace....3-21
3.7 Dropping a Tablespace....3-22
4 Working with Tables....4-1
4.1 Creating a Table....4-2
Starting the Create Table Wizard....4-2
Setting Table, Tablespace and Schema Options....4-3
Setting Column Options....4-5
Setting Column and Table Constraints....4-6
Adding a UNIQUE Constraint....4-8
Adding a Foreign Key while Creating a Table....4-10
Setting Advanced Table Options....4-15
Completing Table Creation....4-17
4.2 Modifying Table Schema....4-19
Altering to another tablespace....4-20
Changing the Column Order....4-20
Changing Column Names....4-21
Changing Column Type....4-21
Changing Precision and Scale....4-22
Changing Column Null / Not Null....4-22
Changing the Primary Key....4-22
Changing the Column Default Value....4-23
Adding Column Constraints....4-24
Changing Column Encryption....4-24
Adding a Column....4-24

©Copyright 1995-2026 CASEMaker Inc.

Deleting a Column 4-25
Tracing a Table 4-26
4.3 Setting Table Properties 4-27
4.4 Granting and Revoking Table Privileges 4-29
4.5 Checking Dependencies and Statistics 4-31
Checking Dependencies 4-31
Checking and Updating Table Statistics 4-32
4.6 Editing Data in a Table 4-36
4.7 Renaming a Table 4-39
4.8 Dropping a Table 4-41
4.9 Constraint Syntax 4-42
4.10 Filtering tables 4-44
4.11 Copying tables 4-45
4.12 Duplicate tables 4-46
4.13 Truncating tables 4-48
4.14 Deleting from tables 4-49
5 Working with Views 5-1
5.1 Creating a View 5-2
5.2 Checking View Properties 5-8
5.3 Granting View Privileges 5-9
5.4 Browsing Views 5-10
5.5 Dropping a View 5-13
6 Working with Indexes 6-1
6.1 Creating an Index 6-2
6.2 Rebuilding an Index 6-7
6.3 Renaming an Index 6-8
6.4 Updating Index Statistics 6-9
6.5 Dropping an Index 6-11

©Copyright 1995-2026 CASEMaker Inc.

6.6 Disabling an Index 6-12

7 Working with Signature Text Indexes 7-1

7.1 Creating a Signature Text Index 7-2

7.2 Rebuilding a Signature Text Index 7-7

7.3 Dropping a Signature Text Index 7-9

7.4 Checking Signature Text Index Statistics 7-10

8 Working with IVF Text Indexes 8-1

8.1 Creating an IVF Text Index 8-2

8.2 Rebuilding an IVF Text Index 8-7

8.3 Dropping an IVF Text Index 8-9

8.4 Checking IVF Text Index Statistics 8-11

9 Using Foreign Keys 9-1

9.1 Creating a Foreign Key 9-2

Adding a Foreign Key 9-2

Setting Foreign Key Options 9-5

9.2 Dropping a Foreign key 9-8

10 Using Triggers 10-1

10.1 Creating a Trigger 10-2

Assigning a Trigger Name and Table 10-2

Specifying Trigger Action Settings 10-4

Indicating the Referencing Clause 10-6

Entering the WHEN Condition Clause 10-7

Entering SQL statements for the trigger action 10-8

10.2 Modifying a Trigger 10-11

10.3 Dropping a Trigger 10-13

11 Working with Stored Commands 11-1

11.1 Creating a Stored Command 11-2

11.2 Executing a Stored Command 11-4

©Copyright 1995-2026 CASEMaker Inc.

11.3 Rebuilding a Stored Command 11-7
11.4 Granting Stored Command Privileges 11-9
11.5 Dropping a Stored Command 11-11

12 Working with Stored Procedures 12-1

12.1 Creating a Stored Procedure 12-2
12.2 Executing a Stored Procedure 12-8
12.3 Granting Privileges on Stored Procedures 12-10
12.4 Dropping a Stored Procedure 12-12
12.5 Add a Jar File for Java Stored Procedure 12-14
12.6 Dropping Jar Files 12-17

13 Working with Functions 13-1

13.1 Creating a Function 13-2
13.2 Dropping a Function 13-6

14 Working with Database Links 14-1

14.1 Creating a Database Link 14-2
14.2 Dropping a Database Link 14-4

15 Working with Replication Schedule 15-1

15.1 Creating a Replication Schedule 15-2
15.2 Modifying a Replication Schedule 15-8
15.3 Dropping a Replication Schedule 15-10

16 Managing Table Replication 16-1

16.1 Synchronous Table Replication 16-3
Adding Synchronous Table Replication 16-3
Adding Subscribers to a Synchronous Replication 16-13
16.2 Asynchronous Table Replication 16-17
Adding Asynchronous Table Replication 16-17
Adding Subscribers to an Asynchronous Replication ..16-26

©Copyright 1995-2026 CASEMaker Inc.

16.3 Dropping a Subscriber 16-31
16.4 Dropping a Replication..... 16-33
17 Working with Domains 17-1
17.1 Creatng a Domain..... 17-2
17.2 Dropping a Domain 17-5
18 Working with Synonyms.....18-1
18.1 Creating a Synonym 18-2
18.2 Dropping a Synonym 18-4
19 Managing Users 19-1
19.1 Creating a New User Account..... 19-2
19.2 Changing Security Level 19-6
Setting Authority Level 19-6
Granting and Removing Object Privileges..... 19-8
19.3 Changing a User's Password..... 19-12
19.4 Browsing Assigned Schema 19-13
19.5 Deleting a User Account 19-14
20 Managing Groups.....20-1
20.1 Creating a New Group 20-2
20.2 Adding and Removing Group Members 20-6
20.3 Managing Group Object Privileges 20-8
21 Working with Schema.....21-1
21.1 Browsing Schema 21-2
21.2 Creating a Schema 21-3
21.3 Dropping a Schema 21-6
22 Managing Access Control List.....22-1
22.1 Browsing Access Control List..... 22-2
22.2 Creating Access Control List..... 22-3

©Copyright 1995-2026 CASEMaker Inc.

22.3 Altering Access Control List 22-7
22.4 Dropping Access Control List 22-10

23 SQL Query 23-1

23.1 The main functional components of SQL Query 23-2
Command Entry Area 23-3
Command Result Area 23-3
Common options 23-3

23.2 Using SQL Query 23-4
Executing Commands 23-4
Saving commands 23-5
Loading Commands 23-7
Clearing commands 23-8
Execute Batch 23-8
Preference setting 23-9
Execute commands as one SQL 23-10

23.3 Exit SQL Query 23-10

24 Data Transfer 24-1

24.1 Importing Data from Text 24-3
24.2 Importing Data from XML 24-12
24.3 Importing Data from ODBC 24-16
Importing ODBC Data from Tables 24-20
Importing ODBC Data Using SQL SELECT Statements 24-27
Importing ODBC Data through an XML Batch File 24-33
24.4 Importing Data from JSON 24-39
24.5 Exporting Data to Text 24-43
24.6 Exporting Data to XML 24-51
24.7 Exporting Data to JSON 24-62

25 Monitoring a Database 25-1

25.1 Database Monitor Tool 25-2

©Copyright 1995-2026 CASEMaker Inc.

25.2 Export to File Options 25-10
25.3 Database Monitor (Tree) 25-13
Monitoring Database Sessions 25-14
Monitoring Database Locks 25-25
Monitoring Database Status 25-28
Monitoring Pending Transactions 25-40
Monitoring System Configurations 25-42
Monitoring Update Statistics 25-43
Monitoring Journal Status 25-45

26 Setting JDBA Options 26-1

26.1 Altering Run Time Settings 26-2
Changing Backup Settings 26-4
Changing Differential and Incremental Backup Settings 26-10
Changing File Object Settings 26-13
Changing System Control Settings 26-15
Changing Log System Settings 26-18
Changing Character Set Encoding Settings 26-26
Changing Statistics Settings 26-30
Changing Auto Index Settings 26-33
Changing Schedule Daemon Settings 26-36
Changing Connection Settings 26-39
26.2 Setting Preferences 26-41
Setting General Options 26-42
Setting Table Options 26-44
Setting JSQL Options 26-46
26.3 Query by Example (QBE) 26-49
26.4 Manage Transparent Data Encryption (TDE) 26-64
Setting TDE options 26-65

Glossary 1

Index 1

©Copyright 1995-2026 CASEMaker Inc.

1 Introduction

Welcome to the JDBA Tool User's Guide. JDBA Tool is a cross-platform user-friendly graphical user interface (GUI) that helps users easily manage database objects in DBMaker, a powerful and flexible SQL Database Management System. JDBA Tool hides the complexity of the DBMS and query language and provides an easy to understand and convenient to use interface. You can perform database functions without having to learn SQL. JDBA Tool's monitoring functions also provides statistical data and information on who is using your database.

This manual gives systematic instructions for using the JDBA Tool to manage database objects in DBMaker. JDBA Tool features are first briefly described followed by procedures that guide you through proper use of the feature.

This book is for users who will design and administer DBMaker databases using JDBA Tool. It teaches people unfamiliar with using DBMaker, but who are familiar with the operation of relational databases. We assumed the user has some knowledge of their database platform. Experienced users may also find the information in this manual to be a helpful reference.

To perform JDBA Tool tasks, please refer to specific sections within the chapters. You should read any descriptions included before the procedures and follow each of the steps in the procedures. We recommended that you read other sections within the same chapter, as this will increase your understanding of JDBA Tool.

©Copyright 1995-2026 CASEMaker Inc.

1.1 Additional Resources

DBMaker provides a complete set of DBMS manuals in addition to this one. For more information on a particular subject, consult one of the books listed below:

- For an introduction to DBMaker's capabilities and functions, refer to the *DBMaker Tutorial*.
- For more information on designing, administering, and maintaining a DBMaker database, refer to the *Database Administrator's Guide*.
- For more information on database server management, refer to the *JServer Manager User's Guide*.
- For more information on configuring DBMaker, refer to the *JConfiguration Tool Reference*.
- For more information on the native ODBC API and JDBC API, refer to the *ODBC Programmer's Guide and JDBC Programmer's Guide*.
- For more information on the dmSQL interface tool, refer to the *dmSQL User's Guide*.
- For more information on the SQL language used in dmSQL, refer to the *SQL Command and Function Reference*.
- For more information on the ESQL/C programming, refer to the *ESQL/C User's Guide*.
- For more information on error and warning messages, refer to the *Error and Message Reference*.
- For more information on the DBMaker COBOL Interface, refer to the *DCI User's Guide*.

©Copyright 1995-2026 CASEMaker Inc.

1.2 Technical Support

CASEMaker provides thirty days of complimentary email and phone support during the evaluation period. When software is registered an additional thirty days of support is included extending the total support period for software to sixty days. However, CASEMaker will continue to provide email support for any bugs reported after the complimentary support or registered support has expired (free of charge).

For most products, support is available beyond sixty days and may be purchased for twenty percent of the retail price of the product. Please contact sales@casemaker.com for details and prices.

CASEMaker support contact information, by post mail, phone, or email, for your area is at: www.casemaker.com/support. We recommend searching the most current database of FAQ's before contacting CASEMaker support staff.

Please have the following information available when phoning support for a troubleshooting enquiry or include this information in your correspondence:

- Product name and version number
- Registration number
- Registered customer name and address
- Supplier/distributor where product was purchased
- Platform and computer system configuration
- Specific action(s) performed before error(s) occurred
- Error message and number, if any
- Any additional information deemed pertinent

©Copyright 1995-2026 CASEMaker Inc.

1.3 Document Conventions

This book uses a standard set of typographical conventions for clarity and ease of use. The NOTE, Procedure, Example, and CommandLine conventions also have a second setting used with indentation.

CONVENTION	DESCRIPTION
Italics	Italics indicate placeholders for information that must be supplied, such as user and table names. The word in italics should not be typed, but is replaced by the actual name. Italics also introduce new words, and are occasionally used for emphasis in text.
Boldface	Boldface indicates filenames, database names, table names, column names, user names, and other database schema objects. It is also used to emphasize menu commands in procedural steps.
KEYWORDS	All keywords used by the SQL language appear in uppercase when used in normal paragraph text.
SMALL CAPS	Small capital letters indicate keys on the keyboard. A plus sign (+) between two key names indicates to hold down the first key while pressing the second. A comma (,) between two key names indicates to release the first key before pressing the second key.
NOTE	Contains important information.
➲ Procedure	Indicates that procedural steps or sequential items will follow. Many tasks are described using this format to provide a logical sequence of steps for the user to follow
➲ Example	Examples are given to clarify descriptions, and commonly include text, as it will appear on the screen. Other forms of this convention include Prototype and Syntax.
`CommandLine`	Indicates text, as it should appear on a text-delimited screen. This format is commonly used to show input and output for dmSQL commands or the content in the dmconfig.ini file

Table 1-1Document Conventions

©Copyright 1995-2026 CASEMaker Inc.

2 Getting to know JDBA Tool

This chapter introduces the layout of the JDBA Tool workspace, and contains general information about many of its components. This chapter also describes how to connect and disconnect from a database.

JDBA Tool, like all the other DBMaker Java tools (e.g., JServer Manager and Java Configuration Tool), operates in the Java Run Time Environment. To start these programs the Java Development Kit (JDK) or the Java Run Time Environment (JRE) should be installed on the system. DBMaker release 5.4, when installed from disk, automatically installs JRE 1.7 on Windows systems.

©Copyright 1995-2026 CASEMaker Inc.

2.1 Summary of Features

JDBA Tool has a wide variety of features for effectively managing your databases.

- You can easily create table schema in a database and add columns to it. You can also define constraints for data in a table.
- After creating the table schema and adding columns, you can insert records to a table. Moreover, you can define a constraint to ensure that each inserted record falls within a specified range.
- You can define a column using a domain. The column inherits the properties of the domain (e.g., data type, default value, and value constraint) so explicit specification is not required.
- After successfully creating a table, it is easy to create an index, primary key, or foreign key in a table. Foreign key establish relationships between different tables in a database.
- DBMaker provides easily querying by creating a view from various tables or specifying a synonym for a specific table. Views and synonyms let you refer to and manipulate data from tables quickly and easily.
- After creating all database objects, you can easily drop a table, view, synonym, or domain. Please consider the mutual relationships between objects before dropping some objects. For example, drop foreign keys before dropping the referenced primary keys.
- Any authorized user can create database links to access database objects on remote databases.
- Use the Table Replication command to transfer table data between local and remote table objects.
- Frequently used SQL commands are easily managed by creating stored commands that can be reused whenever needed.

©Copyright 1995-2026 CASEMaker Inc.

- Stored procedures can help make performing frequently repeated tasks even easily. Thus, eliminating the need for repeated SQL compilation and optimization.
- DBMaker allow user setting one connection's priority and affinity to improve the performance of the whole system.
- Triggers can handle complex transactions in the database.Triggers can automatically execute when the user inserts, deletes, or updates data in the database.
- A trigger actions can call stored procedures.
- Text indices can be created for columns with CHAR, VARCHAR, LONG VARCHAR, and FILE data types.
- System administrator can add users or groups to a database. After adding users and groups to a database, privileges can be assigned to users and groups. Groups provide an efficient method for assigning identical privileges to many users (i.e., all of the users in the group).
- Monitor the current system environment such as which database users are currently connected to the database, locks placed on data objects, and all current runtime settings.
- You can set default values for creating tables and choose to display BLOB data using SQL statements.

©Copyright 1995-2026 CASEMaker Inc.

2.2 Explanation of Privilege

At present, JDBA Tool supports five levels of authority to access database: CONNECT, RESOURCE, DBA, SYSDBA and SYSADM. Users who exist but are not granted RESOURCE, DBA, SYSDBA or SYSADM authority will have CONNECT authority by default. The user interface and the operative function are different for users with different authorities. SYSADM supports all functions, but SYSDBA, DBA, RESOURCE and CONNECT support part of the functions. By default, samples in the following chapters base on the user with DBA authority.

©Copyright 1995-2026 CASEMaker Inc.

2.3 About the JDBA Tool Workspace

Figure 2-1 shows the information required for logging on to the JDBA Tool. We recommend configuring all database settings before logging into the database. The database settings are changed using the configuration tool. The **Setup** button launches the JConfiguration Tool. Enter the **Database Name**, **User Name** and **Password** and click the **Setup** button to check or modify the database's advanced settings.

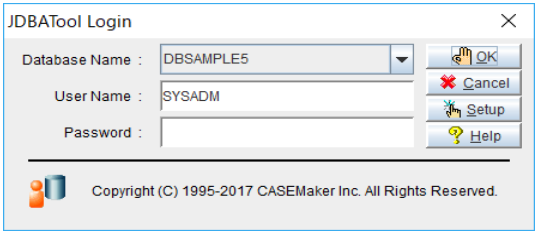


Figure 2-1 JDBA Tool Login window

For details regarding settings, please refer to the *JConfiguration Tool Reference*. After login, the elements of the GUI appear on the screen. These elements are summarized in Figure 2-2 and described in the following subsections.

©Copyright 1995-2026 CASEMaker Inc.

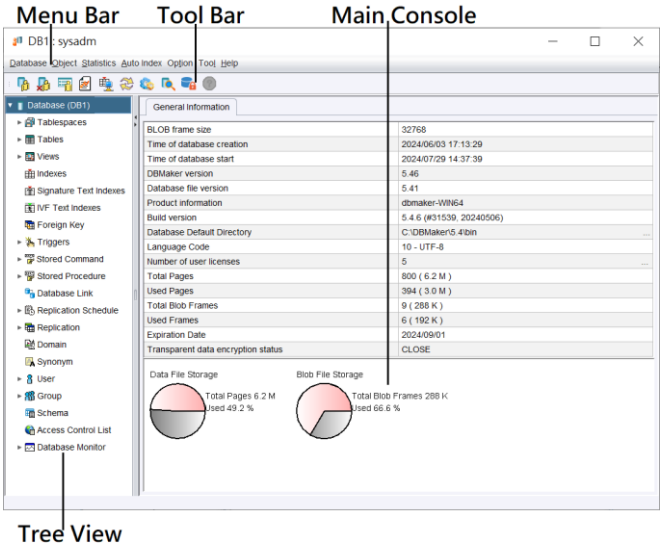


Figure 2-2 Elements of the JDBA Tool Graphical User Interface

Tool Bar

The icons and usages about the toolbar are shown as follows:

Connect Database - Select the database you want to connect to

Disconnect Database - Select the database you want to disconnect from

Create - Select the object that you want to create, such as tablespaces, tables, and so on

Update Statistics - Select automatically and periodically recalculating the statistics for database

Sync Auto Index - Synchronous Update the Auto Index

Refresh - Refresh the database information

Preference - Setting the general parameters that you need. ex, language Choice, Auto Refresh time, Stored Procedure's recursive Depth, the number of records displayed per page and so on)

©Copyright 1995-2026 CASEMaker Inc.

Query - Enter the SQL query command in the command input area and see the command execution results in the command result area

Manage Transparent Data Encryption (TDE) – Manage Transparent Data Encryption settings. This icon only appears when database is encrypted

Help - Called out the JDBA TOOL User's Guide

Users with DBA or higher authority will see information listed in the above image when access to the database, Users with RESOURCE and CONNECT authority will only see the following general information when access to the database:

- DBMaker version
- File version
- Database default directory
- Language code
- Number of users who have permission to access the database

At the same time, when users right-click on the root node in the database, **Check Database** and **Check Catalog** menu items should be disable.

Menu Bar

The menu bar displays JDBA Tool's pull-down menus. Each menu performs various functions.

Users with DBA authority or higher can use all menu items.

There are five list items under **Object** menu item can't be used by users with RESOURCE authority. The five list items are **Create tablespace**, **Create user**, **Create group**, **Create schedule for replication**, and **Create access control list**. In addition, **Statistics** menu item, **Sync Auto Index** menu under **Auto Index** menu item, **Run time setting** menu under **Option** menu item and **Auto refresh for database monitor** check box in the **User preferences** dialog also can't be used.

©Copyright 1995-2026 CASEMaker Inc.

Users with CONNECT authority cannot use not only menu items beyond RESOURCE authority, but also all options under **Object** menu item.

The Database in Tree View

The database tree displays all the objects in the database. The root of the tree is the name of the database, the rest of the objects follow down the tree. The name of each object is also displayed. With the help of the database monitor in the tree, the user can monitor the database and keep track of users and their transactions.

Users with DBA authority or higher can display all objects in the database. However, **Replication schedule**, **Group**, **Database Monitor** tree nodes and all its sub tree nodes will not be shown for users with RESOURCE or CONNECT authority, in addition, **Tablespace** tree nodes and all its sub tree nodes also will not be shown for users with CONNECT authority.

Single-clicking an object in the tree view will open that page in the primary tab and replace the existing content. Double-clicking an object in the tree will open a new tab in the main console.

When focusing in an object, user can use type to search. For example, in the **Tables** tree node, user can search for a table by typing part of table name. Please notice that the searching rule uses contain, and will search the item matches from the current position.

Main Console

The main console is an area to show the opened tab from the tree. The leftmost tab is the primary tab, this tab cannot be closed and displays database information by default. Single-clicking the item loads the content in the primary tab instead of opening a new tab, if the item is already opened in the main console, single-clicking will focus on that tab. Double-clicking the item opens a new tab in the main console.

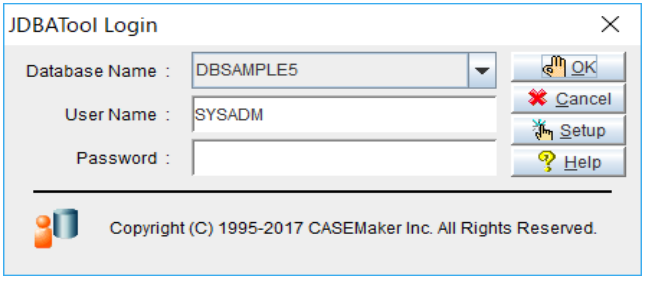
©Copyright 1995-2026 CASEMaker Inc.

2.4 Connecting to a Database

To manipulate data in a database, you must first connect (i.e., login) to the database. JDBA Tool allows users to connect to multiple databases, include single database and multi-user database. To connect to a multi-user database with JDBA Tool, a database server program must already be running. The database can be started with JServer Manager or DBMaker Server. Once started, proceed with the following steps to connect to the database.

➲ Connecting to a database:

1. Select a database from the **Database Name** menu in the **JDBA Tool Login** window.



2. Enter your user ID in the **User Name** field, user ID has three user privilege.

   NOTE *If a default user ID has already been set, then the default user ID is automatically displayed in the User Name field.*

3. If you have been assigned a password, enter it in the **Password** field.
4. Click the **Setup button** to open the **Connect Database Advanced Setting** window.
5. Set options in the following categories:

   - Connection
   - DmServer
   - Data Manipulation

- SQL Attributes

NOTE *Refer to the JConfiguration Tool Reference for more information about advanced settings.*

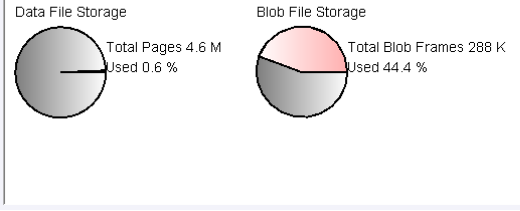
6. Click the **OK button** after you have finished setting the options. The **General Information** window appears next and displays the following information:

- BLOB frame size
- Time of database creation
- Time of database start
- DBMaker version
- Database file version
- Product information
- Build version
- Database Default Directory
- Language Code
- Number of user licenses
- Total Pages
- Used Pages
- Total Blob Frames
- Used Frames
- Expiration Date
- Transparent data encryption status

©Copyright 1995-2026 CASEMaker Inc.

General Information

BLOB frame size	32768
Time of database creation	2024/05/06 16:20:59
Time of database start	2024/05/09 16:51:10
DBMaker version	5.46
Database file version	5.41
Product information	dbmaker-WIN64
Build version	5.4.6 (#31512, 20240423)
Database Default Directory	C:\DBMaker\5.4\bin ...
Language Code	10 - UTF-8
Number of user licenses	5 ...
Total Pages	600 (4.6 M)
Used Pages	4 (32 K)
Total Blob Frames	9 (288 K)
Used Frames	4 (128 K)
Expiration Date	2024/08/04
Transparent data encryption status	CLOSE



NOTE *A graphical representation of the Data and BLOB File storage is also displayed.*

7. If you wish to connect to another database, click the **Database** pull-down menu and select **Connect Database**. Repeat steps 1 through 4.

NOTE *Databases must be started by the server program before you can connect to them.*

©Copyright 1995-2026 CASEMaker Inc.

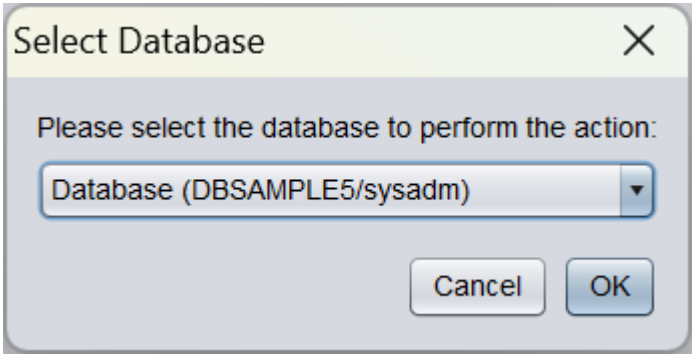
2.5 Disconnecting from a Database

Once you have finished, you can disconnect from the database.

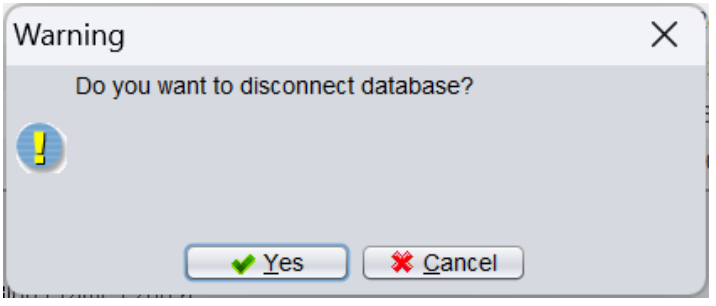
NOTE *Since JDBA Tool works in an AUTOCOMMIT mode, all database transactions are automatically written to the disk before disconnecting from the database.*

➲ Disconnecting from a database:

1. Select **Disconnect Database** from the **Database** menu, or click the button. If user is connecting to multiple databases, the following window will appear. If user is connecting to one database, skip to step3.



2. Select the database to disconnect. The following warning box will appear.



3. Click **YES** to disconnect database

©Copyright 1995-2026 CASEMaker Inc.

3 Working with Tablespaces

DBMaker databases are composed of one or more logical divisions known as *tablespaces*. Tablespaces help divide the database into manageable areas. In the logical view, a tablespace contains one or more tables and indexes. In the physical view, a tablespace is the physical storage (i.e., directory path) that consists of one or more files. By the proper use of tablespaces, data is better organized both at the physical and logical level, and storage space is used more efficiently.

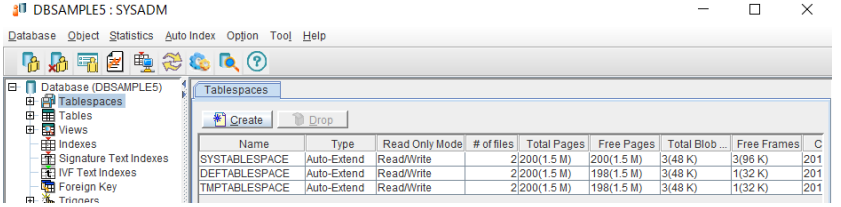
©Copyright 1995-2026 CASEMaker Inc.

3.1 Browsing Tablespaces

All DBMaker databases have at least three tablespaces. The first is the system tablespace (SYSTABLESPACE), the second is the default user tablespace (DEFTABLESPACE), and the third is the temporary tablespace (TMPTABLESPACE). Whenever you create a database, DBMaker generates a system tablespace to record the *system catalog tables*. The system catalog tables store information about the entire database. The default tablespace (DEFTABLESPACE) stores user tables if users do not define their own tablespace. The temporary tablespace (TMPTABLESPACE) is only used to store external temp tables (ETT). You can view information about all the tablespaces in a database by selecting the **Tablespace** item from the tree.

➲ Browsing a tablespace:

1. Click the **Tablespace** object in the tree to display a list of all tablespaces.



2. Browse the tablespaces listed in the right panel using the scroll bar. All of the tablespace information appears in following fields.

   - **Name** – tablespace name
   - **Type** – tablespaces can be regular or autoextend
   - **Read Only Mode** – specifies whether a tablespace is Read-Only or Read/Write. Read-only tablespaces do not allow adding new objects or editing existing ones.
   - **# of files** – total number of data files and BLOB files in the tablespace

©Copyright 1995-2026 CASEMaker Inc.

- **Total Pages** – number of pages in data files
- **Free Pages** – number of free pages in data files
- **Total Blob Frames** – number of frames in BLOB files
- **Free Frames** – number of free frames in BLOB files
- **Creation Time** – tablespace creation time

©Copyright 1995-2026 CASEMaker Inc.

3.2 Creating a Tablespace

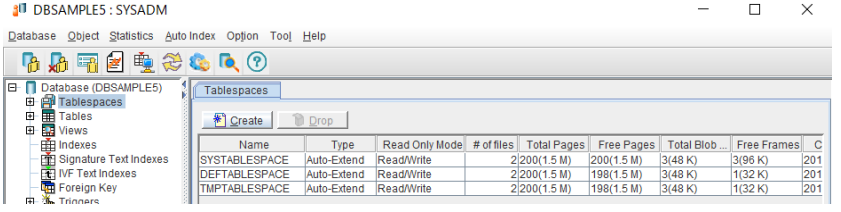
Smaller tables that contain data of a similar nature can be grouped in a single tablespace. Alternatively, very large tables should be placed in their own tablespaces. A regular tablespace has a fixed size and contains one or more data files. If the physical file of a regular tablespace becomes insufficient, you must enlarge it manually. An autoextend tablespace, given this same situation, automatically grows as needed. By default, users with DBA authority or higher have CREATE privilege for the tablespace.

Consider the following when creating a tablespace:

- The logical name must be unique
- Files all have a file type of either DATA or BLOB
- The size of DATA files is measured and reported in pages. The size of BLOB files is measured and reported in frames.
- Tablespaces must contain at least one BLOB file and at least one DATA file. Tablespace creation fails when at least one DATA file is not included.

➲ Creating a tablespace:

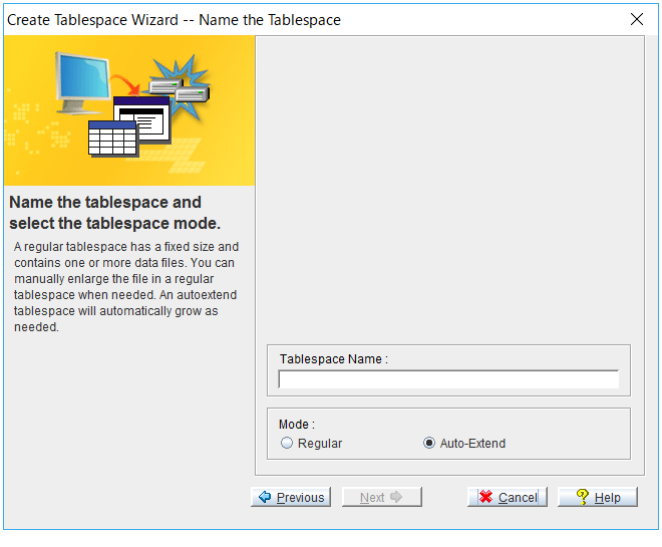
1. Click the **Tablespaces** object in the tree to display the **Tablespaces** page.



©Copyright 1995-2026 CASEMaker Inc.

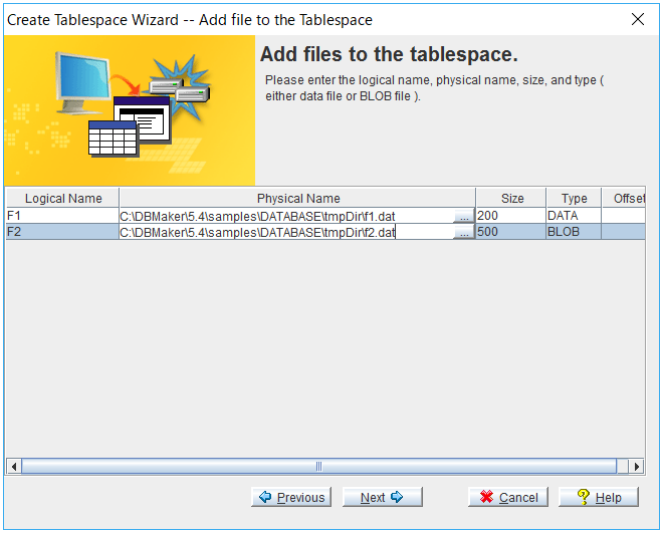
2. Click the **Create** button in the **Tablespaces** window. The **Create Tablespace Wizard** window appears.

3. Click the **Next** button. The **Name the Tablespace** window appears.



©Copyright 1995-2026 CASEMaker Inc.

4. Enter the name for the tablespace in the **Tablespace Name** field.
5. Select a tablespace type:
   - To select a regular tablespace, select the **Regular** option button.
   - To select an Auto-Extend tablespace, select the **Auto-Extend** option button.
6. Click the **Next** button. The **Add file to the Tablespace** window appears.



7. Enter a logical name for the tablespace in the **Logical Name** column.
8. Enter a physical name for the tablespace in the **Physical Name** column.

   NOTE *You can also select the physical name of the tablespace from a list by clicking the Browse button.*

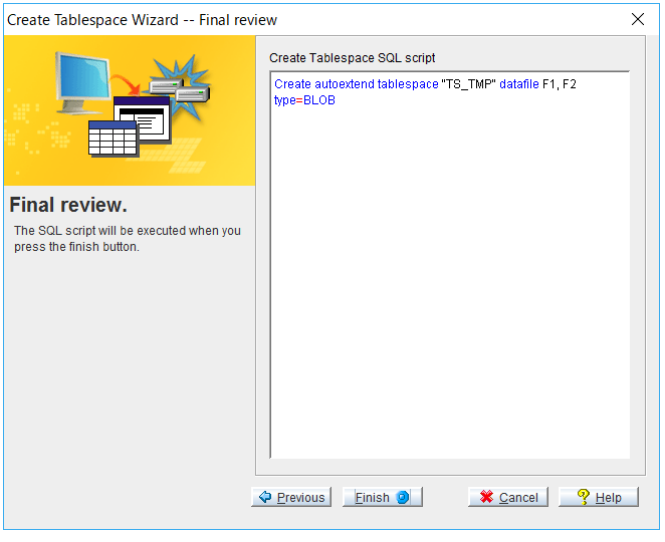
9. Enter a size for the tablespace in the **Size** column.
10. Select a tablespace type from the **Type** column.
11. Set a file offset for raw devices in the **Offset** column.

    NOTE *This option is only for databases running in a UNIX environment.*

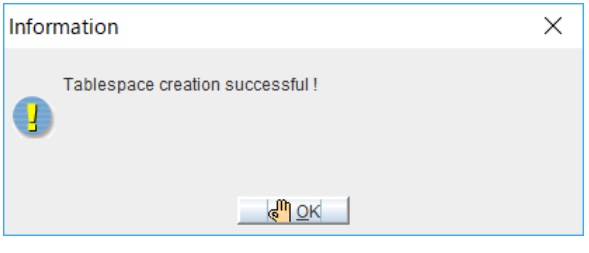
12. Click the **Next** button to proceed. Next, the SQL command for creating

©Copyright 1995-2026 CASEMaker Inc.

the new tablespace appears. Users can edit this command.



13. Edit the command by entering or deleting command syntax in the command line area.
14. Click the **Finish** button after you have finished making an optional edits to the SQL command. The **Information** window of the **Create Tablespace Wizard** appears.



©Copyright 1995-2026 CASEMaker Inc.

3.3 Altering a Tablespace

It may be necessary to alter a tablespace after its creation. Tablespaces can be alterted in the following ways:

- Add DATA or BLOB files
- Compression
- Add pages to files (in the tablespace)
- Change from autoextend to regular or vice versa
- Change from read only to read/write or vice versa

Adding Files to a Tablespace

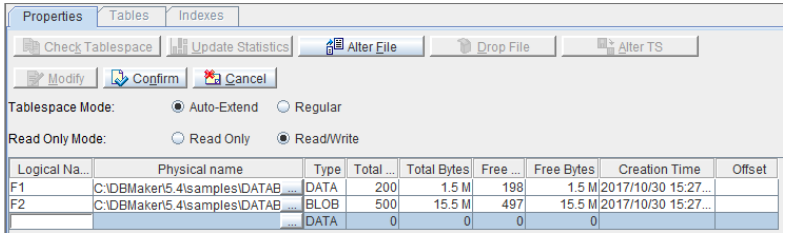
Once you have created a tablespace, you can later add DATA or BLOB files to the tablespace. Please consider the following when adding files to an existing tablespace:

- The logical name must be unique name
- There are two types of files available: DATA or BLOB
- DATA file sizes are measured and reported in pages. BLOB file sizes are measured and reported in frames.

➲ **Adding files to a tablespace:**

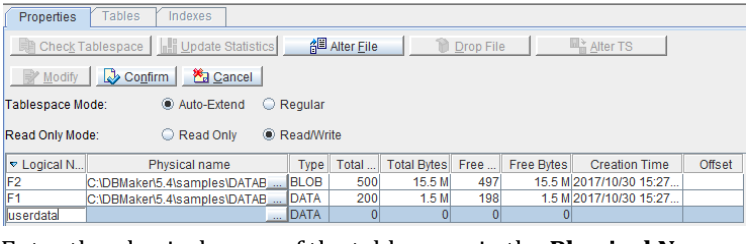
1. Select the tablespace to which you want to add one or more files. The properties for the tablespace appear in the main console.
2. Click the **Modify** button. An empty row appears in the tablespace list.

©Copyright 1995-2026 CASEMaker Inc.

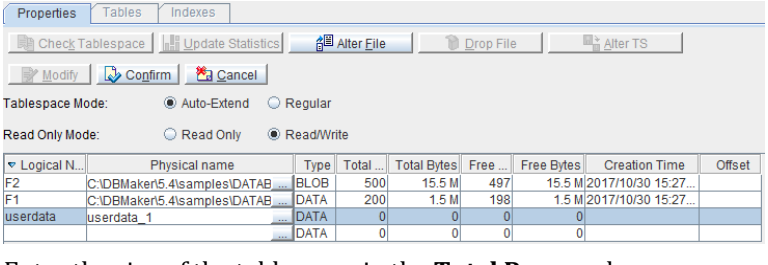


NOTE *You can check the index directly via the tablespace information panel.*

3. To change the **Tablespace Mode** from Auto-Extend to Regular, select the **Regular** option button above the list.
4. To change the **Tablespace Mode** from Regular to Autoextend, select the **Auto-Extend** option button above the list.
5. To change the **Read Only Mode** from Read Only to Read/Write, select the **Read/Write** option button above the list.
6. To change the **Read Only Mode** from Read/Write to Read-Only, select the **Read Only** option button above the list.
7. Enter the logical name of the tablespace in the **Logical Name** column.

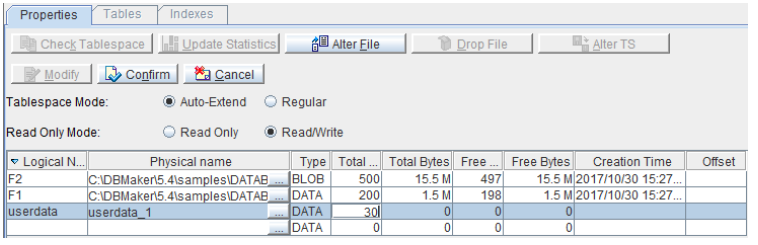


8. Enter the physical name of the tablespace in the **Physical Name** column.

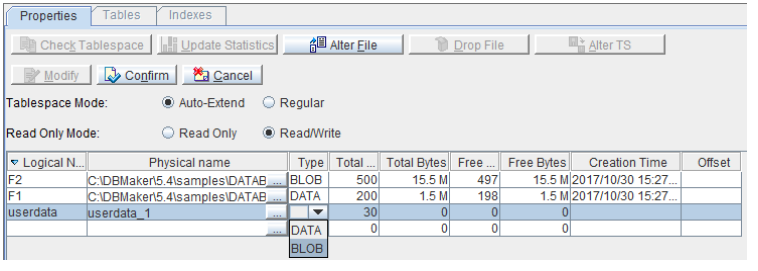


9. Enter the size of the tablespace in the **Total Pages** column.

©Copyright 1995-2026 CASEMaker Inc.



10. Select the tablespace type from the **Type** column.



11. Click the **Confirm** button. The list of files in the tablespace now includes the newly added file.

Shrinking and Compressing a Tablespace

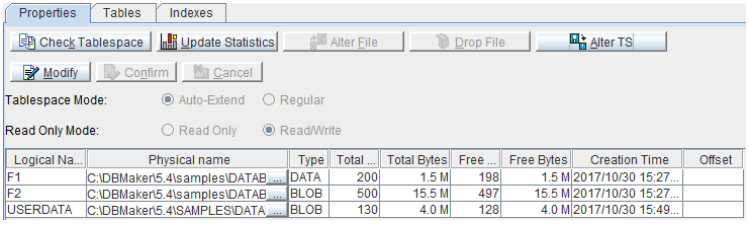
Two ways to optimize storage space are shrinking and compressing tablespaces. Compressing the tablespace moves the DATA pages and BLOB frames to the beginning of the file. When shrinking, the tablespace reduces by the number of free pages remaining in each of the files in the tablespace.

The desired number of tailing free pages contained in the tablespace should be less than the number of free pages being replaced. Replacing the number of tailing free pages in a tablespace altered the total number of free pages in each of the files in the tablespace.

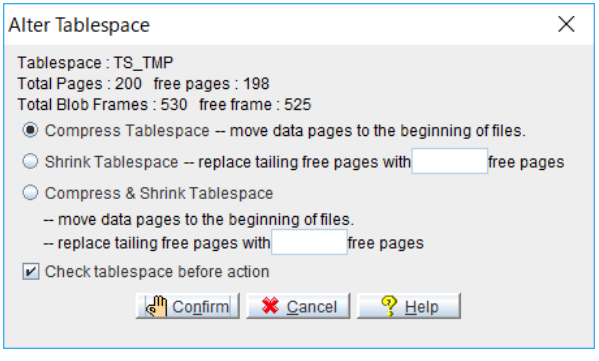
➲ Shrinking and/or compressing a tablespace:

1. Select the tablespace that you want to compress or shrink. Next, the **Properties** page appears.

©Copyright 1995-2026 CASEMaker Inc.



2. Click the **Alter TS** button located toward the top of the window. The **Alter Tablespace** dialog box appears.



3. To compress the tablespace, click the **Compress Tablespace** radio button.
4. To shrink the tablespace, click the **Shrink Tablespace** radio button. This option reduces the tailing or remaining free pages in tablespace files.
5. If you selected Shrink Tablespace in step 4, then it is necessary to enter the desired number of tailing free pages into the **free pages** text field thereby reducing the tailing or remaining free pages in the files of the tablespace.
6. To compress and shrink the tablespace, click the **Compress and Shrink Tablespace** radio button. Next, enter the number of tailing free pages you want the tablespace to contain in the **free pages** text field.
7. Click the **Check tablespace before action** checkbox to execute a consistency check of the database. Do this before applying changes to the tablespace.
8. Click the **Confirm** button. The **Alter Tablespace** confirmation dialog box appears to confirm the successful modifications.

©Copyright 1995-2026 CASEMaker Inc.

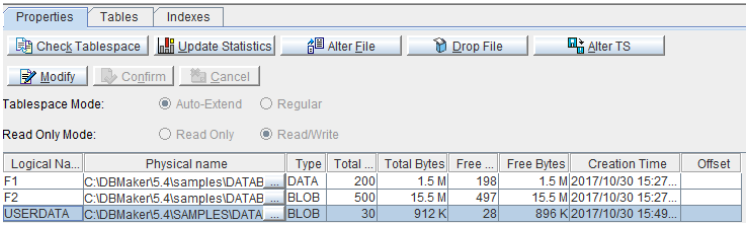
9. Click the **OK** button.

Modifying a File in a Tablespace

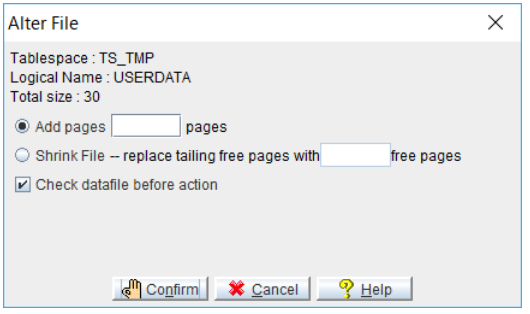
After adding files to a tablespace, it may be necessary to change a particular file in the tablespace or alter the tablespace itself.

➲ Modifying a single file in a tablespace:

1. Select the tablespace from the tree or the **Tablespaces** page that contains the file you want to modify. The properties of the tablespace appear in the main window.



2. Click the **Alter File** button. The **Alter File** dialog box appears.



NOTE *The Alter File button is disabled until a file is highlighted in the Tablespace. The highlighted file is the target to be altered.*

3. Enter into the **pages** text field the number of pages you want to add to the file.
4. If you want to shrink the file by reducing the tailing free pages, select the **Shrink File** radio button and enter the desired number of tailing free pages into the **free pages** text field.

©Copyright 1995-2026 CASEMaker Inc.

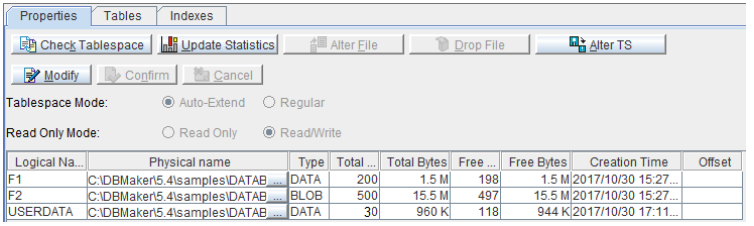
5. Click the **Check datafile before action** check box if you want to check the consistency of the data file before altering the file.
6. Click the **Confirm** button. A confirmation message appears verifying the file modification.
7. Click the **OK** button to dismiss the message dialog window.

Changing Tablespace Modes

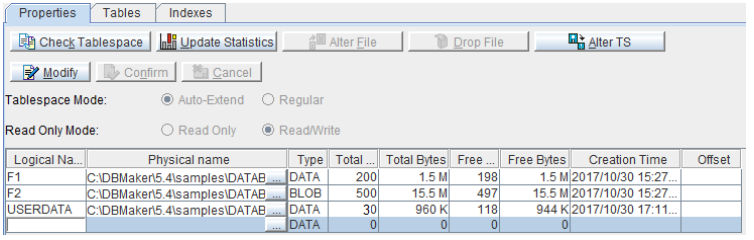
DBMaker supports changing the mode of a tablespace from autoextend to regular or from regular to autoextend. Regular tablespaces generally provide better performance but require close management to prevent them from filling too quickly. Autoextend tablespaces are easier to manage but may become fragmented over time.

➲ Changing the tablespace's mode:

1. Select the tablespace to which you want change the mode. The properties for the tablespace appear in the main console.



2. Click the **Modify** button.



3. Select the desired tablespace mode: **Auto-Extend** or **Regular**.
4. Click the **Confirm** button.

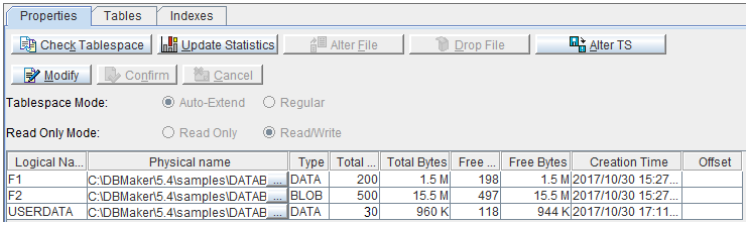
©Copyright 1995-2026 CASEMaker Inc.

Changing the Read Only Mode

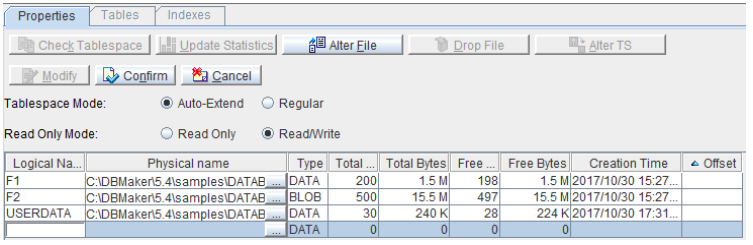
Change the **Read Only Mode** from Read Only to Read/Write or from Read/Write to Read-Only is supported in DBMaker. The Read Only tablespace does not allow any updates or creations of new objects in the tablespace. The **SYSTABLESPACE** cannot be set to **Read Only** mode.

➲ Changing the Read Only mode:

1. Select the tablespace to which you want change. The properties for the tablespace appears in the main console.



2. Click the **Modify** button.



3. Select the desired Read Only Mode: **Read Only** or **Read/Write**.
4. Click the **Confirm** button**.**

Updating Statistics in a Tablespace

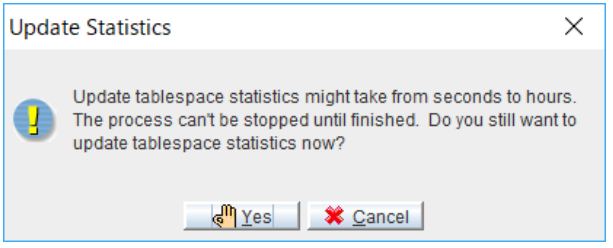
Updating a tablespace's statistics is a significant part of an optimize strategy for tablespace performance.The Sample Ratio value specifies the percentage of data in the object that is used to calculate and update the statistics of the object. If nothing is entered, the default ratio is 100. The sample ratio and the

©Copyright 1995-2026 CASEMaker Inc.

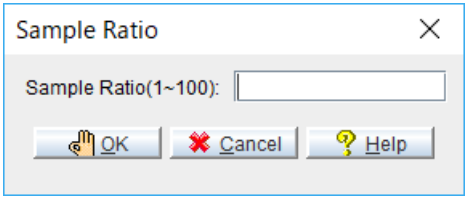
amount of data in the database determine the time needed for updating the tablespace's statistics.

➲ Updating statistics in a tablespace:

1. Select the tablespace you want to update the statistics of from the tree or the tablespaces page. The **Properties** page appears.
2. Click the **Update Statistics** button. The Update Statistics confirmation dialog box appears.



3. Click the **Yes** button, the **Sample Ratio** dialog box appears.



4. Enter a sample ratio in the range of 1 to 100.
5. Click the **OK** button. The statistics of the tablespace are updated.

©Copyright 1995-2026 CASEMaker Inc.

3.4 Altering multiple tables to another tablespace

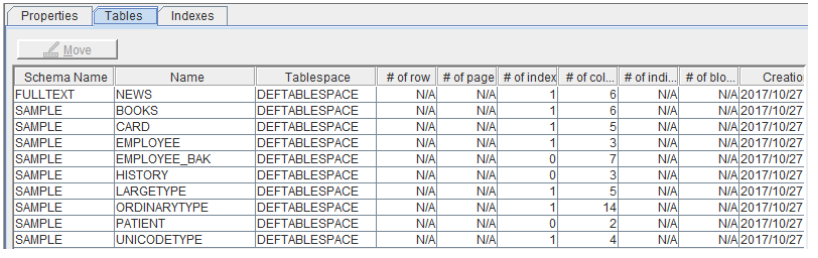
You can move multiple tables to another tablespace, and at the same time move the index to another tablespace if the index and the table in the same tablespace. In addition, if the index and the table in different tablespace, we can rebuild index in another tablespace.

Setting **FASTCOPY ON**, a user can improve execution speed of moving a table to another tablespace. When a table is moved, system will directly copy one data page to another data page, with log files operated only once in a copying and the buffer needless. Therefor the repeated operations of the log will be greatly reduced.

Move table to another tablespace can store the table to other disk, and avoid the table cannot store data while disk full.

➲ Altering multiple tables to another tablespace:

1. Select the tablespace you want to modify from the tree. The **Properties** page appears.
2. Select the **Tables** page, all the tables in the tablespace appear.
3. Select multiple tables that you want to alter from the tablespace.

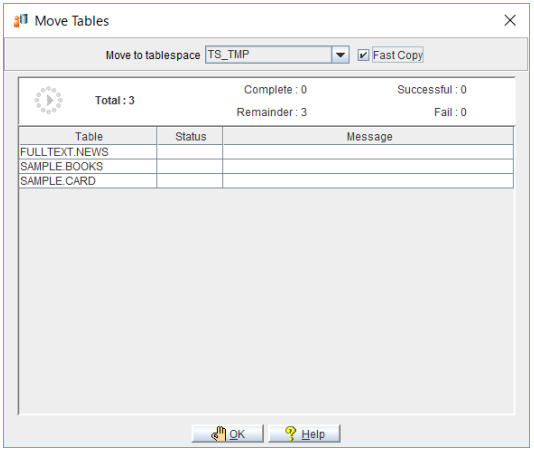


Schema Name	Name	Tablespace	# of row	# of page	# of index	# of col...	# of indi...	# of blo...	Creatio
FULLTEXT	NEWS	DEFTABLESPACE	N/A	N/A	1	6	N/A	N/A	2017/10/27
SAMPLE	BOOKS	DEFTABLESPACE	N/A	N/A	1	6	N/A	N/A	2017/10/27
SAMPLE	CARD	DEFTABLESPACE	N/A	N/A	1	5	N/A	N/A	2017/10/27
SAMPLE	EMPLOYEE	DEFTABLESPACE	N/A	N/A	1	3	N/A	N/A	2017/10/27
SAMPLE	EMPLOYEE_BAK	DEFTABLESPACE	N/A	N/A	0	7	N/A	N/A	2017/10/27
SAMPLE	HISTORY	DEFTABLESPACE	N/A	N/A	0	3	N/A	N/A	2017/10/27
SAMPLE	LARGETYPE	DEFTABLESPACE	N/A	N/A	1	5	N/A	N/A	2017/10/27
SAMPLE	ORDINARYTYPE	DEFTABLESPACE	N/A	N/A	1	14	N/A	N/A	2017/10/27
SAMPLE	PATIENT	DEFTABLESPACE	N/A	N/A	0	2	N/A	N/A	2017/10/27
SAMPLE	UNICODETYPE	DEFTABLESPACE	N/A	N/A	1	4	N/A	N/A	2017/10/27

NOTE *User cannot alter a system table, temporary table or view to another tablespace.*

4. Click the **Move** button, the **Move Tables** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



NOTE *Users can also drag one or more tables to the tablespace that you want to move directly by the mouse.*

5. Select the tablespace name from the **Move to tablespace** drop down list that you want to alter to the tablespace.

   NOTE *User cannot alter a permanent table to SYSTABLESPACE or TMPTABLESPACE.*

6. Click the **OK** button. The table marked icon would moved to another tablespace. If a single table moved failed, it should be marked icon and move the next one.

7. Click the **Finish** button to alter tables to the destination tablespace.

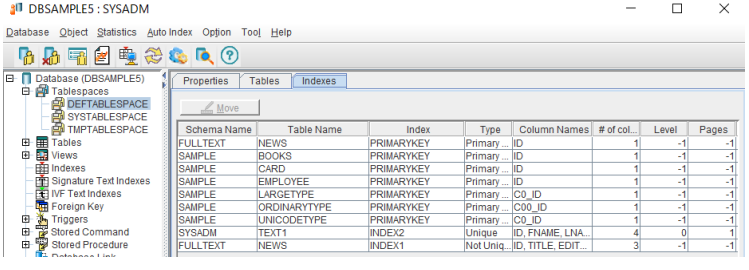
©Copyright 1995-2026 CASEMaker Inc.

3.5 Altering multiple indexes to another tablespace

You can move multiple indexes to another tablespace. Move indexes to another tablespace can store indexes to other disk. By default, owners of indexes and users with RESOURCE authority or higher can move indexes. A user only with CONNECT authority may not move any index.

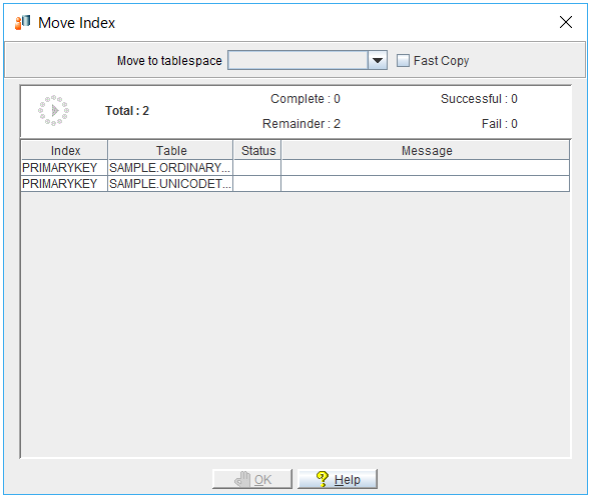
➲ Altering a index to another tablespace:

1. Select the tablespace you want to modify from the tree. The **Properties** page appears
2. Select the **Indexes** page, all the indexes in the tablespace appear.
3. Select one or more indexes that you want to move.



4. Click the **Move** button. The **Move Index** message box appears.

©Copyright 1995-2026 CASEMaker Inc.



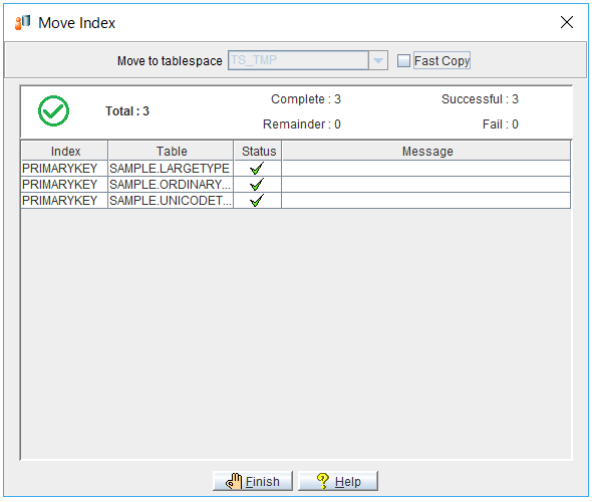
NOTE *Users can also directly drag one or more indexes to the tablespace that you want to move by the mouse.*

5. Select the tablespace name from the **Move to Tablespace Name** drop down list that you want to move.

NOTE *During the initialization for any tablespace, the moving status button is disabled, users cannot move any index belongs to system table or temporary index to another tablespace.*

6. Click the **OK** button. The index marked ✓ icons would moved to another tablespace. If a single index moved failed, it should be marked ✗ icon and move the next one.

©Copyright 1995-2026 CASEMaker Inc.



7. Click the **Finish** button to move index to the destination tablespace.

©Copyright 1995-2026 CASEMaker Inc.

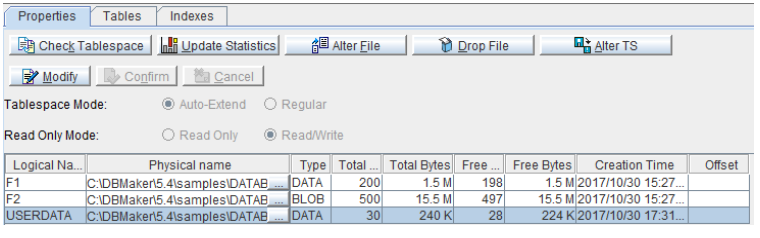
3.6 Dropping a Datafile from a Tablespace

You can drop a datafile file from a tablespace when it is no longer required. To drop a datafile successfully from a tablespace the datafile must be empty and cannot be the single remaining dataspace in the tablespace. The datafiles in the system and default tablespaces cannot be dropped.

➲ Dropping a datafile from a tablespace:

1. Select the tablespace you want to modify from the tree or the tablespaces page. The **Properties** page appears.
2. Select the datafile that you want to remove from the tablespace.

   NOTE *You cannot remove a datafile from a tablespace if it is the only datafile remaining in the tablespace or if it is not empty.*



3. Click the **Drop File** button.

   NOTE *You cannot drop the system or default tablespaces.*

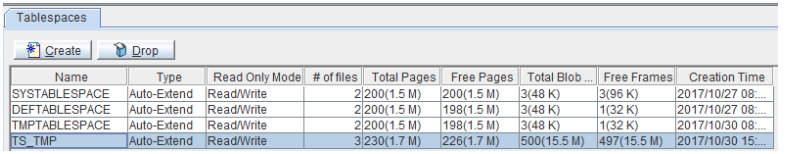
©Copyright 1995-2026 CASEMaker Inc.

3.7 Dropping a Tablespace

You can drop a tablespace that is no longer required by the database. By default, users with DBA authority or higher have DROP privilege for the tablespace.

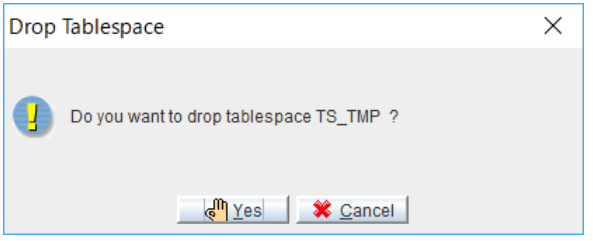
➲ Dropping a tablespace:

1. Click the **Tablespace** object in the tree. A list of all tablespaces appears.



Name	Type	Read Only Mode	# of files	Total Pages	Free Pages	Total Blob ...	Free Frames	Creation Time
SYSTABLESPACE	Auto-Extend	Read/Write	2	200(1.5 M)	200(1.5 M)	3(48 K)	3(96 K)	2017/10/27 08:...
DEFTABLESPACE	Auto-Extend	Read/Write	2	200(1.5 M)	198(1.5 M)	3(48 K)	1(32 K)	2017/10/27 08:...
TMPTABLESPACE	Auto-Extend	Read/Write	2	200(1.5 M)	198(1.5 M)	3(48 K)	1(32 K)	2017/10/30 08:...
TS_TMP	Auto-Extend	Read/Write	3	230(1.7 M)	226(1.7 M)	500(15.5 M)	497(15.5 M)	2017/10/30 15:...

2. Select the tablespace you want to delete from the tree.
3. Click the **Drop** button at the top of the **Tablespace** page. Alternatively, right click the tree node of the tablespace and select **Drop Tablespace** from the popup menu. A window appears to confirm the drop of the selected table.
4. The **Drop Tablespace** confirmation dialog box appears.



5. Click the **Yes** button to drop the tablespace.

©Copyright 1995-2026 CASEMaker Inc.

4 Working with Tables

Tables are the logical unit of storage used by DBMaker to store data. A table consists of several columns and rows. Columns are also referred to as fields or attributes. Rows are also referred to as records or tuples.

DBMaker identifies each table by a unique owner name and table name.

This chapter describes:

- Creating a table
- Modifying table schema
- Setting table properties
- Granting privileges
- Updating table statistics
- Editing data in a table
- Renaming a table
- Dropping a table
- Table and Column Constraint Syntax

NOTE *Select **Preferences** from the **Option** menu to view the system tables. Click the **Show System Tables** radio button on the **Table** page in the **Preferences** window. Please refer to Chapter 24 for more information.*

©Copyright 1995-2026 CASEMaker Inc.

4.1 Creating a Table

Tables provide the essential framework for designing databases. When creating a table, you can import schema from other tables belonging to the same database. By default, users with RESOURCE authority or higher have CREATE privilege for the table. A user only with CONNECT authority may not create any table.

When creating a table, you will select options from the Create Table wizard. The wizard lets you do the following:

- Set table, tablespace, and schema options
- Set column options
- Set column and table constraints
- Create a unique constraint
- Create foreign keys
- Set advanced table options

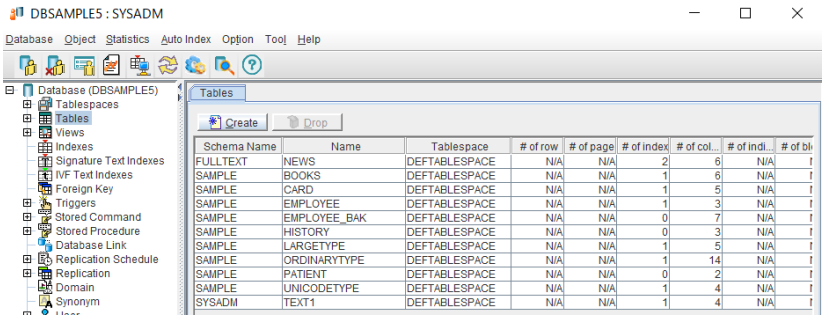
Starting the Create Table Wizard

The **Create Table Wizard** provides an intuitive interface for creating tables. All table creation options are available within the wizard. Options that are important to set at the time of table creation are presented to ensure that the table is created in the way intended.

➲ Using the Create Table Wizard:

1. Select the **Table** object from the tree. The **Table** page appears.

©Copyright 1995-2026 CASEMaker Inc.



2. Click the **Create** button. The **Welcome to Create Table Wizard** window appears.

3. Click the **Next** button. The **Name the Table** window appears.

Setting Table, Tablespace and Schema Options

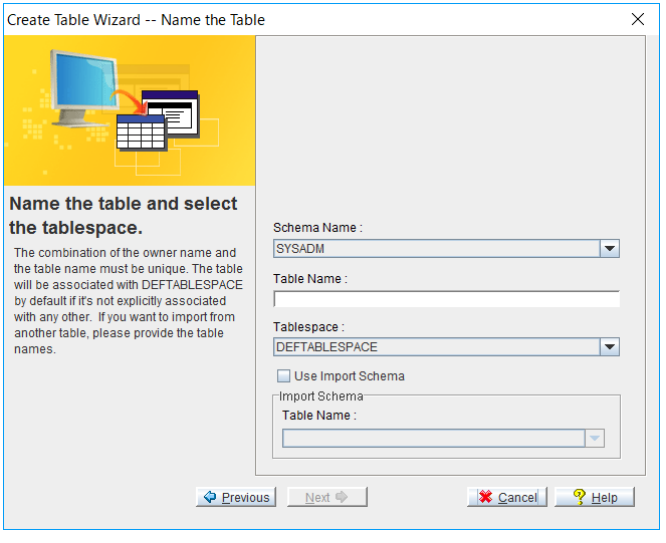
Once you have started the **Create Table Wizard**, you can set table, tablespace, and schema options. Users may import table schema from another table. Imported table schema, all columns (name, length, data type) initially have the

©Copyright 1995-2026 CASEMaker Inc.

same structure as found in the original table. The table schema may be modified after being imported.

➲ Setting table, tablespace, and schema options:

1. Follow the steps in the **Create Table Wizard** until the **Name the Table** window appears.



2. Select the **Schema Name** from the drop down list. Your user login name appears by default.
3. Enter a table name in the **Table Name** field.

   NOTE *The table names must be unique for a single owner.*

4. Select a tablespace from the **Tablespace** menu.

   NOTE *By default, tables are created in the default user tablespace. The default tablespace appears in the **Tablespace** menu as **DEFTABLESPACE**).*

5. Importing schema from another table:
   a) Click the **Use Import Schema** check box if you want the schema of the new table to be the same as the schema of another table in the database.

©Copyright 1995-2026 CASEMaker Inc.

b) Select the table to import from in the **Table Name** menu.

6. Click the **Next** button. The **Add columns to the table** window appears.

Setting Column Options

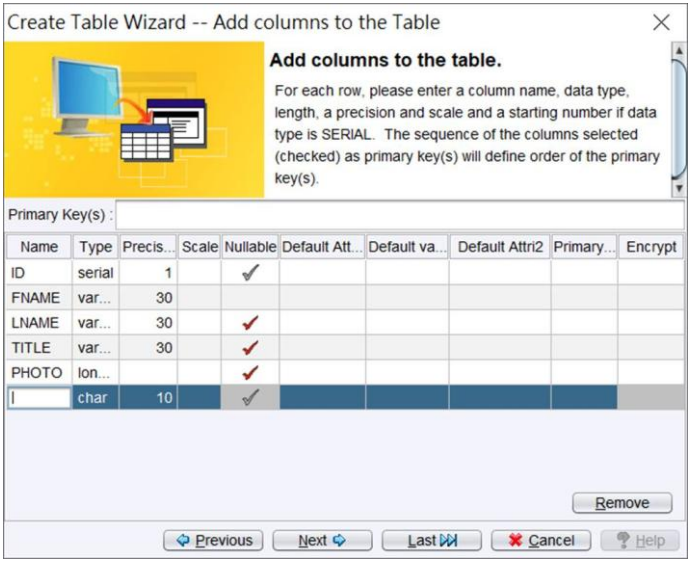
A variety of column options can be set to meet your data requirements. The settings follow:

- Name: specifies a column name.
- Type: specifies the following data types: BIGINT, BIGSERIAL, BINARY, CHAR, NCHAR, DATE, DECIMAL, DOUBLE, FILE, FLOAT, INTEGER, LONG VARBINARY, LONG VARCHAR, OID, SERIAL, SMALLINT, TIME, TIMESTAMP, VARCHAR, NVARCHAR, BLOB, CLOB, NCLOB, REAL, HTMLFILETYPE, HTMLTYPE, JSONCOLS, MSEXCELFILETYPE, MSEXCELTYPE, MSPPTFILETYPE, MSPPTTYPE, MSWORDFILETYPE, MSWORDTYPE, PDFFILETYPE, PDFTYPE, XMLFILETYPE, XMLTYPE. For complete descriptions of the data types, please refer to the *SQL Command and Function Reference*.
- Encrypt: specifies the column as an encrypted column. To create an encrypted column, TDE status must be opened.
- Precision: specifies the length of the columns. This is necessary for the following data types that have variable length: VARCHAR, NVARCHAR, CHAR, NCHAR, DECIMAL, SERIAL, BIGSERIAL and BINARY.
- Scale: specifies the scale of the data type. This is required for the DECIMAL data type.
- Nullable: restricts or permits null values on columns. By specifying NOT NULL, null values cannot be added to the column. The default value is **NULL**. NOT NULL is usually specified for primary key columns.
- Primary Key: specifies a column as a primary key in a table.
- Default Value: specifies a default value for the column data.

©Copyright 1995-2026 CASEMaker Inc.

➲ Setting column options:

1. Follow the steps in the Create Table wizard until the **Name the Table** window appears.
2. Enter column information under the appropriate headings. Both the **USER/SYSTEM** option and the **ON UPDATE** option are optional.



3. Click the **Next** button. The **Column constraints** window appears. If you do not need to add constraints, a foreign key, or set advanced options for the table you may click the **Last** button. The **Final Review** window appears.

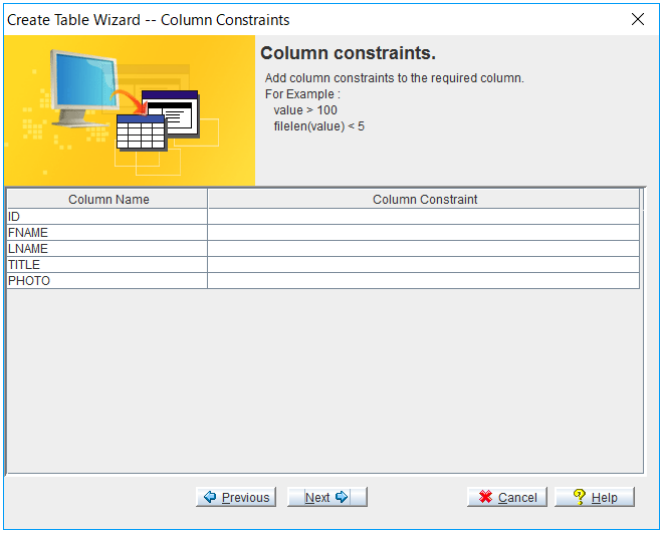
Setting Column and Table Constraints

You can further customize your data requirements by setting column and table constraints. Column and table constraints must satisfy SQL syntax rules. Please see *Constraint Syntax* at the end of this chapter for additional information. Table constraints cannot be altered after the table has been created.

➲ Setting column and table constraints:

©Copyright 1995-2026 CASEMaker Inc.

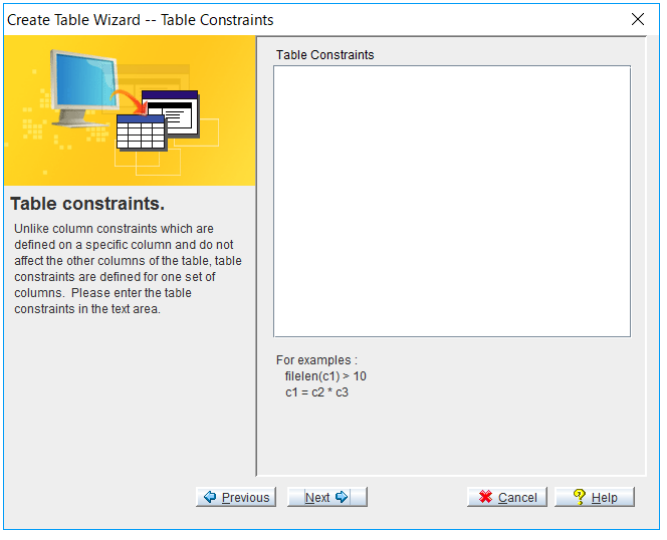
1. Follow the steps in the **Create Table** wizard until the **Column Constraints** window appears.



2. Enter column constraints in the corresponding rows under the **Column Constraint** heading.

©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Table Constraint** window appears.



4. Enter table constraints in the **Table Constraints** field.
5. Click the **Next** button. The **Create Unique Constraint** window appears.

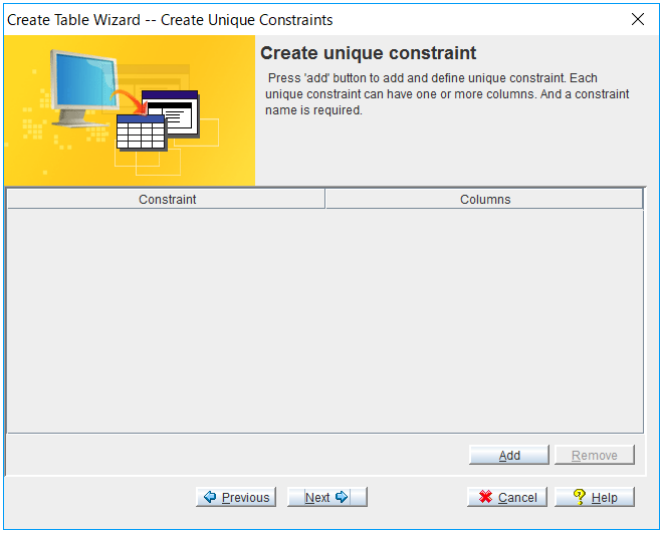
Adding a UNIQUE Constraint

UNIQUE constraints are applied to one or more columns of a table at the time it is created. A column having a UNIQUE constraint cannot contain any two tuples with like values, but it may contain null values. The UNIQUE constraint must be applied when a column is initially created.

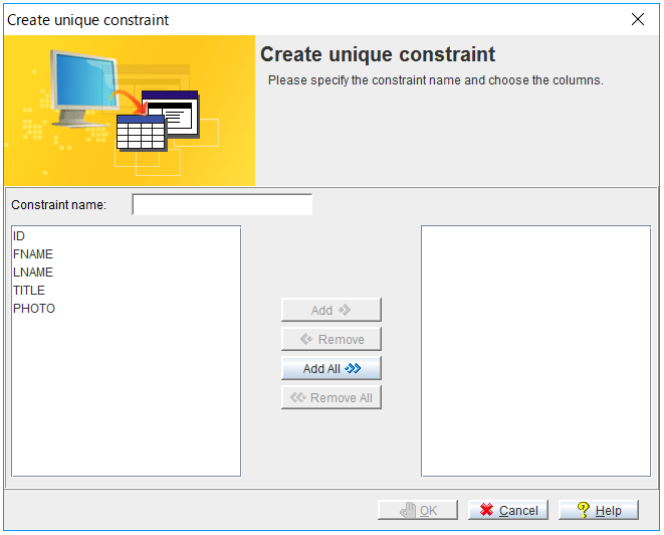
➲ Adding a unique constraint

1. Follow the steps in the **Create Table Wizard** until the **Create Unique Constraint** window appears.

©Copyright 1995-2026 CASEMaker Inc.



2. Click the **Add** button. The **Create Unique Constraint** window appears.

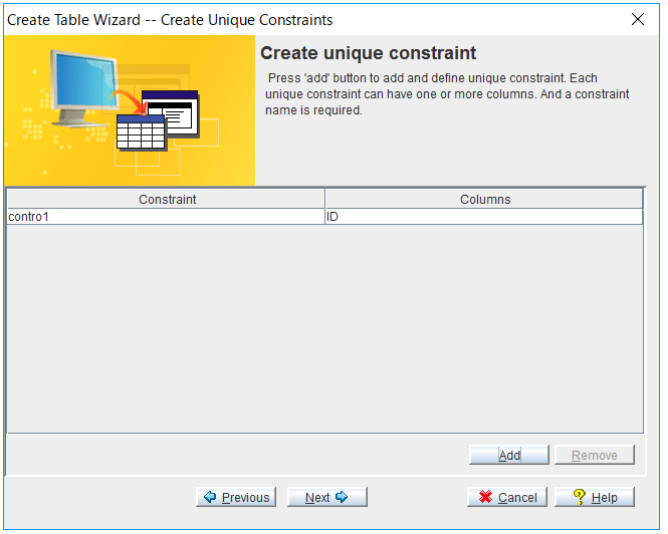


3. Enter a constraint name into the **constraint name** text field.
4. Select columns to apply the UNIQUE constraint to by double clicking the column names appearing on the left side of the window. Alternatively, use the **Add**, **Add All**, **Remove**, and **Remove All** buttons located in the

©Copyright 1995-2026 CASEMaker Inc.

center area of the window to add or remove columns. Selected columns appear on the right side of the window.

5. Click the **OK** button. The **Create unique constraint** window appears.



6. Click the **Next** button. The **Create Foreign Key to identify Column(s)** window appears.

Adding a Foreign Key while Creating a Table

You can choose a foreign key as well as select foreign key options. Foreign key columns in the referencing table must map exactly to the primary key columns or unique index columns of the referenced table. The number of columns used in the foreign key and the schema of those columns must be identical in both the referencing and referenced tables.

A foreign key ensures referential integrity between two tables. The foreign key accomplishes this task according to the action chosen for the foreign key. When a record in the referenced column(s) is deleted or updated, the foreign key performs the action on the referencing column(s). The possible actions are:

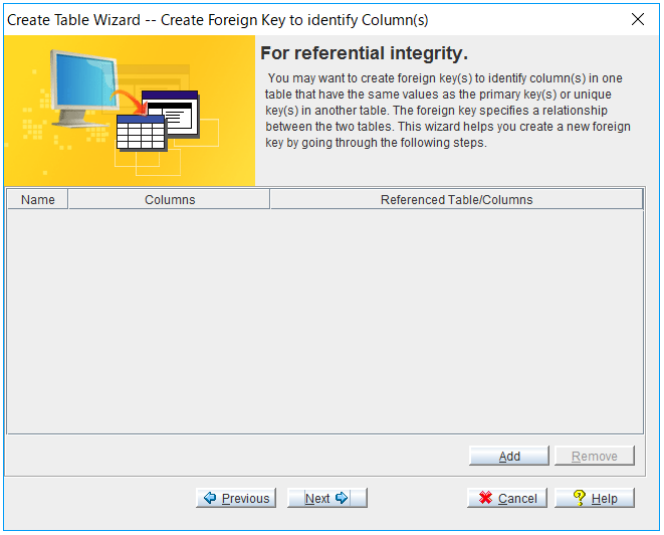
©Copyright 1995-2026 CASEMaker Inc.

- Set Null: Records in the foreign key column are set to null if the user updates or deletes the corresponding records in the primary key or unique index column
- Cascade: The corresponding records in the referenced table are also updated or deleted
- Set Default: The foreign key record is set to a default value if the user updates or deletes the corresponding records in the primary key or unique index column
- No Action: No action is taken on the foreign key column if the user updates or deletes the corresponding records in the primary key or unique index column

Please note that the delete action of a foreign key may be different from the update action.

➲ Adding a foreign key during table creation:

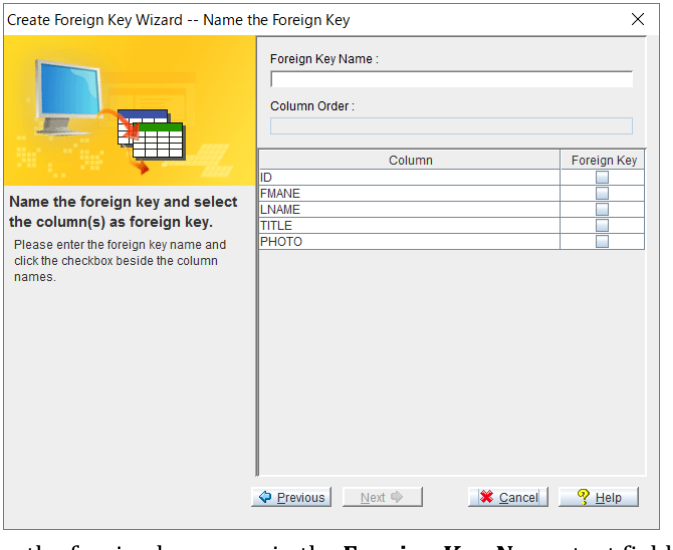
1. Follow the steps in the **Create Table Wizard** until the **Create Foreign Key to identify Column(s)** window appears.



©Copyright 1995-2026 CASEMaker Inc.

2. Click the **Add** button. The **Create Foreign Key – Introduction** window appears.

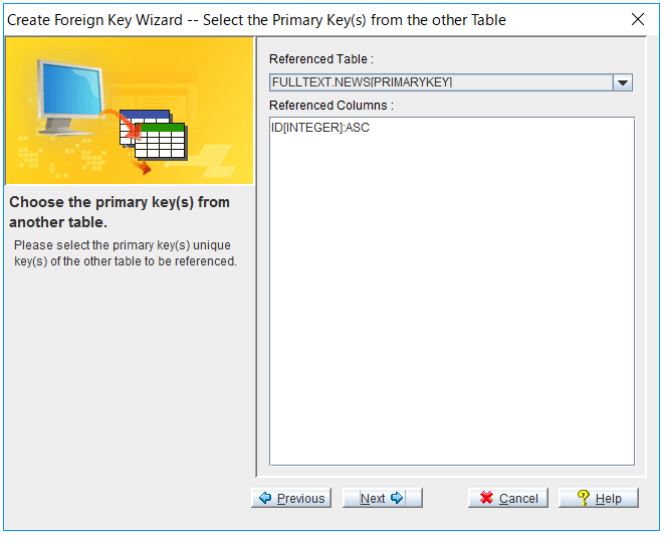
3. Click the **Next** button. The **Name the Foreign Key** window appears.



4. Enter the foreign key name in the **Foreign Key Name** text field.

©Copyright 1995-2026 CASEMaker Inc.

5. Click checkboxes beside the column names to specify which columns to include in the foreign key.
6. Click the **Next** button. The **Choose the primary key(s) from another table** window of the **Create Foreign Key Wizard** appears.

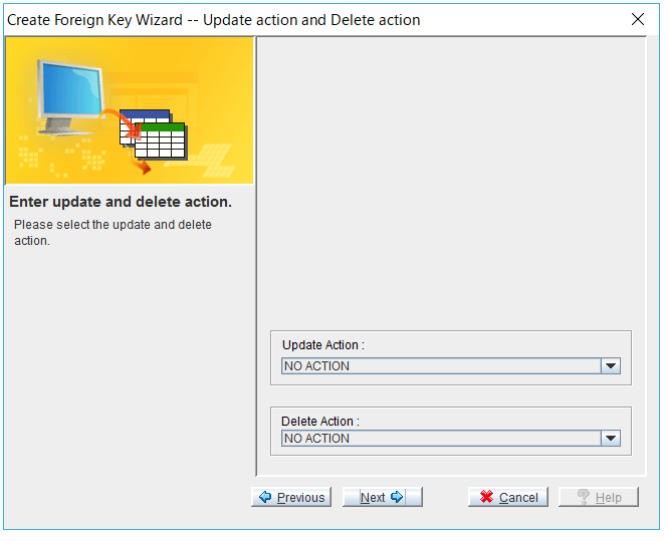


7. Select the name of the referenced table from the **Referenced Table** menu.

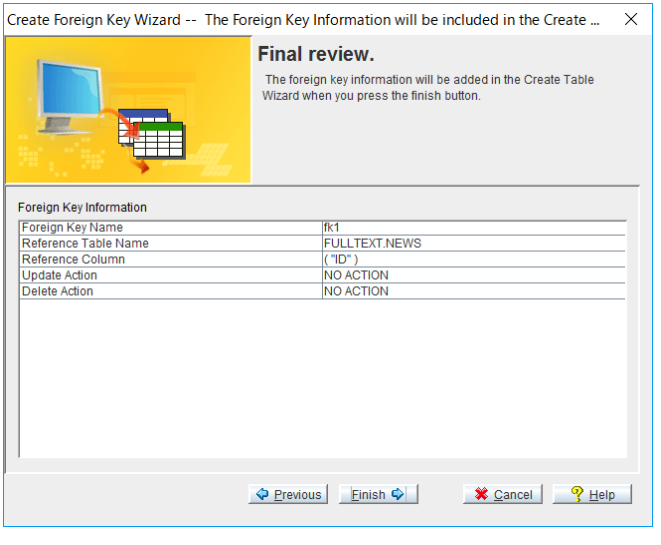
NOTE *Only tables containing primary keys or unique indexes that match the schema of the foreign key are shown.*

8. All the selected table columns that are primary keys or unique indexes are displayed.
9. Click the **Next** button. The **Update action and Delete action** window appears.

©Copyright 1995-2026 CASEMaker Inc.

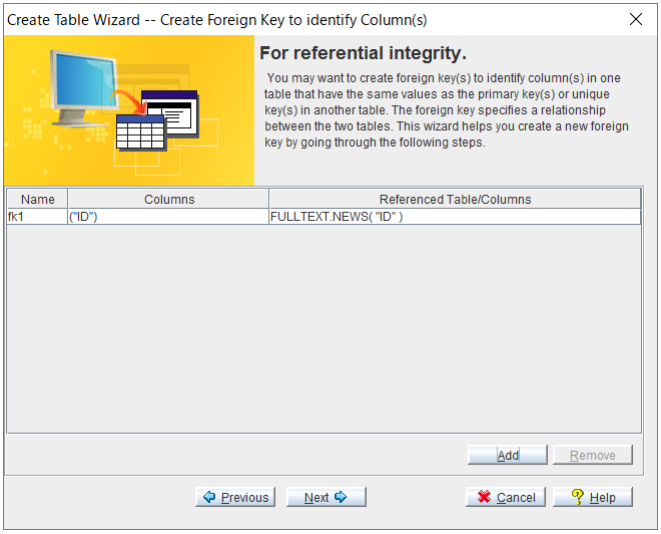


10. Choose a foreign key setting from the **Update Action** pop-up menu.
11. Choose a foreign key setting from the **Delete Action** pop-up menu.
12. Click the **Next** button. The **Final Review** window for foreign key creation appears.



©Copyright 1995-2026 CASEMaker Inc.

13. Click the **Finish** button. The **Create Foreign Key to identify Column(s)** window appears.



14. To delete a foreign key:
    a) Select a foreign key.
    b) Click the **Remove** button.
15. Click the **Next** button. The **Advanced Information** window appears.

Setting Advanced Table Options

Many performance options are available when creating a table. These options can have a significant impact on the table's data handling performance. The options are:

- Temporary / permanent tables: Temporary tables support fast data operations but are only accessible by the creator. Temporary tables exist during a single session, and DBMaker automatically drops temporary tables when the user (i.e., creator) disconnects from the database. Permanent tables exist in the database until they are

©Copyright 1995-2026 CASEMaker Inc.

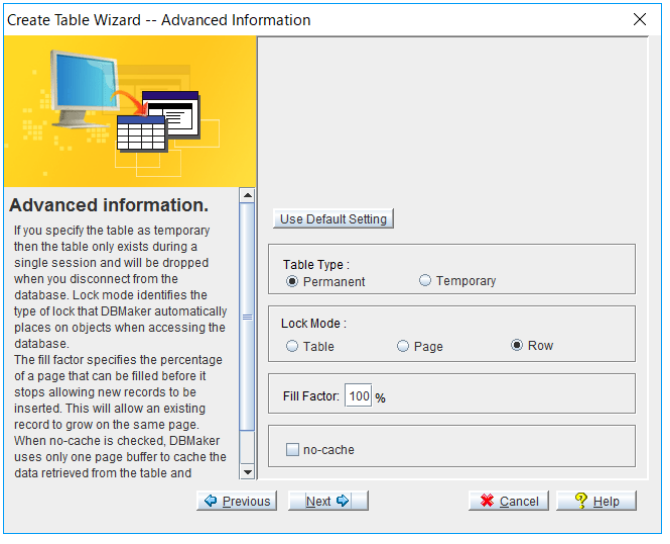
dropped by a user having access and delete privileges. The default table type is ***Permanent***.

- Lock mode: DBMaker locks objects in a table according to the table's lock mode. DBMaker supports three lock modes: TABLE, PAGE, and ROW. The default lock mode is **ROW**. When the lock mode is set to the higher level, TABLE, the number of concurrent database accesses decreases, but the required lock resources (i.e., shared memory) also decreases. When the lock mode is set to the lower level, **ROW**, the number of concurrent database accesses increases, but the required lock resources (i.e., shared memory) also increases.
- Fill Factor: specifies the threshold percent when a page no longer allows new records. Lowering the fill factor value facilitates the growth of existing records in the page.
- No-cache: The no-cache feature is useful when accessing large tables by table scan. Table scans on large tables can cause frequent disk I/O activity. This happens during a table scan on a table with a larger number of data pages than the number of page buffers, which causes all page buffers to be exhausted. For a table created with the no-cache option, DBMaker only uses one page buffer to cache data retrieved from the table during table scans. This prevents a single large table scan from exhausting the page buffers.

➲ Setting advanced table options:

1. Follow the steps in the **Create Table Wizard** until the **Advanced Information** window appears.

©Copyright 1995-2026 CASEMaker Inc.



2. If you want the table to be a temporary table, select the **Temporary** radio button.
3. If you wish to change the lock mode, select a different lock mode radio button. Please note, the default is **Row**. To use table lock mode, click the **Table** radio button. To use page lock mode, click the **Page** radio button.
4. To change the fill factor from its 80% default value, enter a new value in the **Fill Factor** field.
5. Select the **no-cache** check box if you want to use only one page buffer to cache the data retrieved from the table.
6. Click the **Next** button. Next, the **Final review** window appears.

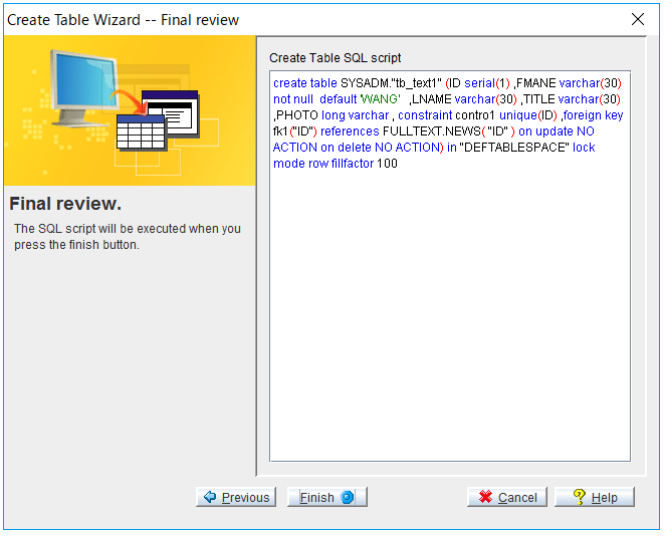
Completing Table Creation

After selecting table options using the **Create Table Wizard**, you are ready to complete the process of creating the table. A full explanation of the CREATE TABLE statement and syntax is in the *SQL Command and Function Reference*.

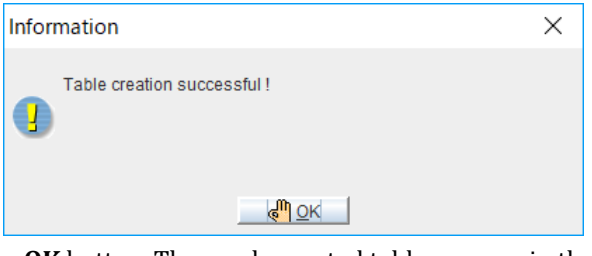
➲ Completing table creation:

©Copyright 1995-2026 CASEMaker Inc.

1. Follow the steps in the **Create Table Wizard** until the **Final Review** window appears.



2. The SQL statement generated by the **Create Table Wizard** can be modified by hand at this point. Simply use the mouse to place the cursor and insert new text. Edit the statement as needed but maintain SQL99 compliance.
3. Review the final SQL statement and click the **Finish** button. A confirmation message appears.



4. Click the **OK** button. The newly created table appears in the tree under the **table** node.

©Copyright 1995-2026 CASEMaker Inc.

4.2 Modifying Table Schema

Once a table is created, you can modify its structure by adding or deleting columns, changing the column order, or modifying schema objects. Some restrictions exist to modifying schema. The following schema objects may be altered:

- Altering to another tablespace
- Column Order
- Name
- Type
- Precision and Scale
- Null / Not Null
- Primary Key(s)
- Default Value
- Column Constraints
- Adding columns
- Deleting columns
- Tracing
- Encrypt

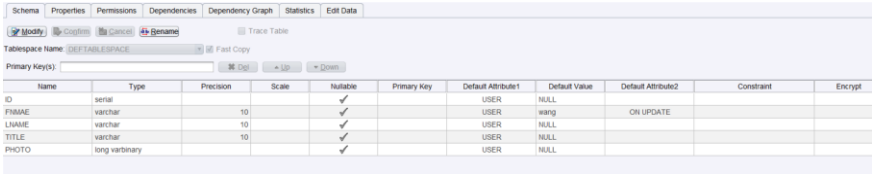
DBMaker allows one of the following users to modify a table.

- Users with DBA authority or higher
- The table owner
- Users granted to ALTER privilege

©Copyright 1995-2026 CASEMaker Inc.

➲ Modifying a table's schema:

1. Click the **Table** node in the tree.
2. From the table page, double-click the table you want to modify. The **Schema** page appears.



NOTE *You can also display the* **Schema** *page by opening the table node in the tree and clicking on a table in the tree.*

3. Click the **Modify** button. The **Del, Up, Down, Confirm** and **Cancel** buttons are activated.

Altering to another tablespace

User can move a table to another tablespace, and at the same time move the index to another tablespace if the index and the table in the same tablespace.

➲ Altering to another tablespace:

1. Select the tablespace name from the **Tablespace name** drop down list that you want to alter to the tablespace.

   NOTE *Connect privilege user can not display the "Tablespace name" combo box, because connect privilege user can not select the systablespace.*

2. Click the **Confirm** button.

Changing the Column Order

The order of columns in the table determines the table's schema. Changing the column order may be necessary to facilitate the use of foreign keys or table replication.

©Copyright 1995-2026 CASEMaker Inc.

➲ Changing column order:

1. Select the column you want to move.
2. Click the **Up** or **Down** buttons to move the column to the desired position.
3. Click the **Confirm** button.

Changing Column Names

New or changed column names must conform to DBMaker's naming conventions. Database configuration determines whether column names are case-sensitive.

➲ Changing a column name:

1. Select a column to rename.
2. Select a field under the **Name** heading.
3. Enter a new name into the field.
4. Click the **Confirm** button.

Changing Column Type

Some data types cannot be converted. For example, CHAR type data cannot be converted to INTEGER type data and DOUBLE type data cannot be converted into FLOAT type data. However, FLOAT type data can be converted into DOUBLE type data. Data conversion follows standard C programming rules.

➲ Changing a column's data type:

1. Select a column to retype.
2. Click a field under the **Type** heading, a drop down menu of available data types appears.
3. Select a new data type from the drop down menu.
4. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.

Changing Precision and Scale

The precision (i.e., length) of CHAR, DECIMAL and SERIAL type columns may be altered. Be aware that decreasing the scale of a column results in concatenation at the end of the string. The scale of DECIMAL type data may also be altered.

➲ Changing a column's precision or scale:

1. Select the column to retype.
2. Click the field under the **Precision** heading and enter a new value into the field.
3. Click the field under the **Scale** heading and enter a new value into the field.
4. Click the **Confirm** button.

Changing Column Null / Not Null

This setting specifies whether a column may accept null values. A column can only be changed to **Not Null** if the table is empty or the **GIVE** keyword is used to input values into the empty records.

➲ Making a column nullable/not nullable:

1. Select a column you want to change.
2. Click a field under the **Nullable** heading. A check mark indicates that the column accepts null values.
3. Click the **Confirm** button.

Changing the Primary Key

Columns may only be selected as primary keys if the data contained in the column is unique.

➲ Altering primary keys:

1. Select the column you want to modify.

©Copyright 1995-2026 CASEMaker Inc.

2. Select the field under the **Primary Key** heading. A key appears in the field to indicate that it is a primary key. The column name also appears at the end of the list in the **Primary Key(s)** text field at the top of the **Schema** page.
3. Clicking the field when a key is displayed in the field removes it as a primary key, and likewise removes it from the list.
4. To change the order of the primary keys, remove all primary keys and then add them in the desired order. The order of the primary keys is indicated in the **Primary Key(s)** text field at the top of the **Schema** page.
5. Click the **Confirm** button.

Changing the Column Default Value

Default values are inserted into columns where the user has not input a value for that column in a record. Default values must be valid for the column data type.

The **USER/SYSTEM** keywords are optional. These keywords specify whether users can modify value of the column with a default value by using the INSERT/UPDATE statement. **USER** is used by default. The **USER** keyword specifies that users can modify its value, and the **SYSTEM** keyword specifies that users cannot modify its value.

The **ON UPDATE** keyword is optional. This keyword specifies that value of the column with a default value can be automatically updated when other columns' value is changed.

➲ Changing or adding a default value for the column:

1. Select a column to change.
2. Enter a default value for the column that is valid for its data type in the field under the **Default Value** heading.
3. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.

Adding Column Constraints

Column constraints must conform to SQL syntax. Please see the section *Constraint Syntax* at the end of this chapter for more information.

➲ To change or add column constraints:

1. Select a column to modify.
2. Enter a new constraint for the column in the field under the **Constraint** heading.
3. Click the **Confirm** button.

Changing Column Encryption

This setting specifies whether a column is an encrypted column. A column can only be encrypted when **TDE** status is open.

➲ To Encrypt/Decrypt column:

1. Select a column you want to change.
2. Click a field under the **Encrypt** heading. A check mark indicates that the column changes to an encrypt column.
3. Click the **Confirm** button.

Adding a Column

Tables without data may have columns added to them. Tables containing data or referencing foreign keys may or may not be able to have columns added. Foreign keys will be broken if the schema of the referenced table is changed. Please refer to section 4.5, *Checking Dependencies,* for directions on checking for referencing tables.

➲ Adding a column:

1. Enter a column name into the blank field under the **Name** heading at the bottom of the list.
2. Select a data type from the drop down menu in the blank field under the **Type** heading.

©Copyright 1995-2026 CASEMaker Inc.

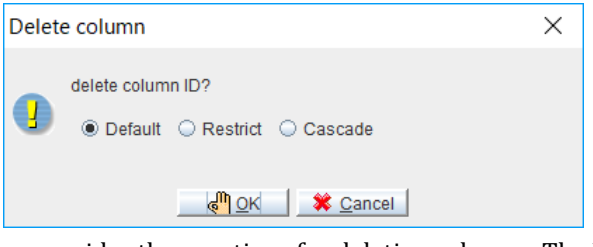
3. Enter a precision,or scale, or both if the data type requires it.
4. Clicking on the blank field under the **Nullable** heading toggles if the column can have null values or not. A check mark indicates that null values are permitted.
5. Clicking on the blank field under the **Primary key** heading toggles if the column is a primary key or not. A key icon indicates that the column is a primary key.
6. Enter a default value into the blank field under the **Default Value** heading at the bottom of the list (optional).
7. Enter column constraints into the blank field under the **Constraints** heading at the bottom of the list.
8. Click the **Confirm** button.

Deleting a Column

Columns may be deleted from a table if they are no longer needed.

➲ Deleting a column:

1. Select the column you want to delete.
2. Click the **Delete** button.
3. Click the **Confirm** button. The **Delete column** confirmation dialog box appears.



4. DBMaker provides three options for deleting columns. The **Default** option drops columns without checking for dependent object. Any dependent objects become invalid after the column drop. The **Restrict** option considers index, text index, and view dependencies. This option does not permit a column drop when one or more dependencies exist. The **Cascade** option drops the column and all of its dependent objects.

©Copyright 1995-2026 CASEMaker Inc.

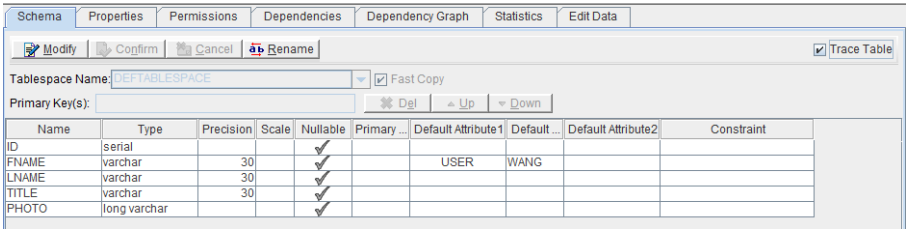
5. Click the **OK** button.

Tracing a Table

While Keywords **DB_LgSvr** is equal or greater than 4, user can adds trace on a single table to log the detail OLD/NEW data.

➲ Tracing a table:

1. Select the table you want to add trace on.
2. Select the **Trace Table** checkbox when the value specified by **DB_LGSVR** keyword is equal or greater than 4.



©Copyright 1995-2026 CASEMaker Inc.

4.3 Setting Table Properties

You can modify the settings you added when you created a table, and view the table constraints. The following properties may be altered:

- Lock Mode: DBMaker locks objects in a table according to the table's lock mode. DBMaker supports three lock modes: TABLE, PAGE, and ROW. The default lock mode is **ROW**. When the lock mode is set to the higher level, **TABLE**, the number of concurrent database accesses decreases, but the required lock resources (i.e., shared memory) also decreases. When the lock mode is set to the lower level, **ROW**, the number of concurrent database accesses increases, but the required lock resources (i.e., shared memory) also increases.
- Fill Factor: specifies the threshold percent when a page no longer allows new records. Lowering the fill factor value facilitates the growth of existing records in the page.
- Cache / No-cache: The no-cache feature is useful when accessing large tables by table scan. Table scans on large tables can cause frequent disk I/O activity. This happens during a table scan on a table with a larger number of data pages than the number of page buffers, which causes all page buffers to be exhausted. For a table created with the no-cache option, DBMaker only uses one page buffer to cache the data retrieved from the table during table scans. This prevents a single large table scan from exhausting the page buffers.
- Serial: Resets the serial number for a serial type column in the table.

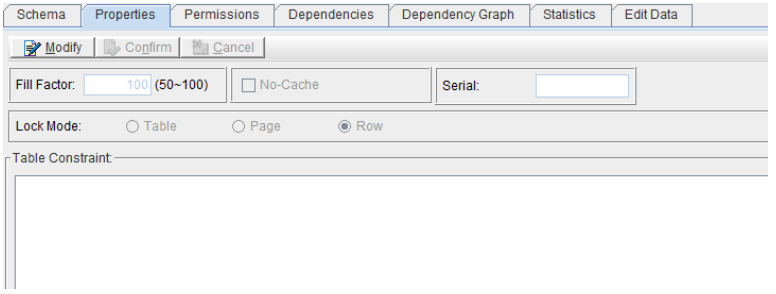
DBMaker allows one of the following users to modify a table.

- Users with DBA authority or higher
- The table owner
- Users granted to ALTER privilege

©Copyright 1995-2026 CASEMaker Inc.

➲ Setting table properties:

1. Double-click the **Table** node in the tree.
2. Double-click the table you want to modify. The **Schema** page appears.
3. Select the **Properties** tab. The **Properties** page appears.



4. Click the **Modify** button. The **Fill Factor, no-cache, Serial,** and **Lock Mode** buttons are activated.
5. Enter a percentage in the **Fill Factor** field.
6. Select the **no-cache** check box if you want only one page buffer to cache the data retrieved from the table.
7. Enter the starting number of the column in the **Serial** box to reset the serial number for a serial type column in the table.
8. If you wish to change the lock mode, choose a different lock mode from the default setting.
   - Click the **Table** radio button to choose table lock mode
   - Click the **Page** radio button to choose page lock mode
   - Click the **Row** radio button to choose row lock mode
9. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.

4.4 Granting and Revoking Table Privileges

To allow other users or groups to access tables in the database, you can grant privileges to them. You can also revoke privileges to deny groups and users access to tables. Privileges are granted to specific functions such as Insert, Update, Delete, Index, and Reference, or to specific columns of the table.

Users with RESOURCE or CONNECT authority cannot have their privileges revoked on objects they own. These privileges are only revoked by changing the user's authority.

The following list summarizes table privileges:

- **Select**: Users have permission to select any records from the table
- **Delete**: Users have permission to delete any records from the table
- **Insert**: Users have permission to insert new records into the table or column
- **Update**: Users have permission to update any records in the table or column
- **Alter**: Users have permission to alter the schema of the table
- **Index**: Users have permission to create an index on the table
- **Reference**: Users have permission to create referential constraints on the table or column

DBMaker allows one of the following users to modify a table privileges.

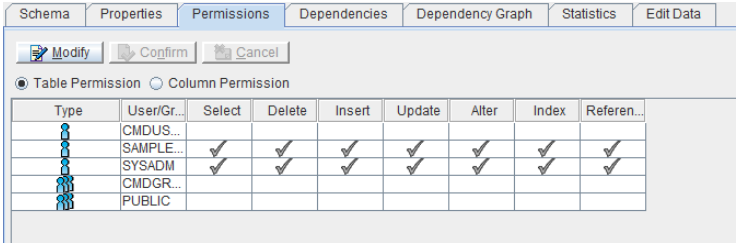
- Users with DBA authority or higher
- The table owner

➲ Granting and revoking table privileges:

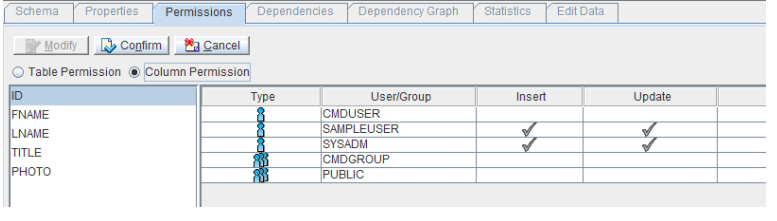
1. Select a table. The **Schema** page appears.

©Copyright 1995-2026 CASEMaker Inc.

2. Click the **Permissions** tab at the top of the window. The **Permissions** page appears.



3. Click the **Modify** button. The **Confirm** and **Cancel** buttons are now active.
4. Select one of the permissions in the cells for the user or group.
5. Click any of the checked columns to revoke that privilege.
6. To grant or revoke permission on columns, click the **Column Permissio**n option button. The following window appears.



7. Select any of the following permissions in the cells for your user/group:
   - **Insert**: users have permission to insert new records into the column
   - **Update**: users have permission to update any records in the column
   - **Reference**: users have permission to create referential constraints on the column
8. Click any of the checked columns to revoke a privilege.
9. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.

4.5 Checking Dependencies and Statistics

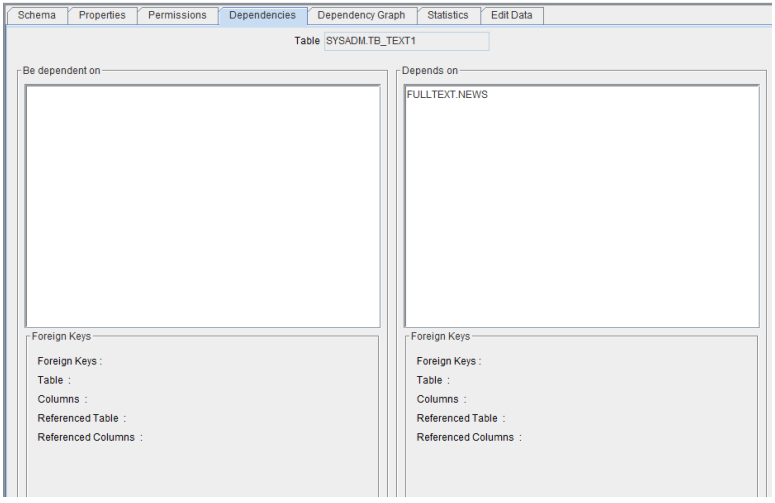
Before handling data in a table or modifying a table's schema, it is recommended that you check dependencies and refer to table statistics.

Checking Dependencies

You can check table names, owners, as well as foreign and primary keys in a table.

➲ Checking dependencies:

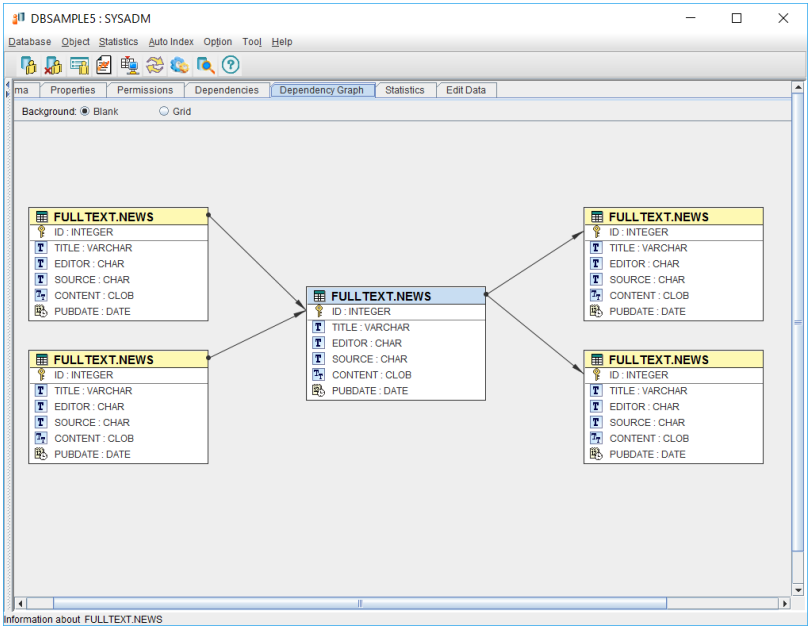
1. Select a table. The **Schema** page appears.
2. Click the **Dependencies** tab at the top of the window. The names of tables that foreign keys identify as referencing the selected table appear in the **Be dependent on** column. The names of tables that are referenced by the selected table appear in the **Depends on** column.



3. Selecting a table from the **Be dependent on** list displays foreign key information on the bottom.

©Copyright 1995-2026 CASEMaker Inc.

4. Selecting a table from the **Depends on** list displays foreign key information on the bottom.
5. Double-click the table name on either list to change the selected table in the **Table** field at the top of the page.
6. Click the **Dependency Graph** tab at the top of the page to check the dependencies relationship of tables.



NOTE *Users can move the picture position and adjust the picture size by the mouse. When the mouse moves to the line, it will display the specific dependence information between the tables.*

Checking and Updating Table Statistics

You can check statistics for both tables and columns. You can also alter the sample ratio, automatically update statistics daemon and use the Update Statistics function.

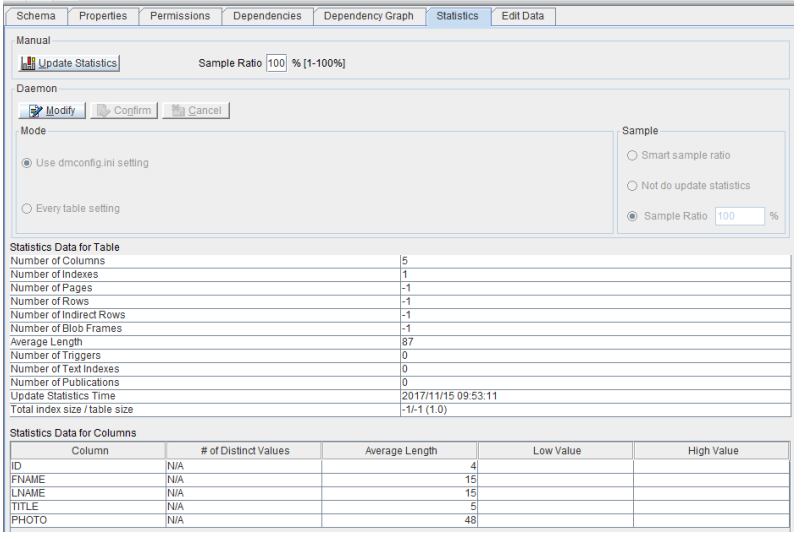
©Copyright 1995-2026 CASEMaker Inc.

DBMaker allows one of the following users to update table statistics, modify update statistics daemon mode and set the update statistics sample ratio.

- Users with DBA authority or higher
- The table owner

➲ Checking and updating table statistics:

1. Select a table. The **Schema** page appears.
2. Select the **Statistics** tab at the top of the window. The statistics appear as shown below.



3. Select the **Sample Ratio** box and enter a sample ratio percentage to change the percentage of the data used to calculate updated statistics manually.
4. Click the **Update Statistics** button. The updated statistics appear.
5. Click the **Modify** button. Choose a different statistics daemon mode from the **Mode** heading.
   - Click the **Use dmconfig.ini setting** radio button to through configure keywords to activate update statistics daemon mode.

©Copyright 1995-2026 CASEMaker Inc.

- Click the **Every table setting** radio button to enable startup update statistics daemon and support set sample for all tables mode.

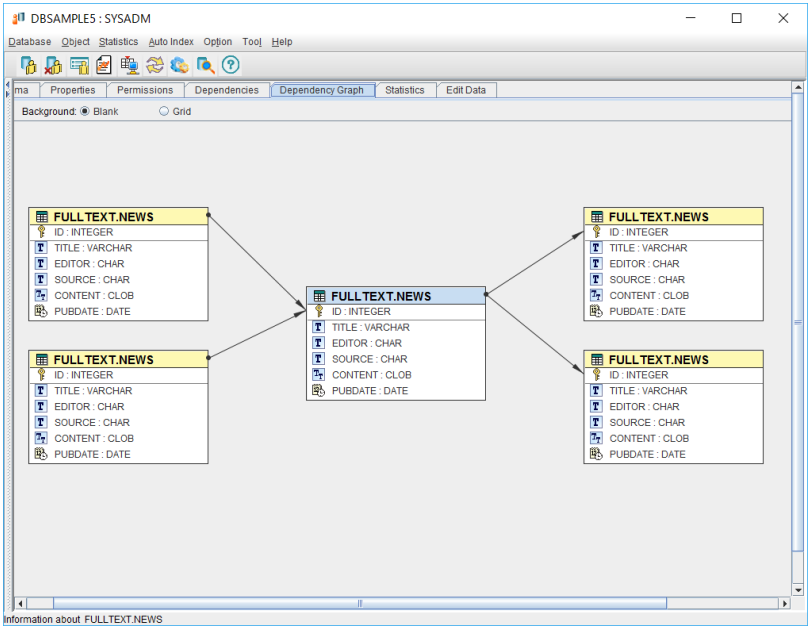
6. Set the Sample Ratio from the **Sample** heading.

   - If users select the **Use dmconfig.ini setting** radio button under the **Mode** heading, then DBMaker will use the **Sample Ratio** radio button under the **Sample** heading by default.

   - If users select the **Every table setting** radio button under the **Mode** heading, then user can select **Smart sample ratio**, **Not do update statistics** or **Sample Ratio** radio button. Click the **Smart sample ratio** radio button to enables startup update statistics daemon and support smart mode. Click the **Not do update statistics** radio button to enable NON-STARTUP update statistics daemon mode. Enter a sample ratio percentage in the **Sample Ratio** area to set the percentage of the data used to calculate updated statistics.

7. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.



NOTE *Users can move the picture position and adjust the picture size by the mouse. When the mouse moves to the line, it will display the specific dependence information between the tables.*

©Copyright 1995-2026 CASEMaker Inc.

4.6 Editing Data in a Table

Once you add records to a table, you may also update existing records in that table. Object privilege is required for table editing. Please refer to *Granting and Revoking Table Privileges* and *Granting and Removing Object Privileges* for information about managing table and column user privileges.

DBMaker allows one of the following users to activate the **Edit Data** tab.

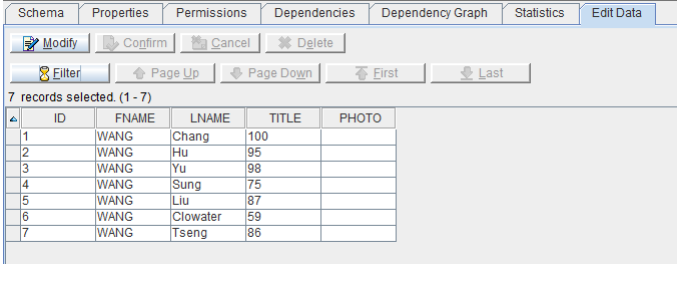
- Users with DBA authority or higher
- The table owner
- Users granted to SELECT privilege

DBMaker allows one of the following users to update data.

- Users with DBA authority or higher
- The table owner
- Users both granted to SELECT privilege and have one of INSERT/UPDATE/DELETE privileges.

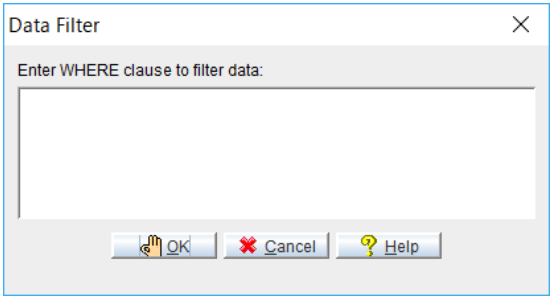
➲ Editing data in a table:

1. Double-click the **Table** node in the tree.
2. Double-click the table you want to modify. The **Schema** page appears.
3. Click the **Edit Data** tab on top of the window. The **Edit Data** page appears.



©Copyright 1995-2026 CASEMaker Inc.

4. To filter the data, click the **Filter** button. The **Data Filter** window appears as shown below.

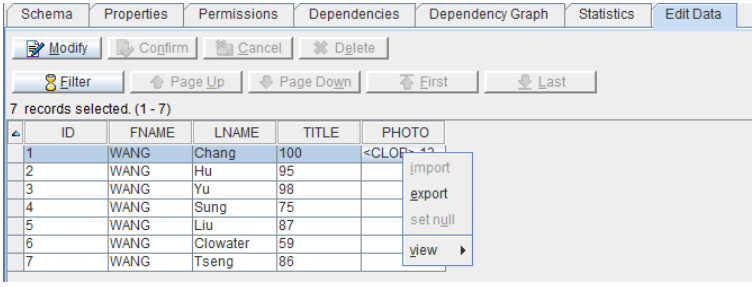


5. Enter a **WHERE** condition to filter the data and click the **OK** button. The main window for the table appears.

NOTE *WHERE conditions must conform to proper SQL syntax. Please refer to the section on Constraint Syntax at the end of this chapter for more information.*

6. Updating data:
   - **a)** Locate the data on the screen using the **Page Up** and **Page Down** buttons.
   - **b)** Click the **Modify** button.
   - **c)** Select the data on the screen.
   - **d)** Edit the data. If the column contains BLOB data, a pop-up menu appears with the option to import, export, set null, or view. BLOB type columns allow content to be stored. FILE type columns give the option to store content as a system file object or to store the file name by creating a user file object. Export allows the user to output BLOB or system file objects. Set null deletes the BLOB or file object. View opens the BLOB or file object in an application according to the selected view.

©Copyright 1995-2026 CASEMaker Inc.



e) Click the **Confirm** button.

7. Deleting data:

a) Locate the data on the screen using the **Page Up**, **Page Down**, **First**, **Last** buttons.

b) Click the **Modify** button.

c) Select the data on the screen.

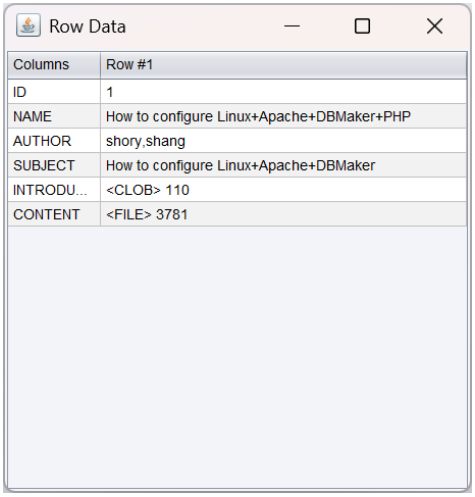
d) Click the **Delete** button.

e) Click the **Confirm** button.

8. Viewing data by row:

a) Locate the data on the screen using the **Page Up**, **Page Down**, **First**, **Last** buttons.

b) Double click the data on the screen.



©Copyright 1995-2026 CASEMaker Inc.

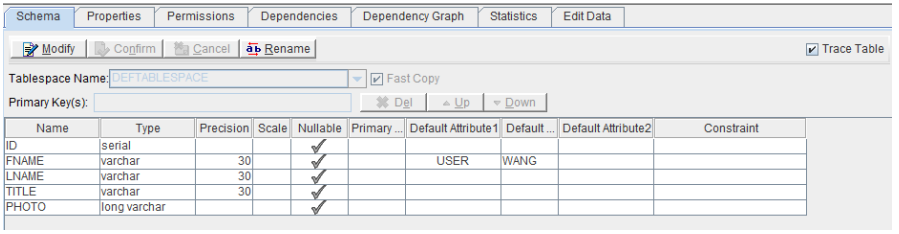
4.7 Renaming a Table

Tables may be renamed in any time after creation. Be sure to check dependencies before renaming a table. Renaming a table will remove all dependencies on the table. DBMaker allows one of the following users to rename a table.

- Users with DBA authority or higher
- The table owner
- Users granted to ALTER privilege

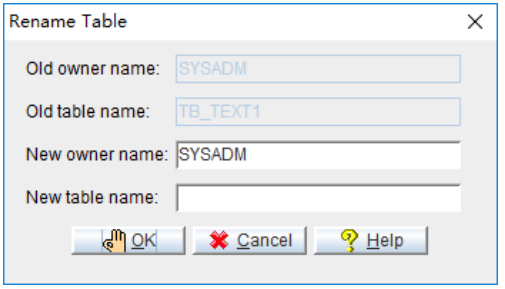
➲ To rename a table:

1. Expand the table node of the tree.
2. Select a table to change the name from the tree. The table **Schema** page appears.



3. Click the **Rename** button. The **Rename Table** window will open.

©Copyright 1995-2026 CASEMaker Inc.



4. Enter a new owner name and table name into the appropriate fields and click the **OK** button.

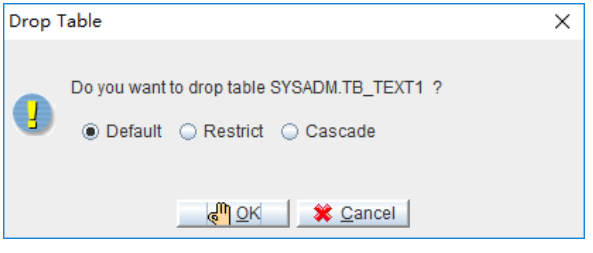
©Copyright 1995-2026 CASEMaker Inc.

4.8 Dropping a Table

You can drop a table that is no longer required in the database. By default, owners of a table and users with RESOURCE authority or higher have drop privilege for the table. A user only with CONNECT authority may not DROP any table.

➲ Dropping a table:

1. Select the table you want to delete from the tree.
2. Click the **Drop** button. Alternatively, right click the tree node of the table and select **Drop Table** from the popup menu. A window appears to confirm that you want to drop the selected table.



3. There are three options for dropping tables. The **Default** option deletes the table without considering dependent object within the table. Dependent objects become invalid after the table is dropped. The **Restrict** option considers dependencies. This option does not permit a table drop when one or more objects depend on the table. The **Cascade** option drops the table and all of its dependent objects. Click a radio button to select a table dropping option.
4. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

4.9 Constraint Syntax

The following table lists comparators to use with the VALUE key word for creating valid table, column, domain, or replication constraints. These comparators can create WHERE conditions for filtering data or creating views, or WHEN conditions for creating triggers.

You can combine simple conditions with logical operators like AND, OR, and NOT to form compound conditions. You can use the keyword AND to combine two search conditions that must both be true. You can use the keyword OR to combine two search conditions when one or the other, but not both, must be true. Finally, you can use the keyword NOT to select rows where a search condition is false.

COMPARATOR	DESCRIPTION
Relational Operators	These may be one of the following: >, >=, <=, <, =, and <>. The relational operator condition is satisfied when the expression on either side of the relational operator fulfills the relation set up by the operator.
BETWEEN	This comparison takes the form: x BETWEEN y AND z. The BETWEEN condition is satisfied when the value or expression to the left of the BETWEEN keyword lies in the range (denoted by the AND keyword) of the two expressions on the right of the keyword.
IN	This comparison takes the form: x IN (y, z, ...). The IN condition is satisfied when the value or expression to the left of the IN keyword is included in the list of values to the right of the keyword.
IS NULL	This takes the form: x IS NULL. The IS NULL condition is satisfied when the value or expression to the left of the IS NULL keywords is a NULL value.
IS NOT NULL	This takes the form: x IS NOT NULL. The IS NOT NULL condition is satisfied when the value or expression to the left of the IS NOT NULL keywords contains a value other than a NULL value.

©Copyright 1995-2026 CASEMaker Inc.

COMPARATOR	DESCRIPTION
LIKE	This takes the form: x LIKE ‘y’ ESCAPE ‘z’. The LIKE condition is satisfied when the string value or expression to the left of the LIKE keyword meets the criteria specified in the case-sensitive quoted string to the right of the keyword. You can use the percent symbol (%) and the underscore symbol (_) as wildcards in the quoted strings. The percent symbol matches zero or more characters, and the underscore symbol matches exactly one character. The ESCAPE clause is optional and allows you to define an escape character so you can include the percent and underscore symbols in the quoted string without having them interpreted as wildcards. To include a single-quote character in a quoted string, use two consecutive single-quotes.
MATCH	This takes the form: x NOT CASE MATCH ‘y’. The MATCH condition is satisfied when the quoted string to the right of the MATCH keyword matches the entire string value or expression to the left of the keyword. The NOT and CASE keywords are optional. The NOT keyword inverts the search results, and the CASE keyword makes the search case-sensitive.
CONTAIN	This takes the form: x NOT CASE CONTAIN ‘y’. The CONTAIN condition is satisfied when the quoted string to the right of the CONTAIN keyword matches any part of the string value or expression to the left of the keyword. The NOT and CASE keywords are optional. The NOT keyword inverts the search results, and the CASE keyword makes the search case-sensitive.

Table 4-1 Constraint Syntax Comparators

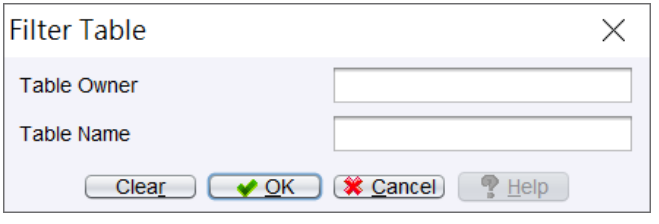
©Copyright 1995-2026 CASEMaker Inc.

4.10 Filtering tables

You can filter tables if there are a large number of tables in your database. Filter tables will search the tables by table owner, table name or both.

➲ Filtering tables:

1. Right click the **Tables** object from the tree.
2. Select the Filter Table from the **popup** menu. A window appears to input table owner and table name to **filter tables**.



3. Input the table owner or table name to **filter tables**.
4. Click **OK**. The Tables object will show filtered and the tree node will only show filtered tables.

➲ Remove filter:

1. Right click the Tables object from the tree.
2. Select the **Remove Filter** from the **popup** menu.

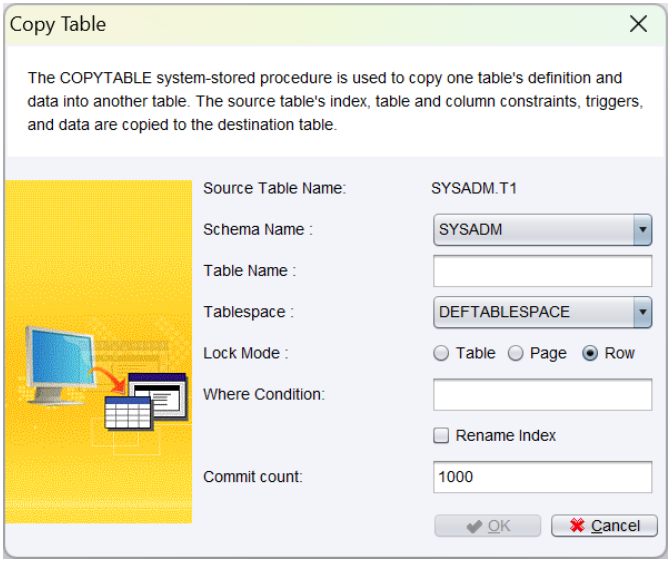
©Copyright 1995-2026 CASEMaker Inc.

4.11 Copying tables

Users can use the **Copy Table** to copy tables, this based on the stored procedure **COPYTABLE**. Copy table definition, data, index, trigger and create a new table.

➲ Copying a table:

1. Click the **Tables** object from the tree.
2. Right click the tree node of table and select **Copy table** from the popup menu. The **Copy Table** window appears.



3. Select the **Schema Name**, **Tablespace** and enter the **Table Name**. The **Where Condition** textbox allows users to filter data when creating the new table. The **Commit count** textbox will separate the commit by the defined value.
4. Click the **OK** button to copy and create a new table.

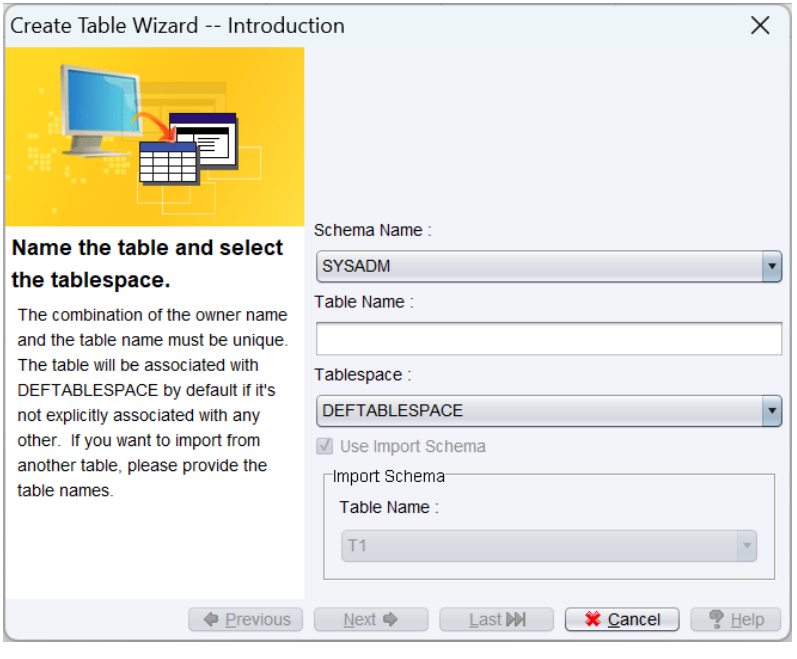
©Copyright 1995-2026 CASEMaker Inc.

4.12 Duplicate tables

Users can use the **Duplicate Table** to create a new table. The Duplicate Table is used to help users to simplify the creation of a new table. It can help users to import schema from the selected table, means the column name and data type will be automatically defined. **Copy Table** is used when users need an exactly same table; **Duplicate Table** will refer on the original table and can modify some of the content.

➲ Duplicate tables:

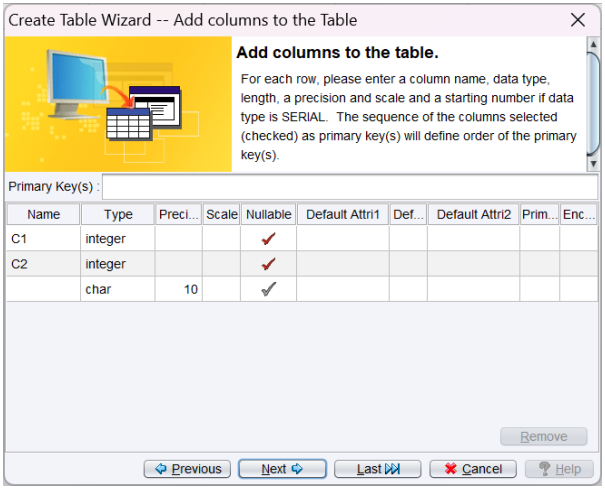
1. Click the **Tables** object from the tree.
2. Right click the tree node of table and select **Duplicate Table** from the popup menu. The **Create Table Wizard** window appears.



©Copyright 1995-2026 CASEMaker Inc.

3. The **Use Import Schema** checkbox will be automatically selected and the **Table Name** will fill in with the selected table. Users only need to decide the **Schema Name**, **Table Name** and **Tablespace**.

4. Click the **Next** button, the **Add columns to the Table** window appears, the column from the original table will be added automatically, users can modify column or add new column in this window.



NOTE: The Duplicate Table will only add the column name and data type from original table, indexes, column constraints, primary key, foreign key will not be added.

5. Click the **Next** button. The **Column constraints** window appears. If users do not need to add constraints, a foreign key, or set advanced options for the table, please click the **Last** button. The **Final Review** window appears.

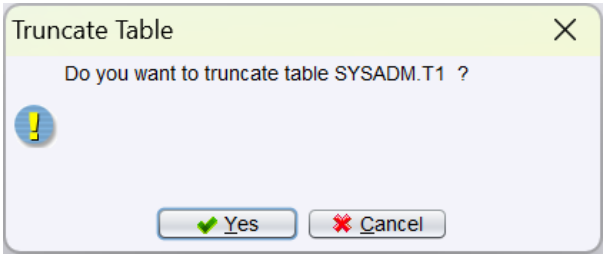
©Copyright 1995-2026 CASEMaker Inc.

4.13 Truncating tables

Users can use the **Truncate Table** to delete all data in the table. This is based on the **TRUNCATE** command. **TRUNCATE** is similar to the **DELETE** statement with no WHERE clause; however, TRUNCATE is faster and uses fewer resources.

➲ Truncating a table:

1. Click the **Tables** object from the tree.
2. Right click the tree node of table and select **Truncate Table** from the popup menu. A window appears to confirm truncating the selected table.



3. Click the **Yes** button.

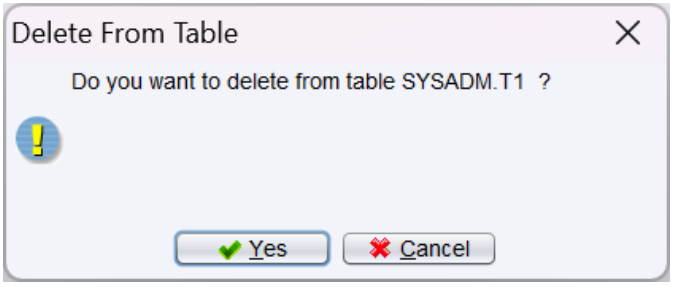
©Copyright 1995-2026 CASEMaker Inc.

4.14 Deleting from tables

Users can use the **Delete From Table** to delete all data from a table.

➲ Deleting a table:

1. Click the **Tables** object from the tree.
2. Right click the tree node of table and select **Delete From Table** from the popup menu. A window appears to confirm deleting data from the selected table.



3. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.

5 Working with Views

DBMaker provides the ability to define a virtual table, called a *view.* The view definition is a logical construct that is computed or collated (i.e., derived) from one or more tables and functions like a table but it is not stored in the database as a table itself. You can create views for flexible data queries on tables. You can use views to browse data in multiple tables. JDBA Tool allows you to create, browse, update, and drop views.

©Copyright 1995-2026 CASEMaker Inc.

5.1 Creating a View

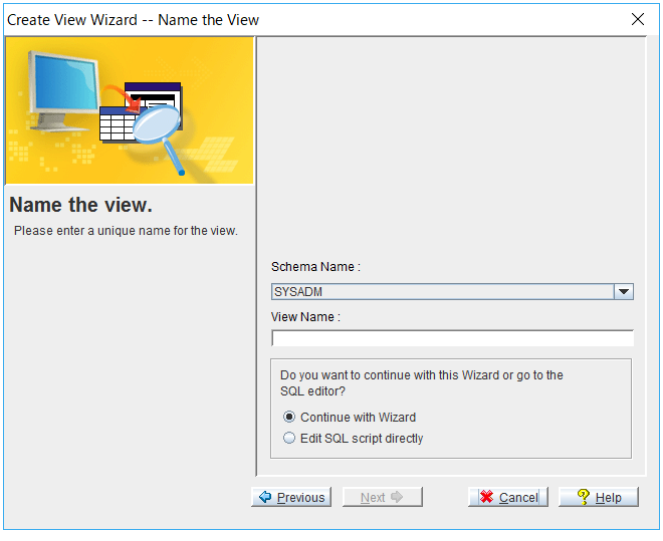
After you enter data into the database tables, you can use the tables to create views. By default, users with RESOURCE authority or higher have CREATE privilege for the view. A user only with CONNECT authority may not create any view.

➲ Creating a view:

1. Select the **View** object from the tree. The owner name, view name and time of creation for all views on the database appear on a table in the **View** page.
2. Click the **Create** button. The **Create View Wizard – Introduction** window appears.

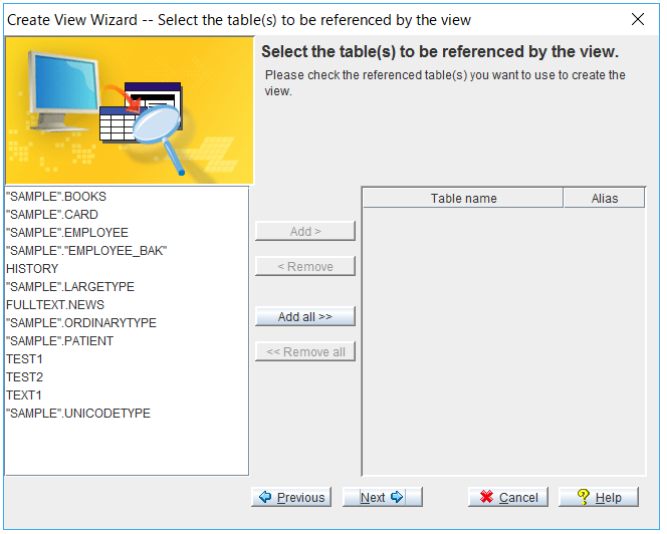
3. Click the **Next** button. The **Name the View** window appears.

©Copyright 1995-2026 CASEMaker Inc.



4. Select the schema name from the **Schema Name** drop down list.
5. Enter the name of the view in the **View Name** field.
6. Select the **Continue with Wizard** radio button or select the **Edit SQL script directly** radio button to jump directly to the last step of the wizard (see step 14).
7. Click the **Next** button. The **Select the table(s) to be referenced by the view** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



8. Select tables to use in the view by double-clicking the table names on the left. Alternatively, use the buttons in the middle to add or remove columns. Selected tables appear on the right.

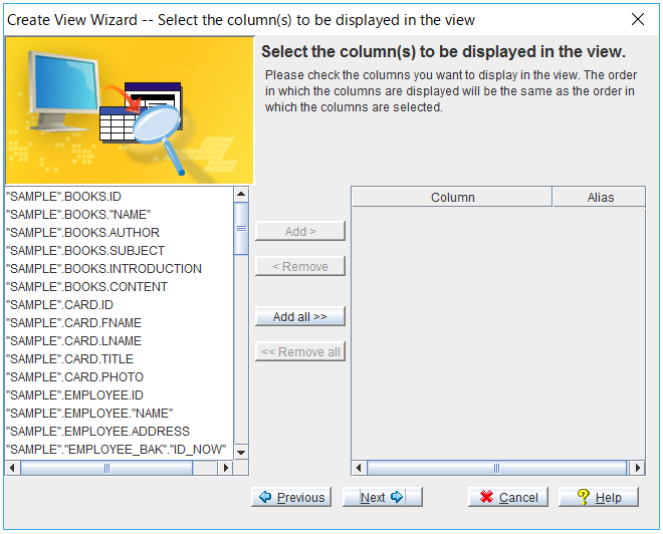
9. You can add a table to a view more than once, however, an alias for the table is required. The alias is a new name for the table that allows referencing columns in the table more than once. To add an alias:

   a) Select a table already added to the view.

   b) Enter an alias name into the field under the **Alias** heading. The default name appears as **TableAliasName**.

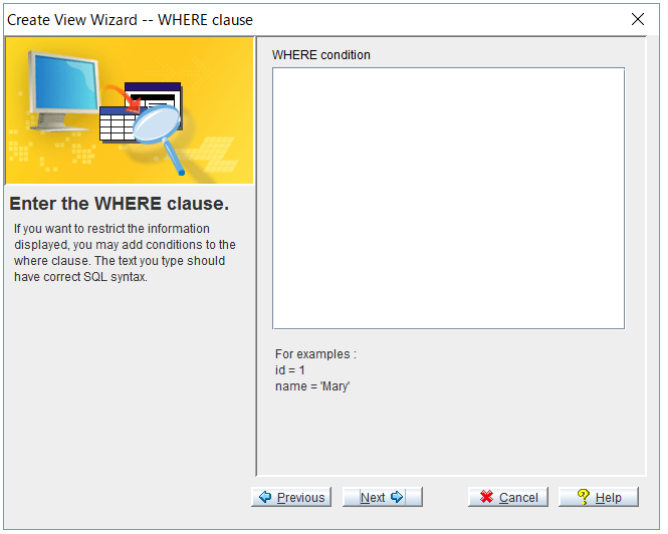
10. Click **Next**. The **Select the column(s) to be displayed in the view** window appears.

©Copyright 1995-2026 CASEMaker Inc.



11. Select columns to include in the view by double-clicking the column names on the left. Alternatively, use the buttons in the middle to add or remove columns. Selected columns appear on the right. Columns added more than once must have an alias. Create column aliases the same way as table aliases.
12. Click the **Next** button. The WHERE clause window appears.

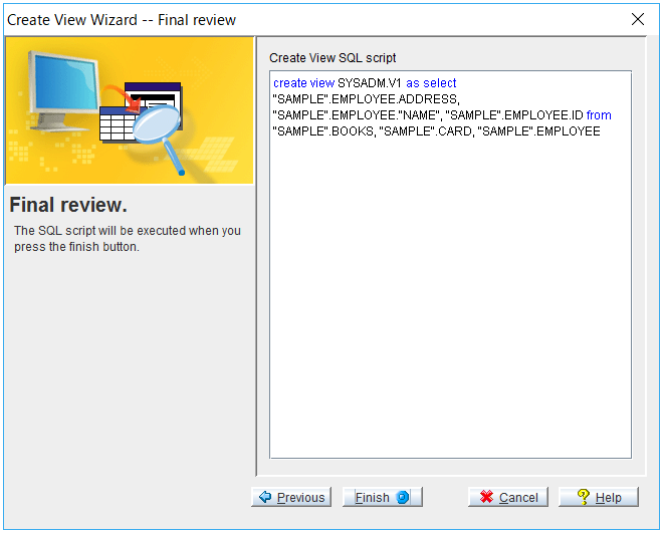
©Copyright 1995-2026 CASEMaker Inc.



13. Enter a condition by which the data in the view is to be filtered.

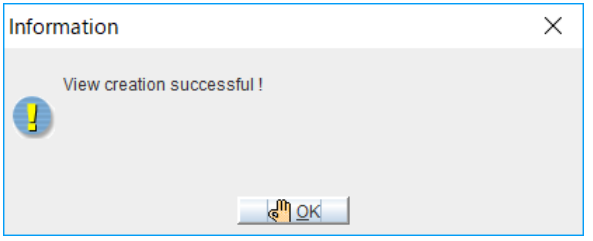
NOTE *WHERE statements must conform to SQL syntax rules. Please see the section on Constraint Syntax near the end of Chapter 4.*

14. Click the **Next** button. The **Final review** window appears.



©Copyright 1995-2026 CASEMaker Inc.

15. If you wish to alter the view data after reviewing the SQL script, enter a new script into the **Create View SQL script** field or click the **Previous** button to display previous wizard screens.
16. Click the **Finish** button after reviewing the SQL script.
17. Click the **OK** button.



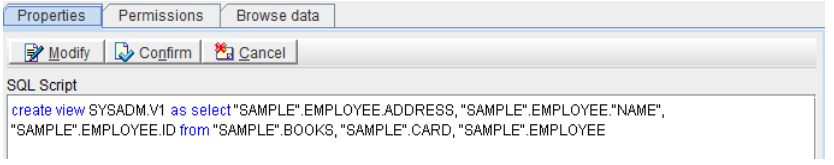
©Copyright 1995-2026 CASEMaker Inc.

5.2 Checking View Properties

You can check the properties of a view including the tables and columns used as well as ownership.

➲ Checking view properties:

1. Click the **View** object in the tree. All the views in the database appear.
2. Select a view. The **Properties** page appears.



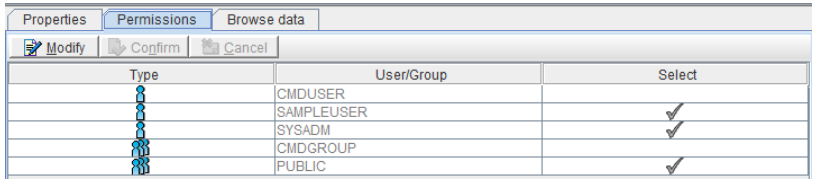
©Copyright 1995-2026 CASEMaker Inc.

5.3 Granting View Privileges

You can allow other users and groups to access a view by granting privileges. You can also revoke access to a view for a user/group at any time. By default, owners of a view and users with RESOURCE authority or higher have grant/revoke privilege for the view. A user only with CONNECT authority may not grant or revoke SELECT privilege to the any view.

➲ Granting view privileges:

1. Select the **View** object in the tree.
2. Select a view. The **Properties** page appears.
3. Click the **Permissions** tab on the top of the window. The permission details appear.



4. Click the **Modify** button.
5. Select the user/group that you want to grant or revoke select privilege to the view.
6. Click the **Select** column next to the selected User/Group. A check mark indicates that the user or group has permission on the view.
7. Click the **Confirm** button.

©Copyright 1995-2026 CASEMaker Inc.

5.4 Browsing Views

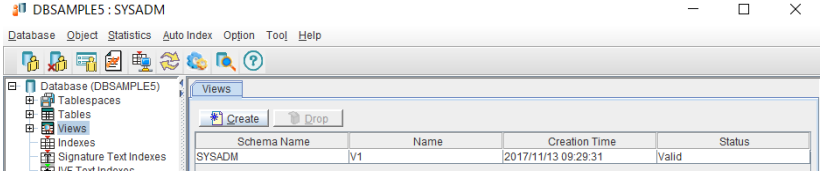
You can browse the data contained in a view quickly and efficiently. You can sort this data using the filter data function in JDBA Tool. For optimal efficiency, break exceptional large views into several parts. This will increase the ease and speed that you will be able to browse the view's data. The maximum number of records that can be browsed in a single instance is 5,000.

DBMaker allows one of the following users to activate the **Browse Data** tab.

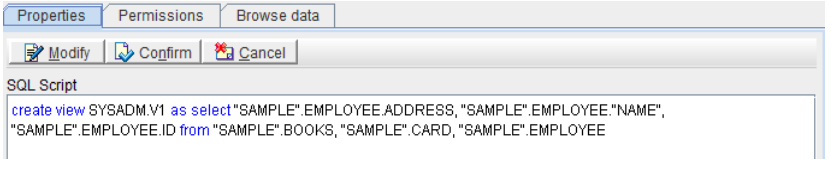
- Users with DBA authority or higher
- The view owner
- Users granted to SELECT privilege

➲ Browsing a view:

1. Click the **Views** icon from the tree. A list of all the views in the database appears beneath the **Views** icon.

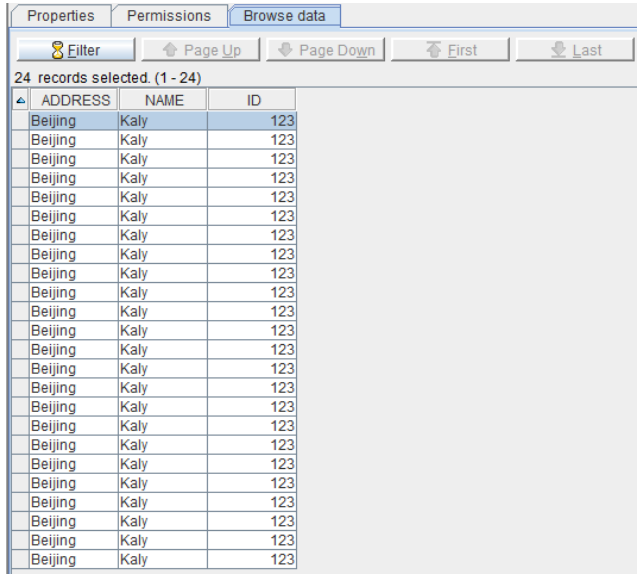


2. Select the view you want to browse from the list of views. The view appears in the panel to the right.



3. Click the **Browse data** tab in the panel on the right. The **Browse data** window opens in the panel on the right.

©Copyright 1995-2026 CASEMaker Inc.



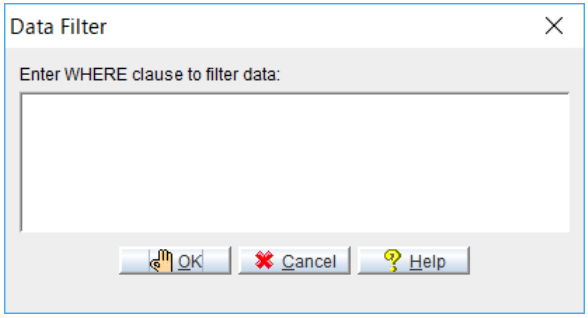
ADDRESS	NAME	ID
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123
Beijing	Kaly	123

a) To quickly browse the view by page, use the **Page Up** and **Page Down** buttons.

b) Click the **First** button to jump to the first entry in the view.

c) Click the **Last** button to jump to the last entry in the view.

4. Click the **Filter** button to set sort rules for browsing the view. The **Data Filter** dialog box appears.



a) Enter a WHERE clause or an ORDER BY clause to filter the data when browsing a view.

©Copyright 1995-2026 CASEMaker Inc.

NOTE *WHERE statements must comply with SQL syntax rules. Please see the section on Constraint Syntax near the end of Chapter 4 for information.*

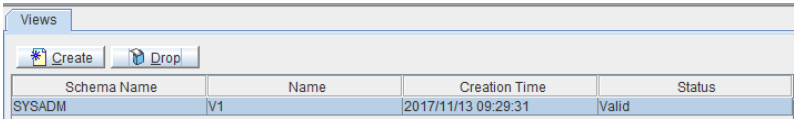
©Copyright 1995-2026 CASEMaker Inc.

5.5 Dropping a View

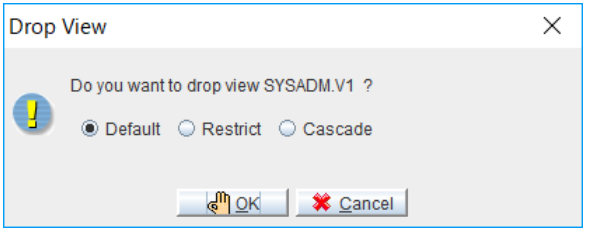
Delete unwanted views using the following procedure. By default, owners of a view and users with RESOURCE authority or higher have DROP privilege for the view. A user only with CONNECT authority may not drop any view.

➲ Dropping a view:

1. Click the **View** object in the tree. All views in the database are displayed.
2. Select a view.



3. Click the **Drop** button at the top of the window. The **Drop View** dialog box appears. Alternatively, right click the view to drop from the tree, a pop-up menu appears. Select **Drop** from the pop-up menu.



4. There are three options for dropping views. The **Default** option deletes the views without considering dependent object within the views. Dependent objects become invalid after the view is dropped. The **Restrict** option considers dependencies. This option does not permit a views drop when one or more objects depend on the views. The **Cascade** option drops the views and all of its dependent objects. Click a radio button to select a views dropping option.
5. Click the **OK** button. The remaining views in the database are displayed.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

6 Working with Indexes

An index facilitates random access of table rows. An index is a sub-table of the table containing the accessing rows. The index table contains the key columns of its parent table. Indexes are particularly useful for tables that hold large amounts of data, or tables that are frequently accessed. An index can be *unique* or *non-unique*. In a unique index, no more than one row can have the same key value, with the exception that any number of rows may have NULL values.

DBMaker supports auto indexes, its behavior is similar to non-unique index, but it can be automatically created or dropped by the auto index daemon. If set AUTOCOMMIT ON, DBMaker only requires *Update(U)* lock when creating an auto index, which means DBMaker allows other users to query the table simultaneously.

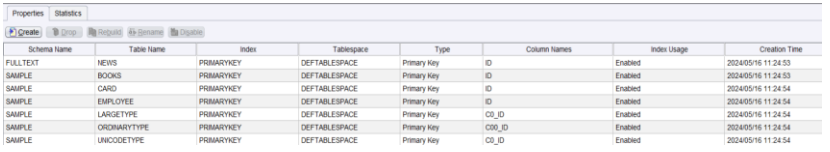
©Copyright 1995-2026 CASEMaker Inc.

6.1 Creating an Index

Create an index when one or more rows in a table will be used in search queries. Indexing the rows will also allow sorting of data to be performed. By default, users with RESOURCE authority or higher have CREATE privilege for the index. A user only with CONNECT authority may not create any index.

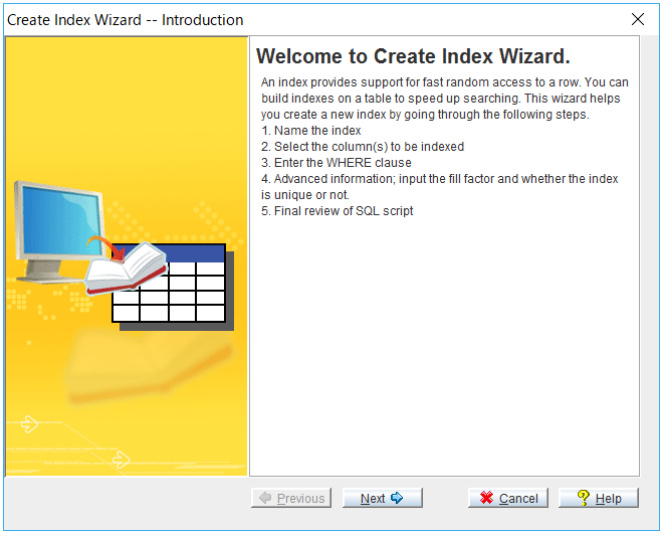
➲ Creating an index:

1. Click the **Indexes** object in the tree. DBMaker lists the database's indexes.



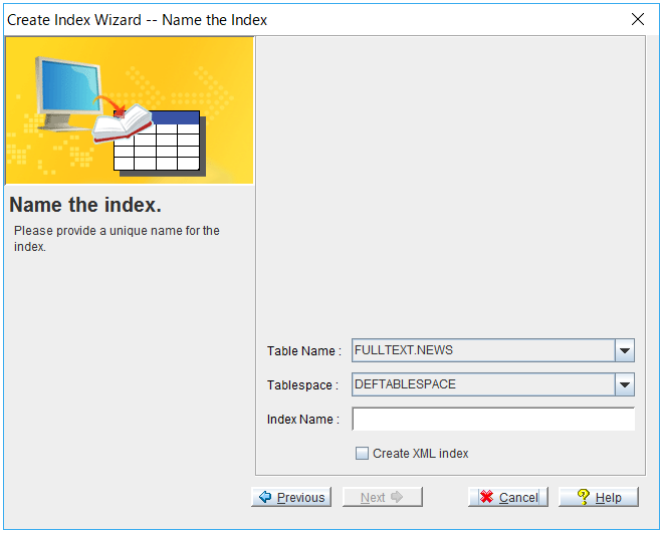
Schema Name	Table Name	Index	Tablespace	Type	Column Names	Index Usage	Creation Time
FULLTEXT	NEWS	PRIMARYKEY	DEFTABLESPACE	Primary Key	ID	Enabled	2024/05/16 11:24:53
SAMPLE	BOOKS	PRIMARYKEY	DEFTABLESPACE	Primary Key	ID	Enabled	2024/05/16 11:24:53
SAMPLE	CARD	PRIMARYKEY	DEFTABLESPACE	Primary Key	ID	Enabled	2024/05/16 11:24:54
SAMPLE	EMPLOYEE	PRIMARYKEY	DEFTABLESPACE	Primary Key	ID	Enabled	2024/05/16 11:24:54
SAMPLE	LARGETYPE	PRIMARYKEY	DEFTABLESPACE	Primary Key	C0_ID	Enabled	2024/05/16 11:24:54
SAMPLE	ORDINARYTYPE	PRIMARYKEY	DEFTABLESPACE	Primary Key	C00_ID	Enabled	2024/05/16 11:24:54
SAMPLE	UNICODETYPE	PRIMARYKEY	DEFTABLESPACE	Primary Key	C0_ID	Enabled	2024/05/16 11:24:54

2. Click the **Create** button. The **Create Index Wizard** appears.



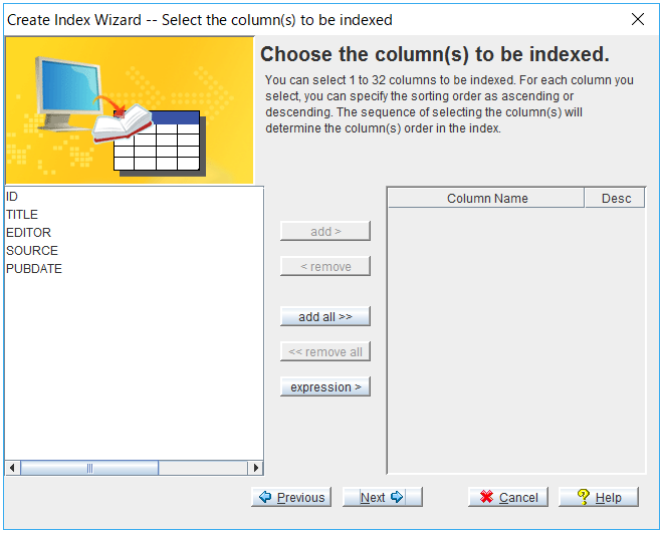
©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Name the Index** window appears.



4. The index will refer to the table name that you select from the **Table Name** menu.
5. Select a tablespace to build the index in. Building an index on a different tablespace may result in better disk I/O, especially if the tablespaces reside on separate disks.
6. Enter a name for the Index in the **Index Name** field.
7. To improve the performance of XML queries, select the **Create XML Index** radio button. This creates a special XML index on the XML columns.
8. Click the **Next** button. The **Select the column(s) to be indexed** window appears.

©Copyright 1995-2026 CASEMaker Inc.

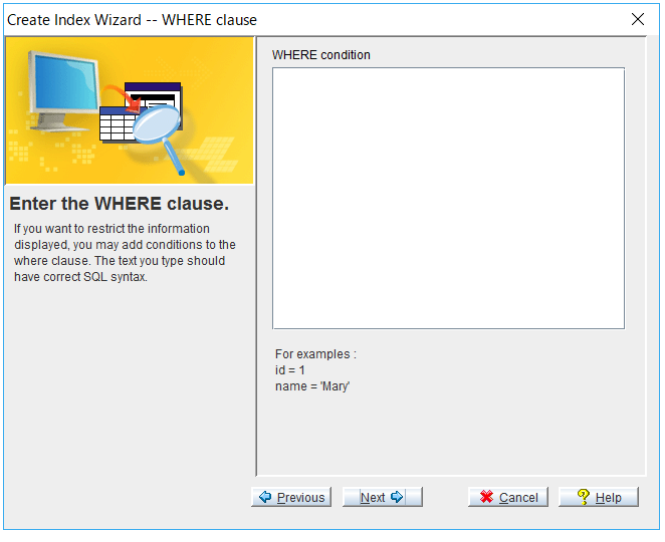


9. A list of available columns on the selected table appears on the left hand list box. Select the columns that you want to use in the index by double clicking on the column name or selecting the column and clicking the **Add** button. The **Add all** button adds all columns to the index. All the columns selected for the creation of the index are displayed in the list box to the right. Columns in the index can be removed by double-clicking the column name or selecting the column and clicking the **Remove** button.
10. The **expression** buttion allows you to input an express for the index.
11. To change the sorting order to descending for a column:
    a) Select a column from the list box to the right.
    b) Click the cell in the **Desc** column for the selected column in the list box to the right. A check mark in the box indicates that the order for that column's index is descending.

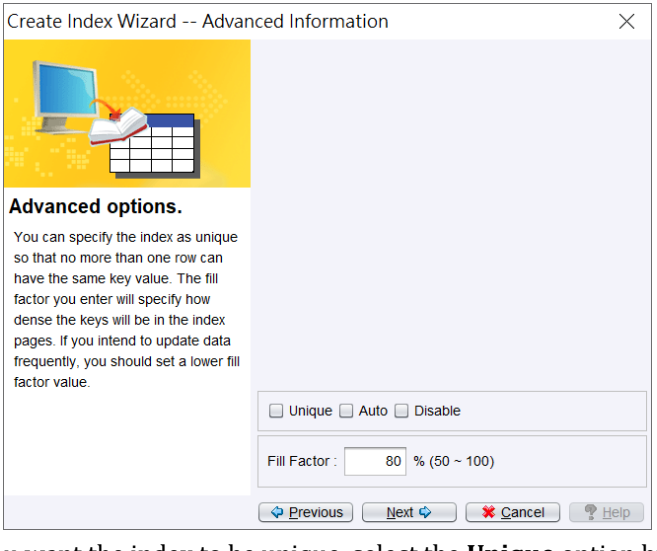
    NOTE *By default, the sort order is ascending.*

©Copyright 1995-2026 CASEMaker Inc.

12. Click the **Next** button. The **WHERE clause** window appears.



13. Click the **Next** button. The **Advanced Information** window appears.

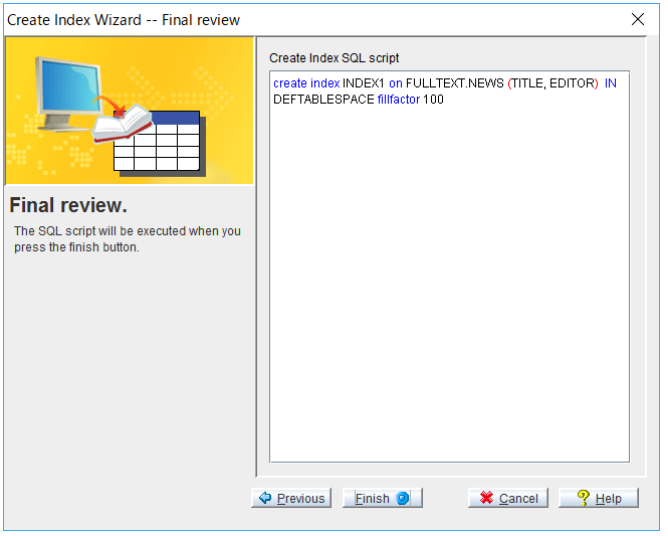


14. If you want the index to be unique, select the **Unique** option button. If you want the index to be an auto index, select the **Auto** option button. If

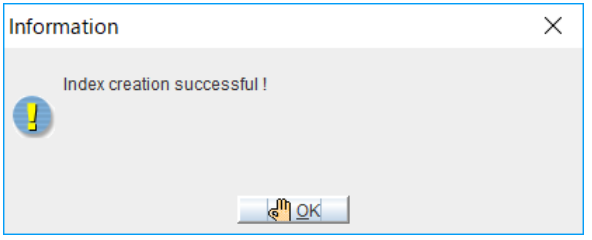
©Copyright 1995-2026 CASEMaker Inc.

you want the index disabled when it's created, select the **Disable** option button.

15. To change the default fill factor, enter a percentage in the **Fill Factor** field.
16. Click the **Next** button. The **Final review** dialog box appears.



17. Click the **Finish** button after checking that the SQL script is correct. The **Information** dialog box appears.



18. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

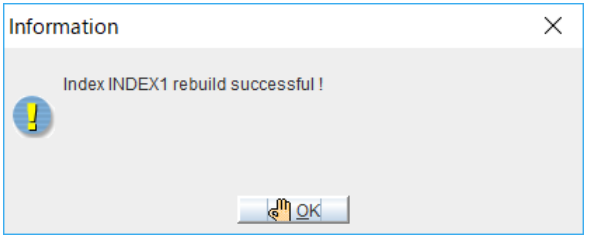
6.2 Rebuilding an Index

If you have made a number of changes to the table or disabled index, you should rebuild the index. Through the rebuilding process, the old index replaced with a new one. By default, owners of an index and users with RESOURCE authority or higher can rebuild the index. A user only with CONNECT authority may not rebuild any index.

➲ Rebuilding an index:

1. Click the **Index** object in the tree. DBMaker lists of all indexes in the database on the **Properties** page.
2. Select an index to rebuild.
3. Click the **Rebuild** button. The **Rebuild** message box appears.

4. Click the **Yes** button. The **Information** message box appears.



5. Click the **OK** button.

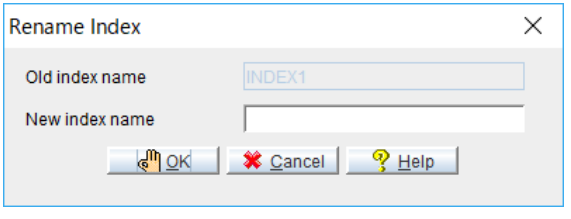
©Copyright 1995-2026 CASEMaker Inc.

6.3 Renaming an Index

Any index may be renamed. The renaming only affects the index name in the system catalog. Renaming does not rebuild the index in the database. By default, owners of an index and users with RESOURCE authority or higher can rename the index. A user only with CONNECT authority may not rename any index.

➲ Renaming an index:

1. Click the **Index** object in the tree. A list of all indexes in the database is displayed on the **Properties** page.
2. Select an index to be renamed.
3. Click the **Rename** button. The Rename Index message box appears.



4. Input the new index name in the proper filed.
5. Click the **OK** button.

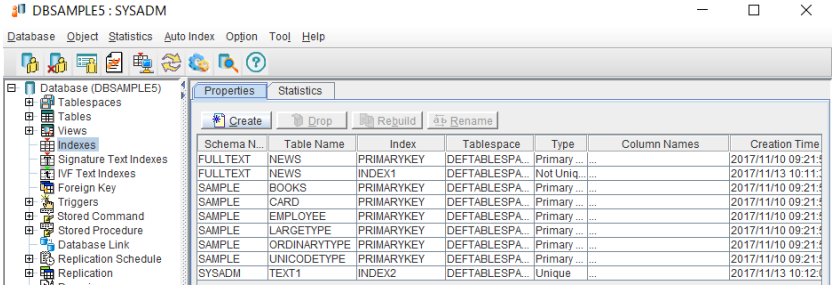
©Copyright 1995-2026 CASEMaker Inc.

6.4 Updating Index Statistics

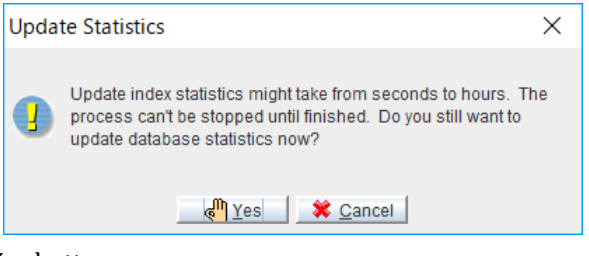
If your database is extremely large, it will take a lot of time to update the statistics values of all schema objects. An alternative method is to update statistics on only those specific schema objects that have been modified since the last update. You can also set the sampling rate, which is the percentage of the records used to calculate the statistics.

➲ Updating index statistics:

1. Click the **Index** object in the tree. A list of all the indexes in the database is displayed.
2. Click the **Statistics** tab. The **Statistics** page appears.

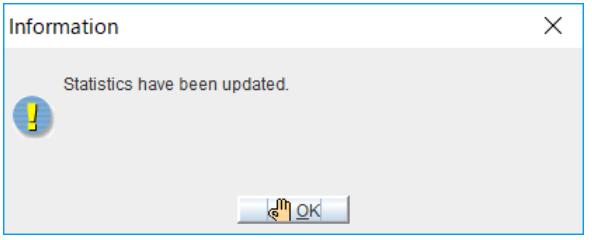


3. Select an index, select the **Sample Ratio** box and enter a percentage to sample when you want to change the percentage of the data used to calculate updated statistics.
4. Click the **Update Statistics** button. The following window appears.



5. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.



6. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

6.5 Dropping an Index

You can drop an index that is no longer required in a database. By default, owners of an index and users with RESOURCE authority or higher have DROP privilege for the index. A user only with CONNECT authority may not drop any index.

➲ Dropping an Index:

1. Click the Index object in the tree. The indexes in the database are listed.
2. Select the index to be dropped and click the **Drop** button. The Drop Index dialog box appears.

3. Click the **Yes** button. All the remaining indexes in the database are displayed on the **Properties** page.

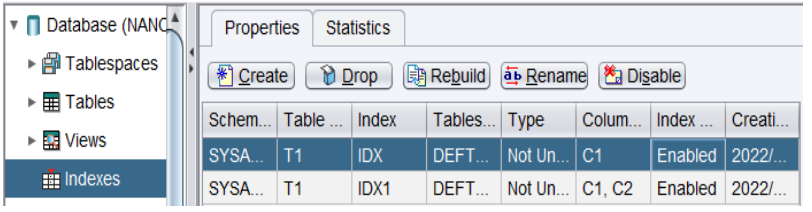
©Copyright 1995-2026 CASEMaker Inc.

6.6 Disabling an Index

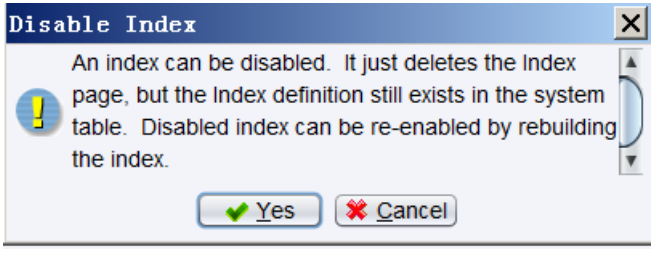
If an index is temporarily unnecessary, users can disable the index. Disable an index won't drop the index, only make the index deactivated. Insert, delete, update on column won't affect the disabled index. By default, owners of an index and users with RESOURCE authority or higher can disable the index. A user only with CONNECT authority may not disable any index.

➲ Disabling an index:

1. Click the Index object in the tree. A list of all indexes in the database is displayed on the **Properties** page.

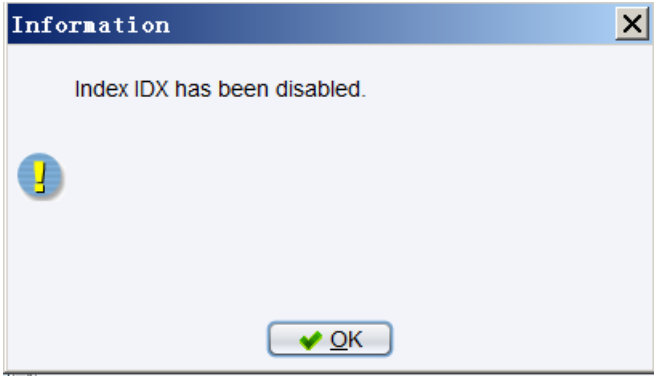


2. Select an index to disable.
3. Click the **Disable** button. The Disable Index message box appears.



4. Click **YES**. The Information message box will appear.

©Copyright 1995-2026 CASEMaker Inc.



5. Click **OK**.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

7 Working with Signature Text Indexes

A text index is a mechanism that provides fast access to rows of a table containing columns with one or more words or phrases as text. Text indexes contain a representation of all the text found in the text columns on which they are based. However, the data is encoded and structured to make retrieval much faster than would be otherwise possible when handled directly from the table. Once you create a text index on a table, its operation is transparent to users of the database. The DBMS uses the text index to improve full-text query performance whenever possible.

Signature text indexes exist within the same tablespace as the columns that are indexed. Signature text indexes are well suited for building on columns that contain a combined total less than 200 MB. For larger text indexes, consider building an inverted file (IVF) text index. For detailed information on IVF text indexes, please refer to *Working with IVF Text Indexes*.

©Copyright 1995-2026 CASEMaker Inc.

7.1 Creating a Signature Text Index

Signature text indexes can be built on all character type columns, including CHAR, VARCHAR, LONG VARCHAR, NCHAR, NVARCHAR, NCLOB, and FILE types. A table can have multiple text indexes, and text indexes can be built on multiple columns. By default, users with RESOURCE authority or higher have CREATE privilege for the signature text index. A user only with CONNECT authority may not create any signature text index.

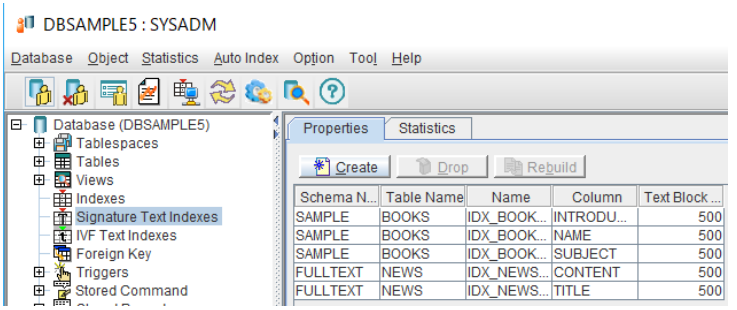
To ensure optimal performance for a signature text index, specify the total text size and scale. The total text size approximates the total number of megabytes of storage that used by the columns to be indexed. The total text size can range from 10 to 200 MB, and the default total text size is 32 MB.

The scale is the relative size of the text index to the total text size expressed as a percentage. Generally, query performance improves as the scale increases. The default scale is 50 percent.

Text indexes may be sorted using the **order by** feature.

➲ Creating a Signature Text Index:

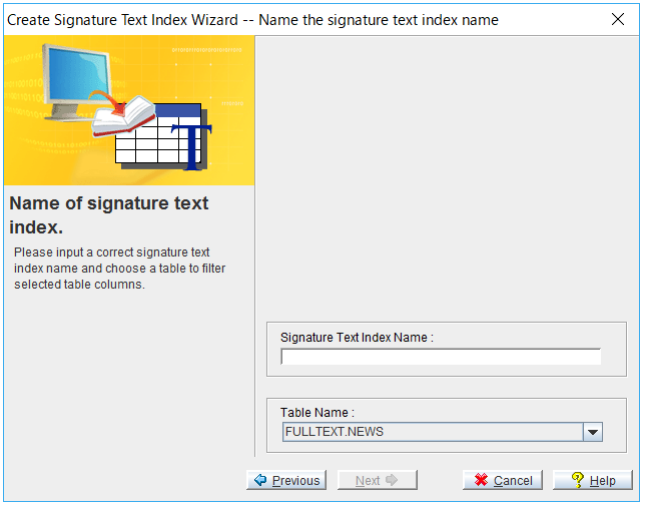
1. Click the **Signature Text Indexes** object in the tree. The **Properties** page appears.



2. Click the **Create** button. The **Introduction** window for the **Create Signature Text Index Wizard** appears.

©Copyright 1995-2026 CASEMaker Inc.

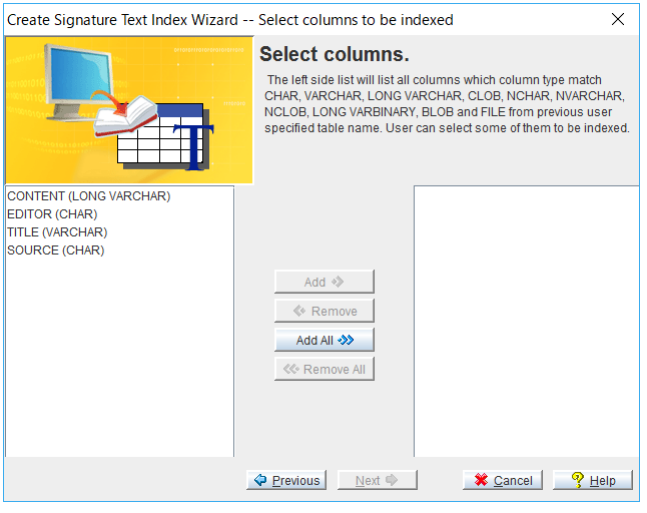
3. Click the **Next** button. The **Name the signature text index window** appears.



4. Select the name of the table from which the text index is to be created from the **Table Name** menu.
5. Enter the name of the text index in the **Text Index Name** field.
6. Click the **next** button**.** A list of available columns on the selected table appears on the left hand list box. Select the columns that you want to

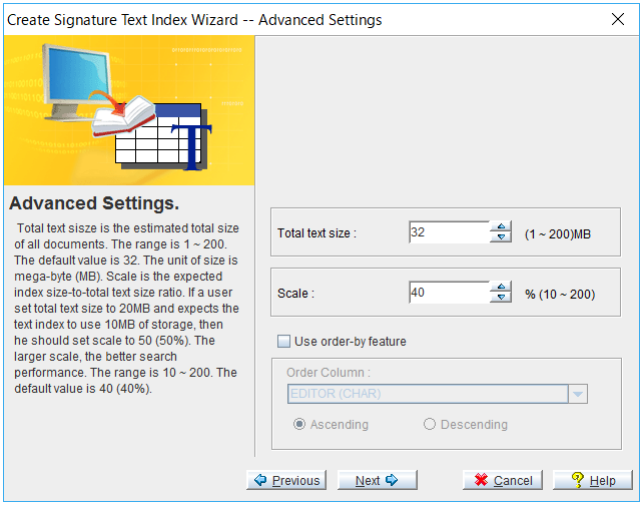
©Copyright 1995-2026 CASEMaker Inc.

use in the signature text index by double-clicking the column name or selecting the column and clicking the **Add** button. The **Add all** button adds all columns to the index. All the columns selected for the creation of the text index are displayed in the list box to the right. Columns in the text index can be likewise removed by double-clicking the column name or selecting the column and clicking the **Remove** button.



7. Click the **Next** button. The **Advanced Settings** window appears.

©Copyright 1995-2026 CASEMaker Inc.



8. Changing the signature text index settings:

   - To change the total text size, enter a value for **Total Text Size** between 1 and 200.
   - To change the scale, enter a value for **Scale** between 10 and 200.

9. To display data as it relates to another column in the table, select the **Use order-by feature** check box.

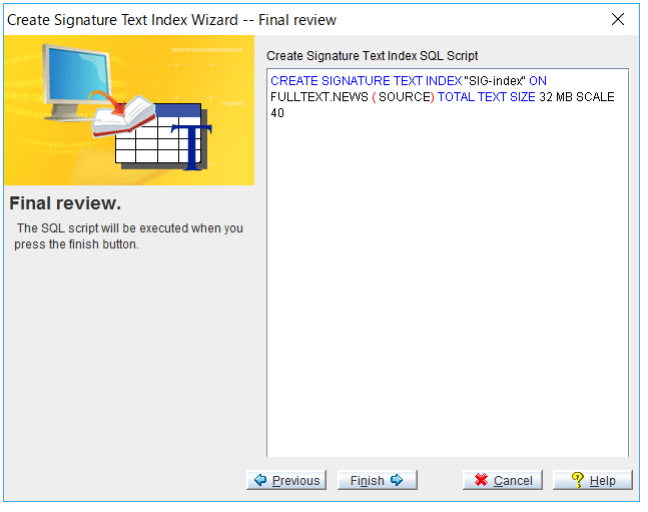
   a) Select the column that the text index will follow the order of from the **Order Column** menu.

   b) Select the **Ascending** option button to order data in the text index by the Order Column's ascending order.

   c) Select the **Descending** option button to order data in the text index by the Order Column's descending order.

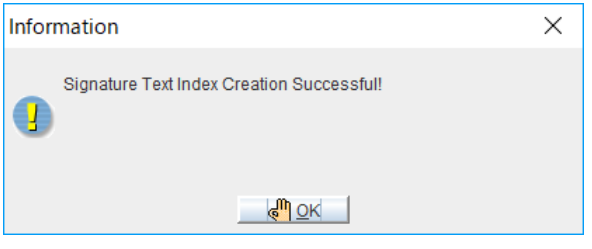
10. Click the **Next** button. The **Final review** window appears.

©Copyright 1995-2026 CASEMaker Inc.



11. To alter the SQL script, enter a new script into the **Create Signature Text Index SQL script** field or click the **Previous** button to display the previous wizard screens.

12. To complete the view creation after reviewing the SQL script, click the **Finish** button. The **Information** dialog box appears.



13. Click the **OK** button.

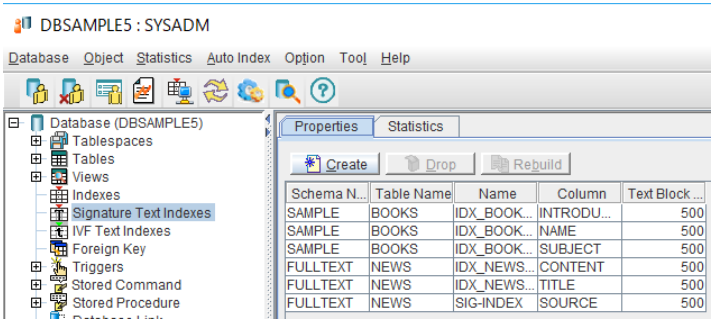
©Copyright 1995-2026 CASEMaker Inc.

7.2 Rebuilding a Signature Text Index

If a number of changes have been made to the table or to the column on which a text index was created, then the text index should be rebuilt. When a text index is rebuilt, it will be updated to include all changes to the table or column. By default, owners of a signature text index and users with RESOURCE authority or higher can rebuild the signature text index. A user only with CONNECT authority may not rebuild any signature text index.

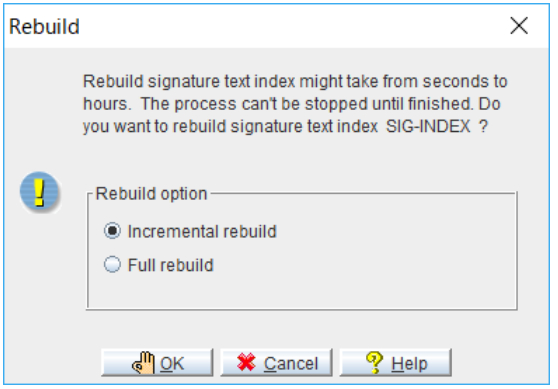
➲ Rebuilding a signature text index:

1. Click the **Signature Text Indexes** object in the tree. The **Properties** page appears.
2. Select a signature text index to rebuild.

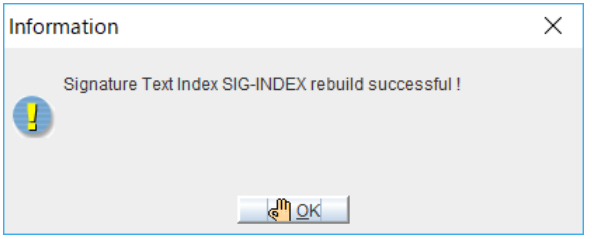


3. Click the **Rebuild** button. The **Rebuild** dialog appears.

©Copyright 1995-2026 CASEMaker Inc.



4. Select either the **Incremental rebuild** or **Full rebuild** radio button.
5. Click the **OK** button. A confirmation window appears as shown below.



6. Click the **OK** button.

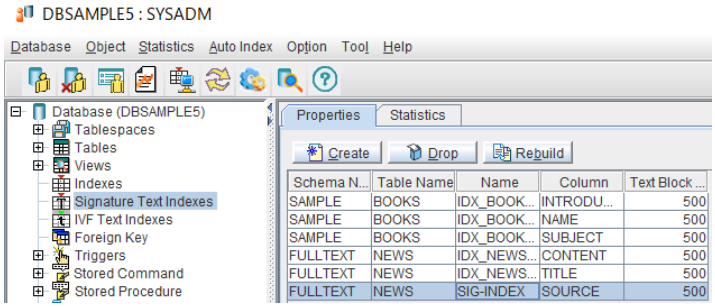
©Copyright 1995-2026 CASEMaker Inc.

7.3 Dropping a Signature Text Index

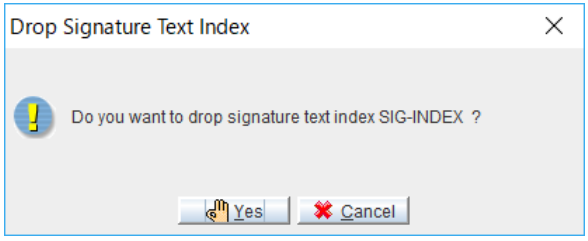
You can drop a text index that is no longer required. By default, owners of a signature text index and users with RESOURCE authority or higher have DROP privilege for the signature text index. A user only with CONNECT authority may not drop any signature text index.

➲ Dropping a signature text index:

1. Click the **Signature Text Indexes** object in the tree. All the text indexes in the database are displayed in the **Properties** page.
2. Select a **Signature Text Index** to be dropped.



3. Click the **Drop** button. A confirmation window appears to confirm that you want to drop the signature text index.



4. Click the **Yes** button. The remaining signature text indexes in the database are displayed.

©Copyright 1995-2026 CASEMaker Inc.

7.4 Checking Signature Text Index Statistics

The statistics of a signature text index may periodically be checked. Statistics include the table owner and name, the text index name, the number of text blocks and the average block size.

➲ Viewing text index statistics:

1. Select **Signature Text Indexes** from the tree. The **Properties** page appears.
2. Click the **Statistics** tab. The statistics of the text indexes appears. These include the names of the signature text indexes, the average block size of the indexes and the number of text blocks.

Properties | Statistics

Statistics for Signature Text Index(es):

Schema Name	Table Name	Name	# Text Block	Average Block Size
SAMPLE	BOOKS	IDX_BOOKS_INTRO	3	512
SAMPLE	BOOKS	IDX_BOOKS_NAME	3	34
SAMPLE	BOOKS	IDX_BOOKS_SUBJECT	3	44
FULLTEXT	NEWS	IDX_NEWS_CONTENT	9	1327
FULLTEXT	NEWS	IDX_NEWS_TITLE	7	55
FULLTEXT	NEWS	SIG-INDEX	7	40

©Copyright 1995-2026 CASEMaker Inc.

8 Working with IVF Text Indexes

DBMaker builds inverted file (IVF) text indexes on files separate from the tablespace where the columns are stored. As a result, query performance is significantly better for larger text indexes. You need to create a logical path for storing the IVF text index files before creating an IVF text index. Specify the logical paths with the JConfiguration Tool or by modifying the DMConfig.ini file. First specify a path, and then DBMaker handles the creation of files for use in the IVF text index.

Since IVF text indexes are a part of the operating system's file system, they may be susceptible to fragmentation. Be sure to specify a logical path on a reserved partition and periodically defragment that partition to ensure optimal performance.

Once you create an IVF text index on a table, its operation is transparent to users of the database. The DBMS uses the IVF text index to improve full-text query performance whenever possible.

©Copyright 1995-2026 CASEMaker Inc.

8.1 Creating an IVF Text Index

IVF text indexes can be built on all character type columns, including CHAR, VARCHAR, LONG VARCHAR, NCHAR, NVARCHAR, NCLOB, and FILE types. A table can have multiple text indexes, but a text index can only be built on a column. If you need to perform full-text search for other columns, you should build an index for each column. By default, users with RESOURCE authority or higher have CREATE privilege for the IVF text index. A user only with CONNECT authority may not create any IVF text index.

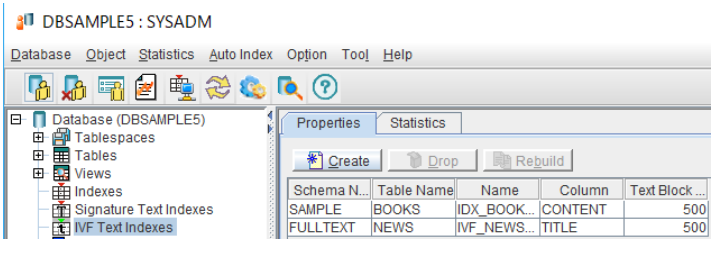
The three parameters used for IVF text indexes include the storage path, total text size, and scale. The storage path is the location of the IVF text index in the file system. You must specify a path in the configuration file before using an IVF text index. For more information on the storage path for IVF text indexes, please refer to the *JConfiguration Tool Reference*.

The total text size approximates the total number of megabytes of storage that used by the columns on which the text index is to be built. The total text size ranges from 10 to 10,000 MB. The default total text size is ***32*** MB.

IVF text indexes may be sorted using the **order by** feature.

➲ Creating an IVF text index:

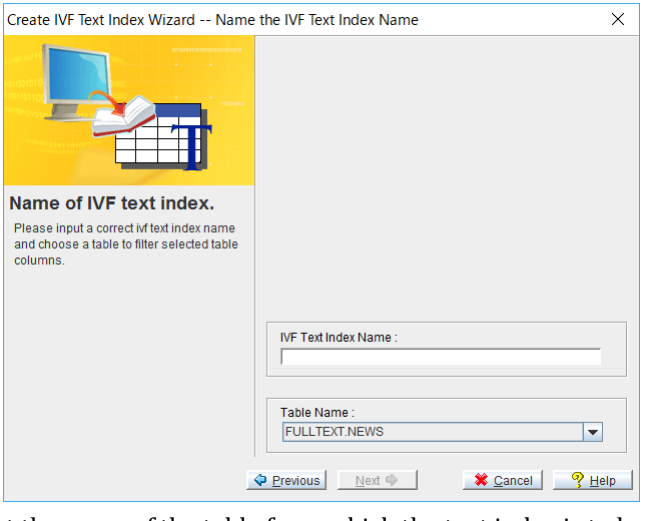
1. Click the **IVF Text Index** object in the tree. The **Properties** page appears.



2. Click the **Create** button. The **Introduction** window for the **Create IVF Text Index Wizard** appears.

©Copyright 1995-2026 CASEMaker Inc.

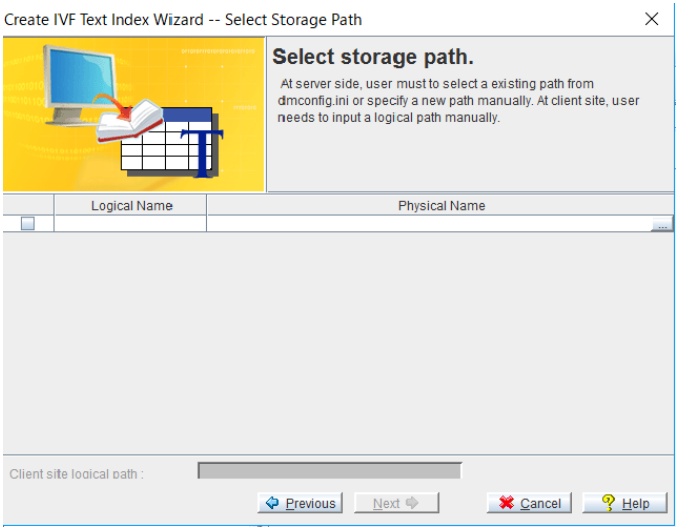
3. Click the **Next** button. The **Name the IVF Text Index Name** window appears.



4. Select the name of the table from which the text index is to be created from the **Table Name** menu.
5. Enter the name of the text index in the **IVF Text Index Name** field.

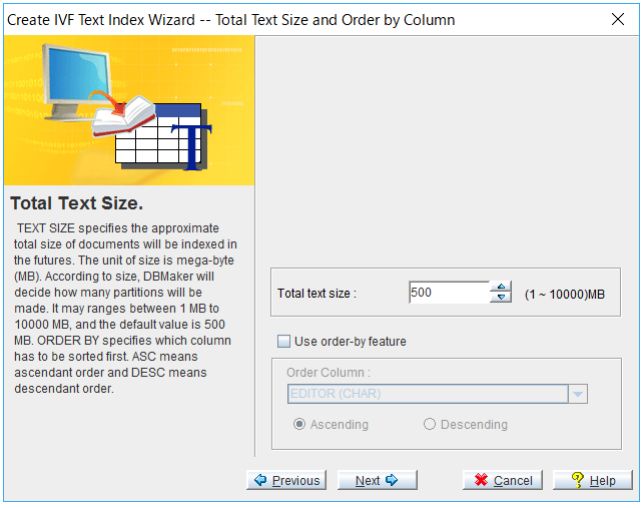
©Copyright 1995-2026 CASEMaker Inc.

6. A list of available columns on the selected table appears on the left list box. Select the columns to use in the IVF text index by double-clicking the column name or selecting the column and clicking the **Add** button. The **Add all** button adds all columns to the index. All the columns selected for the creation of the text index are displayed in the list box to the right. Columns in the text index can likewise be removed by double-clicking on the column name or selecting the column and clicking the **Remove** button.
7. Click the **Next** button. The **Select Storage Path** window appears.



8. Select a storage path for the IVF text index by selecting the check box in the row corresponding to the IVF text index path. The paths and logical names correspond to those specified in the configuration file. If you have not yet specified a path for storing IVF text indexes, open the JConfiguration Tool and add a new **IVF Text Index Storage Path** on the **User Files** page. For detailed instructions on using the JConfiguration Tool, please refer to the *JConfiguration Tool Reference*.
9. Click the **Next** button. The **Total Text Size and Order by Column** window appears.

©Copyright 1995-2026 CASEMaker Inc.



10. To change the IVF text index settings:

    - To change the total text size, enter a value into **Total Text Size** between 1 and 10,000.

11. To present data as it relates to another column in the table select the **Use order-by feature** check box.

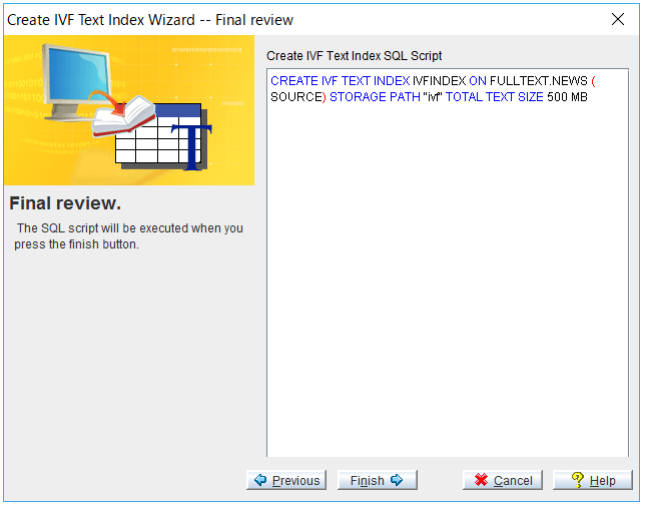
    a) Select the column that the text index will follow the order of from the **Order Column** menu.

    b) Select the **Ascending** option button to order data in the text index by the Order Column's ascending order.

    c) Select the **Descending** option button to order data in the text index by the Order Column's descending order.

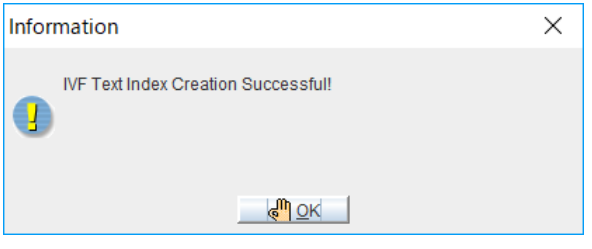
12. Click the **Next** button. The **Final review** window appears.

©Copyright 1995-2026 CASEMaker Inc.



13. To alter the SQL script, enter a new script into the **Create Text Index SQL script** field or click the **Previous** button to display the previous wizard screens.

14. To complete the view creation after reviewing the SQL script, click the **Finish** button. The **Information** dialog box appears.



15. Click the **OK** button.

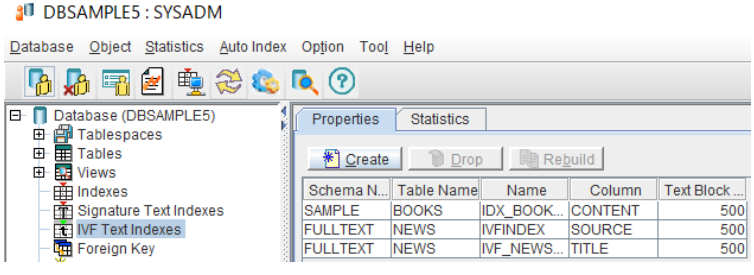
©Copyright 1995-2026 CASEMaker Inc.

8.2 Rebuilding an IVF Text Index

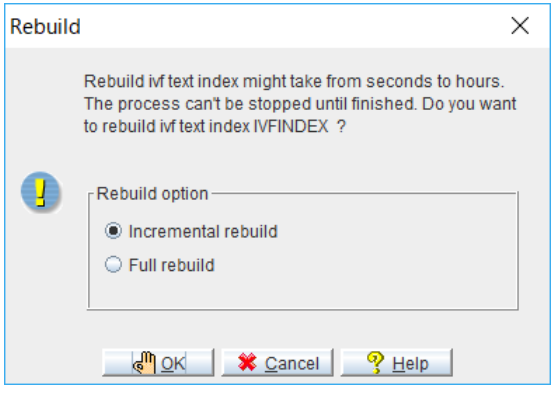
When a number of changes have been made to the table or to the column on which a text index was created, then the text index should be rebuilt. When an IVF text index is rebuilt, it will be updated to include all changes to the table or column. By default, owners of an IVF text index and users with RESOURCE authority or higher can rebuild the IVF text index. A user only with CONNECT authority may not rebuild any IVF text index.

➲ Rebuilding an IVF text index:

1. Click the **Text Indexes** object in the tree. The **Properties** page appears.
2. Select an IVF text index to rebuild.



3. Click the **Rebuild** button. The **Rebuild** dialog appears.



©Copyright 1995-2026 CASEMaker Inc.

4. Click either the **Incremental** rebuild or **Full rebuild** radio button.
5. Click the **OK** button. A confirmation message appears.
6. Click the **OK** button.

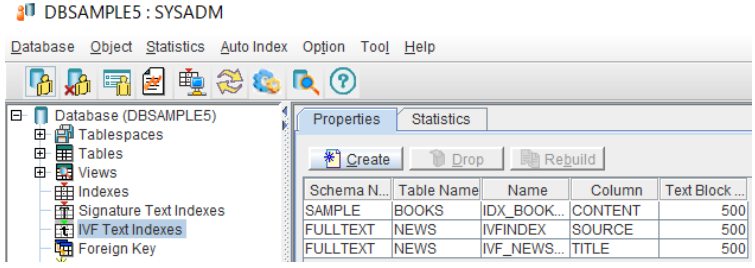
©Copyright 1995-2026 CASEMaker Inc.

8.3 Dropping an IVF Text Index

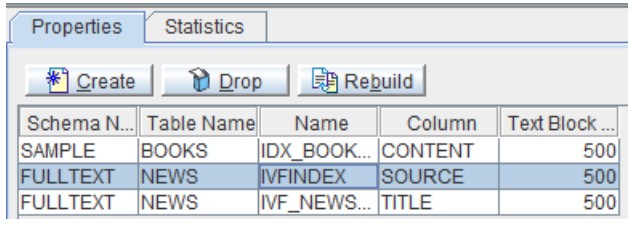
You can drop a text index that is no longer required. By default, owners of an IVF text index and users with RESOURCE authority or higher have DROP privilege for the IVF text index. A user only with CONNECT authority may not drop any IVF text index.

➲ Dropping an IVF text index:

1. Click the **IVF Text Indexes** object in the tree. All the text indexes in the database are displayed in the **Properties** page.

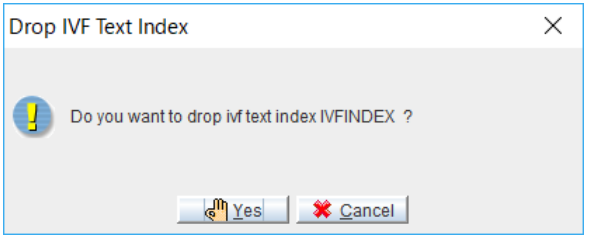


2. Select the **IVF Text Index** to be dropped.



3. Click the **Drop** button. A confirmation window appears to confirm that you want to drop the IVF text index.

©Copyright 1995-2026 CASEMaker Inc.



4. Click the **YES** button. The remaining IVF text indexes in the database are displayed.

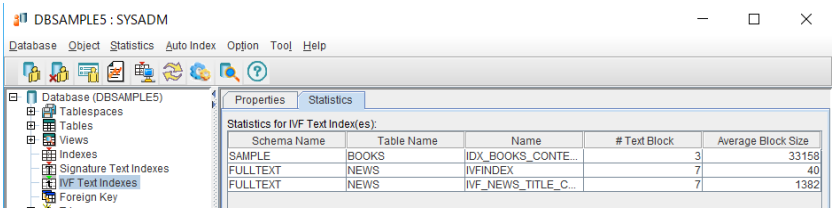
©Copyright 1995-2026 CASEMaker Inc.

8.4 Checking IVF Text Index Statistics

The statistics of an IVF text index may periodically be checked. Statistics include the table owner and name, the text index name, the number of text blocks and the average block size.

➲ Viewing text index statistics:

1. Select **IVF Text** Indexes from the tree. The **Properties** page appears.
2. Click the **Statistics** tab. The statistics of the text indexes are displayed. These include the names of the text indexes, the average block size of the IVF text indexes and the number of text blocks.



©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

9 Using Foreign Keys

A foreign key is the relationship between a column or combination of columns in a table has and the primary key or unique index from another table. A foreign key contains the same values and defines the relationship between the two tables.

This section illustrates creating foreign keys for tables and then dropping foreign keys that are no longer required.

©Copyright 1995-2026 CASEMaker Inc.

9.1 Creating a Foreign Key

A foreign key refers to another table by specifying the referencing and referenced columns. Both the referencing and referenced columns should map to each other. The mapping columns should be the same type and length. The referenced columns should be NOT NULL, but the referencing columns can be NULL. The referenced columns should be primary key or unique index. By default, users with RESOURCE authority or higher can creat a foreign key. A user only with CONNECT authority may not create any foreign key.

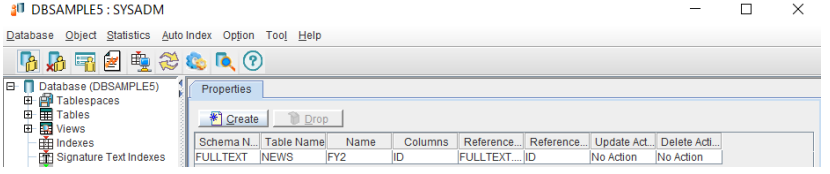
Adding a Foreign Key

Creating a foreign key comprises several phases. First, includes selecting a table that contains referencing columns, selecting columns, and naming a foreign key.

The foreign key columns in the referencing table must map exactly to the primary key columns or unique index columns of the referenced table. The number of columns used in the foreign key and the schema of those columns must be the same in both the referencing and referenced tables.

➲ Adding a foreign key:

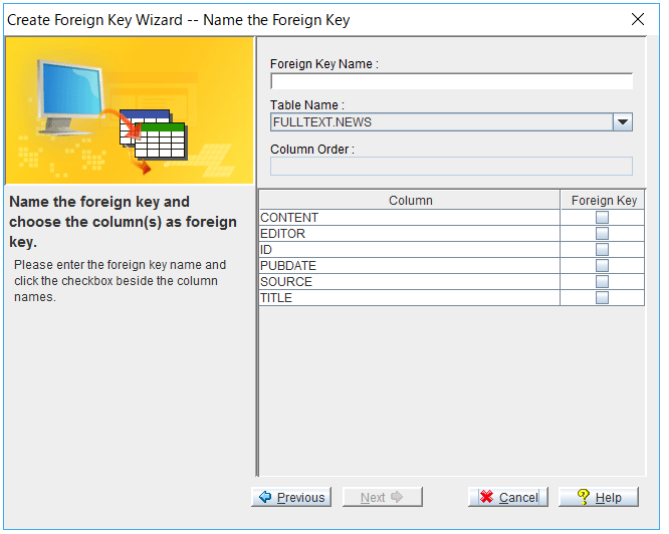
1. Select **Foreign Key** from the tree. The **Properties** page appears.



2. Click the **Create** button. The **Create Foreign Key – Introduction** window appears.

©Copyright 1995-2026 CASEMaker Inc.

3. Select the **Next** button, the **Name the Foreign Key** window appears.

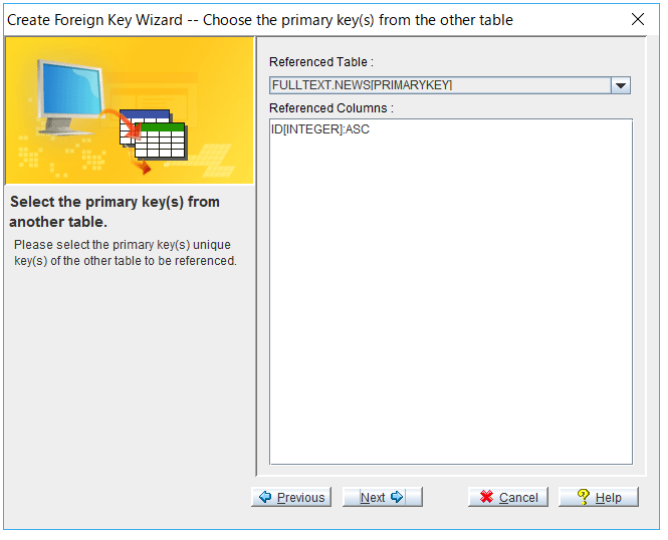


4. Enter a name for the foreign key in the **Foreign Key Name** text field.
5. Select the referencing table name.
6. Click checkboxes beside the column names to specify the columns

©Copyright 1995-2026 CASEMaker Inc.

included in the foreign key.

7. Click the **Next** button. The **Choose the primary key(s) from the other table** window of the **Create Foreign Key Wizard** appears.



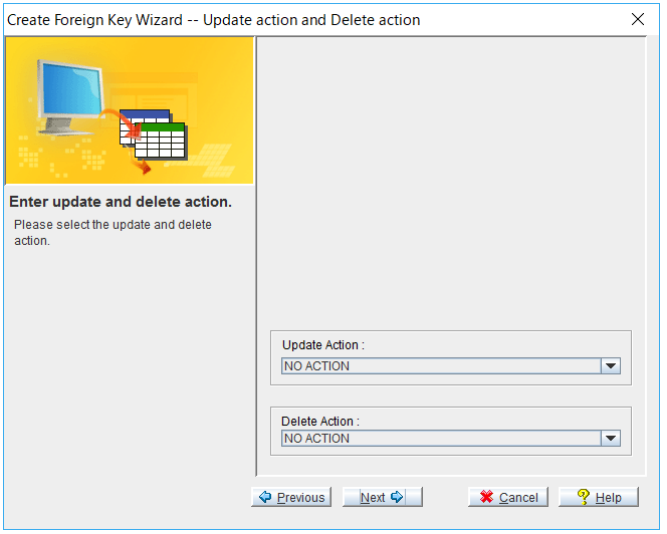
8. Select the name of the referenced table from the **Referenced Table** menu.

NOTE *Only tables containing primary keys or unique indexes matching the schema of the foreign key are displayed.*

9. All of the selected table columns that are primary keys or unique indexes are displayed.

10. Click the **Next** button. The **Update action and Delete action** window appears.

©Copyright 1995-2026 CASEMaker Inc.



Setting Foreign Key Options

Foreign key and their options can be specified. The following table shows the effect of available foreign key options.

FOREIGN KEY OPTION	DESCRIPTION
Set Null	Records in the foreign key column are set to **Null** if the corresponding records in the primary key or unique index column are updated or deleted.
Cascade	The corresponding records in the referencing table are also updated or deleted.
Set Default	The foreign key record is set to a default value if corresponding records in the primary key or unique index column are updated or deleted.

©Copyright 1995-2026 CASEMaker Inc.

FOREIGN KEY OPTION	DESCRIPTION
No Action	There is no action on the foreign key column if the corresponding records in the primary key or unique index column are updated or deleted.

Table 9-1Foreign Key Options

➲ Setting foreign key options:

1. Choose one of the following foreign key settings from the **Update Action** menu:

   - **No Action**
   - **Cascade**
   - **Set Default**
   - **Set Null**

   Upon update of the referenced table the chosen update action will occur.

2. Choose one of the following foreign key settings from the **Delete Action** menu:

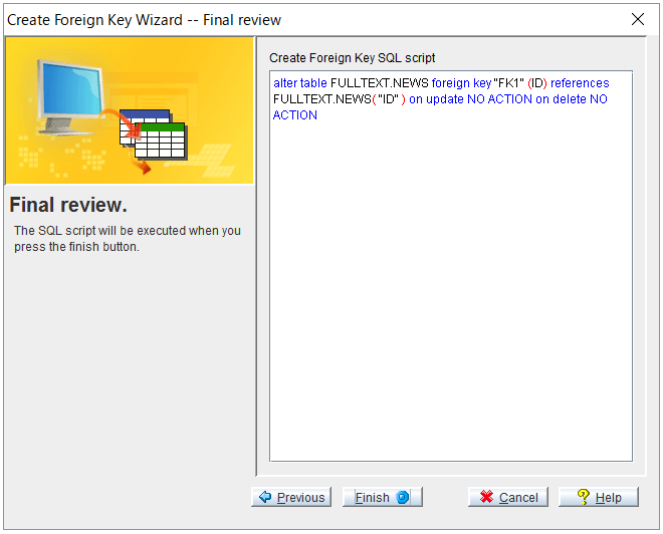
   - **No Action**
   - **Cascade**
   - **Set Default**
   - **Set Null**

   Upon update of the referenced table the chosen delete action will occur.

   NOTE *The default setting is **No Action**.*

3. Click the **Next** button. The window shown below appears. The SQL command generated by the wizard can be edited as needed.

©Copyright 1995-2026 CASEMaker Inc.



4. Click the **Finish** button. The **Properties** page reappeard with the new foreign key in the list.

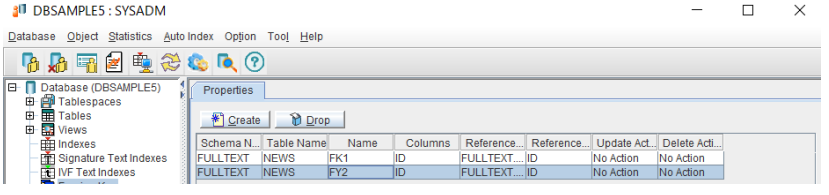
©Copyright 1995-2026 CASEMaker Inc.

9.2 Dropping a Foreign key

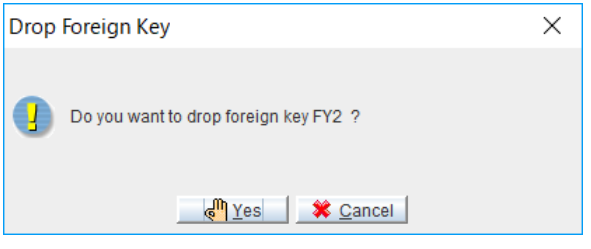
If a foreign key is no longer required, it can be dropped from the database. By default, owners of a table and users with RESOURCE authority or higher can drop a foreign key. A user only with CONNECT authority may not drop any foreign key.

➲ Dropping a foreign key:

1. Select the **Foreign Key** object from the tree. The **Properties** page appears.
2. Click the foreign key to drop.



3. Click the **Drop** button. The **Drop Foreign Key** dialog box appears.



4. Click the **YES** button. The remaining foreign keys in the database are displayed.

©Copyright 1995-2026 CASEMaker Inc.

10 Using Triggers

Triggers are predefined commands that automatically execute in response to specific events, regardless of which user or application program causes the events. Triggers provide dtabase customization beyond what is possible with standard SQL commands. Triggers can:

- Implement business rules
- Create database activity audit trails
- Generate additional value from existing data
- Replicate data across multiple tables
- Perform security authorization procedures
- Control data integrity
- Define unconventional integrity constraints

This section illustrates creating triggers with various trigger options and then dropping triggers from a database.

©Copyright 1995-2026 CASEMaker Inc.

10.1 Creating a Trigger

To create a trigger on a table, you must be the owner of the table or have DBA authority or higher. You must also have the necessary object privileges on all objects referenced by the trigger definition. By default, users with RESOURCE authority or higher have CREATE privilege for the trigger. A user only with CONNECT authority cannot create any trigger.

Creating a trigger involves the following set of actions:

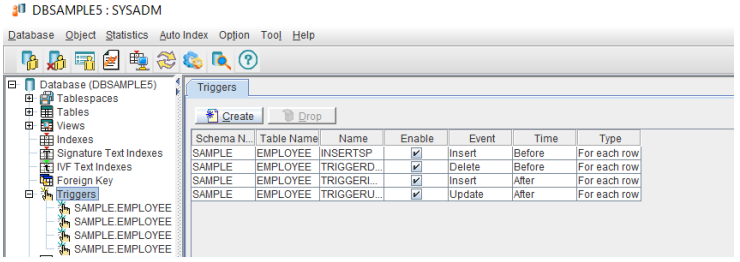
- Assigning a trigger name and table
- Specifying a trigger action time
- Defining a referencing clause
- Entering a WHEN condition clause
- Entering SQL statements for the trigger action

Assigning a Trigger Name and Table

A trigger name must be assigned and a table selected when creating a new trigger.

➲ Specifying a trigger name and table:

1. Click the **Trigger** object in the tree. The **Triggers** page appears.



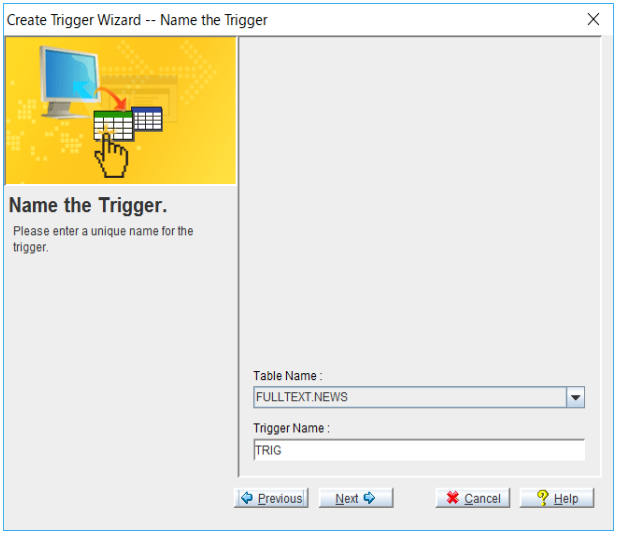
2. Click the **Create** button. The **Introduction** window of the **Create**

©Copyright 1995-2026 CASEMaker Inc.

Trigger Wizard appears.

3. Click the **Next** button. The **Name the Trigger** window appears.
4. Select the table name in which the Trigger will be based on from the **Table Name** menu.
5. Enter the name of the trigger in the **Trigger Name** field.

©Copyright 1995-2026 CASEMaker Inc.



6. Click the **Next** button. The **Choose the Trigger Action Time** window appears.

Specifying Trigger Action Settings

After identified the trigger's name and table, specify the type of actions the trigger will perform. The settings are as follows:

Action Time: The trigger action time specifies whether a trigger should fire (i.e., begin executing) before or after the SQL statement that activated the trigger. The BEFORE and AFTER keywords specify the trigger's action time. The BEFORE keyword indicates the trigger action fires before the trigger statement. The AFTER keyword indicates the trigger action fires after the trigger statement. A trigger cannot have more than a single trigger time.

Trigger Event: The trigger event is the database operation that causes a trigger to execute (i.e., fire). The trigger event may be an INSERT, UPDATE, or DELETE statement that operates on the trigger's table. Only a single trigger event can be specified for each trigger.

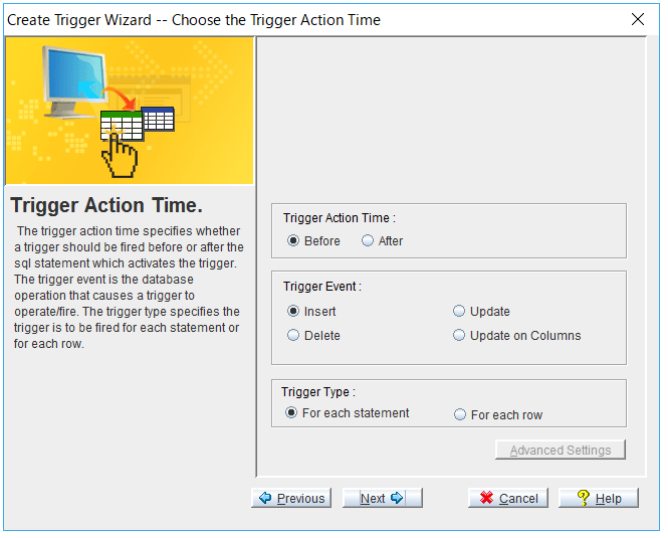
©Copyright 1995-2026 CASEMaker Inc.

Trigger Type: Specifies how many times the trigger fires for each trigger event. There are two types of triggers: row triggers and statement triggers. The FOR EACH ROW keyword specifies a row trigger. Row triggers fire a trigger action once for each row modified by the trigger event. The FOR EACH STATEMENT keyword specifies a statement trigger. Statement triggers firea trigger action once for each trigger event.

Referencing Clause: The advanced setting tab is highlighted when the selected Trigger type is FOR EACH ROW. Click this tab to define the REFERENCING clause. The REFERENCING clause defines correlation names for the old and new values of a column. This is primarily used when you cannot use the default OLD and NEW names because of a conflict with a table with the same name.

➲ specifying trigger action settings:

1. From the **Name the Trigger** window of the **Create Trigger Wizard**, click the **Next** button. The **Choose the Trigger Action Time** window appears.



2. Select one of the following trigger action time options:

©Copyright 1995-2026 CASEMaker Inc.

- Click the **After** radio button to fire the trigger after the SQL statement.
- Click the **Before** radio button to fire the trigger before the SQL statement.

3. Select one of the following trigger event options:
   - To choose the INSERT command as the trigger event, choose the **Insert** option button in the **Trigger Event** group.
   - To choose the UPDATE command as the trigger event, choose the **Update** option button in the **Trigger Event** group.
   - To choose the DELETE command as the trigger event, choose the **Delete** option button in the **Trigger Event** group.
4. Select one of the following trigger type options from the **Trigger Type** field.
   - Selecting the **For each row** option button sets the trigger statement to execute on each row modified by the trigger event.
   - Selecting the **For each statement** option button sets the trigger statement to execute upon each instance of the trigger event.
5. If the **For each statement** option button is selected, clicking **Next** opens the **Specify a WHEN condition** clause window. If the **For each row** option button is selected, clicking **Next** opens the **Create Trigger Advanced Settings** window.

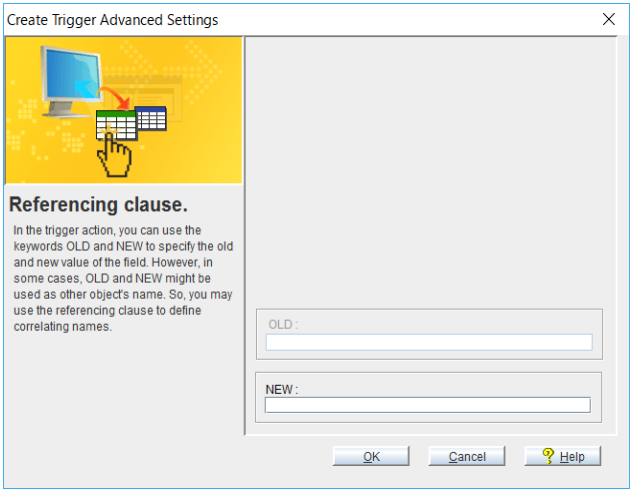
Indicating the Referencing Clause

After defining the trigger options, specify the referencing clause of the trigger. These options are required when FOR EACH ROW is the trigger type. The REFERENCING clause defines correlation names for the old and new values of a column. Use this primarily when you cannot use the default OLD and NEW names because a column has the same name.

➲ Defining the referencing clause:

©Copyright 1995-2026 CASEMaker Inc.

1. From the **Choose the Trigger Action Time** window of the **Create Trigger Wizard** click the **Next** button. The **Create Trigger Advanced Settings** window appears.



2. When you create a **For each row** type trigger, enter a substitute name to refer to the old value in the **OLD** field and enter a substitute name to refer to the new value in the **NEW** field.
3. Click the **Next** button. The **Specify a WHEN condition clause** window appears.

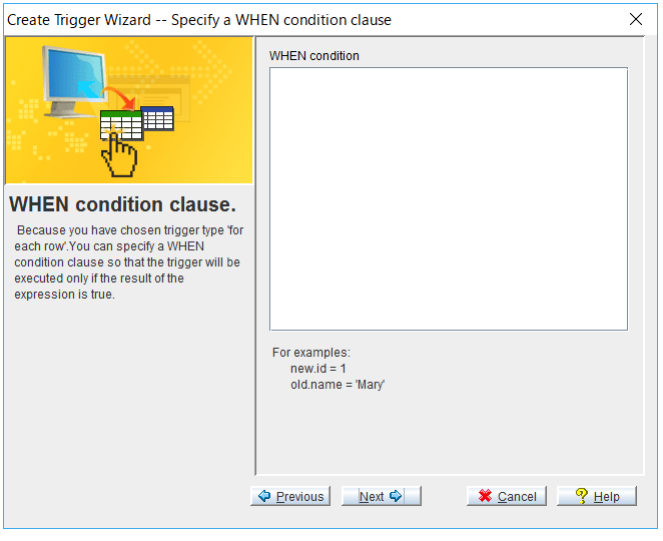
Entering the WHEN Condition Clause

Once the settings for the trigger action are complete, specify a WHEN clause to place constraints on the actions that cause the trigger to fire. The WHEN statement must follow proper SQL syntax rules. Please see *Constraint Syntax,* near the end of Chapter 4 for more information.

➲ Specifying a WHEN condition:

1. From the **Choose the Trigger Action Time** window of the Create Trigger Wizard (if "**For each statement**" has been specified) or from the **Create Trigger Advanced Settings** window, click the **Next** button. The **Specify a WHEN condition clause** window is opened.

©Copyright 1995-2026 CASEMaker Inc.



2. Enter the WHEN condition and click **Next**. The **Trigger Action** window appears.

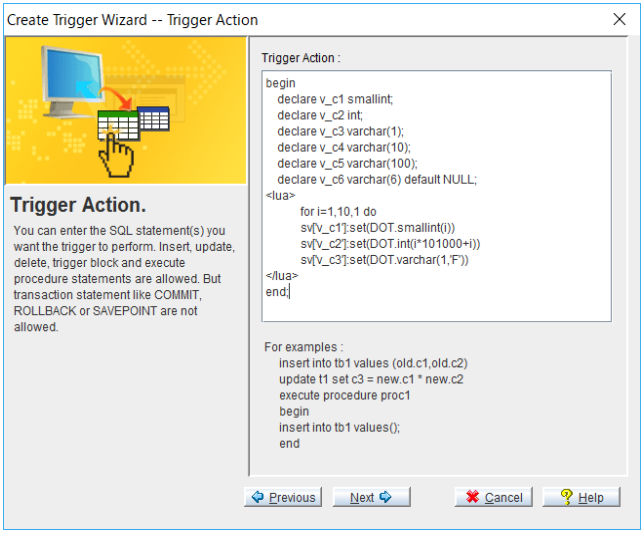
Entering SQL statements for the trigger action

After defining the trigger event and its constraints, the trigger action is set. The trigger action is the command or set of commands that the trigger will carry out on all data that meet the constraints when the trigger fires.

➲ Specifying SQL statements for the trigger action:

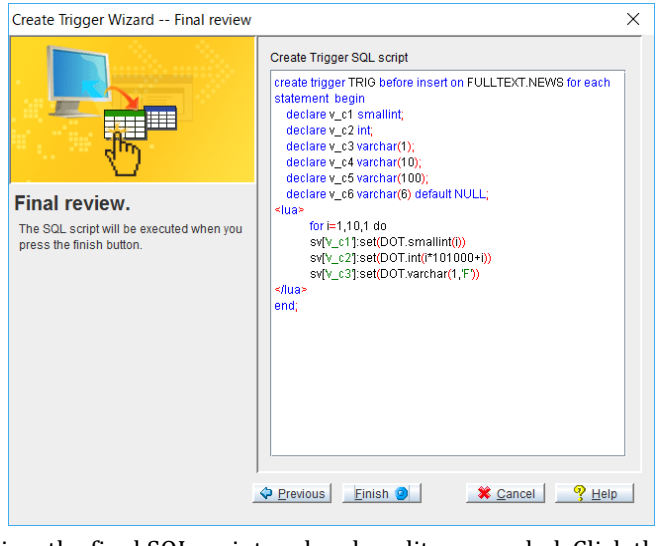
1. From the **Specify a WHEN condition clause**, click the **Next** button. The **Trigger Action** window appears.
2. Enter the SQL statements, SQL block or LUA script that the trigger will execute.

©Copyright 1995-2026 CASEMaker Inc.



NOTE *A trigger SQL block' compound statements are bounded by the keywords BEGIN and END.*

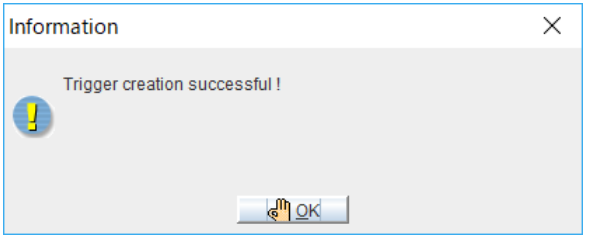
3. Click the **Next** button. The **Final review** window appears.



4. Review the final SQL script and make edits as needed. Click the **Finish**

©Copyright 1995-2026 CASEMaker Inc.

button when you are satisfied with the script. A confirmation window appears.



5. Click the **OK** button.

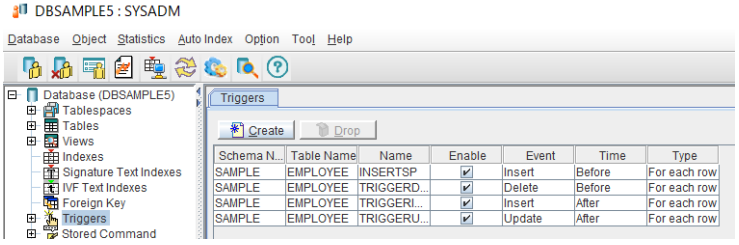
©Copyright 1995-2026 CASEMaker Inc.

10.2 Modifying a Trigger

After creating a trigger, it can be modified or disabled. Changing the trigger's SQL code can modify the triggering event, trigger action, and trigger type. By default, owners of a trigger and users with RESOURCE authority or higher can modify a trigger's SQL code. A user only with CONNECT authority may not modify any trigger.

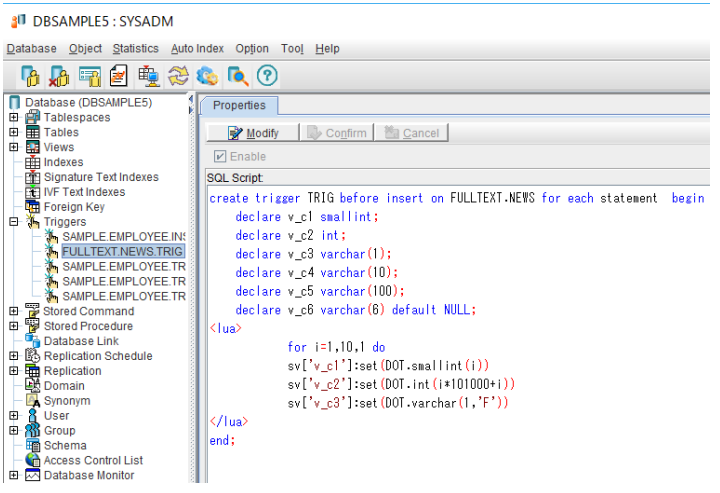
➲ Modifing a trigger:

1. Click the object **Trigger** in the tree. All the triggers in the database are displayed on the **Triggers** page.

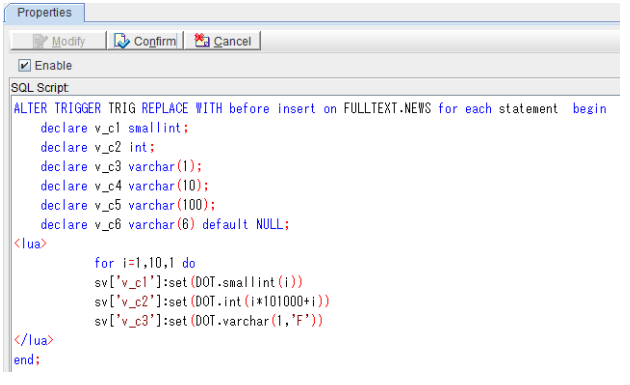


2. Double-click the trigger to be modified, or expand the Triggers node in the tree and select a trigger from the tree. The **Properties** page appears.

©Copyright 1995-2026 CASEMaker Inc.



3. Click the **Modify** button.



4. To disable the trigger, remove the check mark next to **Enable**.
5. To make changes to the SQL script, click in the appropriate place in the **SQL Script** field and edit the statement.
6. Click the **Confirm** button. The modified trigger appears.

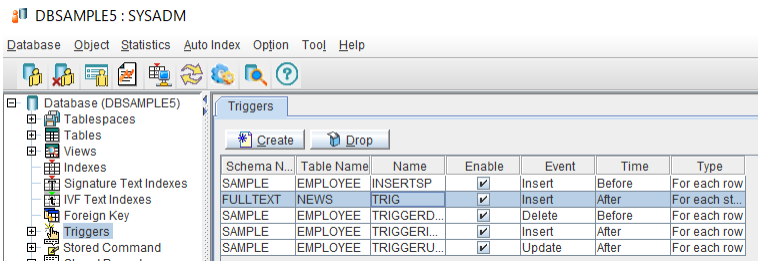
©Copyright 1995-2026 CASEMaker Inc.

10.3 Dropping a Trigger

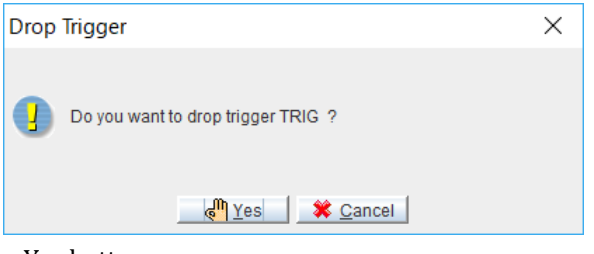
When a table's schema is altered and then a trigger on that table is fired, DBMaker executes the trigger according to the new table definition. However, if the specified column in a triggering event or action was dropped, the trigger execution fails, as does the triggering statement. In this situation, users need to drop the trigger or modify the trigger definition according to the new table schema. Upon deleting a table, the trigger that references that table is also dropped. By default, owners of a trigger and users with RESOURCE authority or higher have DROP privilege for the trigger. A user only with CONNECT authority may not drop any trigger.

➲ Deleting a trigger:

1. Click the object **Trigger** in the tree. All the triggers in the database are displayed as shown.
2. Select a trigger.



3. Click the **Drop** button. A confirmation window appears.



4. Click the Yes button

©Copyright 1995-2026 CASEMaker Inc.

11 Working with Stored Commands

A stored command is a compiled SQL DML statement that is stored in the database. Stored commands are precompiled in an executable form. They can execute the same commands without repeated compiling and optimizing. It is possible to create a stored command for any frequently used SQL statement.

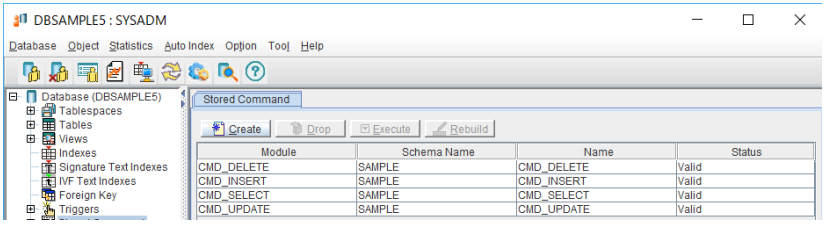
©Copyright 1995-2026 CASEMaker Inc.

11.1 Creating a Stored Command

Create stored commands to achieve improved performance of frequently used statements. By default, users with RESOURCE authority or higher have CREATE privilege for the stored command. A user only with CONNECT authority may not create any stored command.

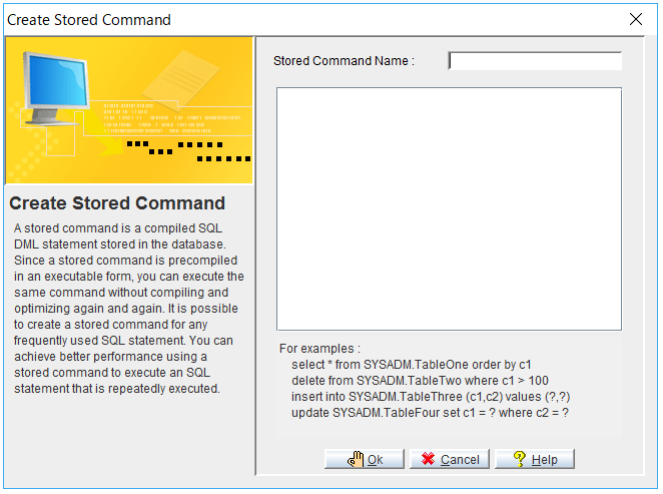
➲ Creating a stored command:

1. Click the object **Stored Command** in the tree. The **Stored Command** window appears.



2. Click the **Create** button. The **Create Stored Command** window appears.

©Copyright 1995-2026 CASEMaker Inc.



3. Enter a name for the stored command into the **Stored Command Name** field.
4. Enter the SQL command.
5. Click the **OK** button. The newly created stored command appears with the database's existing stored commands.

©Copyright 1995-2026 CASEMaker Inc.

11.2 Executing a Stored Command

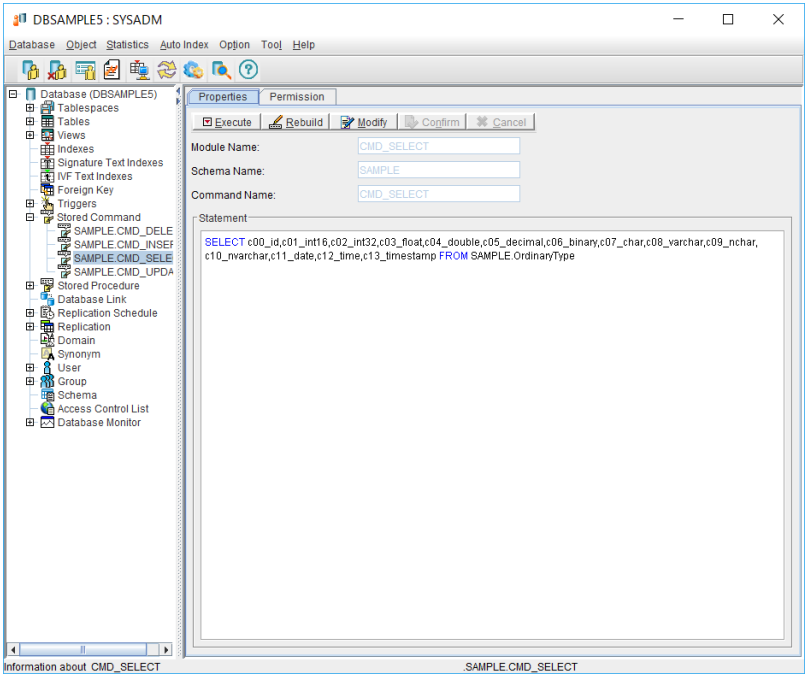
After creating a stored command, you can execute it directly or in an application program. If you execute a stored command that has input parameters, JDBA tool prompts for its value during execution of the stored command. When you execute a stored command, the number of input parameters should be equal to number of input parameters in the stored command.

By default, owners of a stored command and users with RESOURCE authority or higher can execute a stored command. A user only with CONNECT authority may not execute any stored command.

➲ Executing a stored command:

1. Open the object **Stored Command** in the tree and then select the stored command to execute. The stored command **Properties** pageappears.

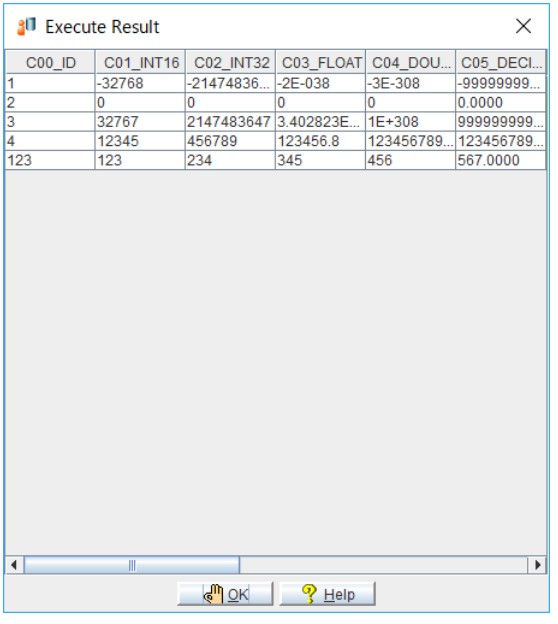
©Copyright 1995-2026 CASEMaker Inc.



NOTE *Double-clicking the Stored Command node of the tree also displays the Properties window.*

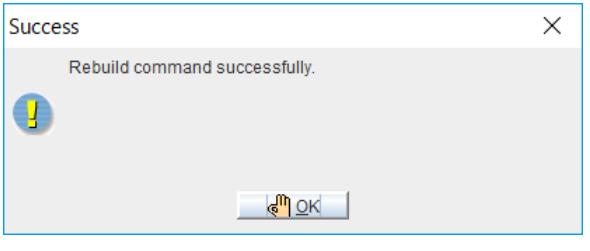
2. Click the **Execute** button. The result of the executed stored command are displayed.

©Copyright 1995-2026 CASEMaker Inc.



C00_ID	C01_INT16	C02_INT32	C03_FLOAT	C04_DOU...	C05_DECI...
1	-32768	-21474836...	-2E-038	-3E-308	-99999999...
2	0	0	0	0	0.0000
3	32767	2147483647	3.402823E...	1E+308	999999999...
4	12345	456789	123456.8	123456789...	123456789...
123	123	234	345	456	567.0000

3. Click the **OK** button to dismiss the result window.
4. Click the **Rebuild** button to rebuilded the stored command. A confirmation dialog box appears.



©Copyright 1995-2026 CASEMaker Inc.

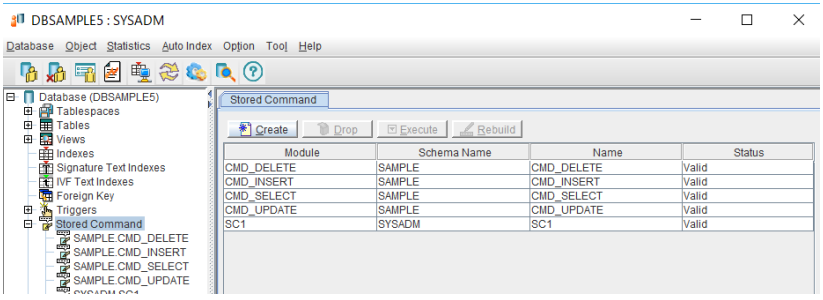
11.3 Rebuilding a Stored Command

Many times users create one Stored Command with litter table data, but table data grow up day and day, it will make the Stored Command execution efficiency worse and worse. Therefore, you should to rebuild the Stored Command to improve the performance efficiency.

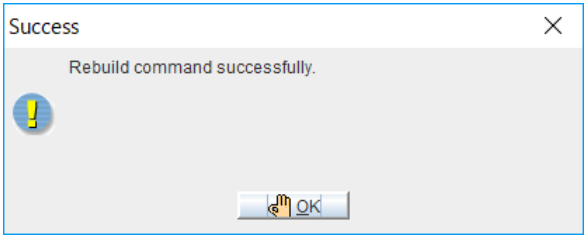
By default, owners of a stored command and users with RESOURCE authority or higher have Rebuild privilege for the stored command. A user only with CONNECT authority may not Rebuild any stored command.

➲ Rebuilding a stored command:

1. Click the object **Stored Command** in the tree. The **Stored Command** page appears.
2. Click the stored command to rebuild. It will highlight blue.



3. Click the **Rebuild** button. The **Rebuild Stored Command** dialog box appears.



©Copyright 1995-2026 CASEMaker Inc.

4. Click the **OK** button. The stored commands in the database are rebuilded.

©Copyright 1995-2026 CASEMaker Inc.

11.4 Granting Stored Command Privileges

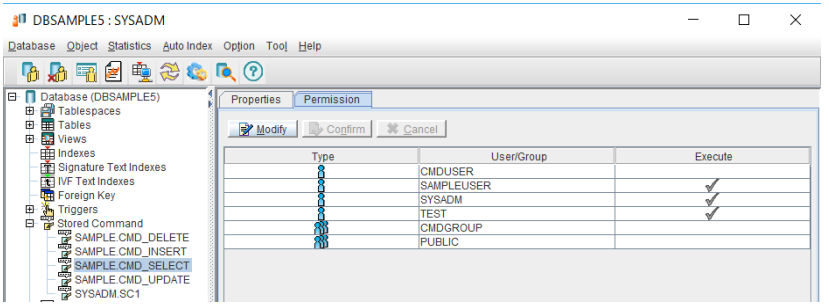
By default, owners of a stored command and users with CONNECT authority or higher, have execute privilege for the stored command. The owner and a user with RESOURCE authority or higher may grant execute privileges to other users. The execute privilege can also be revoked from other users with resource authority by the owner of the stored command. A user with DBA authority or higher has all privileges on all stored commands and can revoke those privileges even from the owner of the stored command unless the owner has DBA authority or higher. However, you can only grant or revoke the execute privilege of stored commands.

➲ Granting and revoking privileges on a stored command:

1. Open the **Stored Command** node and then select a stored command. The stored command **Properties** page appears.

   NOTE *Double-clicking on the stored command node in the tree also displays the **Properties** window.*

2. Click the **Permission** tab. The stored command **Permission** page appears.



3. Select a User/Group for granting or revoking privileges.
4. Click the **Modify** button.
5. Click the **Execute** column beside the **User/Group** to grant or revoke execute privileges. A check mark indicates that the user/group has execute privilege.

©Copyright 1995-2026 CASEMaker Inc.

6. Click the **Confirm** button.

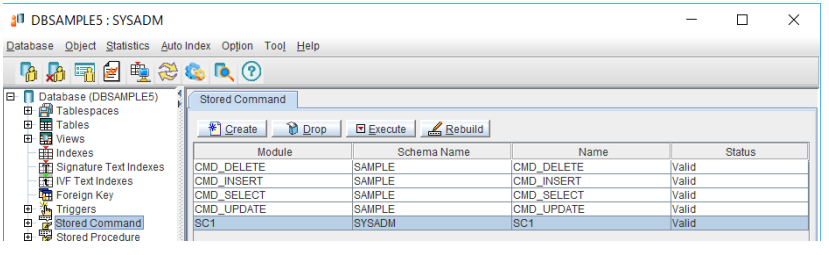
©Copyright 1995-2026 CASEMaker Inc.

11.5 Dropping a Stored Command

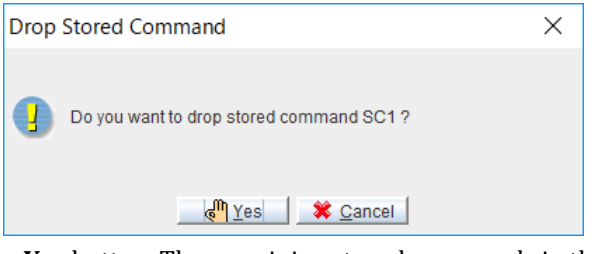
If a stored command is no longer required, it can be dropped. By default, owners of a stored command and users with RESOURCE authority or higher have DROP privilege for the stored command. A user only with CONNECT authority may not drop any stored command.

➲ Dropping a stored command:

1. Click the object **Stored Command** in the tree. The **Stored Command** page appears.
2. Click the stored command to drop. It will highlight blue.



3. Click the **Drop** button. The **Drop Stored Command** dialog box appears.



4. Click the **Yes** button. The remaining stored commands in the database are displayed.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

12 Working with Stored Procedures

A stored procedure is a special kind of user-defined function. This increases the performance of frequently repeated tasks by bypassing repeated SQL compilation and optimization. Stored procedures are executed as commands in interactive SQL, or invoke it in application programs, trigger actions, or other stored procedures.

You can accomplish a wide range of objectives with stored procedures, including improving database performance, simplifying the writing of applications, and limiting or monitoring database access.

Stored procedures are stored as executable objects in the database and are available to every application running on the database. Application development time decreases because stored procedures are available to any application.

©Copyright 1995-2026 CASEMaker Inc.

12.1 Creating a Stored Procedure

DBMaker provides three languages for creating stored procedures: ESQL/C, Java and SQL. By default, users with RESOURCE authority or higher have CREATE privilege for the stored procedure. A user only with CONNECT authority may not create any stored procedure.

An ESQL/C stored procedure can perform any function a C application can perform, including calling other C functions and system calls. Writing ESQL/C stored procedures requires a C compiler. C compilers are usually already available in UNIX environments. Simply include the compiler in your path. Install Visual C++ version 4.2 or greater when using Windows.

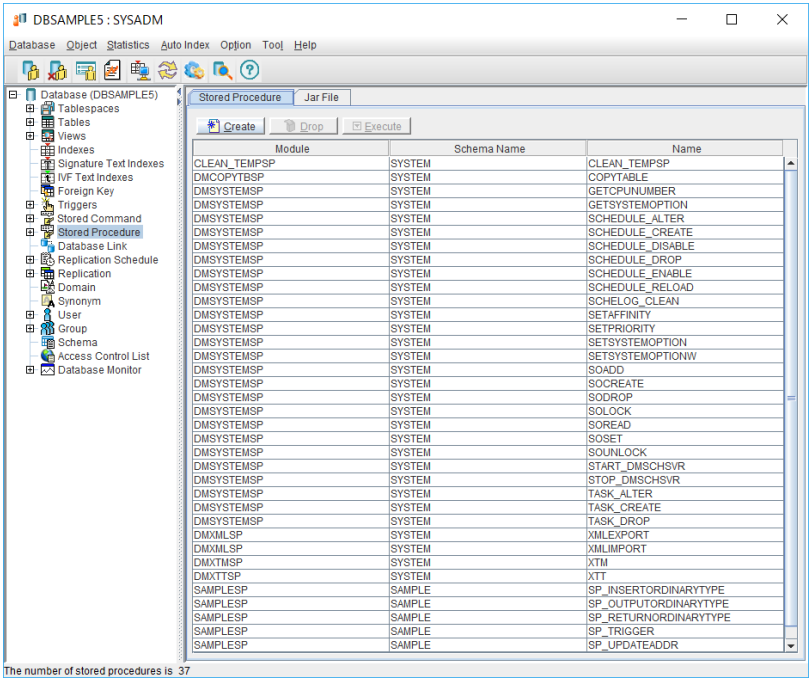
Given Java's popularity today, it is certainly possible that members of a development team are more proficient in Java than ESQL. To create Java stored procedures, the value of the **DB_SPDir** in the dmconfig.ini must define and next copy the physical jar file to the *DB_SPDir/jar/username/directory.* Finally, execute the add jar file SQL statement to register the jar file with the database. For details about adding jar file SQL statements, please refer to chapter 12 of *Database Administrator's Guide*.

Creating stored procedures with ESQL and Java is inefficient. Stored procedures made directly from SQL statements make very efficient SQL stored procedures. A SQL stored procedure is a set of SQL statements that can be stored in a server. Once created, the user can refer to the SQL stored procedure rather than reissuing many individual statements. For more information on SQL stored procedures, please see the *DBMaker SQL Stored Procedure User's Guide*.

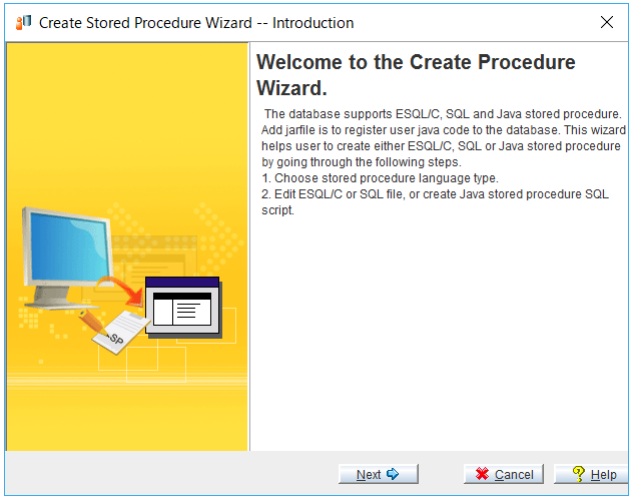
➲ Creating a stored procedure:

1. Click the object **Stored Procedure** in the tree. The **Stored Procedures** page appears.

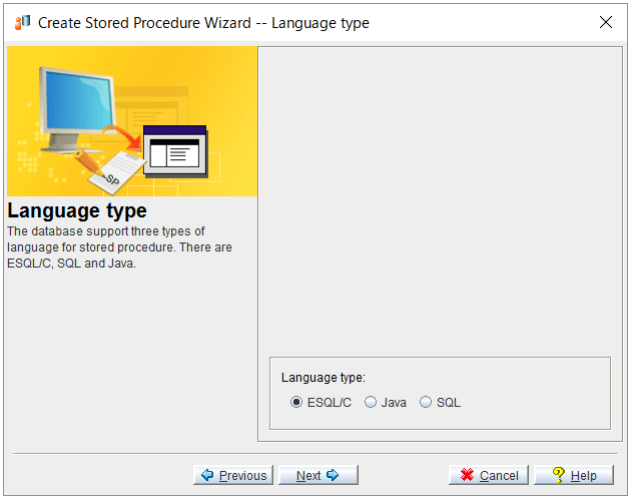
©Copyright 1995-2026 CASEMaker Inc.



2. Click the **Create** button. The **Introduction** window of the **Create Stored Procedure** wizard appears.

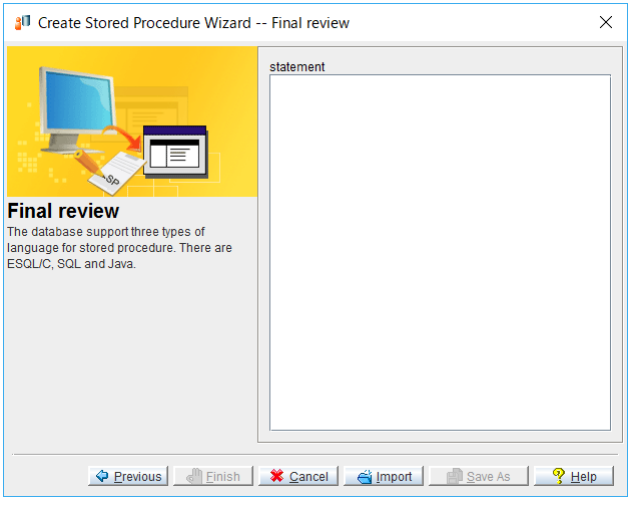


3. Click the **Next** button. The **Language Type** wizard appears.

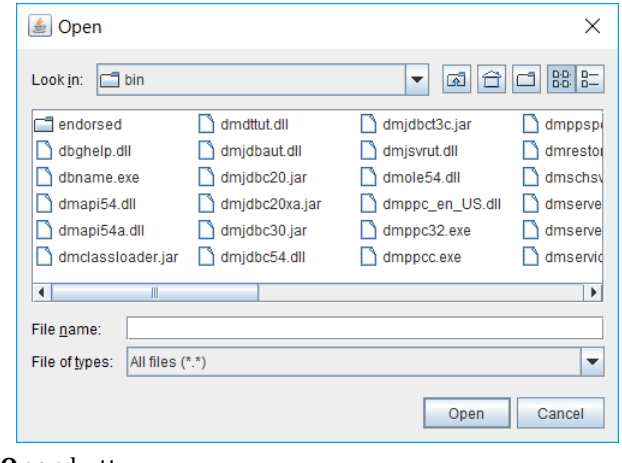


4. Select a language to use for writing the stored procedure by clicking either the **ESQL/C** radio button, **Java** radio button or the **SQL** radio button. Skip to step five when **ESQL/C** is selected, skip to step nine when **Java** is selected and skip to step 10 when **SQL** is selected.
5. Click the **Next** button. The **Final Review** window appears. Input the ESQL/C statement or click the **Import** button to import an ESQL/C statement from a file.

©Copyright 1995-2026 CASEMaker Inc.



6. Click the **Import** button. The **Open** window appears. Import files from any source, including the SPDIR directory of other databases on the server or network drives. Enter the path in the **File name** field, or browse the directory tree until the desired path is found.

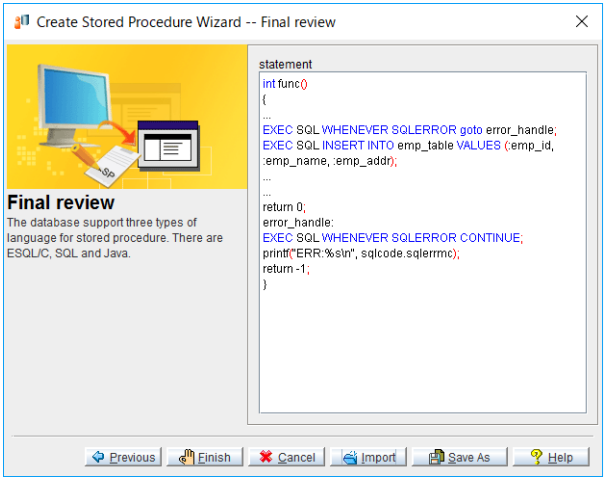


7. Click the **Open** button.

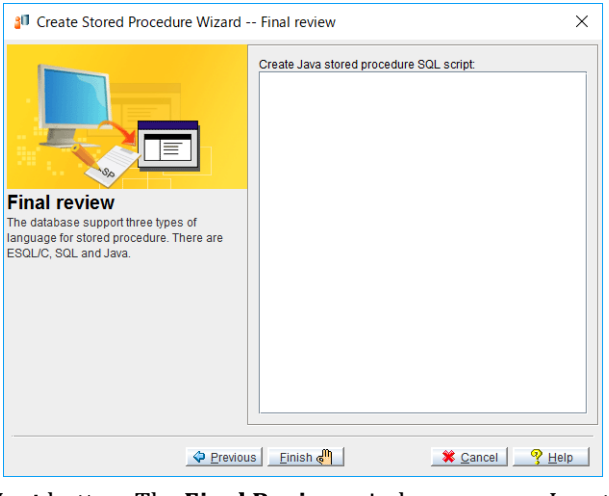
8. The **Final review** window reappears as in the example below if the imported file contains properly formatted (ASCII) text, or if you choose to manually enter the code. Click the **Save As** button to save the stored procedure to another location, or click the **Finish** button to compile and save the stored procedure in

©Copyright 1995-2026 CASEMaker Inc.

the database.

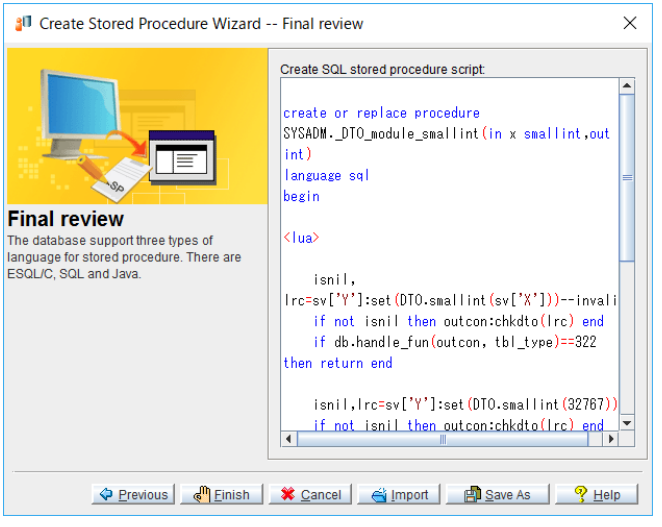


9. Click the **Next** button. The **Final Review** window appears. Input the Java stored procedure SQL script. Click the **Finish** button.

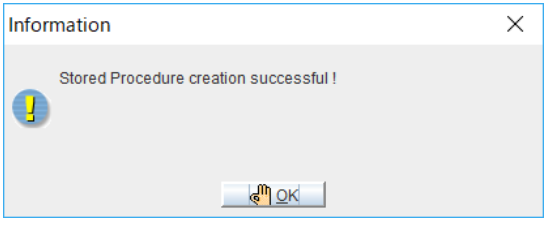


10. Click the **Next** button. The **Final Review** window appears. Input the SQL statement or LUA script, and you also can click the **Import** button to import a SQL statement from a file, then enter or copy the LUA script into the SQL statement.

©Copyright 1995-2026 CASEMaker Inc.



11. If the Stored Procedure compiles correctly, the following message appears.



12. Click the **OK** button.

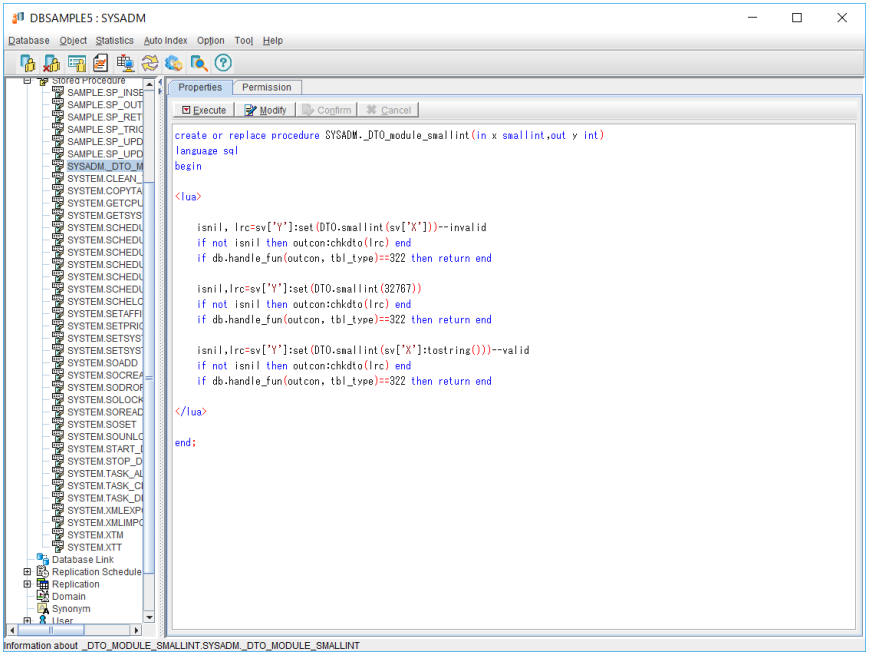
©Copyright 1995-2026 CASEMaker Inc.

12.2 Executing a Stored Procedure

After creating a stored procedure, you can execute it directly or in an application program. If you execute a stored procedure that has input parameters, JDBA tool prompts for the parameter's value during execution of the stored procedure. The number of input parameters should be equal to number of input parameters in the stored procedure.

➲ Executing a stored procedure:

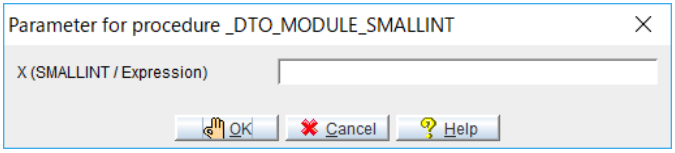
1. Open the **Stored Procedure** node and select a stored procedure. The stored procedure **Properties** page appears.



NOTE *Double-clicking a stored procedure in the right panel displays the same window.*

©Copyright 1995-2026 CASEMaker Inc.

2. Click the **Modify** button, the stored procedure that to be executed is modified.
3. Click the **Execute** button, the parameter dialog box that to be executed stored procedure appears.



4. Enter the parameter, and click the **OK** button, the result of the executed stored procedure appears.

©Copyright 1995-2026 CASEMaker Inc.

12.3 Granting Privileges on Stored Procedures

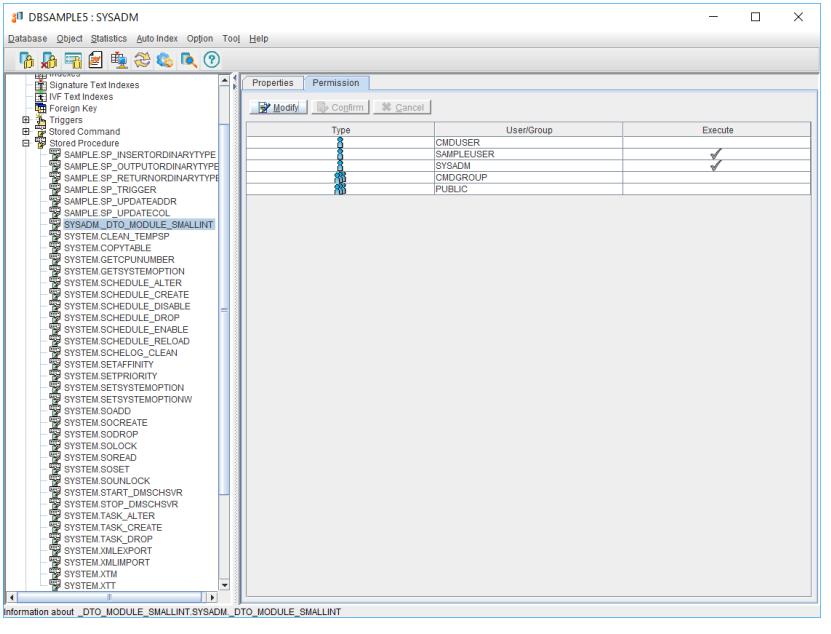
Users must have EXECUTE privilege for a stored procedure to execute it. In order to allow a stored procedure to be used by others, you can grant execute privilege on a stored procedure to other database users. If you do not wish others to execute a stored procedure, you can revoke execute privilege of the stored procedure from them.

➲ Granting and revoking privileges on a stored procedure:

1. Open the **Stored Procedure** node and then select a stored procedure. The stored procedure **Properties** page appears.

   NOTE *Double-clicking on the Stored Procedure node from the tree also displays the **Properties** window.*

2. Click the **Permission** tab. The **Permission** page appears.



3. Select the **User/Group** to whom you want to grant or revoke the privilege.

©Copyright 1995-2026 CASEMaker Inc.

4. Click the **Modify** button.
5. Click the **Execute** column beside the **User/Group** to grant the execute privilege. A check mark indicates that the user has the execute privilege. By clicking on it again, you can revoke the privilege.
6. Click the **Confirm** button.

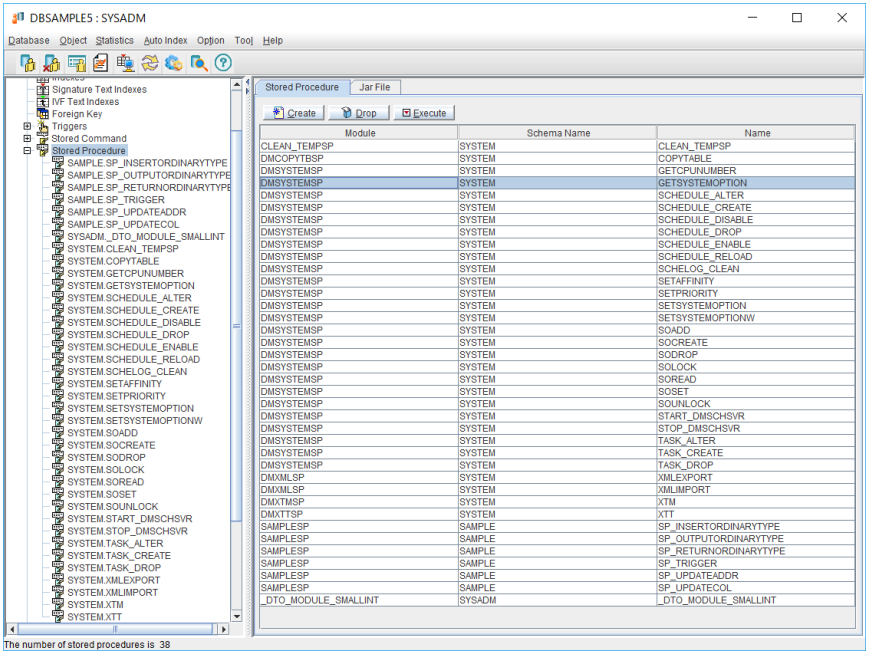
©Copyright 1995-2026 CASEMaker Inc.

12.4 Dropping a Stored Procedure

If a stored procedure is no longer required, it can be dropped. By default, owners of a stored procedure and users with RESOURCE authority or higher have DROP privilege for the stored procedure. A user only with CONNECT authority may not drop any stored procedure.

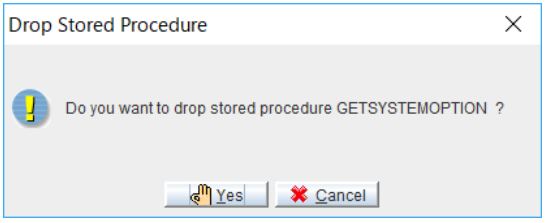
➲ Dropping a stored procedure:

1. Click the **Stored Procedure** object in the tree. All the stored procedures in the database appear.



2. Select a Stored Procedure.
3. Click the **Drop** button. A confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



4. Click the **Yes** button to drop the stored procedure or click the **Cancel** button to terminate the dropping process.

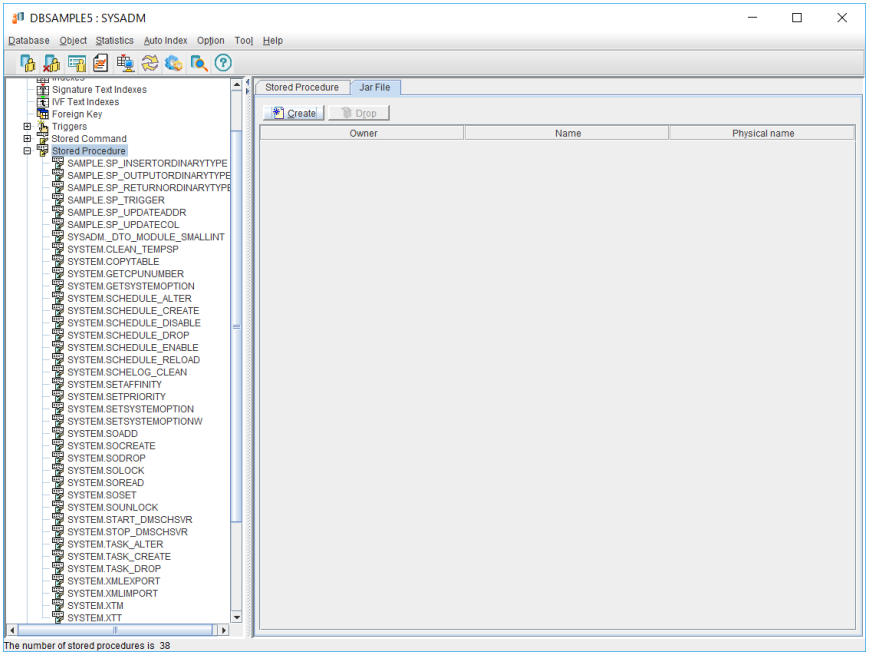
©Copyright 1995-2026 CASEMaker Inc.

12.5 Add a Jar File for Java Stored Procedure

Before creating Java stored procedure, you should register you jar file into the database. By default, users with RESOURCE authority or higher have CREATE privilege for the java stored procedure. A user only with CONNECT authority may not create any java stored procedure.

➲ Creating a Jarfile:

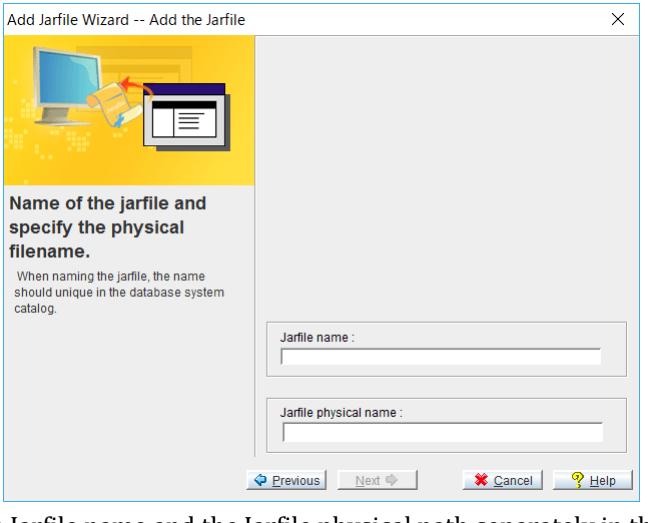
1. Click the object **Stored Procedure** in the tree. The Stored Procedures page will appear.
2. Click the **Jar File**. The Jar File page appears.



3. Click the **Create** button. The **Add Jarfile** wizard appears.

©Copyright 1995-2026 CASEMaker Inc.

4. Click the **Next** button. The **Name of the jarfile and specify the physical filename** wizard appears.

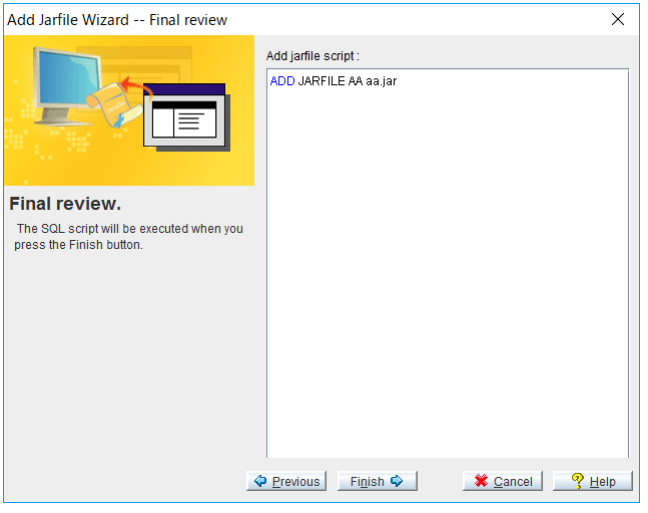


5. Input the Jarfile name and the Jarfile physical path separately in the proper filed.

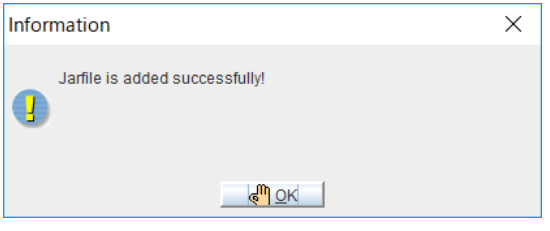
NOTE *Users need compile the Java stored procedure with Javac tool before this step. For details, please refer to the readme file under the JavaSP directory pointed by DBMaker's home intallation directory:*

C:\DBMaker\5.4\samples\JavaSP.

6. Click the **Next** button. The Final review wizard appears.



7. Review the script in the right console and if you are sure about it, click the **Finish** button**.** The added Jarfile window will appear.



8. Click the **OK** button.

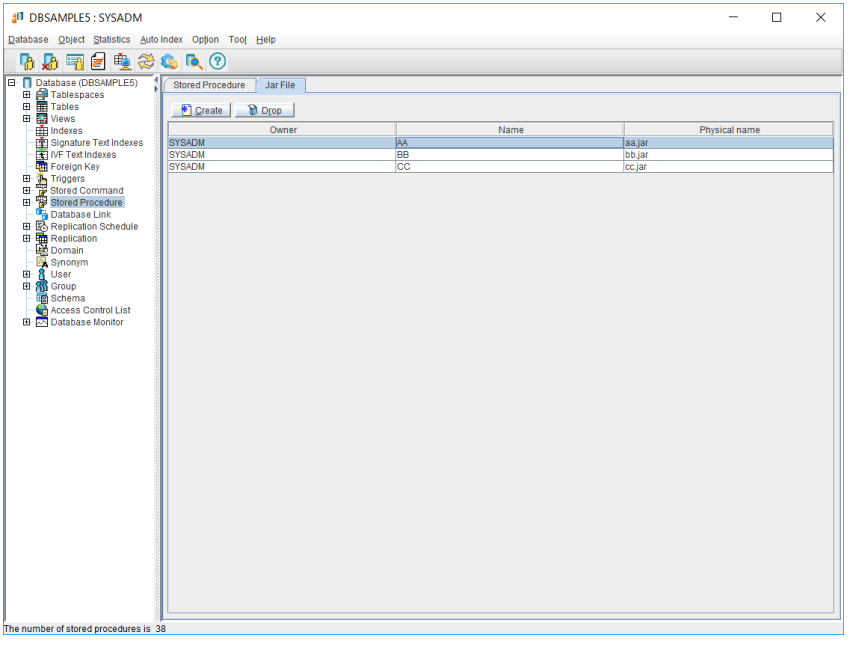
©Copyright 1995-2026 CASEMaker Inc.

12.6 Dropping Jar Files

When a jarfile is no longer required, you can drop it. By default, owners of a stored procedure and users with RESOURCE authority or higher have DROP privilege for the java stored procedure. A user only with CONNECT authority may not drop any java stored procedure.

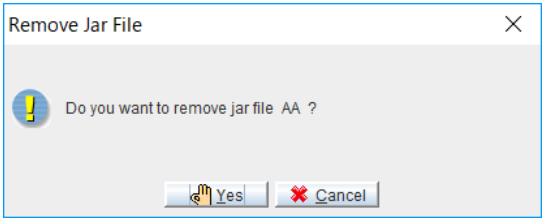
➲ Dropping a jar file:

1. Click the **Stored Procedure** icon from the tree, the Stored Procedure page will appear in the right panel, and then click the **Jar File** tab. A list of the all the Jarfile you have added is displayed.



2. Choose the Jarfile you want to drop, Click the **Drop** in the top of the page. The **Remove Jar File** confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



3. Click the **Yes** button. The remaining Jarfiles in the database are displayed.

©Copyright 1995-2026 CASEMaker Inc.

13 Working with Functions

After DBMaker 5.4.7, **CREATE FUNCTION** syntax is released. Users can design and create their own SQL functions. This chapter will show users how to create/delete SQL functions with JDBA Tool.

SQL function is a type of user-defined function (a subroutine that can return a single value) can be executed regularly in the database. SQL function provided by DBMaker is based on the syntax of stored procedure, but different from stored procedure, SQL functions cannot do DML/DDL operations except SELECT.

SQL Functions are similar to operators in that they operate on data items and return results. Functions differ from operators in the format of their arguments. This format enables them to operate on zero, one, two or more arguments.

Only users with DBA, SYSDBA or SYSADM privilege can see the **Function** in the object list and manage functions. But after the function is created, every user can execute that function (if that function selects data from table, users may also need table privilege).

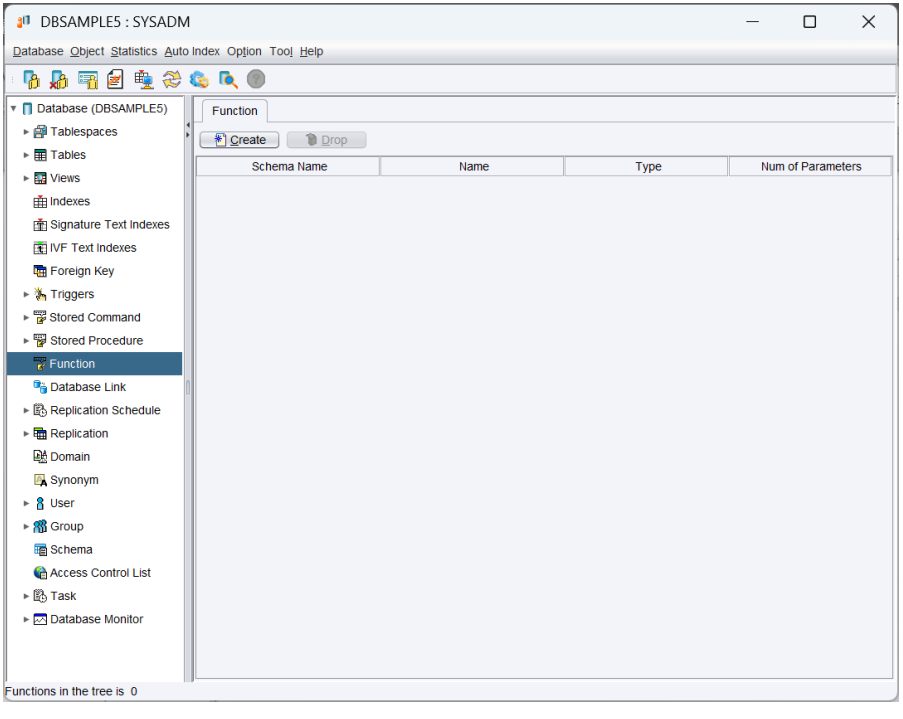
©Copyright 1995-2026 CASEMaker Inc.

13.1 Creating a Function

DBMaker provides three kinds of user-defined functions: SQL function, C function and Lua function. Only users with DBA, SYSDBA or SYSADM authority can create function.

➲ Creating a function:

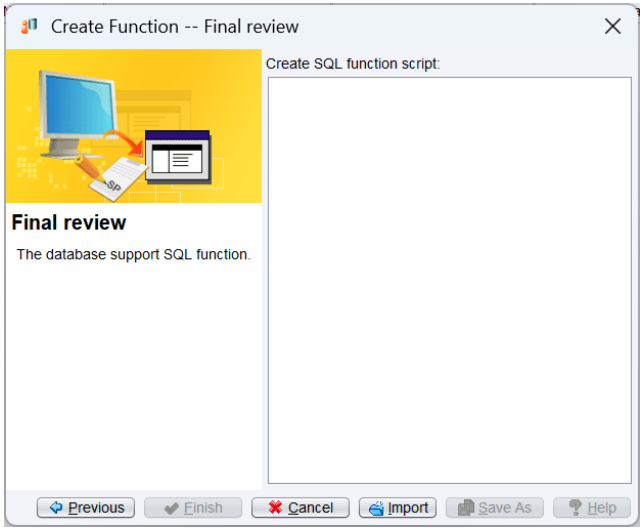
1. Click the **Function** object in the tree, the Function page will appear in the right panel.



©Copyright 1995-2026 CASEMaker Inc.

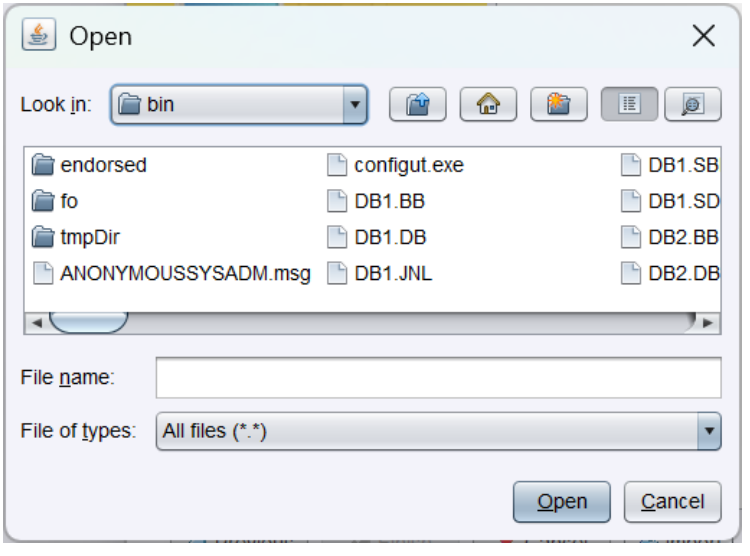
2. Click the **Create** button. The **Introduction** window of the **Create Function** wizard appears.

3. Click the **Next** button. The **Final Review** window appears. Input the function script or click the **Import** button to import SQL function script from a file.



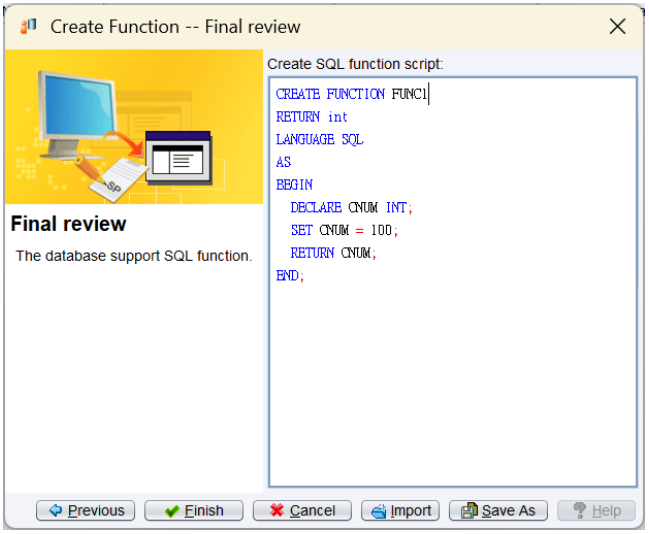
©Copyright 1995-2026 CASEMaker Inc.

4. Click the **Import** button. The **Open** window appears. Import files from any source, including the SPDIR directory of other databases on the server or network drives. Enter the path in the **File name** field, or browse the directory tree until the desired path is found.

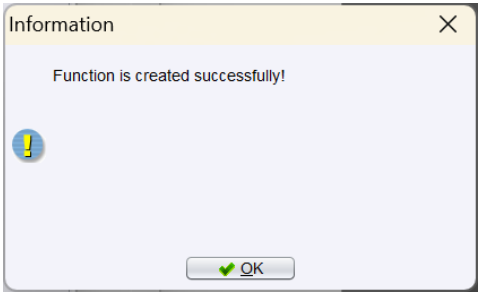


5. Click the **Open** button.
6. The **Final review** window reappears as in the example below or if users choose to manually enter the code. Click the **Save As** button to save the function script to another location, or click the **Finish** button to create and save the SQL function in the database.

©Copyright 1995-2026 CASEMaker Inc.



7. If the function is created successfully, the following message appears.



8. Click the **OK** button.

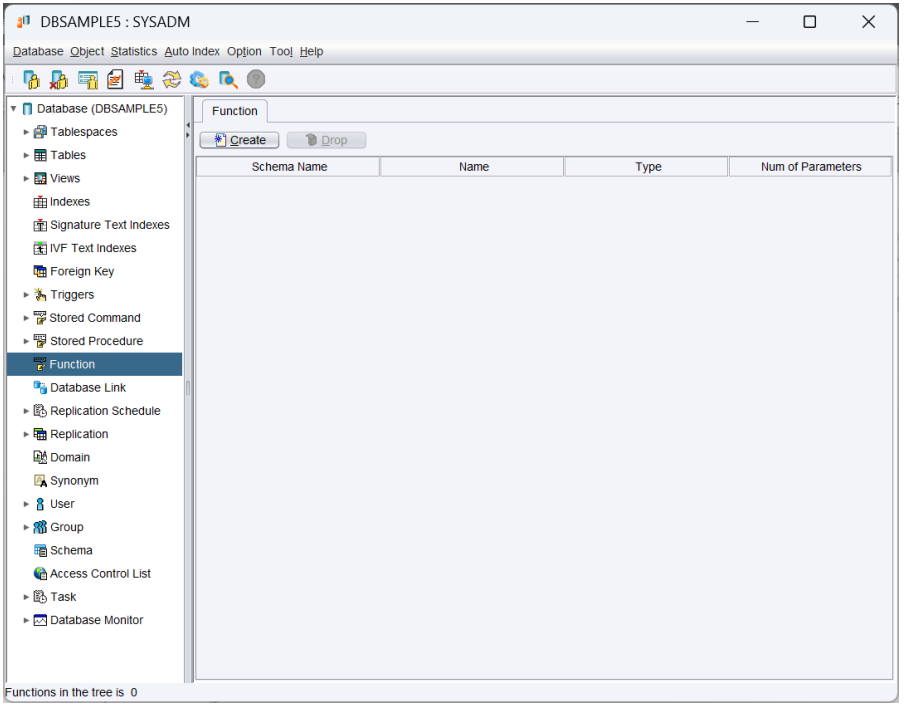
©Copyright 1995-2026 CASEMaker Inc.

13.2 Dropping a Function

If a function is no longer required, it can be dropped. By default, users with DBA privilege can drop their own function; users with SYSDBA privilege can drop all DBA function and their own function, SYSADM can drop every function.

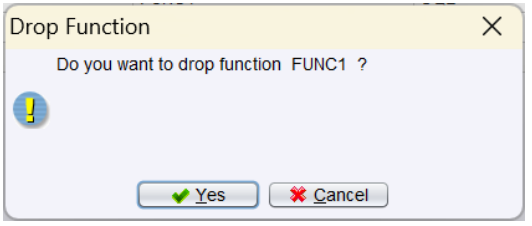
➲ Dropping a function:

1. Click the **Function** object in the tree, the Function page will appear in the right panel.



2. Select a function.
3. Click the **Drop** button. A confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



4. Click the **Yes** button to drop the function or click the **Cancel** button to terminate the dropping process.

©Copyright 1995-2026 CASEMaker Inc.

14 Working with Database Links

DBMaker effectively links multiple client applications and multiple database servers. Each client has a direct connection to a single database server, which is known as that client's Coordinator Database. Through the Coordinator Database, the client can connect to other remote databases, which are called Participant Databases. You can access a remote database by using database links defined in the Coordinator Database.

A database link creates a connection to a remote database, and contains the login information and password necessary for connecting to the remote database. This allows you to connect to a remote database with a different user name than you are using in the Coordinator Database, or to connect to a remote database where you do not have an account but can still connect using a public link. It also makes data in a distributed database environment location transparent. The link definition, which also contains the login information and password, is stored in the Coordinator Database.

Database links are also useful for the management of synchronous table replication, and can be used to define the connection between source and destination tables.

©Copyright 1995-2026 CASEMaker Inc.

14.1 Creating a Database Link

Only users with DBA authority or higher can create public database links for all users of that database. Users with RESOURCE authority or higher can create private database links for themselves. A user only with CONNECT authority may not create any database link. Multiple users may create private database links using the same name. If you create a private database link with a name identical to a public database link, the private database link will override the public database link.

DBMaker will create a public link by default if you do not specify the type of link to create. If you do not specify the User Name and password, your current login name and password are used by default.

©Copyright 1995-2026 CASEMaker Inc.

15 Working with Functions

After DBMaker 5.4.7, **CREATE FUNCTION** syntax is released. Users can design and create their own SQL functions. This chapter will show users how to create/delete SQL functions with JDBA Tool.

SQL function is a type of user-defined function (a subroutine that can return a single value) can be executed regularly in the database. SQL function provided by DBMaker is based on the syntax of stored procedure, but different from stored procedure, SQL functions cannot do DML/DDL operations except SELECT.

SQL Functions are similar to operators in that they operate on data items and return results. Functions differ from operators in the format of their arguments. This format enables them to operate on zero, one, two or more arguments.

Only users with DBA, SYSDBA or SYSADM privilege can see the **Function** in the object list and manage functions. But after the function is created, every user can execute that function (if that function selects data from table, users may also need table privilege).

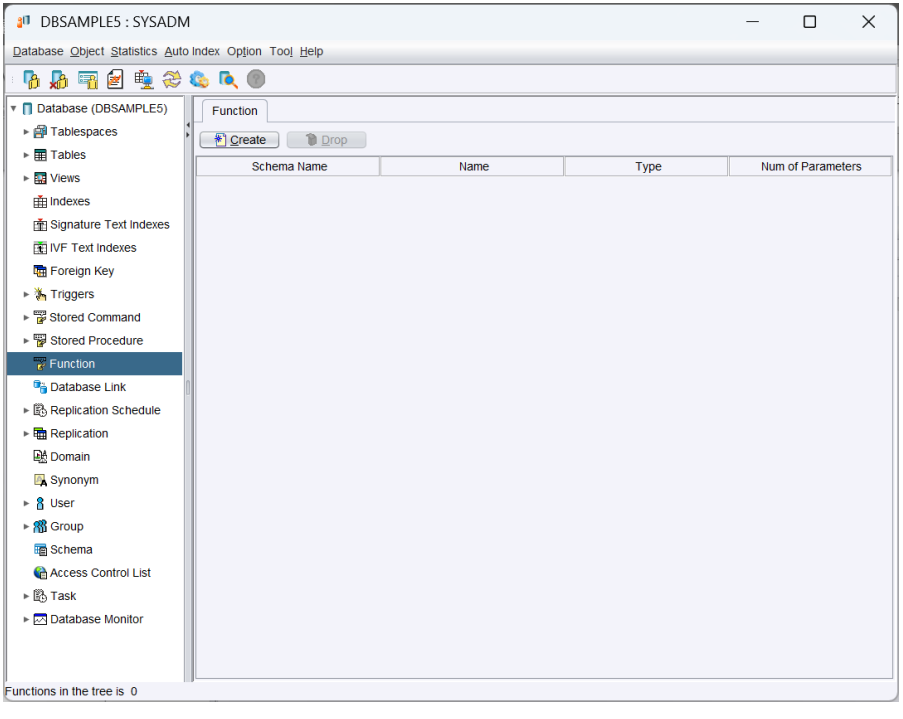
©Copyright 1995-2026 CASEMaker Inc.

15.1 Creating a Function

DBMaker provides three kinds of user-defined functions: SQL function, C function and Lua function. Only users with DBA, SYSDBA or SYSADM authority can create function.

➲ Creating a function:

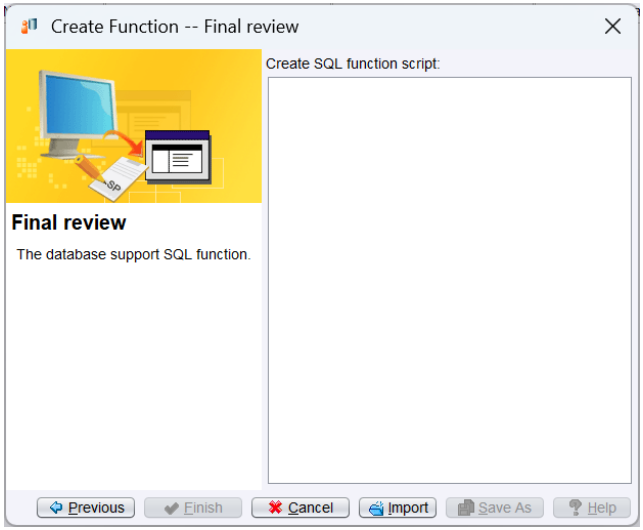
5. Click the **Function** object in the tree, the Function page will appear in the right panel.



©Copyright 1995-2026 CASEMaker Inc.

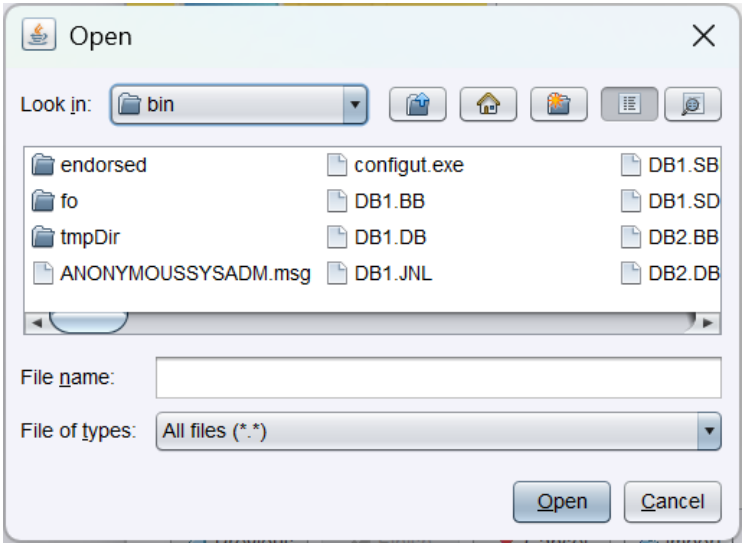
6. Click the **Create** button. The **Introduction** window of the **Create Function** wizard appears.

7. Click the **Next** button. The **Final Review** window appears. Input the function script or click the **Import** button to import SQL function script from a file.



©Copyright 1995-2026 CASEMaker Inc.

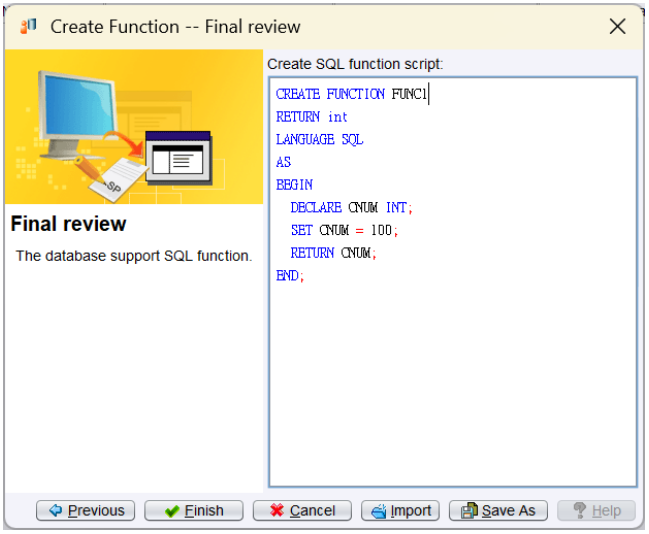
8. Click the **Import** button. The **Open** window appears. Import files from any source, including the SPDIR directory of other databases on the server or network drives. Enter the path in the **File name** field, or browse the directory tree until the desired path is found.



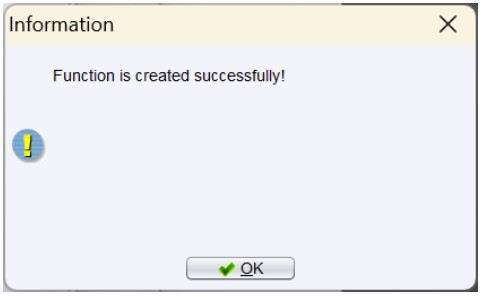
9. Click the **Open** button.

10. The **Final review** window reappears as in the example below or if users choose to manually enter the code. Click the **Save As** button to save the function script to another location, or click the **Finish** button to create and save the SQL function in the database.

©Copyright 1995-2026 CASEMaker Inc.



11. If the function is created successfully, the following message appears.



12. Click the **OK** button.

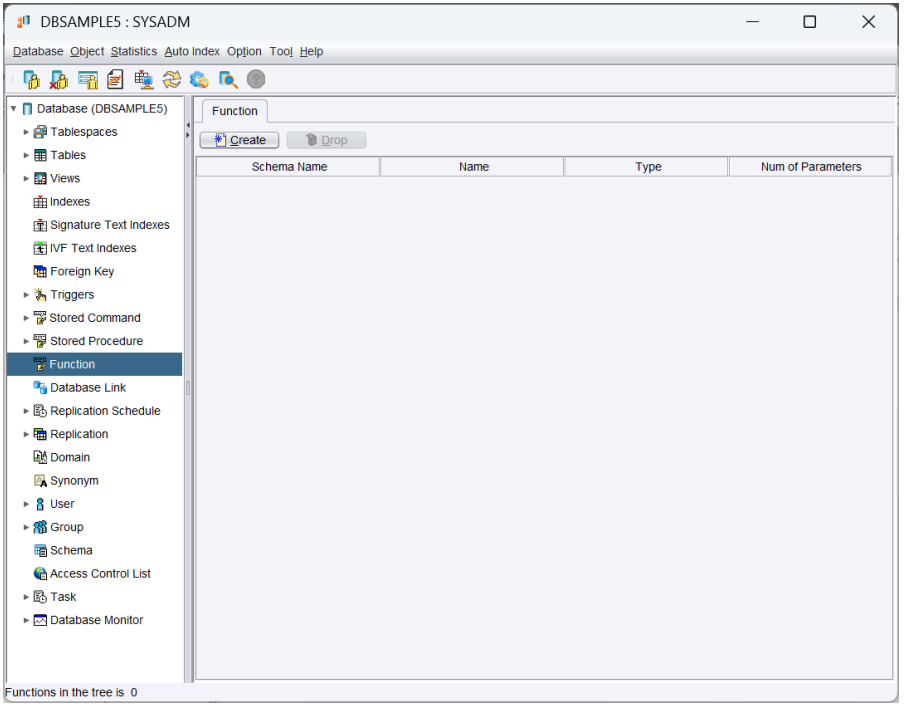
©Copyright 1995-2026 CASEMaker Inc.

15.2 Dropping a Function

If a function is no longer required, it can be dropped. By default, users with DBA privilege can drop their own function; users with SYSDBA privilege can drop all DBA function and their own function, SYSADM can drop every function.

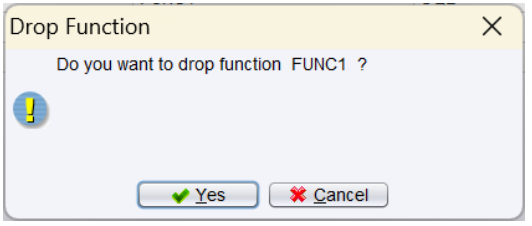
➲ Dropping a function:

13. Click the **Function** object in the tree, the Function page will appear in the right panel.



14. Select a function.
15. Click the **Drop** button. A confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.

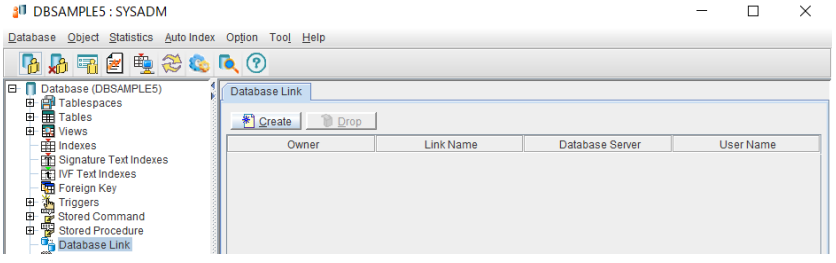


16. Click the **Yes** button to drop the function or click the **Cancel** button to terminate the dropping process.

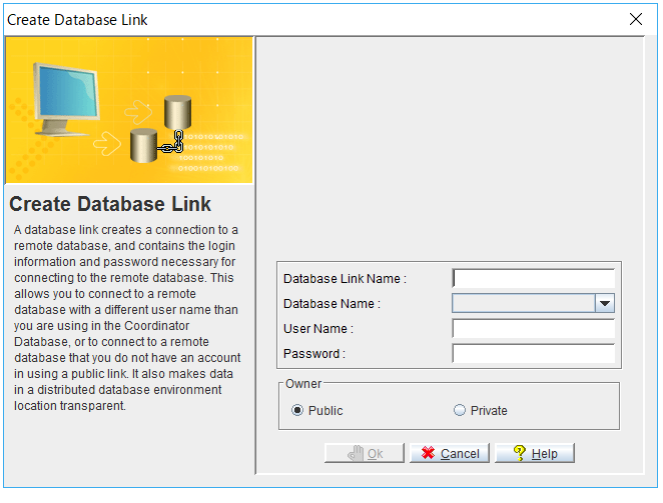
©Copyright 1995-2026 CASEMaker Inc.

➲ Creating a database Link:

1. Select the object **Database Link** in the tree. A list of all database links in the database appears.



2. Click the **Create** button. The **Create Database Link** window appears.



3. Enter the database link name.
4. Select the database to be the coordinator database from the **Database Name** menu.
5. The default user name for the selected coordinator database is displayed in the **User Name** field. A new name can be entered.
6. Enter the password.
7. If the owner of the database link is private (exclusively for a single user), select the **Private** option button. If the database link is to be used

©Copyright 1995-2026 CASEMaker Inc.

by multiple users, select the **Public** option button. The default setting is **Public**.

8. Click the **OK** button. The newly created database link can be seen in the list of all the database links.

©Copyright 1995-2026 CASEMaker Inc.

15.3 Dropping a Database Link

If the database link is no longer required, drop it. By default, owners of a database link and users with RESOURCE authority or higher have DROP privilege for the database link. A user only with CONNECT authority may not drop any database link.

©Copyright 1995-2026 CASEMaker Inc.

16 Working with Functions

After DBMaker 5.4.7, **CREATE FUNCTION** syntax is released. Users can design and create their own SQL functions. This chapter will show users how to create/delete SQL functions with JDBA Tool.

SQL function is a type of user-defined function (a subroutine that can return a single value) can be executed regularly in the database. SQL function provided by DBMaker is based on the syntax of stored procedure, but different from stored procedure, SQL functions cannot do DML/DDL operations except SELECT.

SQL Functions are similar to operators in that they operate on data items and return results. Functions differ from operators in the format of their arguments. This format enables them to operate on zero, one, two or more arguments.

Only users with DBA, SYSDBA or SYSADM privilege can see the **Function** in the object list and manage functions. But after the function is created, every user can execute that function (if that function selects data from table, users may also need table privilege).

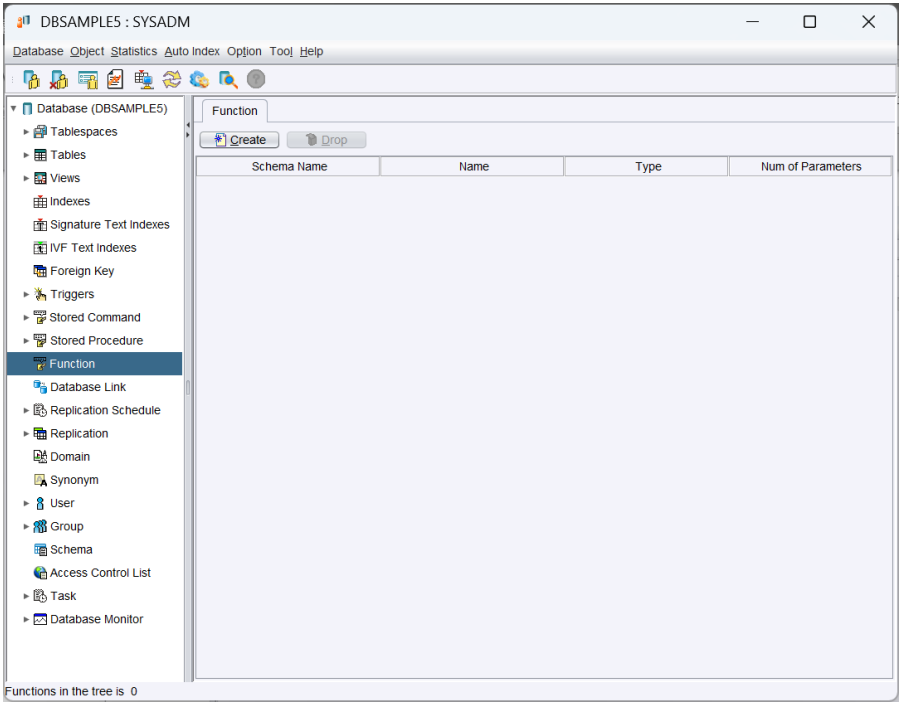
©Copyright 1995-2026 CASEMaker Inc.

16.1 Creating a Function

DBMaker provides three kinds of user-defined functions: SQL function, C function and Lua function. Only users with DBA, SYSDBA or SYSADM authority can create function.

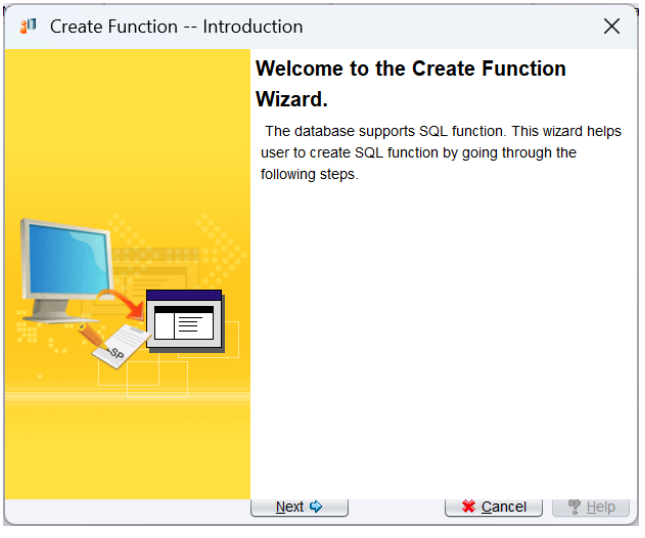
➲ Creating a function:

9. Click the **Function** object in the tree, the Function page will appear in the right panel.

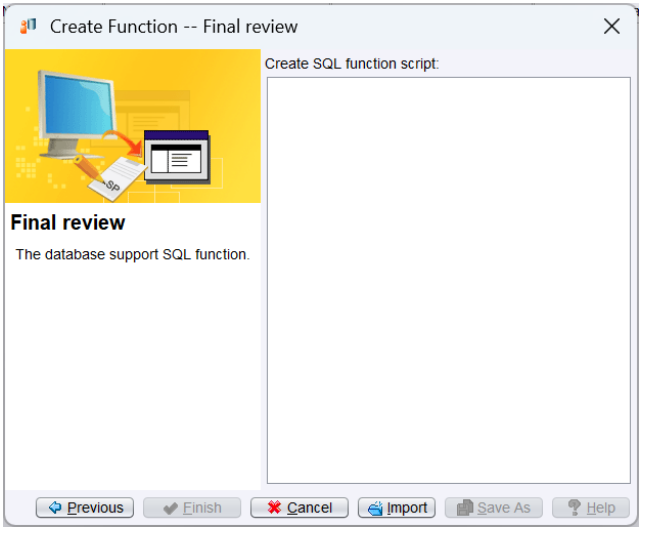


©Copyright 1995-2026 CASEMaker Inc.

10. Click the **Create** button. The **Introduction** window of the **Create Function** wizard appears.

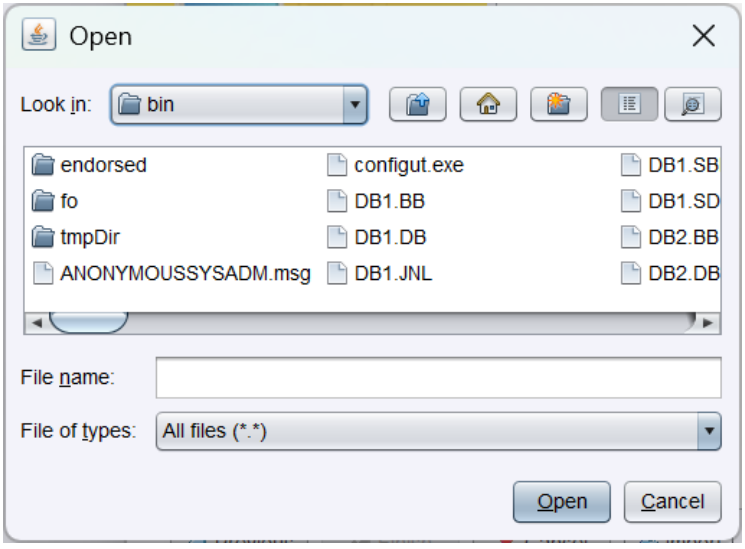


11. Click the **Next** button. The **Final Review** window appears. Input the function script or click the **Import** button to import SQL function script from a file.



©Copyright 1995-2026 CASEMaker Inc.

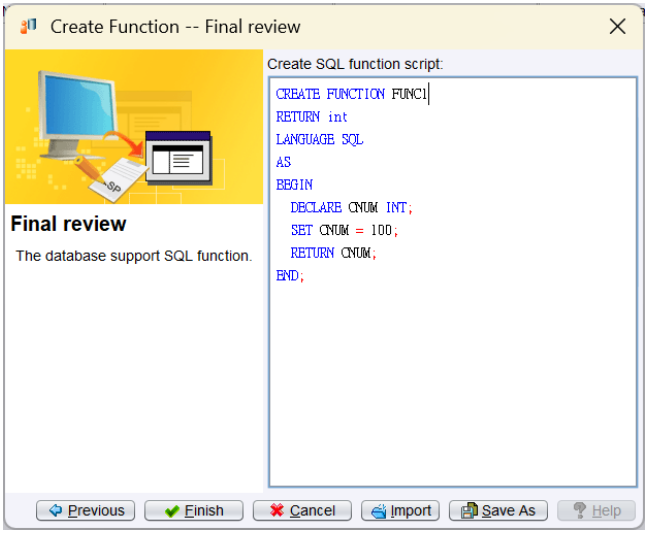
12. Click the **Import** button. The **Open** window appears. Import files from any source, including the SPDIR directory of other databases on the server or network drives. Enter the path in the **File name** field, or browse the directory tree until the desired path is found.



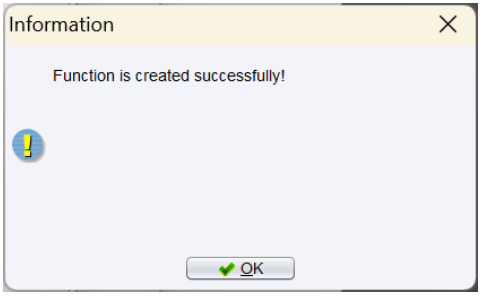
13. Click the **Open** button.

14. The **Final review** window reappears as in the example below or if users choose to manually enter the code. Click the **Save As** button to save the function script to another location, or click the **Finish** button to create and save the SQL function in the database.

©Copyright 1995-2026 CASEMaker Inc.



15. If the function is created successfully, the following message appears.



16. Click the **OK** button.

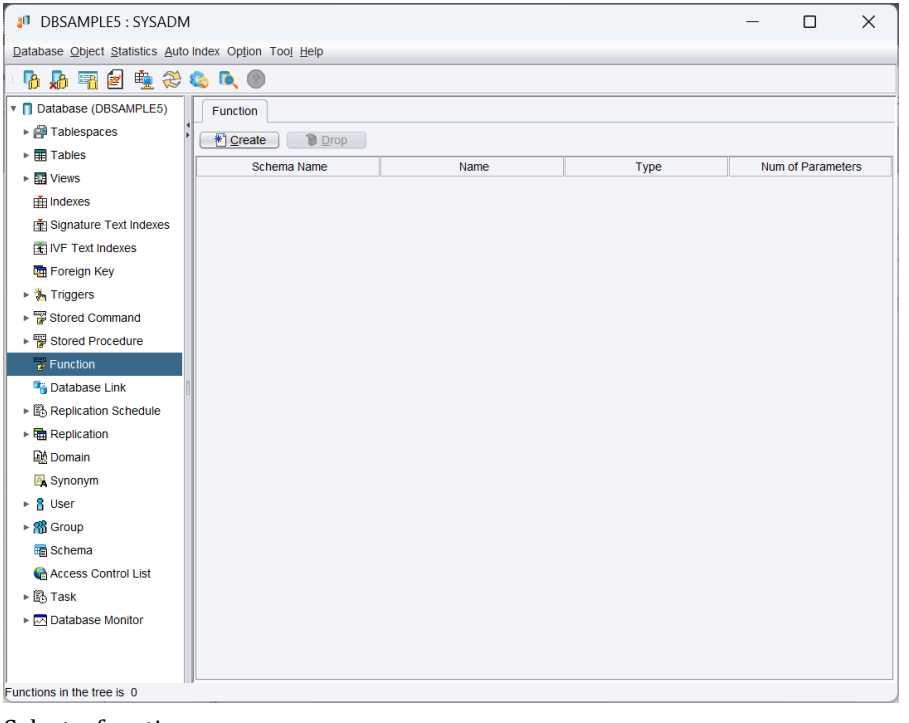
©Copyright 1995-2026 CASEMaker Inc.

16.2 Dropping a Function

If a function is no longer required, it can be dropped. By default, users with DBA privilege can drop their own function; users with SYSDBA privilege can drop all DBA function and their own function, SYSADM can drop every function.

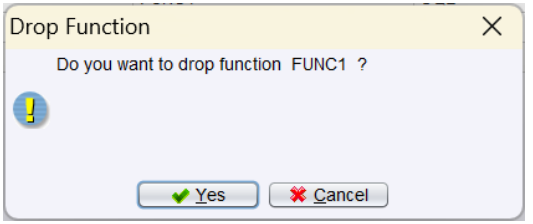
➲ Dropping a function:

17. Click the **Function** object in the tree, the Function page will appear in the right panel.



18. Select a function.
19. Click the **Drop** button. A confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.

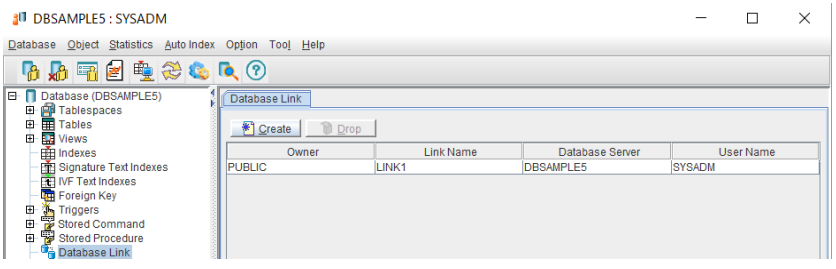


20. Click the **Yes** button to drop the function or click the **Cancel** button to terminate the dropping process.

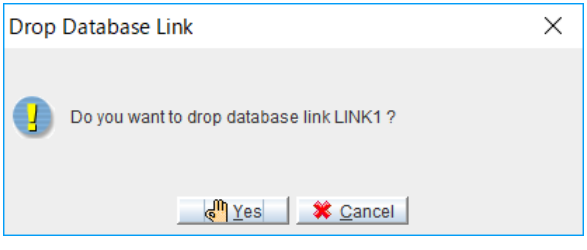
©Copyright 1995-2026 CASEMaker Inc.

➲ Dropping a database link:

1. Select the object **Database Link** in the tree. A list of all database links in the database is displayed.
2. Select a database link.



3. Click the **Drop** button. The **Drop Database Link** dialog box is displayed.



4. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.

17 Working with Replication Schedule

Replication schedules are user-defined schedules that tell the replication daemon when to begin the process of updating replicated data to destination (subscriber) databases. Replication that is carried out by this process is called asynchronous table replication. 'Asynchronous' means the modification to the remote site is delayed. The time difference from source to destination database depends on the replication schedule. Each destination database must have its own replication schedule. Modifications to local tables are stored in replication logs, and are replicated to the destination table(s) according to the replication schedule. Using replication logs enables DBMaker to treat the local transaction and the remote transaction independently, allowing you to update local tables normally even if the remote connection is not available. DBMaker keeps trying to replicate until any failures are corrected or until the user-specified number of attempts to reconnect is reached. This allows asynchronous table replications to tolerate network and remote database failures.

Asynchronous table replication depends on the establishment of a replication schedule. Likewise, developing a replication schedule without an asynchronous table replication that uses it is meaningless. A replication schedule should first be developed. Afterward, the asynchronous table replication can be created.

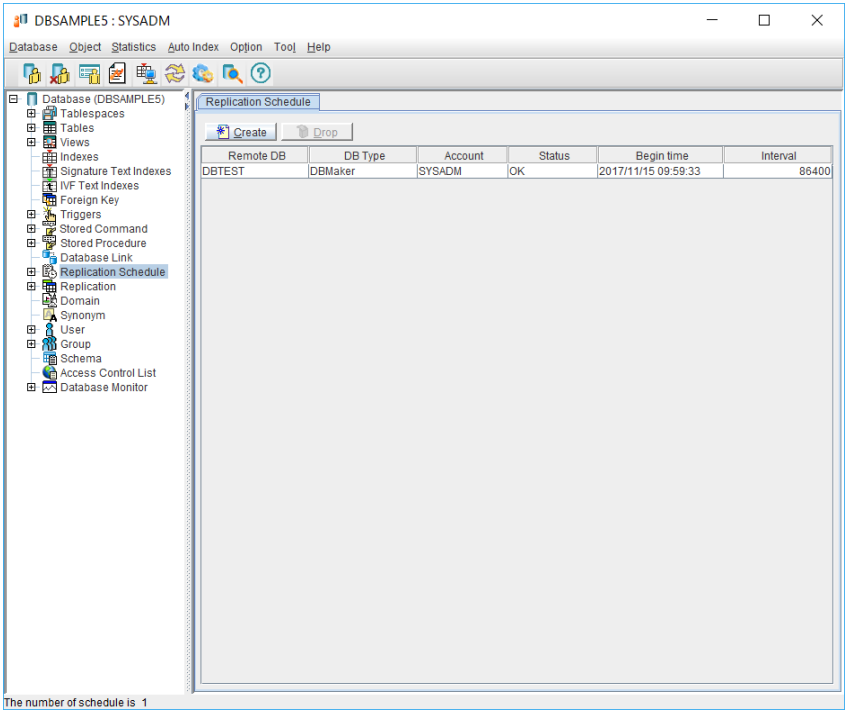
©Copyright 1995-2026 CASEMaker Inc.

17.1 Creating a Replication Schedule

Before creating asynchronous replications to remote tables, a user with DBA authority or higher needs to define a *schedule*. A schedule defines the starting time of replication, the replicating period, and the account and password used to connect to the remote database. Several schedules for different remote databases in the same source database may be created, but no more than one schedule can be made for the same remote database.

➲ Creating a Replication Schedule:

1. Select the object **Replication Schedule** from the tree. The following window is displayed.

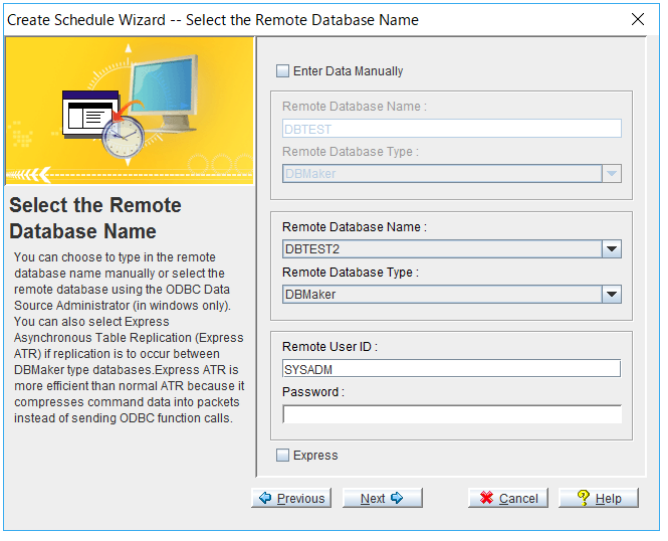


©Copyright 1995-2026 CASEMaker Inc.

2. Click the **Create** button. The **Create Schedule Wizard – Introduction** window is opened.

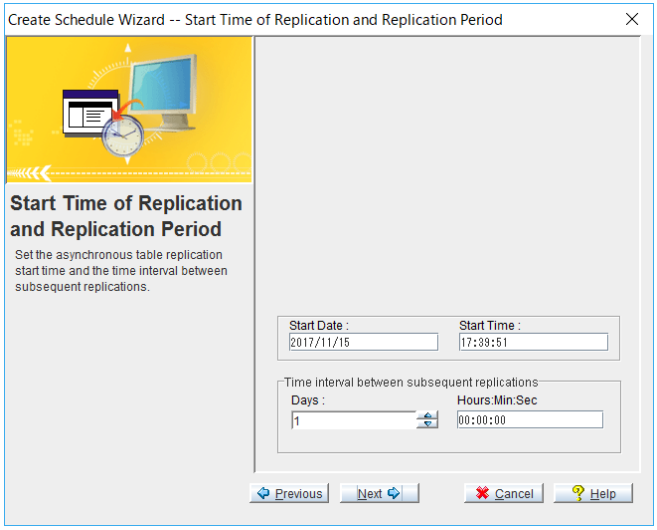
3. Click the **Next** button. The **Select the Remote Database Name** window appears.

©Copyright 1995-2026 CASEMaker Inc.



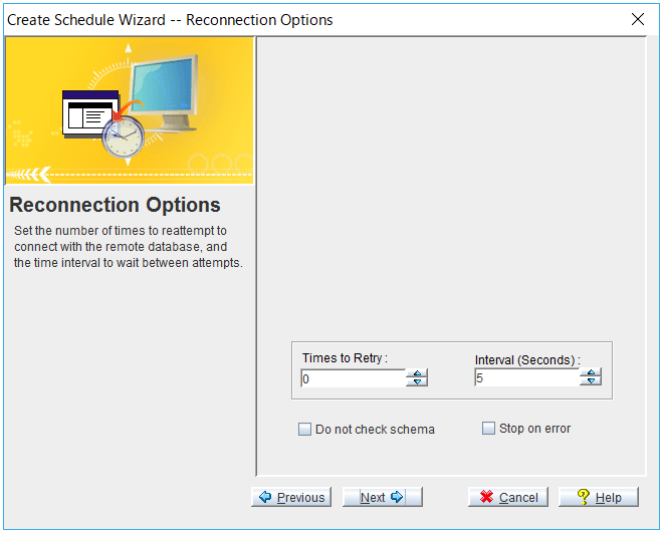
4. If you choose to enter the remote database name manually, click the **Enter Data Manually** check box and enter the Remote Database Name.
5. Select the **Remote Database Type** from the menu.
6. If you do not choose to enter data manually, you can select the **Remote Database Name** and the **Remote Database Type** from the respective menus.
7. Enter the **User ID** for the remote database.
8. Enter the **password.**
9. Click the **Express** check box if you want to enable express asynchronous table replication.
10. Click the **Next** button. The **Start Time of Replication and Replication Period** window appears.

©Copyright 1995-2026 CASEMaker Inc.



11. Enter a date, in the **Start Date** box, when the Asynchronous Table Replication will begin.
12. Enter the starting time of Asynchronous Table Replication in the **Start Time** box.
13. In the **Time interval between subsequent replications** field:
    - **a)** Enter the number of days between which the Asynchronous Table Replication will performed in the Days combo box.
    - **b)** Enter the interval of time in which the Asynchronous Table Replication will be done by entering the time interval in hours, minutes and seconds in the **Hours:Min:Sec** combo box. The total time between replications is equal to the sum of the **Days** and **Hours:Min:Sec** fields.
14. Click the **Next** button. The **Reconnection Options** window appears.

©Copyright 1995-2026 CASEMaker Inc.

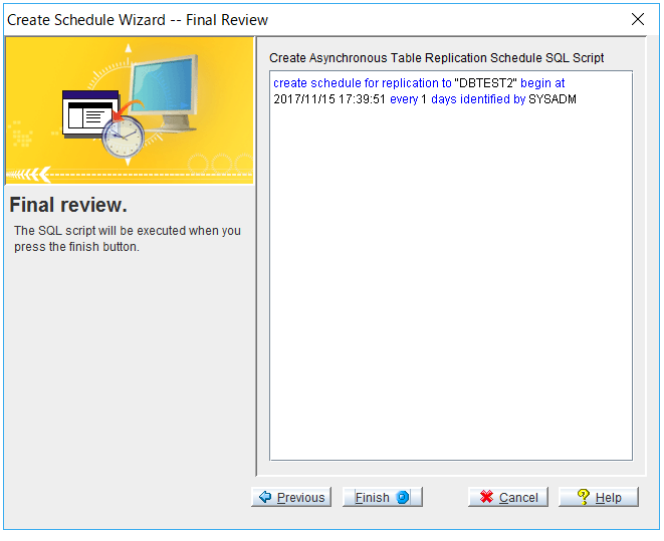


15. Enter the number of times the replication process has to be retried if any failure occurs and replication has not been completed in the **Times to Retry** combo box. Using the upward and downward arrows, you can increase or decrease the number.
16. Enter the time duration the replication process has to wait before trying again in the **Interval** box. Using the upward and downward arrows, you can increase or decrease the number.
17. Click the **Do not check schema** check box if you do not want the Replication process to check the table schema before replicating.
18. Click the **Stop on error** check box if you want the Replication process to stop if any error occures.

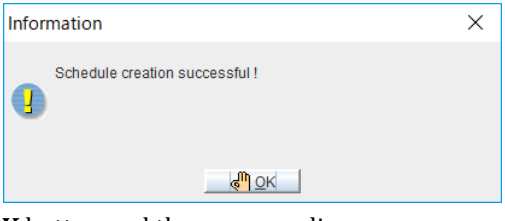
    NOTE *The Stop on error check box is disabled if you want an Express Asynchronous Table Replication.*

19. Click the **Next** button. The following window appears.

©Copyright 1995-2026 CASEMaker Inc.



20. The SQL script for replication schedule creation is displayed. You can modify it if required.
21. Click the **Finish** button. The **Information** dialog box appears.



22. Click the **OK** button and the message disappears.

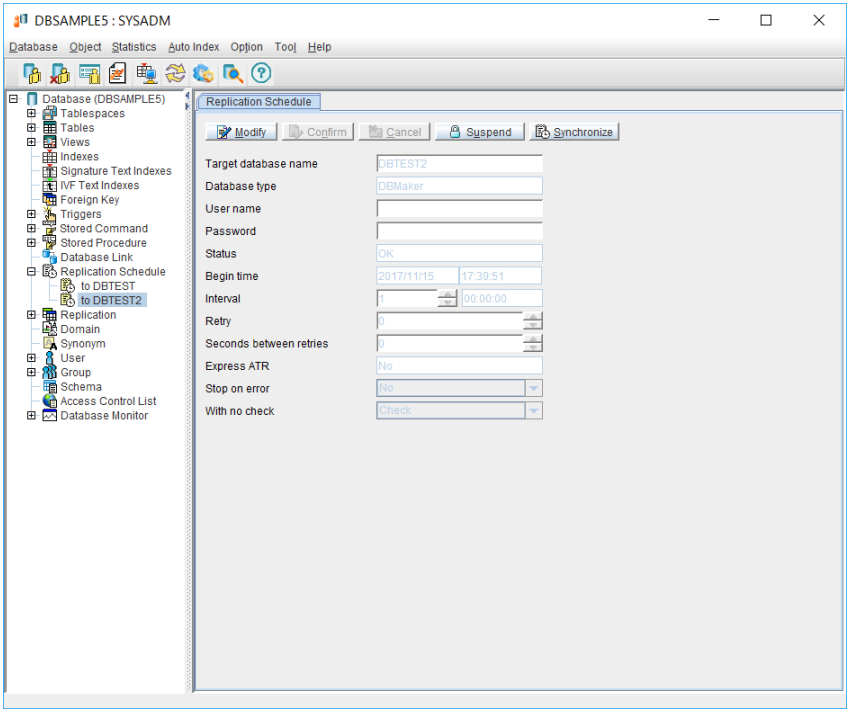
©Copyright 1995-2026 CASEMaker Inc.

17.2 Modifying a Replication Schedule

If a replication schedule has to be changed, it can be modified by the following procedure.

➲ Modifying a replication schedule:

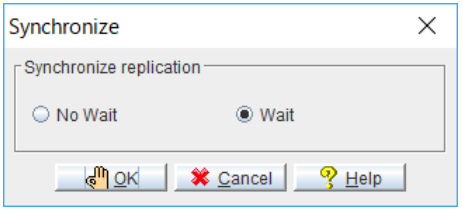
1. Open the node **Replication Schedule** in the tree and select a replication schedule. The following screen appears.



2. Click the **Modify** button. All the replication schedule information is displayed. The **Target Database** name and **Database Type** are disabled and they cannot be modified.

©Copyright 1995-2026 CASEMaker Inc.

3. The **User Name** can be changed.
4. By clicking the **Suspend** button, you can change the status of the replication schedule to suspend. Once the status of the replication schedule is changed to suspend, you can click the **Resume** button to change the status to **OK**.
5. The **Begin Time** of the replication schedule can be modified.
6. The interval of time in which replication has to be done, the number of days and time interval can be modified.
7. The number of times to **Retry** and the interval of time between two consecutive retries can also be modified.
8. Replication is to be **Stopped on Error** can be changed to **Yes** or **No**.
9. The **With no Check** button is disabled; whether or not schema is to be checked before replication cannot be changed.
10. Click the **Confirm** button.
11. Click the **Synchronous** button to synchronize replication. A window is opened as shown.



12. Click the **Wait** radio button or click the **No Wait** radio button.
    - **Wait** – the user must wait for the completion of synchronization before he/she can do other operation.
    - **No Wait** – after triggering the synchronization process, the user can continue other processes instead of waiting for the completion of the synchronization process.
13. Click the **OK** button.

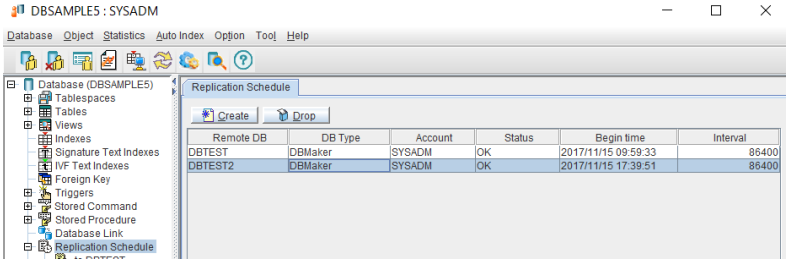
©Copyright 1995-2026 CASEMaker Inc.

17.3 Dropping a Replication Schedule

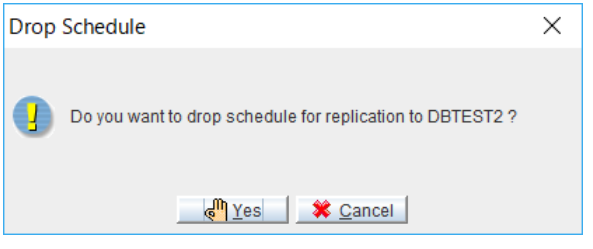
If you do not want to use a replication schedule any more it can be dropped.

➲ Dropping a replication schedule:

1. Select the object **Replication Schedule** from the tree. The list of all the replication schedules in the database is displayed.
2. Select a replication schedule. It will highlight blue.



3. Click the **Drop** button. A confirmation window appears.



4. Click the **Yes** button.

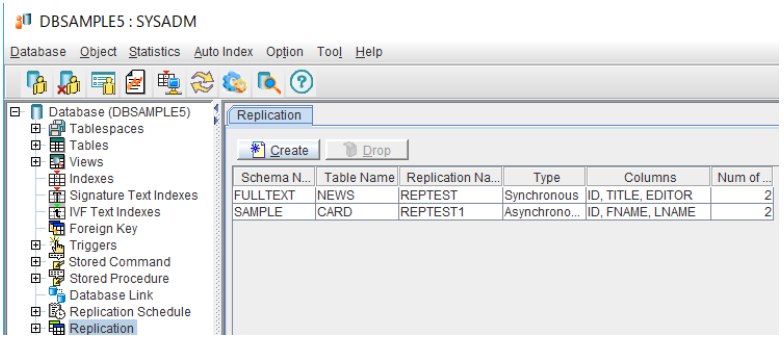
©Copyright 1995-2026 CASEMaker Inc.

18 Managing Table Replication

DBMaker can use table replication to share data between databases. Table replication can take place simultaneously (synchronous table replication), or by means of the distributor daemon, which periodically updates data on the destination (i.e., target) database. Asynchronous Table Replication or ATR are available. JDBA Tool provides easy to use tools for creation and management of synchronous and asynchronous table replication.

➲ Viewing Table Replications:

1. Select **Replication** from the tree, the **Replication** page will display. You can view a list of replications by owner, source table name, replication name, replication type (synchronous or asynchronous), source columns, and the number of subscribers to the replication (the number of destination tables).



©Copyright 1995-2026 CASEMaker Inc.

2. Selecting a replication from the tree or the list on the **Replication** page opens the page for that replication. You can view the link names, owners, table names, column names, and settings for each link to a destination table.

©Copyright 1995-2026 CASEMaker Inc.

18.1 Synchronous Table Replication

Table replication ensures the consistency of data across multiple databases. Synchronous means the modification to the remote site is immediate. The replication modifies the remote table at the same time it modifies the local table. Thus, after establishing a replication, any update on the source table will become a DDB (distributed database) action. This affects the local database's behavior; if the remote database server is cannot be contacted an update on the local database will fail.

©Copyright 1995-2026 CASEMaker Inc.

19 Working with Functions

After DBMaker 5.4.7, **CREATE FUNCTION** syntax is released. Users can design and create their own SQL functions. This chapter will show users how to create/delete SQL functions with JDBA Tool.

SQL function is a type of user-defined function (a subroutine that can return a single value) can be executed regularly in the database. SQL function provided by DBMaker is based on the syntax of stored procedure, but different from stored procedure, SQL functions cannot do DML/DDL operations except SELECT.

SQL Functions are similar to operators in that they operate on data items and return results. Functions differ from operators in the format of their arguments. This format enables them to operate on zero, one, two or more arguments.

Only users with DBA, SYSDBA or SYSADM privilege can see the **Function** in the object list and manage functions. But after the function is created, every user can execute that function (if that function selects data from table, users may also need table privilege).

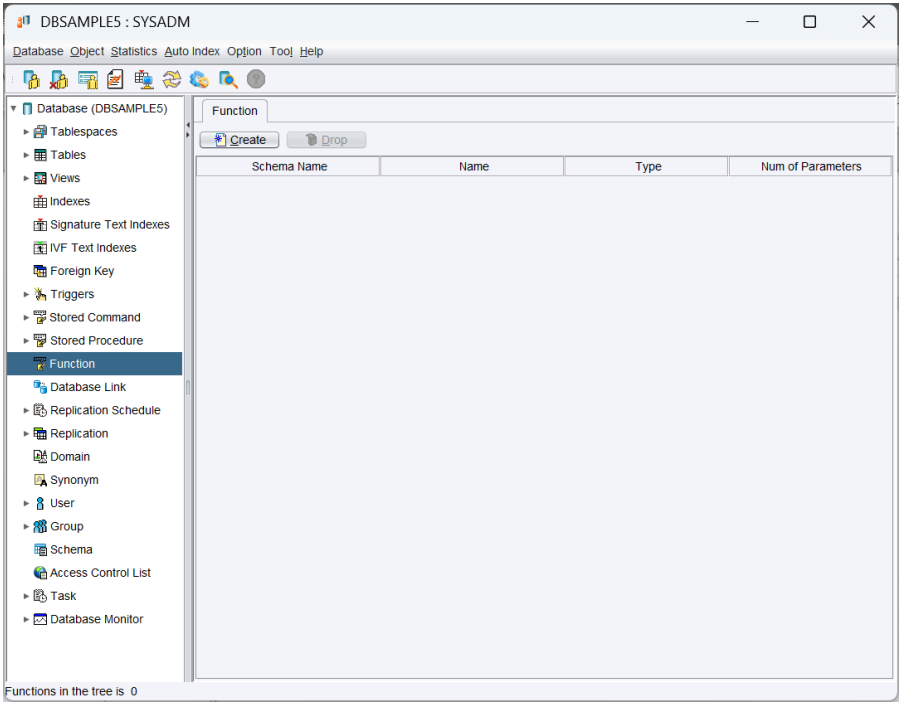
©Copyright 1995-2026 CASEMaker Inc.

19.1 Creating a Function

DBMaker provides three kinds of user-defined functions: SQL function, C function and Lua function. Only users with DBA, SYSDBA or SYSADM authority can create function.

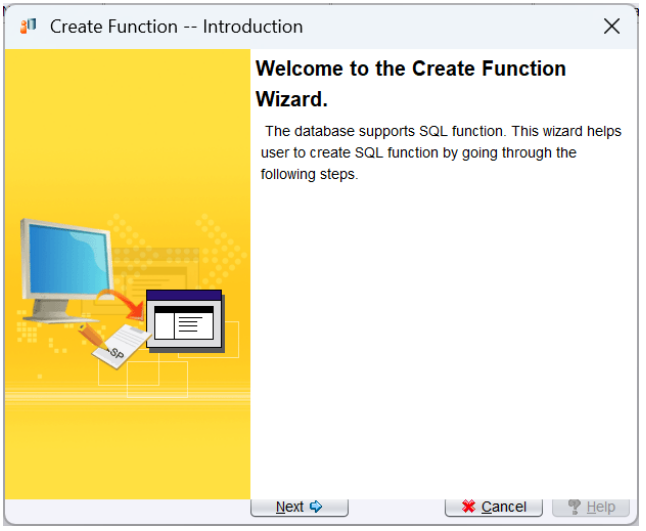
➲ Creating a function:

3. Click the **Function** object in the tree, the Function page will appear in the right panel.

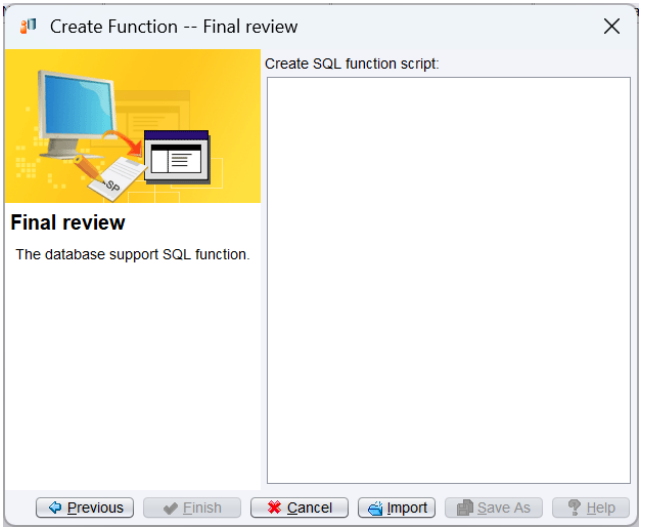


©Copyright 1995-2026 CASEMaker Inc.

4. Click the **Create** button. The **Introduction** window of the **Create Function** wizard appears.

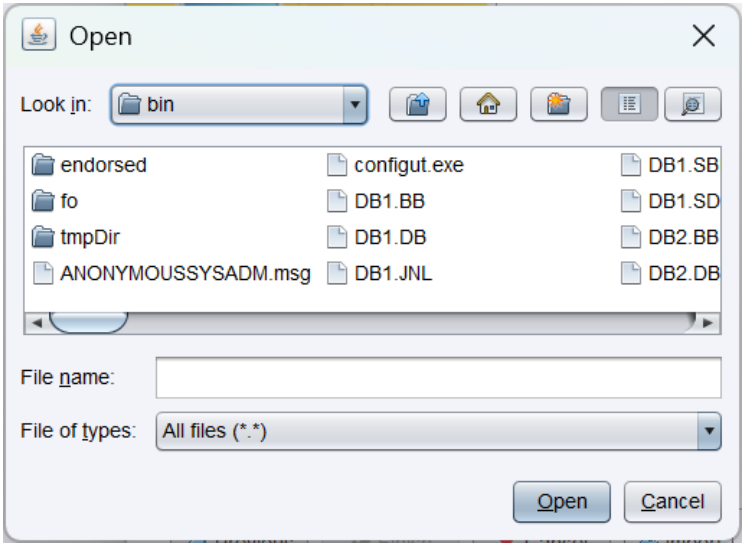


5. Click the **Next** button. The **Final Review** window appears. Input the function script or click the **Import** button to import SQL function script from a file.



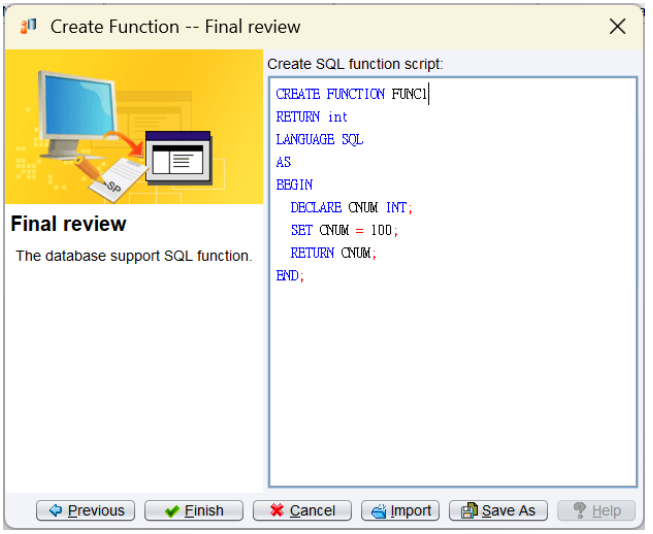
©Copyright 1995-2026 CASEMaker Inc.

6. Click the **Import** button. The **Open** window appears. Import files from any source, including the SPDIR directory of other databases on the server or network drives. Enter the path in the **File name** field, or browse the directory tree until the desired path is found.

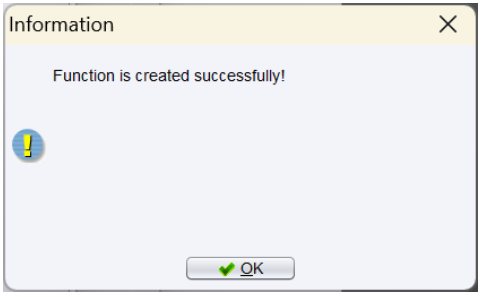


7. Click the **Open** button.
8. The **Final review** window reappears as in the example below or if users choose to manually enter the code. Click the **Save As** button to save the function script to another location, or click the **Finish** button to create and save the SQL function in the database.

©Copyright 1995-2026 CASEMaker Inc.



9. If the function is created successfully, the following message appears.



10. Click the **OK** button.

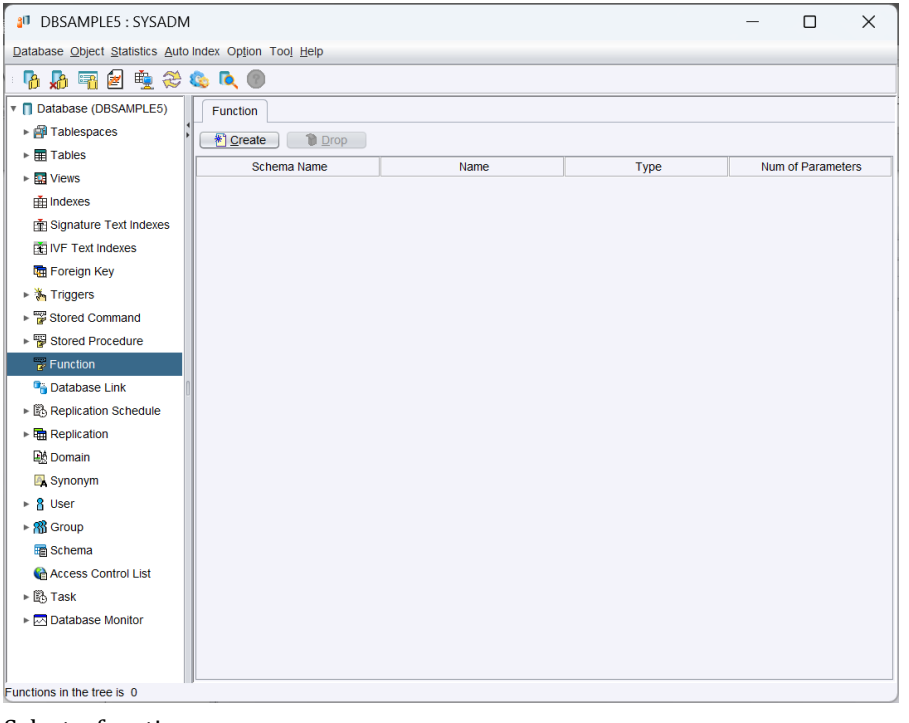
©Copyright 1995-2026 CASEMaker Inc.

19.2 Dropping a Function

If a function is no longer required, it can be dropped. By default, users with DBA privilege can drop their own function; users with SYSDBA privilege can drop all DBA function and their own function, SYSADM can drop every function.

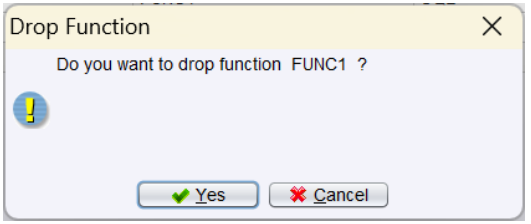
➲ Dropping a function:

11. Click the **Function** object in the tree, the Function page will appear in the right panel.



12. Select a function.
13. Click the **Drop** button. A confirmation dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



14. Click the **Yes** button to drop the function or click the **Cancel** button to terminate the dropping process.

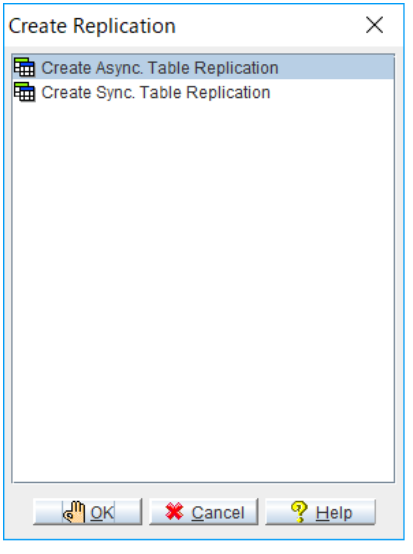
©Copyright 1995-2026 CASEMaker Inc.

Adding Synchronous Table Replication

Synchronous table replication must be created for it to occur. Furthermore, distributed database mode must be enabled in all databases engaged in synchronous table replication. The schema of the destination columns must match the schema of the source columns. Primary key columns in the source table must be replicated. By default, users with RESOURCE authority or higher can create a synchronous table replication. A user only with CONNECT authority may not create any synchronous table replication.

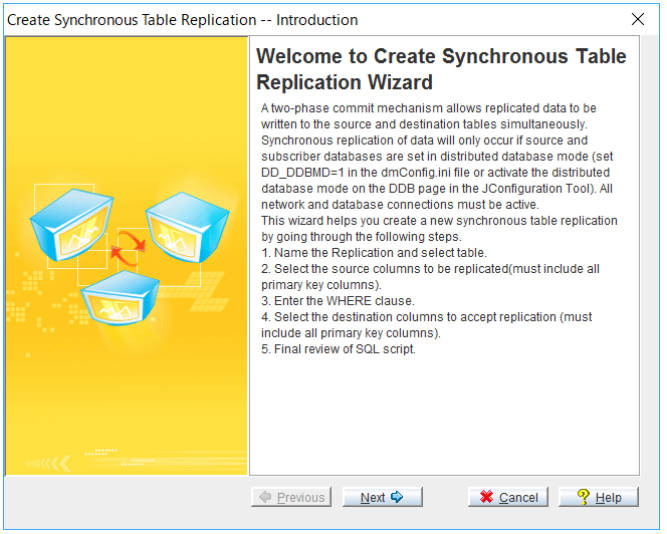
➲ Creating a Synchronous Table Replication:

1. Select **Create** from the top of the **Replication** page. The **Create Replication** window will appear.



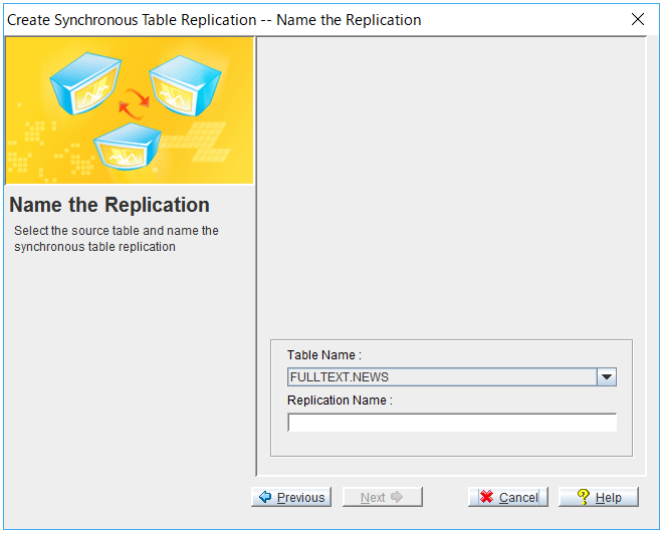
2. Select **Synchronous Table Replication** from the menu, the **Synchronous Table Replication Wizard** will begin.

©Copyright 1995-2026 CASEMaker Inc.

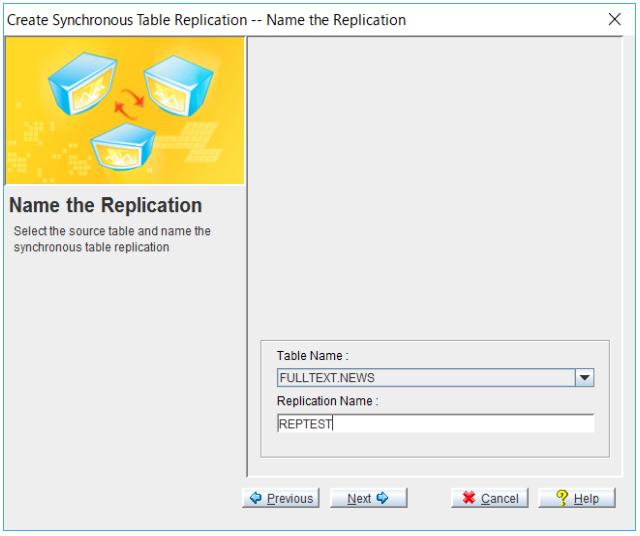


©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Synchronous Table Replication Wizard** will appear.

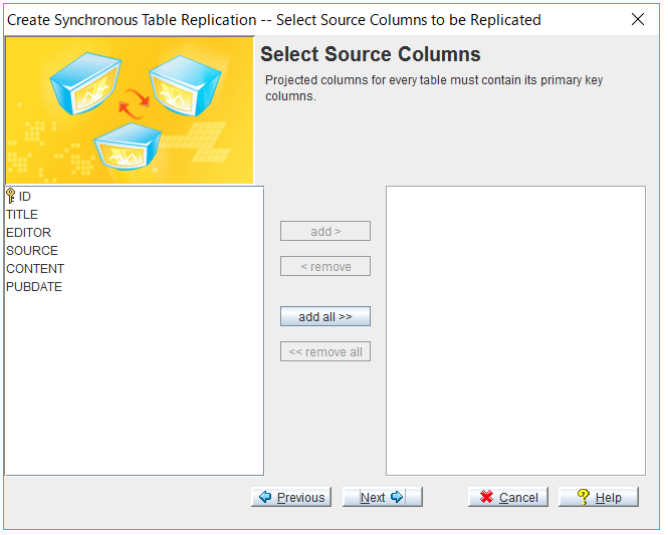


4. Identify the source table to replicate and name the replication. Select the source table from the **Table Name** menu at the bottom of the window. Enter a name into the **Replication Name** field.



©Copyright 1995-2026 CASEMaker Inc.

5. Click the **Next** button. The **Select Source Columns to be Replicated** window will appear.

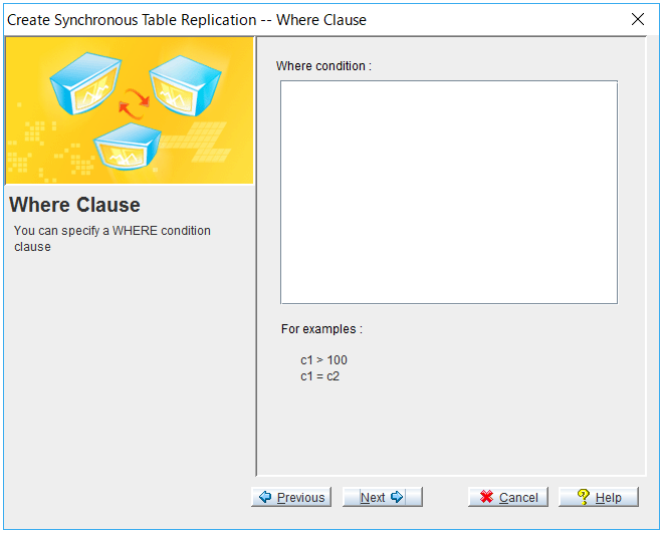


6. Select source columns from the source table. A list of available columns on the selected table appears on the left hand list box. Select source columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button adds all columns to the replication. The list box to the right displays all the selected source columns. Remove columns from the replication by double-clicking the column name or selecting the column and clicking the **remove** button.

NOTE *You must select all primary key columns from the source table for replication.*

7. Click the **Next** button. The **Where Clause** window appears.

©Copyright 1995-2026 CASEMaker Inc.

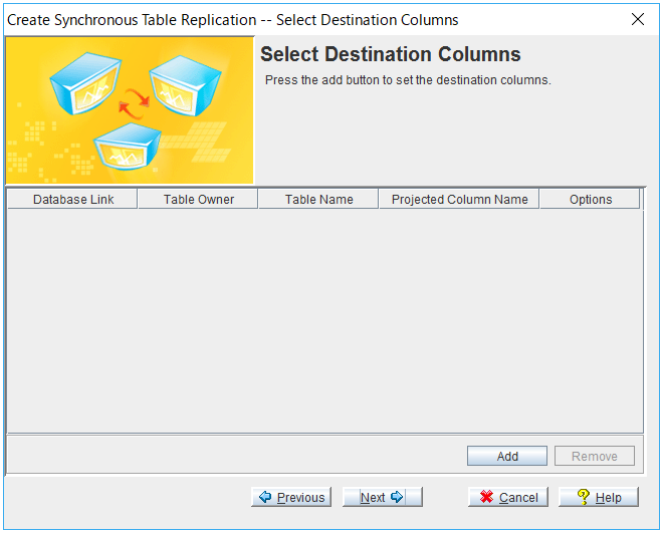


8. Add a WHERE condition clause to the SQL script, if desired. This is useful if you want to place constraints on the data that is to be replicated.

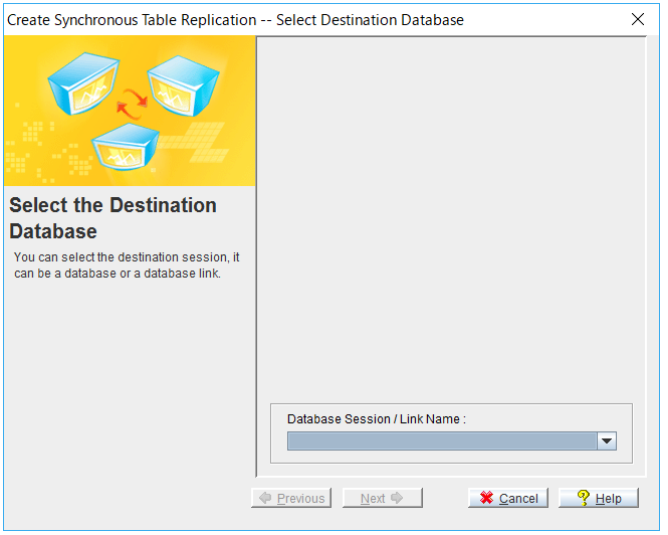
   NOTE *The WHERE clause must follow proper SQL syntax. Refer to the section on Constraint Syntax at the end of chapter 4 for more information.*

9. Click the **Next** button. The **Select Destination Columns** window will appear.

©Copyright 1995-2026 CASEMaker Inc.

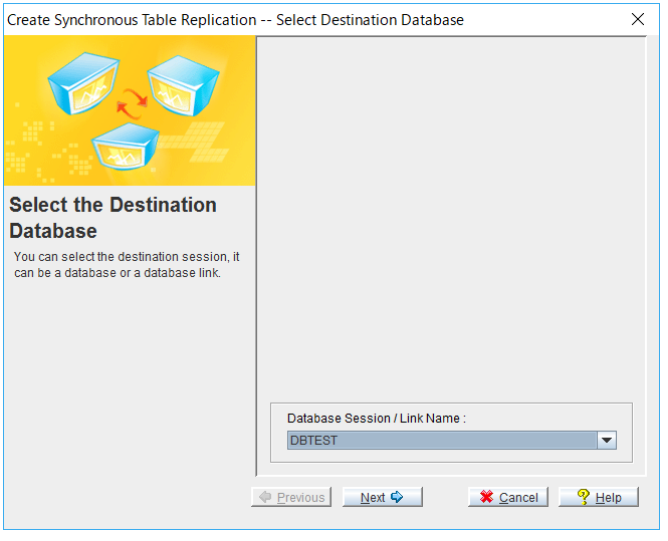


10. Add destination tables to the replication. Click the **Add** button. The **Select Destination Database** window will open.

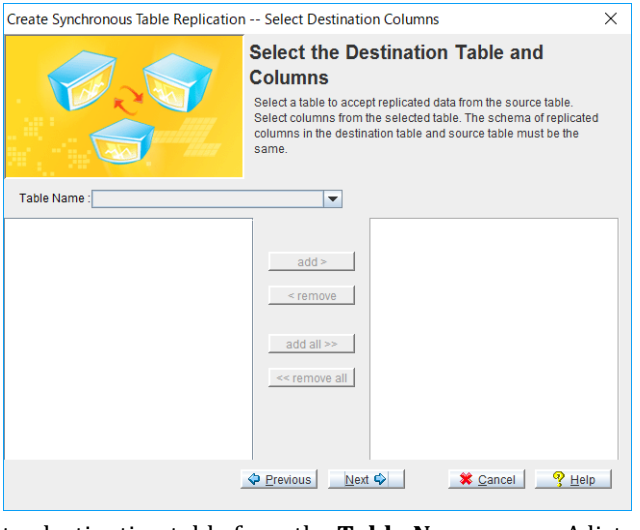


11. Enter the destination database information from the **Database Session / Link Name** menu at the bottom of the page.

©Copyright 1995-2026 CASEMaker Inc.

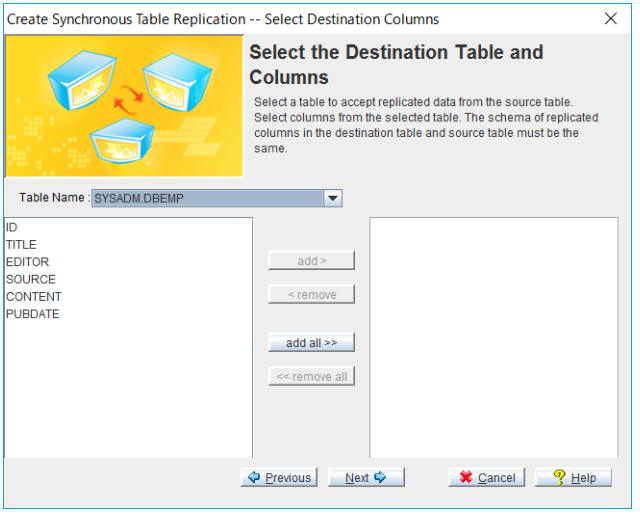


12. Click the **Next** button. The **Select the Destination Table and Columns** window appears.



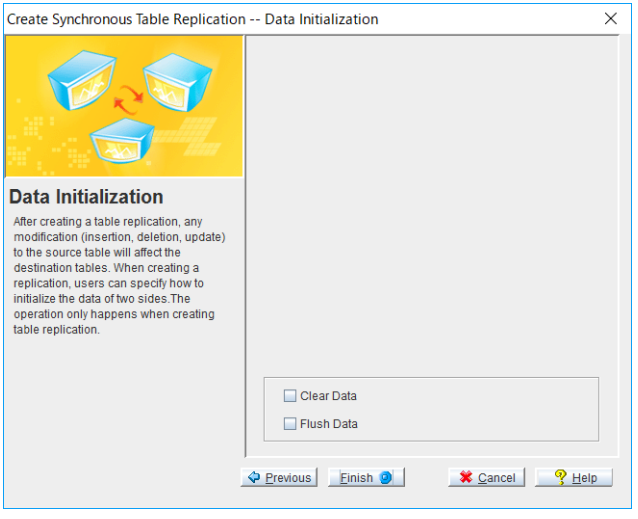
13. Select a destination table from the **Table Name** menu. A list of available columns on the selected table will appear on the left hand list box.

©Copyright 1995-2026 CASEMaker Inc.



14. Select destination columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button adds all columns. All the selected destination columns are displayed in the list box to the right. Columns can be removed from the replication by double clicking on the column name or selecting the column and clicking the **remove** button.
15. Click the **Next** button. The **Data Initialization** window appears.

©Copyright 1995-2026 CASEMaker Inc.



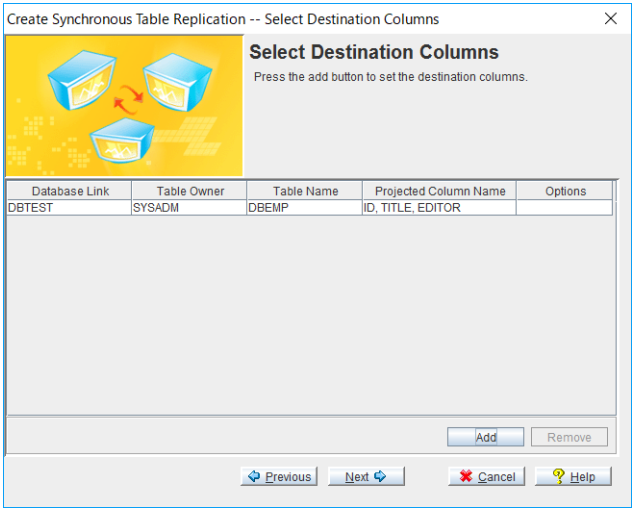
16. Select the desired options by checking the check box next to the appropriate option.

    - **Clear Data**: all data in the destination table is deleted when the replication is created.
    - **Flush Data**: copies all data that matches search criteria into the destination table.

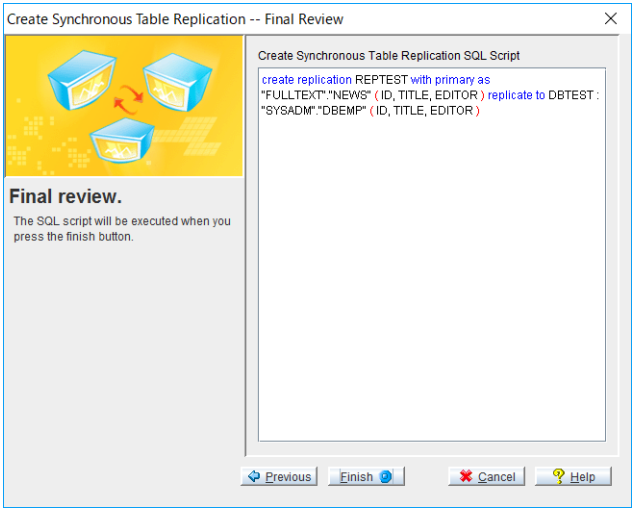
    NOTE *If you choose two options at the same time, it will clear the data of target table firstly and then copy the source table's data which matching the search condition. If the two options are not selected, the default is do nothing.*

17. Click the **Finish** button**.** The **Select Destination Columns** window will reappear.

©Copyright 1995-2026 CASEMaker Inc.

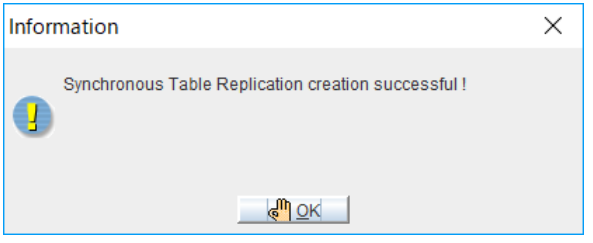


18. Click the **Add** button and repeat steps 7 through 13 to add more columns, otherwise click the **Next** button.



19. Review the SQL script. You can edit the script manually by placing the cursor in the text field or clicking on **Previous** to return to prior windows. Click the **Finish** button when you are satisfied with the SQL statement. The **Information** dialog box appears

©Copyright 1995-2026 CASEMaker Inc.



20. Click the **OK** button.

Adding Subscribers to a Synchronous Replication

After a replication is created, other subscribers may be added to the replication. Other tables in the same database or other databases can be subscribers. Subscriber destination tables must meet all requirements of synchronous table replication: the schema of source and destination columns must match, and the databases must have an active link. By default, users with RESOURCE authority or higher can add a subscriber to a synchronous replication. A user only with CONNECT authority may not add any subscriber to a synchronous replication.

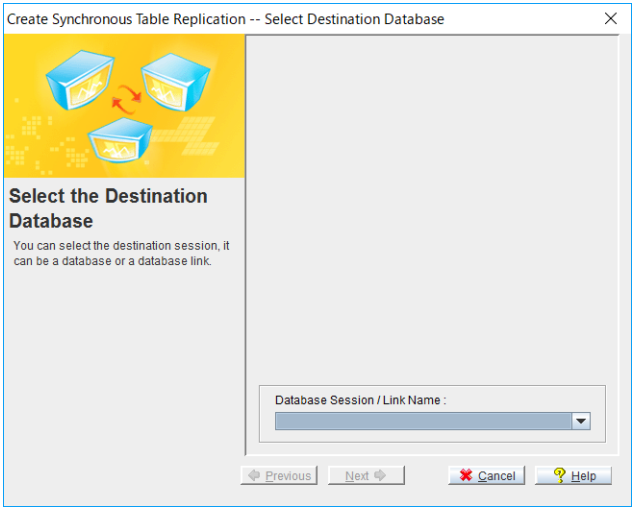
➲ Adding subscribers to a synchronous replication:

1. Select the **Replication** node from the tree. A list of replications appears on the **Replication** page to the right.
2. Select a replication from the list. The replication page will show subscriber information about the selected database in a table. Furthermore, the buttons **Add Subscriber** and **Remove Subscriber** will appear at the top of the page.

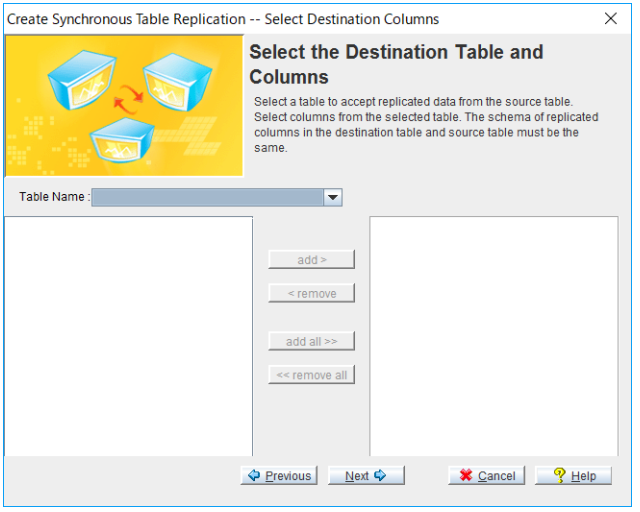
3. Click the **Add Subscriber** button. The **Select Destination Database**

©Copyright 1995-2026 CASEMaker Inc.

window will open.



4. Choose a database session from the **Database Session / Link Name** menu at the bottom of the page. Click the **Next** button.



5. The **Select Destination Columns** window will prompt you to select destination columns and a destination table.
6. Select a destination table from the **Table Name** menu. A list of available

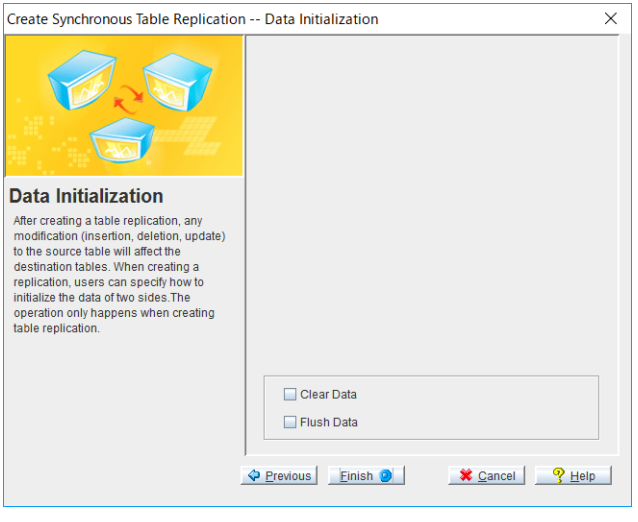
©Copyright 1995-2026 CASEMaker Inc.

columns on the selected table will appear on the left hand list box.

7. Select destination columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button adds all columns. All the selected destination columns will be displayed in the list box to the right. Columns can be likewise removed from the replication by double clicking on the column name or selecting the column and clicking the **remove** button.

NOTE *The schema of the destination columns must match the schema of the source columns.*

8. Select **Next** after the desired columns are selected.



9. The **Data Initialization** window offers you two options. Select the desired options by checking the check box next to the appropriate option.

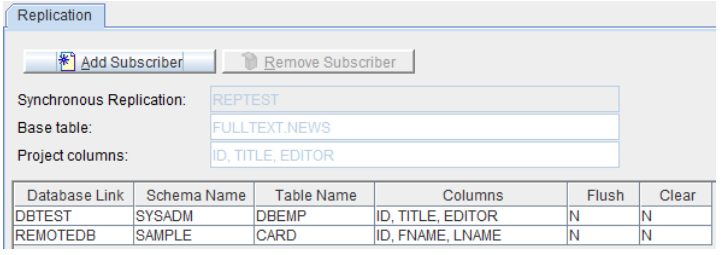
- **Clear Data**: all data in the destination table is deleted when the replication is created.
- **Flush Data**: copies all data matching the search criteria into the destination table.

NOTE *If you choose two options at the same time, it will clear the data of target table firstly and then copy the source table's data*

©Copyright 1995-2026 CASEMaker Inc.

which matching the search condition. If the two options are not selected, the default is do nothing.

10. Select **Finish**. The new replication are added to the list of subscriptions.



©Copyright 1995-2026 CASEMaker Inc.

19.3 Asynchronous Table Replication

Asynchronous means that the modification to the remote site is delayed. It is significantly slower than synchronous table replication. The time difference from source to destination database depends on a user-defined schedule. This is the replication schedule. Modifications to local tables are stored in replication logs, and are replicated to the remote table according to the predefined schedule. Using replication logs enables DBMaker to treat the local transaction and the remote transaction independently, allowing you to update local tables normally even if the remote connection is not available. This allows asynchronous table replications to tolerate network and remote database failures, since DBMaker will retry replication until no failures occur or until the user-specified number of attempts to reconnect is satisfied.

Asynchronous table replication uses ODBC function calls to communicate with remote databases. This might give poor performance in a Wide Area Network (WAN) environment. To achieve better performance in a WAN, DBMaker provides another mechanism named Express Asynchronous table replication whereby SQL Commands and related data applied to the source table are packed and sent to the remote database. Since other DBMS packages do not support this protocol, Express Asynchronous Table Replication cannot work with heterogeneous replication. It also does not support the STOP ON ERROR option.

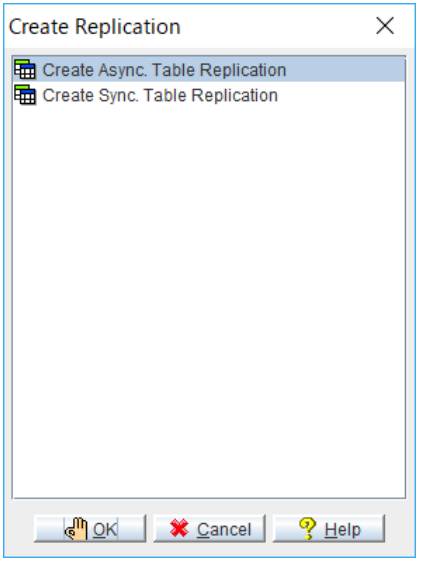
Adding Asynchronous Table Replication

A replication schedule must exist for asynchronous table replication to occur, and all associated settings for the replication schedule set. Furthermore, the schema of the destination columns must match the schema of the source columns and primary key columns in the source table must be replicated. By default, users with RESOURCE authority or higher can create an asynchronous table replication. A user only with CONNECT authority may not create any asynchronous table replication.

©Copyright 1995-2026 CASEMaker Inc.

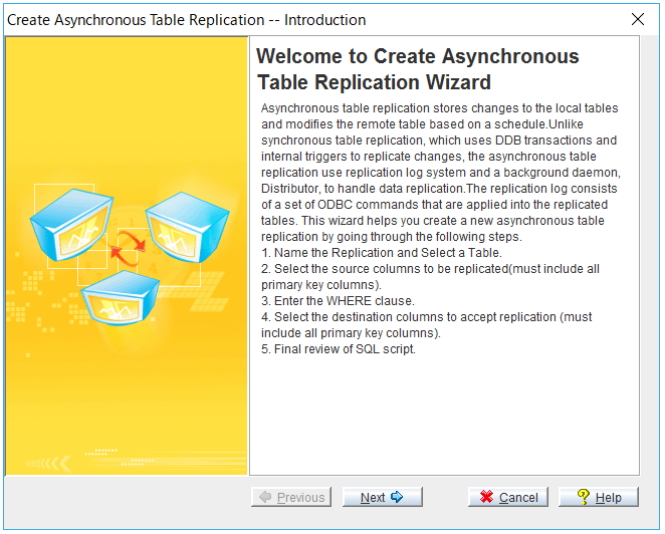
➲ Creating an Asynchronous Table Replication:

1. Select **Create** from the top of the **Replication** page. The **Create Replication** window will appear.

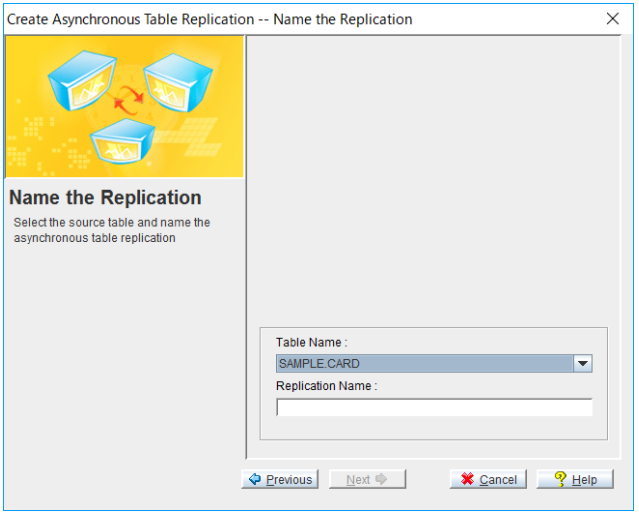


2. Select **Asynchronous Table Replication** from the menu, the **Create Asynchronous Table Replication Wizard** will begin.

©Copyright 1995-2026 CASEMaker Inc.

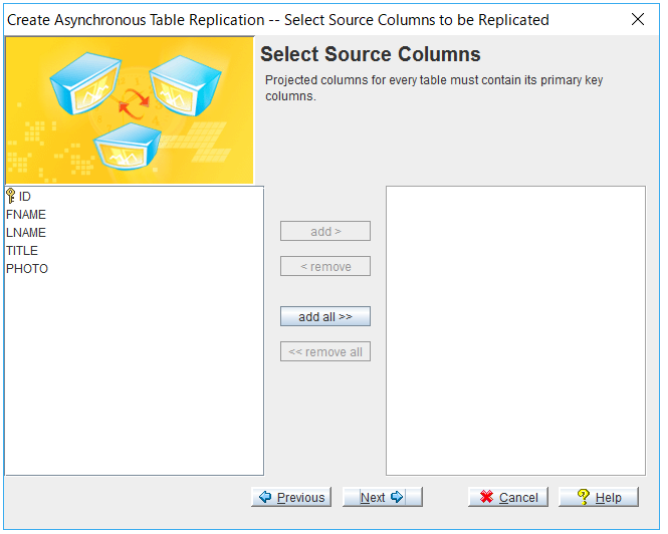


3. Select **Next**. The **Name the Replication** window appears.



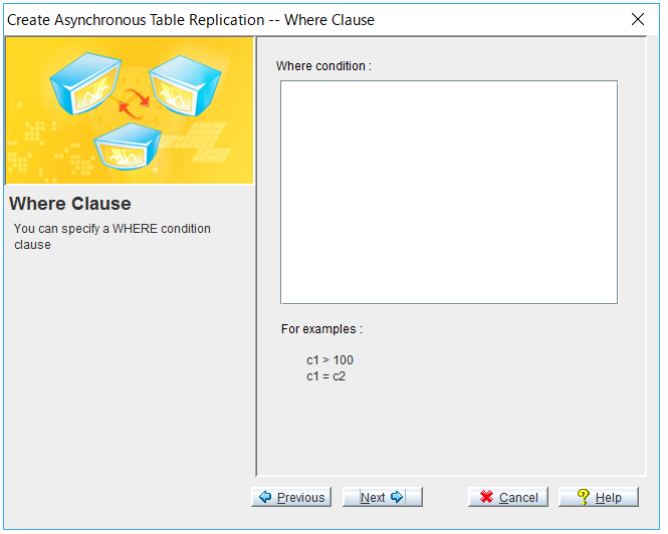
4. The **Asynchronous Table Replication Wizard** will prompt you to identify the source table to replicate and name the replication. Select the source table from the **Table Name** menu at the bottom of the window. Enter a name into the **Replication Name** field and Click the **Next** button.

©Copyright 1995-2026 CASEMaker Inc.



5. The **Select Source Columns to be Replicated** window will prompt you to select source columns from the source table. A list of available columns on the selected table appears on the left hand list box. Select source columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button can be used to add all columns to the replication. All the selected source columns will be displayed in the list box to the right. Columns can be likewise removed from the replication by double clicking on the column name or selecting the column and clicking the **remove** button.
6. Click the **Next** button after the desired columns have been selected.

©Copyright 1995-2026 CASEMaker Inc.



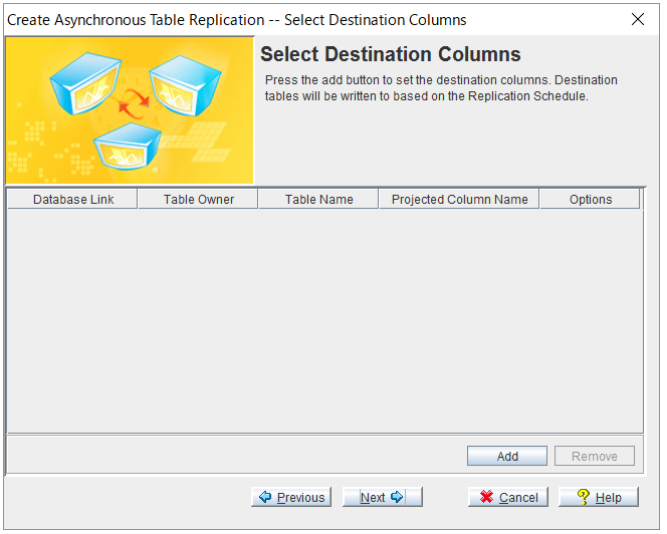
NOTE *Selection of all primary key columns, from the source table is required for replication. .*

7. The **Where Clause** window allows you to add a WHERE condition clause to the SQL script. This is useful if you want to place constraints on the data that is to be replicated. Click the **Next** button to continue.

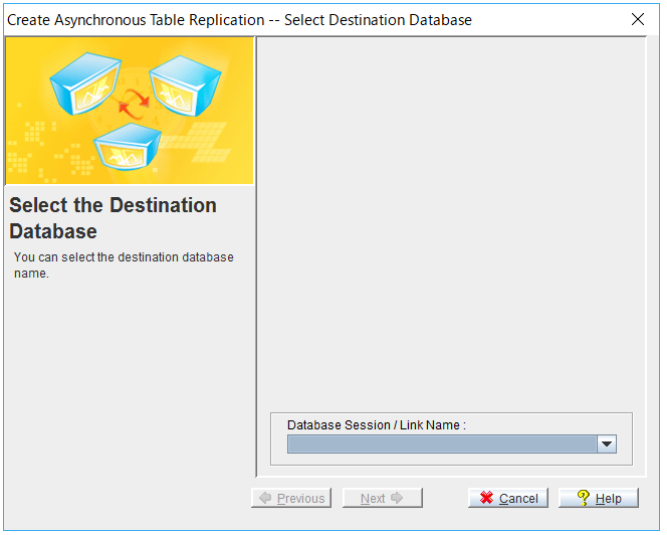
   NOTE *The WHERE clause must follow proper SQL syntax. Refer to the section on Constraint Syntax at the end of chapter 4 for more information.*

8. The **Asynchronous Table Replication Wizard** prompts to add destination columns to the replication.

©Copyright 1995-2026 CASEMaker Inc.



9. Click the **Add** button at the bottom of the **Select Destination Columns** window. The **Select Destination Database** window appears.

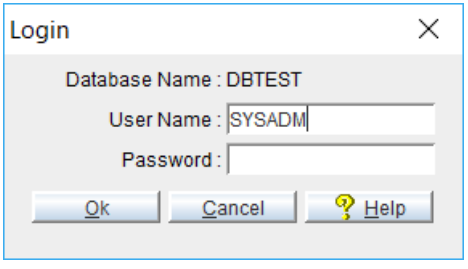


10. Enter the destination database information manually or from the **Database Session / Link Name** menu at the bottom of the page. To enter the information manually, select the **Enter Data Manually** check

©Copyright 1995-2026 CASEMaker Inc.

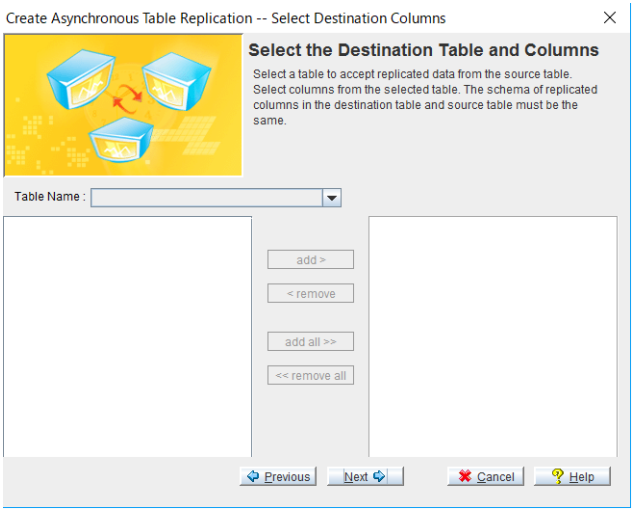
box and provide the required information in the fields below.

11. Click the **Next** button.



12. JDBA will prompt you to log on to the destination database. Enter a username and password for a user with DBA authority or higher.

13. Click the **OK** button, the **Select Destination Columns** window appears.



14. The **Select Destination Columns** window will prompt you to select destination columns and a destination table.

15. Select a destination table from the **Table Name** menu. A list of available columns on the selected table will appear on the left hand list box.

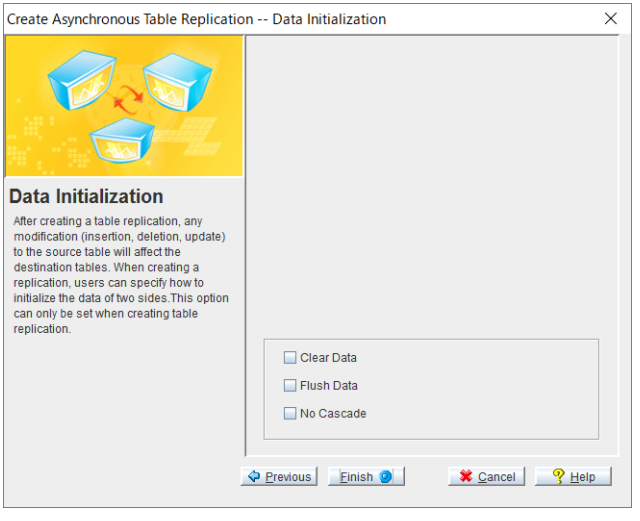
16. Select destination columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button adds all columns. All the selected destination columns are displayed in

©Copyright 1995-2026 CASEMaker Inc.

the list box to the right. Columns can be likewise removed from the replication by double-clicking on the column name or selecting the column and clicking the **remove** button.

NOTE *The schema of the destination columns must match the schema of the source columns.*

17. Click the **Next** button after the desired columns are selected.



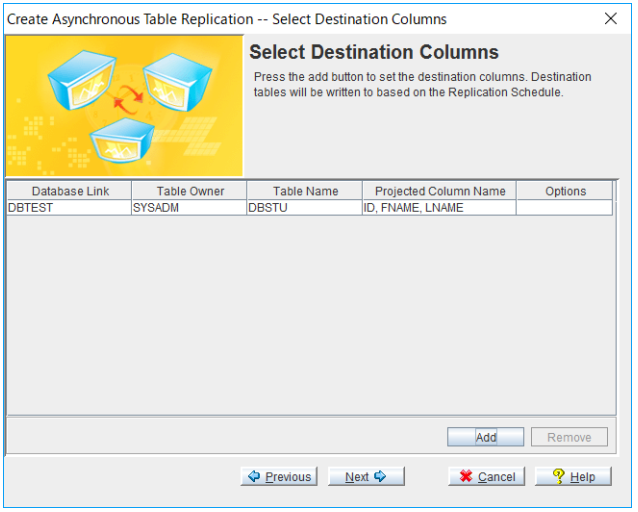
18. The **Data Initialization** window offers you three options. Select the desired options by checking the check box next to the appropriate option.

- **Clear Data**: all data in the destination table is deleted when the replication is created.
- **Flush Data**: copies all data that matches search criteria into the destination table.
- **No Cascade**: prevents replicated data from being replicated from the destination tables. If this option is not selected, data can be replicated from the source database to the destination databases down to other databases that are subscribers to the destination databases.

©Copyright 1995-2026 CASEMaker Inc.

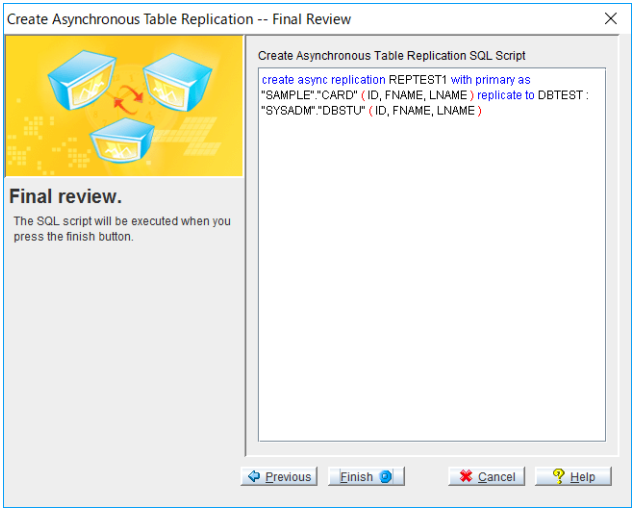
NOTE *If you choose **Clear Data** and **Flush Data** options at the same time, it will clear the data of target table firstly and then copy the source table's data which matching the search condition. If the two options are not selected, the default is do nothing.*

19. Click the **Finish** button. The **Select Destination Columns** window reappears.

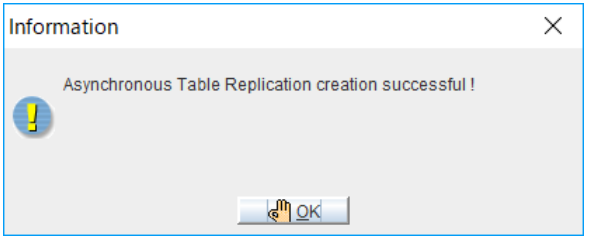


20. Click the **Add** button and repeat steps 7 through 13 to add more columns, otherwise click the **Next** button.

©Copyright 1995-2026 CASEMaker Inc.



21. Review the SQL script. You can edit the script manually by placing the cursor in the text field or click the **Previous** button to return to prior windows. Click the **Finish** button when you are satisfied with the SQL statement. The **Information** dialog box appears.



22. Click the **OK** button.

Adding Subscribers to an Asynchronous Replication

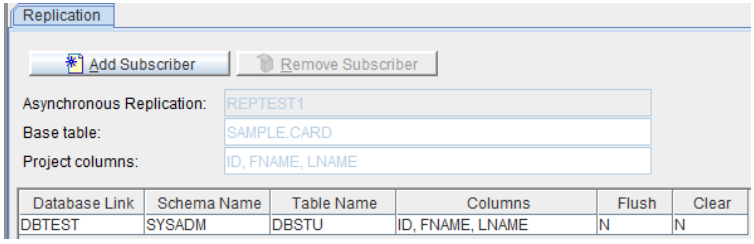
After creating a replication, users can be subscribed. Other tables in the same database or other databases can be subscribers. Subscriber destination tables must meet all requirements of asynchronous table replication: the schema of source and destination columns must match, and a replication schedule must exist. By default, users with RESOURCE authority or higher can add a subscriber to an asynchronous replication. A user only with CONNECT

©Copyright 1995-2026 CASEMaker Inc.

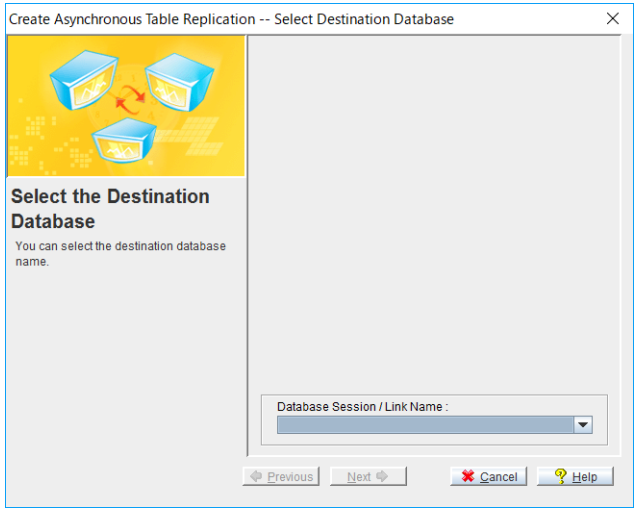
authority may not add any subscriber to an asynchronous replication.

➲ Adding subscriber tables to an asynchronous replication:

1. Select the **Replication** node from the tree. A list of replications appears on the **Replication** page to the right.
2. Select a replication from the list. The replication page will show subscriber information about the selected database in a table. Furthermore, the buttons **Add Subscriber** and **Remove Subscriber** appears at the top of the page.

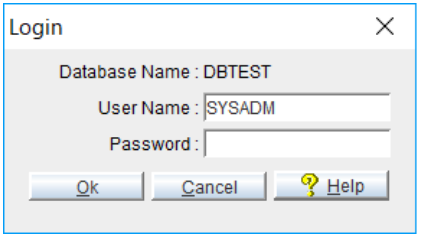


3. Click the **Add Subscriber** button. The **Select Destination Database** wizard appears.

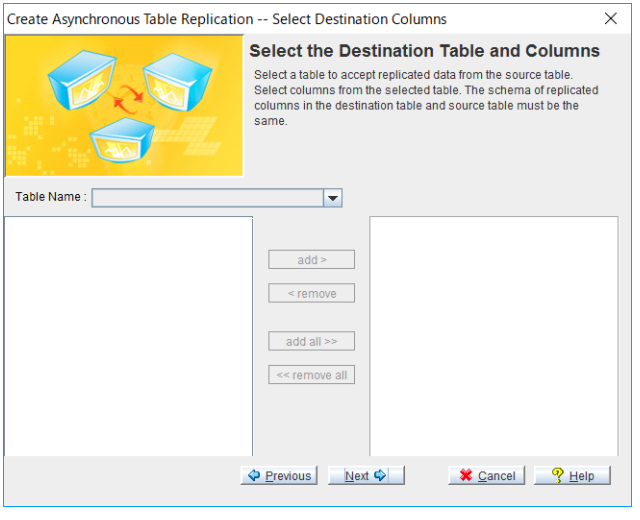


4. Select the destination database from the **Database Session / Link Name** menu. Click the **Next** button.

©Copyright 1995-2026 CASEMaker Inc.



5. JDBA will prompt you to log on to the destination database. Enter a User Name and Password for a user with DBA authority or higher. Click the **OK** button, the **Select Destination Columns** window appears.



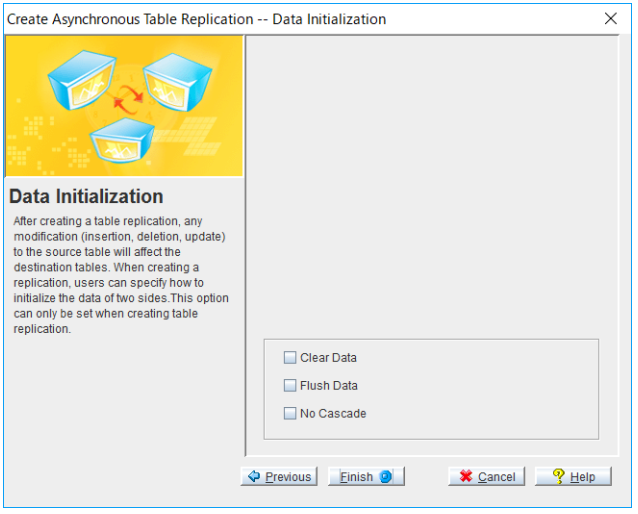
6. The **Select Destination Columns** window prompts for destination columns and a destination table selection.
7. Select a destination table from the **Table Name** menu. A list of available columns on the selected table appears on the left hand list box.
8. Select destination columns by double clicking on the column name or selecting the column and clicking the **add** button. The **add all** button adds all columns. All the selected destination columns are displayed in the list box to the right. Columns can be likewise removed from the replication by double clicking on the column name or selecting the column and clicking the **remove** button.

NOTE *The schema of the destination columns must match the schema*

©Copyright 1995-2026 CASEMaker Inc.

of the source columns.

9. Click the **Next** button after selecting the desired columns.



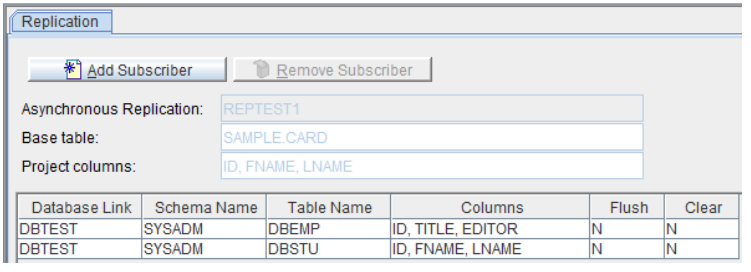
10. The **Data Initialization** window offers you two options. Select the desired options by checking the check box next to the appropriate option.

- **Clear Data**: all data is deleted from the destination table when the replication is created.
- **Flush Data**: copies all data that matches search criteria into the destination table.
- **No Cascade**: prevents replicated data from being replicated from the destination tables. If this option is not selected, data can be replicated from the source database to the destination databases down to other databases that are subscribers to the destination databases.

NOTE *If you choose **Clear Data** and **Flush Data** options at the same time, it will clear the data of target table firstly and then copy the source table's data which matching the search condition. If the two options are not selected, the default is do nothing.*

©Copyright 1995-2026 CASEMaker Inc.

11. Select **Finish**. The new database link will appear in the table on the **Replication** page.



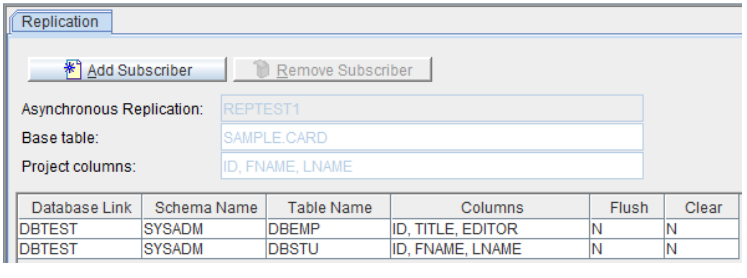
©Copyright 1995-2026 CASEMaker Inc.

19.4 Dropping a Subscriber

Subscribers may be dropped when the data no longer needs to be replicated. Subscribers should be dropped whenever the subscriber's destination columns or databases are deleted. By default, users with RESOURCE authority or higher can drop a subscriber. A user only with CONNECT authority may not drop any subscriber.

➲ Dropping a subscriber:

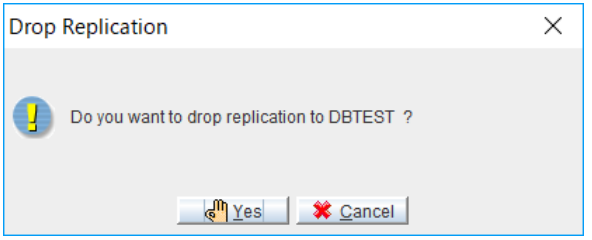
1. Select the **Replication** node from the tree. The tree will expand, showing a list of replications below the **Replication** node on the tree.
2. Select a replication from the tree. The replication page will show subscriber information about the selected database in a table. Furthermore, the buttons **Add Subscriber** and **Remove Subscriber** will appear at the top of the page.



3. Select a database link to drop a subscription (it will highlight blue).

©Copyright 1995-2026 CASEMaker Inc.

4. Click the **Remove Subscriber** button. The **Information** dialog box will appear asking conformation.



5. Click the **Yes** button.

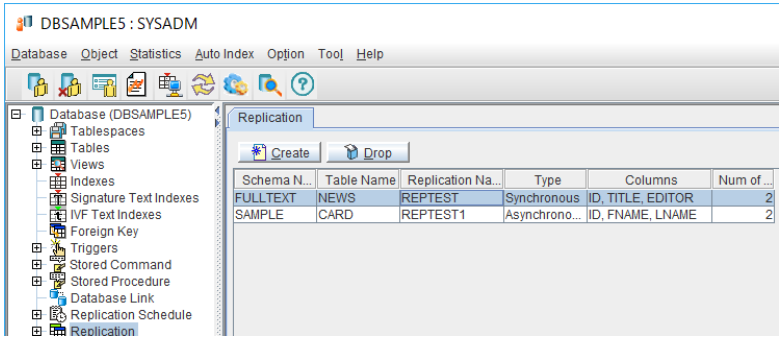
©Copyright 1995-2026 CASEMaker Inc.

19.5 Dropping a Replication

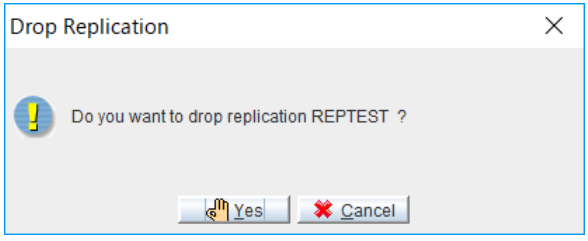
Consider dropping a replication if there are no subscribers and no further need to replicate that data. By default, users with RESOURCE authority or higher can drop a replication. A user only with CONNECT authority may not drop any replication.

➲ Dropping a replication:

1. Select the **Replication** node from the tree. A list of replications appears on the **Replication** page to the right.
2. Select a replication from the list (it will highlight blue). The **Drop** button at the top of the page will activate.



3. Click the **Drop** button. A conformation dialog box appears.



4. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

20 Working with Domains

A domain is a type of integrity constraint you can use when defining a column. Domains specify the data type of the column. Domains may also specify a default value and a value constraint. When defining a column using a domain, the column inherits the properties of the domain (e.g., data type, default value, and value constraint) without requiring explicit specification.

Specifying the default value and value constraint using domains achieves the same results as specifying them in a standard column definition. If you specify a default value for a column, it will override the default value specified in a domain. Value constraints specified in the column definition, are used in addition to the value constraints specified in the domain. If you do define a column using a domain and specify additional value constraints, you should ensure the additional value constraints do not conflict with those defined in the domain.

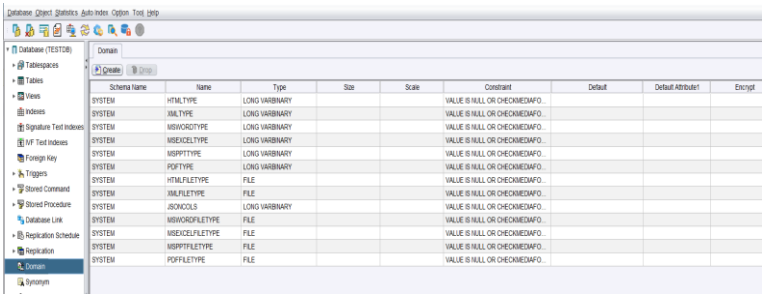
©Copyright 1995-2026 CASEMaker Inc.

20.1 Creatng a Domain

Defining a domain name, an optional default value, and an optional constraint creates a domain. Any data type supported by DBMaker, except the SERIAL type, can be used in domains. By default, users with RESOURCE authority or higher have CREATE privilege for the domain. A user only with CONNECT authority may not create any domain.

➲ Creating a domain:

1. Select the object **Domain** from the tree. The following window appears.

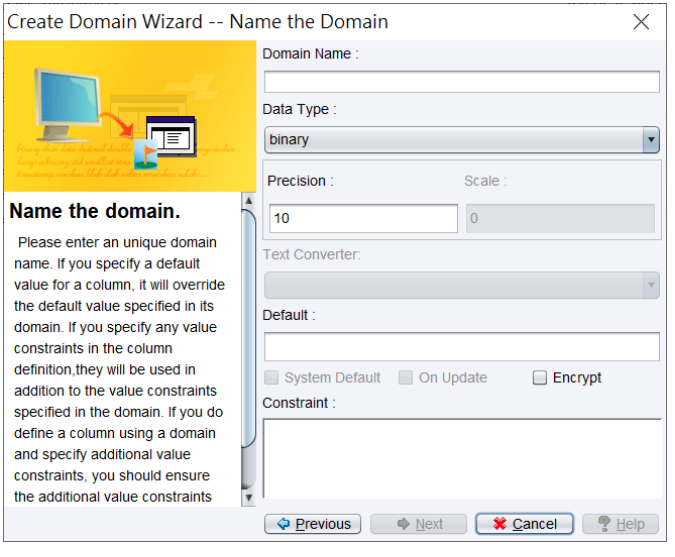


NOTE: The **Encrypt** column only appears in an encrypted database.

2. Click the **Create** button. The **Create Domain Wizard** appears.

©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button.

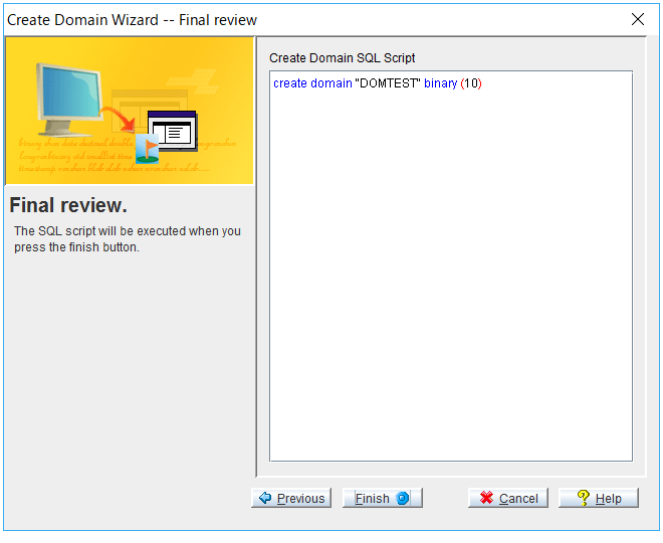


4. Enter the **Domain Name**.
5. Select a data type from the **Data Type** menu.
6. Enter **Precision** and **Scale** information corresponding to the selected data type.
7. Enter a **Default** value. If user wants the **default** value is a **system default** value, select the **System Default** option. If user wants the

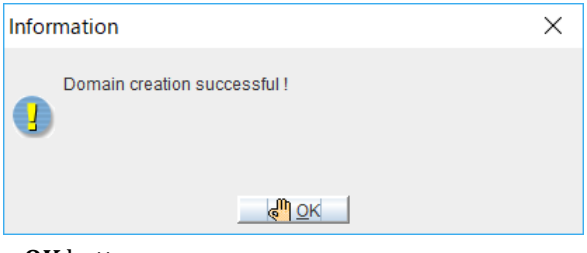
©Copyright 1995-2026 CASEMaker Inc.

default value will be **on update**, select the **On Update** option. If user wants the domain has an **encrypt** keyword, select the **Encrypt** option.

8. Enter the **Constraint** of the column. Constraints must comply with SQL syntax rules. Please see *Constraint Syntax* at the end of Chapter 4 for additional information.
9. Click the **Next** button.



10. Check the SQL script and make changes if required.
11. Click the **Finish** button. The **Information** dialog box appears.



12. Click the **OK** button.

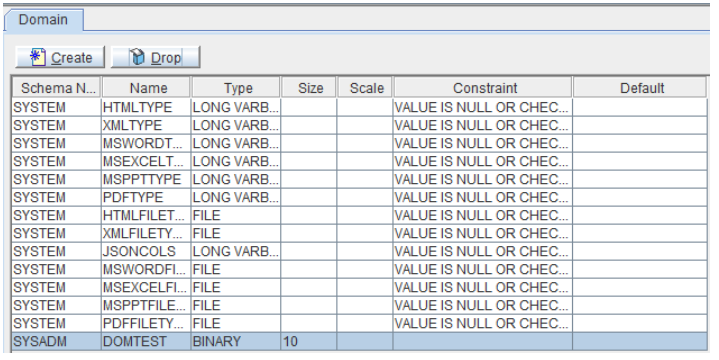
©Copyright 1995-2026 CASEMaker Inc.

20.2 Dropping a Domain

Consider dropping domains that are no longer required. By default, owners of a domain and users with RESOURCE authority or higher have DROP privilege for the domain. A user only with CONNECT authority may not drop any domain.

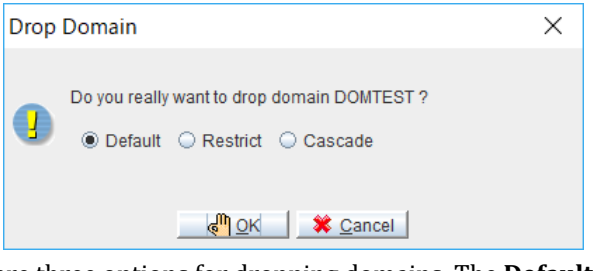
➲ Dropping a Domain:

1. Select the object **Domain** from the tree. The list of all the Domains in the database appears.
2. Select a domain.



Schema N...	Name	Type	Size	Scale	Constraint	Default
SYSTEM	HTMLTYPE	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	XMLTYPE	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	MSWORDT...	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	MSEXCELT...	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	MSPPTTYPE	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	PDFTYPE	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	HTMLFILET...	FILE			VALUE IS NULL OR CHEC...	
SYSTEM	XMLFILETY...	FILE			VALUE IS NULL OR CHEC...	
SYSTEM	JSONCOLS	LONG VARB...			VALUE IS NULL OR CHEC...	
SYSTEM	MSWORDFI...	FILE			VALUE IS NULL OR CHEC...	
SYSTEM	MSEXCELFI...	FILE			VALUE IS NULL OR CHEC...	
SYSTEM	MSPPTFILE...	FILE			VALUE IS NULL OR CHEC...	
SYSTEM	PDFFILETY...	FILE			VALUE IS NULL OR CHEC...	
SYSADM	DOMTEST	BINARY	10			

3. Click the **Drop** button. A confirmation window appears.



4. There are three options for dropping domains. The **Default** option deletes the domain without considering dependent object within the domain. Dropping a domain renders dependent objects become invalid. The **Restrict** option considers table, domain, and procedure

©Copyright 1995-2026 CASEMaker Inc.

dependencies. This option does not permit a domain drop when one or more objects depend on the domain. The **Cascade** option drops the domain and all of its dependent objects. Click a radio button to select a domain dropping option.

5. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

21 Working with Synonyms

A *synonym* is an alias for any table or view. Since a synonym is simply an alias, it requires no storage other than its definition in the system catalog.

Synonyms are useful for simplifying a fully qualified table or view name. DBMaker normally identifies tables and views with fully qualified names that are composites of the owner and object names. By using a synonym, anyone can access a table or view through the corresponding synonym without having to make use of the fully qualified name. Since a synonym has no owner name, all synonyms in the database must be unique for DBMaker to identify them.

©Copyright 1995-2026 CASEMaker Inc.

21.1 Creating a Synonym

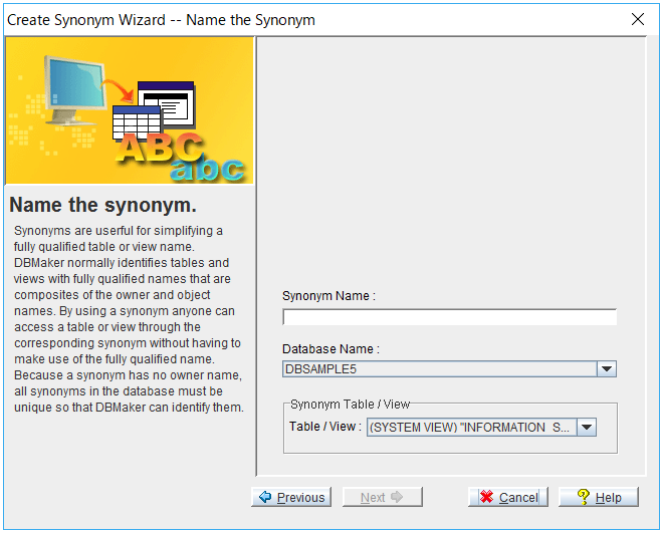
You can create a synonym for a table or view. By default, users with RESOURCE authority or higher have CREATE privilege for the synonym. A user only with CONNECT authority may not create any synonym.

➲ Creating a synonym for a table or view:

1. Select the object **Synonym** from the tree and click the **Create** button.
2. The **Create Synonym Wizard - Introduction** window appears.
3. Click the **Next** button. The **Name the Synonym** window appears.

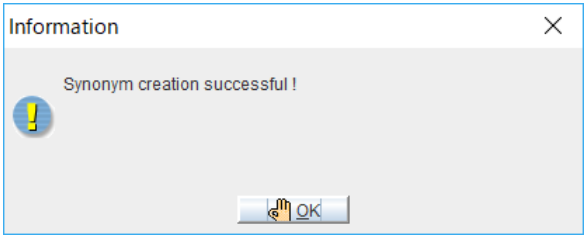
4. Enter the **Synonym Name**.
5. Select the **Table/View** name from the menu.
6. Click the **Next** button. The **Final Review** window appears.

©Copyright 1995-2026 CASEMaker Inc.



7. Check the SQL script and make changes if required. Click the **Finish** button.

8. The **Information** dialog box appears.



9. Click the **OK** button.

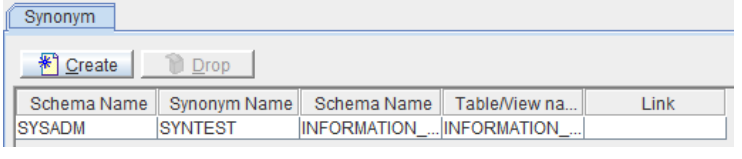
©Copyright 1995-2026 CASEMaker Inc.

21.2 Dropping a Synonym

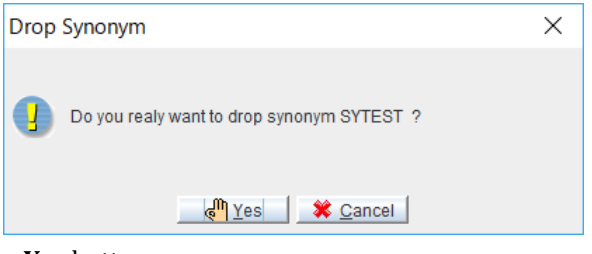
Consider dropping synonyms that are no longer required. By default, owners of a synonym and users with RESOURCE authority or higher have DROP privilege for the synonym. A user only with CONNECT authority may not drop any synonym.

➲ Dropping a synonym:

1. Select the object **Synonym** from the tree and select the Synonym from the list.



2. Click the **Drop** button. The **Drop Synonym** window will open to confirm if you want to drop the window.



3. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.

22 Managing Users

DBMaker supports multiple users and allows database privileges to be granted or revoked to those users. JDBA tool provides an easy to use interface for creating new users, changing user passwords, and adding or revoking user privileges. User names have a maximum length of 128 characters, passwords have a maximum length of sixteen characters, and both may contain letters, numbers, the underscore character, and the symbols $ and #. The first character may not be a number. You must log onto the database as SYSADM or SYSDBA to be able to create a new user or change database authority for any user.

Object privileges include the SELECT, INSERT, DELETE, UPDATE, INDEX, ALTER, and REFERENCE for tables and views, and the EXECUTE privilege for stored commands and stored procedures. Object privileges may be granted or revoked by the user that created the object, or by a user with DBA authority or higher.

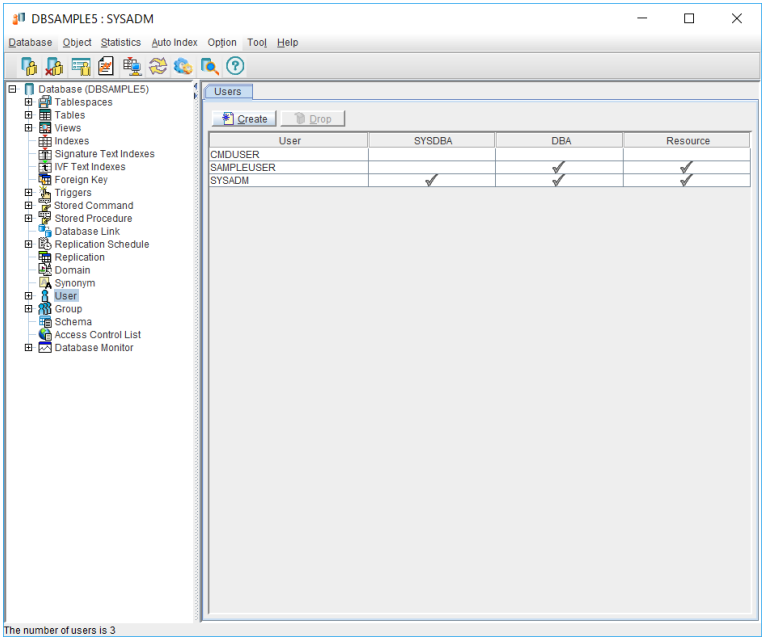
©Copyright 1995-2022 CASEMaker Inc.

22.1 Creating a New User Account

Create new accounts when specific access profiles are required for an individual. It is also useful to create user accounts for all individuals accessing the database to facilitate user management. User names have a maximum length of 128 characters. Passwords have a maximum length of sixteen characters. Both user names and password may contain letters, numbers, the underscore character, and the symbols $ and #. The first character may not be a number. A user only with SYSADM or SYSDBA authority have CREATE privilege for the user account.

➲ Creating a new user account:

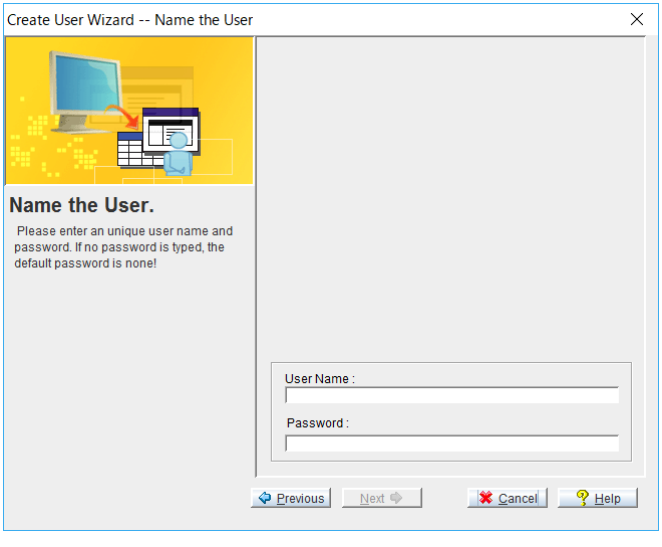
1. Open the **User** node on the object tree. The **Users** window appears.



2. Click the **Create** button from the top of the **Users** page. The **Create User Wizard** starts.

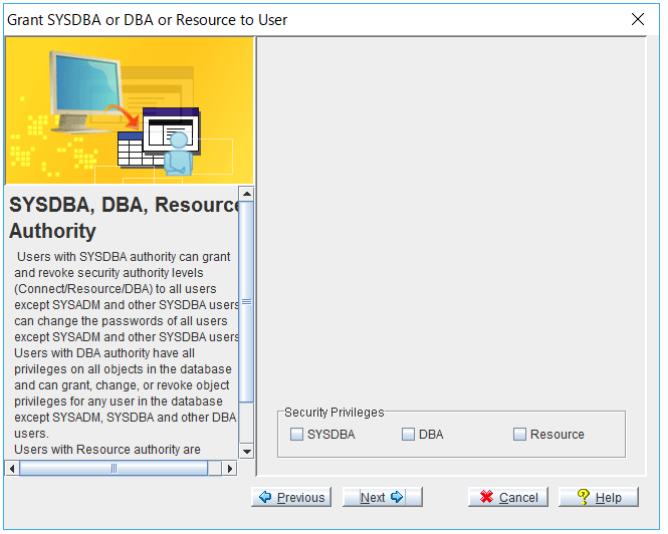
©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Name the User** window appears.

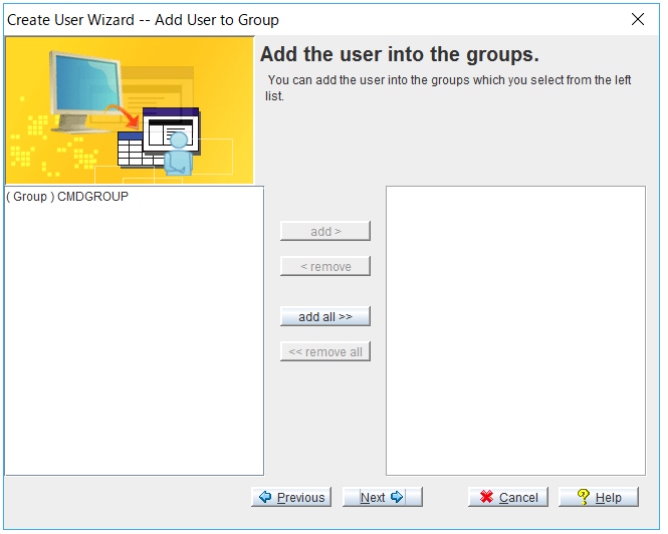


4. Enter a name into the **User Name** field, and a password in the **Password** field.
5. Click the **Next** button. The **Grant SYSDBA or DBA or Resource to User** window appears.

©Copyright 1995-2022 CASEMaker Inc.



6. Select the type of database authority the new user will have. Connect authority is granted to the new user if no authority level is selected. Click the **Next** button, the following window appears.

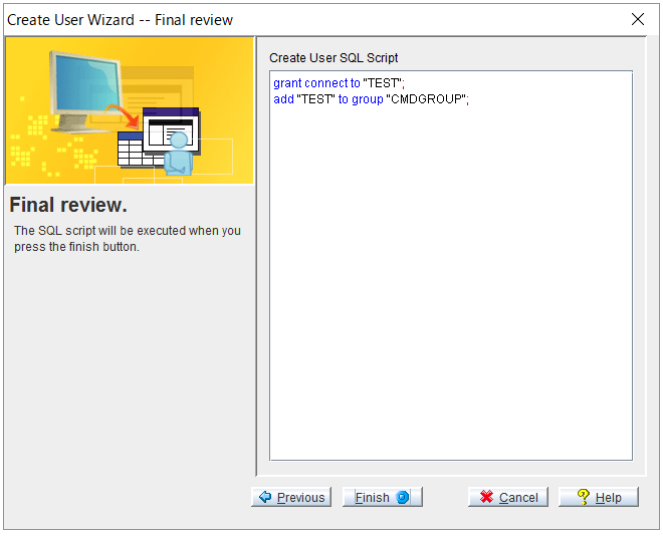


7. A list of available groups appears on the left list box. Select a group by double clicking on the group name or selecting the group(s) and

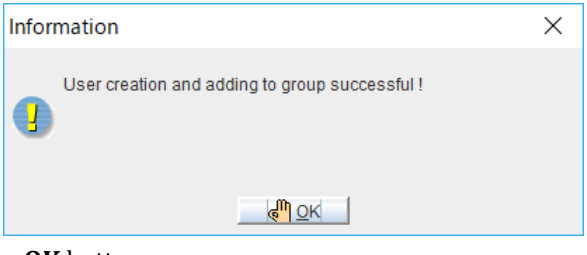
©Copyright 1995-2026 CASEMaker Inc.

clicking the **add** button. The **add all** button can be used to select all groups to add the user into. All the selected groups are displayed in the list box to the right. Groups can be, likewise removed by double clicking on the group name, or by selecting the name and clicking the **remove** button.

8. Click the **Next** button, the **Final review** page appears.



9. Review the new SQL script. You can add to the SQL script in the **Create User SQL Script** field if necessary by clicking on the field. Click the **Finish** button to execute the SQL command. The **Information** dialog box will appear.



10. Click the **OK** button.

©Copyright 1995-2022 CASEMaker Inc.

22.2 Changing Security Level

There are two security mechanisms in DBMaker. The first is user authority level and the second is object privilege. Authority determines the level of access and control the user has to grant or revoke object privileges, as well as the ability to create new objects. Object privilege determines the level of access and control an individual user has over database objects, such as tables, views, stored procedures, and stored commands.

Setting Authority Level

There are five levels of authority: CONNECT, RESOURCE, DBA, SYSDBA and SYSADM.

Users who exist but are not granted RESOURCE, DBA or SYSDBA authority will have CONNECT authority by default. A user with CONNECT authority may perform queries on any data to which they have explicitly been granted permission, but may not create objects.

A user that is created and granted RESOURCE authority can create, alter, and drop tables, domains, and indexes. As the owner of any objects they create, users with RESOURCE authority may grant and revoke object privileges to other users and create synonyms and views for any objects they own.

DBA authority has all same capabilities as RESOURCE authority, but may additionally create tablespaces and files. Users with DBA authority can also grant or revoke object privileges for schema objects owned by other users (except for system schema objects). A user only with DBA authority or higher can change the user's database authority level.

Users with SYSDBA authority can both grant CONNECT, RESOURCE and DBA to other users or revoke those privileges from other users. In addition, they also can change other users' passwords, except users with SYSADM or SYSDBA authority. Users with SYSDBA have all the privileges of DBA authority and only

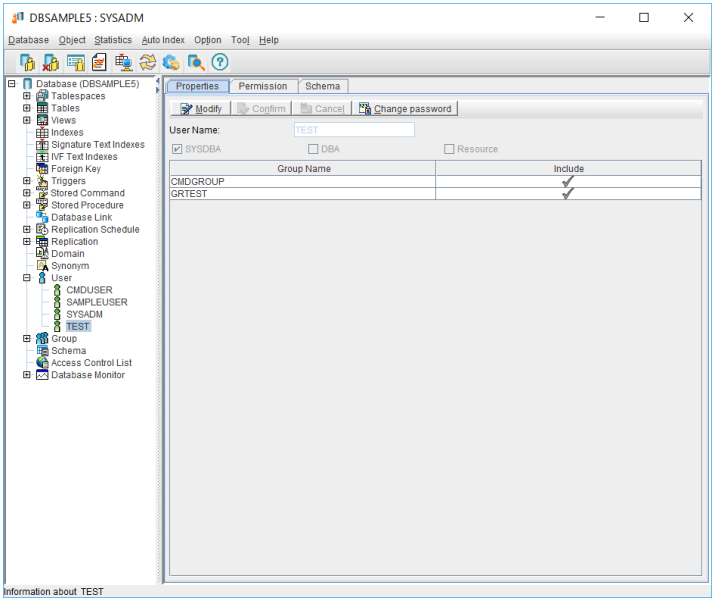
©Copyright 1995-2026 CASEMaker Inc.

users with SYSADM authority can grant or revoke SYSDBA from users. For users with SYSDBA authority, if the SYSADM revokes SYSDBA, the users still has DBA authority, but if the SYSADM revokes DBA, the users has neither SYSDBA nor DBA authority.

The SYSADM is the most powerful authority level in DBMaker. There can be only one SYSADM for every database. A user with SYSADM authority can grant SYSDBA, DBA, RESOURCE or CONNECT authority to other users, and has all privileges of the SYSDBA and DBA authority level on the database.

➲ Setting a user's authority level:

1. Open the **User** node on the object tree. The **Users** page appears.
2. The users table displays the current authority level of the user. Double-click the user name in the tree or on the users table. The **Properties** page appears.



3. Select **Modify** to change the user's database authority.
4. A check mark next to the type of database authority indicates the user has that privilege. Clicking the check box toggles the authority of the

©Copyright 1995-2022 CASEMaker Inc.

user. Click the **Confirm** button to save the changes, or **Cancel** to discard them.

Granting and Removing Object Privileges

Object privileges control which database objects a user can access and the actions they can perform. There are eight object privileges: SELECT, INSERT, DELETE, UPDATE, INDEX, ALTER, REFERENCE and EXECUTE.

SELECT privilege allows you to select the data in a database object. This privilege applies to the entire object. This cannot be granted on specific columns.

INSERT privilege allows you to insert new data into a database object. This privilege can also be restricted to specific columns.

DELETE privilege allows you to delete data from a database object. This privilege applies to the entire object. It cannot be granted on specific columns.

UPDATE privilege allows you to update data in a database object. This privilege can also be restricted to specific columns.

INDEX privilege allows you to create an index on a database object. This privilege applies to the entire object and cannot be granted on specific columns.

ALTER privilege allows you to alter the schema of a database object. This privilege applies to the entire object and cannot be granted on specific columns.

REFERENCE privilege allows you to create referential constraints (such as foreign keys) on a database object. This privilege can also be restricted to specific columns.

EXECUTE privilege allows the user to use a stored procedure or command.

The user who creates a schema object is the owner of that object. The owner and any user with DBA authority or higher automatically have all object privileges on that object. System catalog tables belong to a special virtual user

©Copyright 1995-2026 CASEMaker Inc.

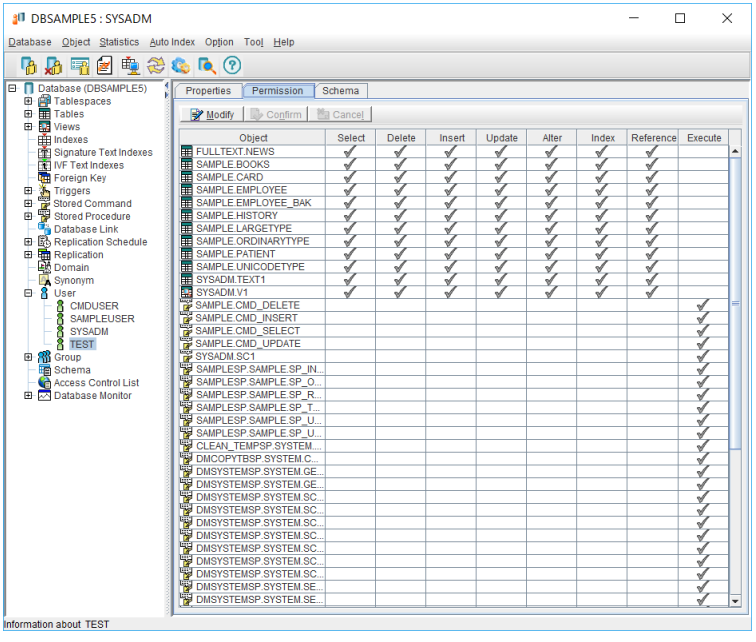
called SYSTEM. All users including the SYSADM have only SELECT privilege on system catalog tables. You cannot grant additional object privileges on the system catalog tables.

You cannot grant privileges on specific columns and on the entire database object at the same time. If you wish to do so you must use the command twice, once to grant privileges on specific columns, and once to grant privileges on the entire table. It is possible to grant object privileges to all users simultaneously by granting the privileges to PUBLIC. All current and future users will then have those privileges on the database object.

➲ Granting and removing object privileges:

1. Open the **User** node on the object tree. The **Users** page appears.
2. Double-click the user name in the tree or on the users table. The **Properties** page appears.
3. Select the **Permission** tab at the top of the page. The **Permission** page appears.

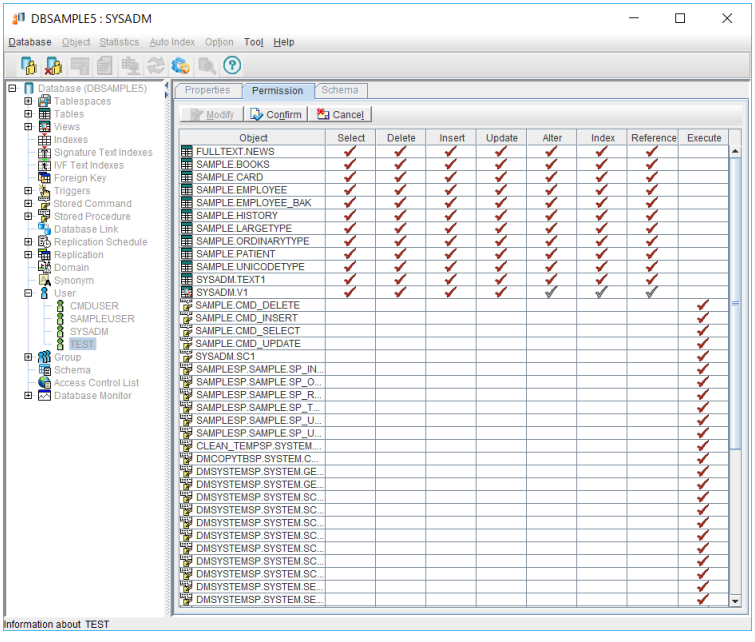
©Copyright 1995-2022 CASEMaker Inc.



NOTE *Users with DBA authority or higher have permission on all objects in the database.*

4. Click the **Modify** button to alter the user's object privilege. Choose which type of privilege to grant to the user on each object. Click the column and row corresponding to the object privilege to be granted and the object to grant the privilege to. Privileges can be likewise removed. A check mark will appear for each privilege selected as in the table below.

©Copyright 1995-2026 CASEMaker Inc.



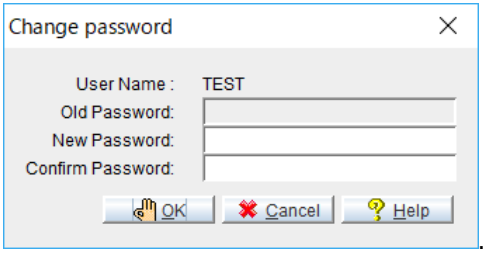
5. Click the **Confirm** button to save the changes, or **Cancel** to discard them.

22.3 Changing a User's Password

Passwords may be changed for a user if they have forgotten their old password or are requesting a new one. Passwords have a maximum length of sixteen characters and may contain letters, numbers, the underscore character, and the symbols "$" and "#". The first character may not be a number.

➲ Changing a user's password:

1. Open the **User** node on the object tree. The **Users** page appears.
2. Double click the user name in the tree or on the users table. The **Properties** page appears.
3. Select the **Change password** button to open the **Change Password** dialog box. Enter a new password and confirm it by reentering it in the **Confirm Password** field.



4. Click the **OK** button to change the password.

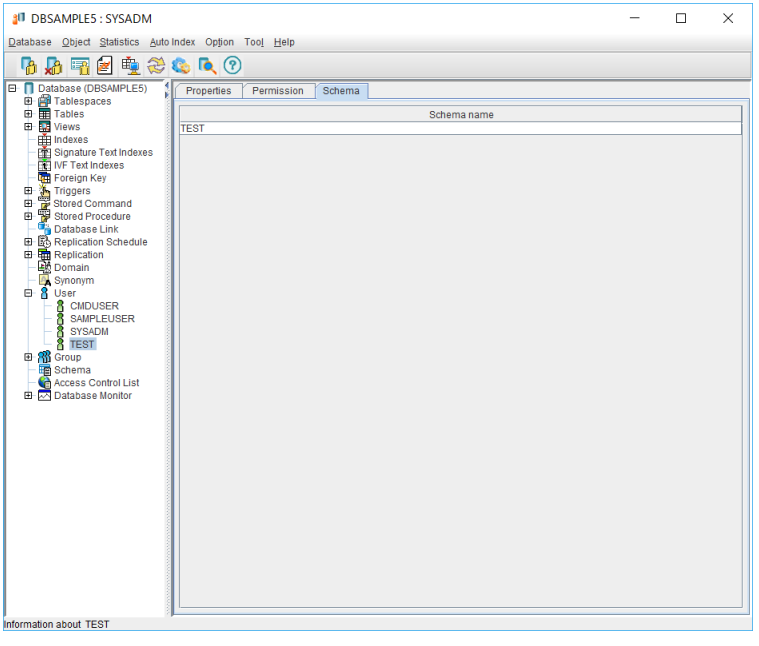
©Copyright 1995-2026 CASEMaker Inc.

22.4 Browsing Assigned Schema

You can quickly browse the schemas that are assigned to each user in the Schema page.

➲ Browsing schema assigned to users:

1. Open the **User** node on the object tree. The **Users** page appears.
2. Double click the user name in the tree or on the users table. The **Properties** page appears.
3. Select the **Schema** tab at the top of the page. The **Schema** page appears.



4. All of the schemas in the database that are assigned to the user are displayed on this page.

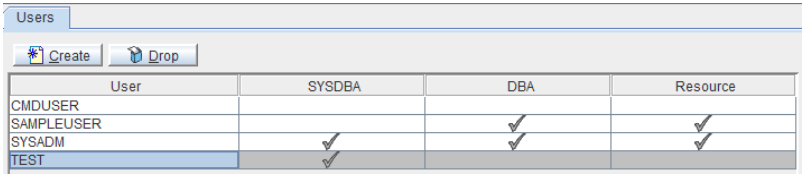
©Copyright 1995-2022 CASEMaker Inc.

22.5 Deleting a User Account

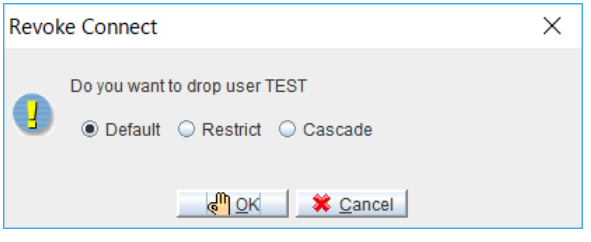
User accounts should be deleted when the account is no longer active. A user only with SYSADM or SYSDBA authority can drop the user account.

➲ Removing a user account from the database:

1. Open the **User** node on the object tree. The Users page appears.
2. Select the user account to be deleted from the table by clicking once (the user account will highlight blue).



3. Click the **Drop** button. A window will appear asking confirmation.



4. DBMaker provides three options for dropping user. The **Default** option drops the user without considering dependent objects. The **Restrict** option checks the user's schema and schema's table for dependencies. The **Cascad** option drops the user and the user's shema and the schema's table together.
5. Click the **OK** button to remove the user from the database.

©Copyright 1995-2026 CASEMaker Inc.

23 Managing Groups

Groups simplify the management of object privileges in a database with a large number of users. You can use a group to collect several users and/or groups. Any object privileges you grant to the group are automatically granted to all members of the group.

Members added to a group after object privileges are granted gain those object privileges in addition to object privileges are granted to them directly. Members removed from a group lose object privileges that are granted to the group, but retain any privileges that are to them directly.

You can also specify a group name in place of a user name, as long as the group you are trying to add does not contain the group you are adding it to. User and group names have a maximum length of 128 characters, and may contain letters, numbers, the underscore character, and the symbols $ and #. The first character may not be a number.

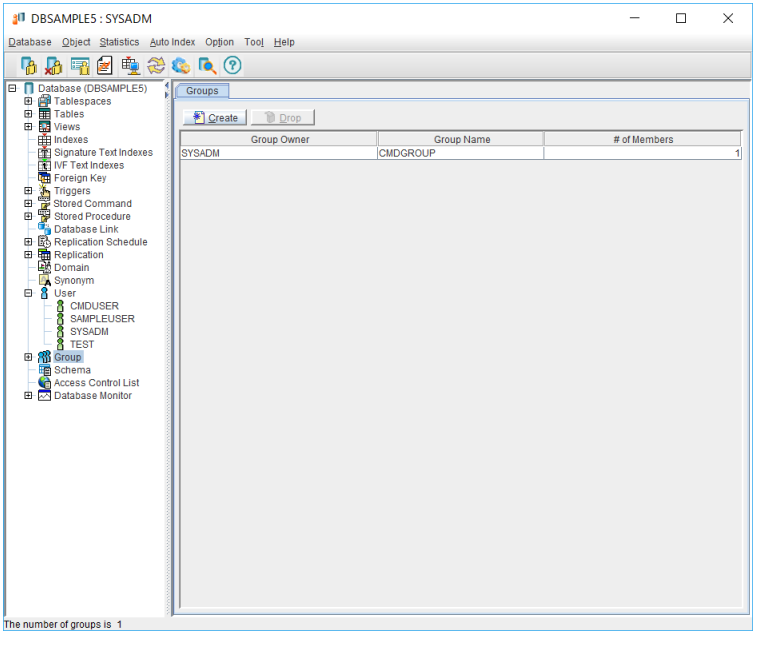
©Copyright 1995-2026 CASEMaker Inc.

23.1 Creating a New Group

New group accounts should always be created when a specific access profile is required for a group of individuals. It is also useful to create group accounts that represent the structure of your organization or business.

➲ Creating a new group:

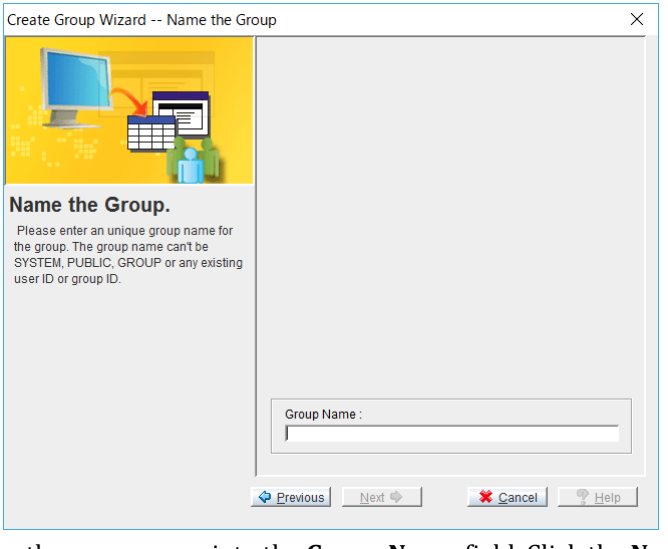
1. Select the **Group** node from the object tree. The **Groups** window appears displaying all existing groups, the group owner, and the number of members in the group.



2. Click the **Create** button. The **Create New Group** wizard starts.

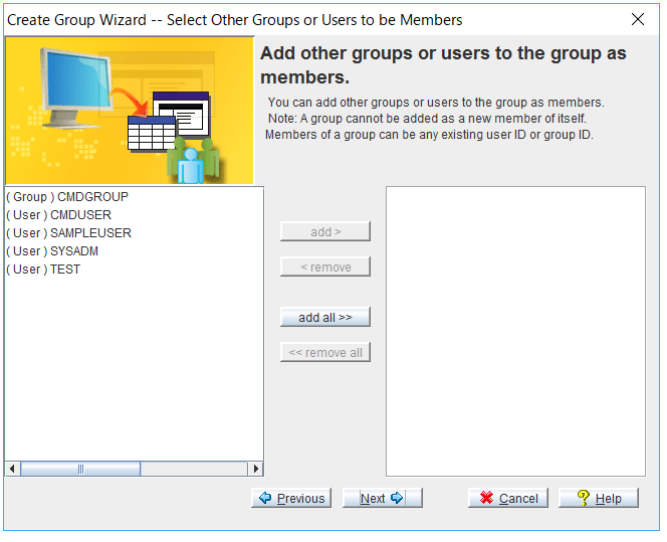
©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Name the Group** window appears.



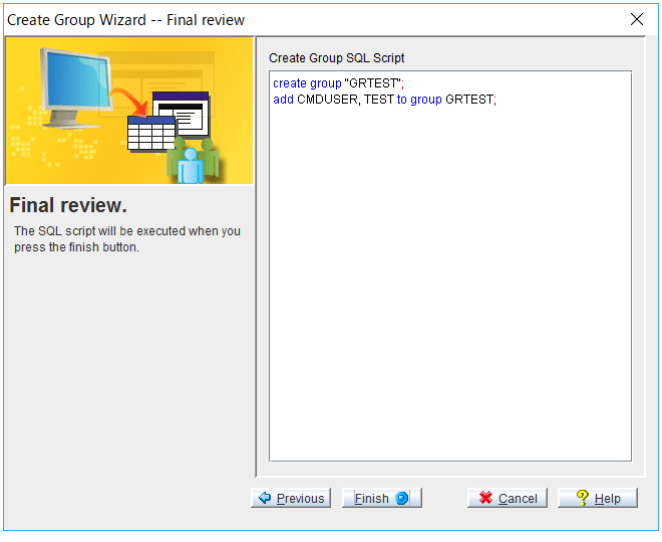
4. Enter the group name into the **Group Name** field. Click the **Next** button, the **Select Other Groups or Users to be Members** window appears.

©Copyright 1995-2026 CASEMaker Inc.

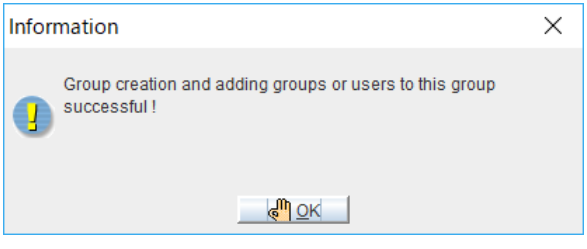


5. A list of available users/groups appears on the left list box. Select a user/ group by double clicking on the name or selecting the user(s)/ group(s) and clicking the **add** button. The **add all** button can be used to select all users/ groups to the newly created group. All the selected users/ groups are displayed in the list box to the right. Selected users/groups can be, likewise removed by double clicking on the name, or by selecting the users/groups and clicking the **remove** button. Click the **Next** button, the **Final review** page appears.

©Copyright 1995-2026 CASEMaker Inc.



6. Review the SQL script that has been created. You can add to the SQL script in the **Create Group SQL Script** field if necessary by clicking on the field. Click **Finish** to execute the SQL command. The **Information** dialog box appears.



7. Click the **OK** button.

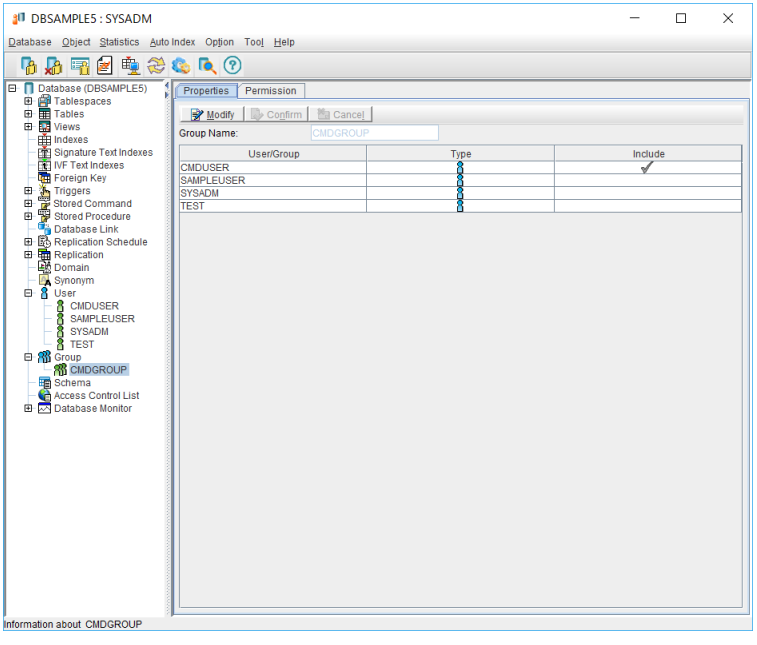
©Copyright 1995-2026 CASEMaker Inc.

23.2 Adding and Removing Group Members

Members should be removed or added to groups according to their access needs or position within your organization. Users may be added to groups from the group properties page or the user's property page.

➲ Adding and removing members from a group:

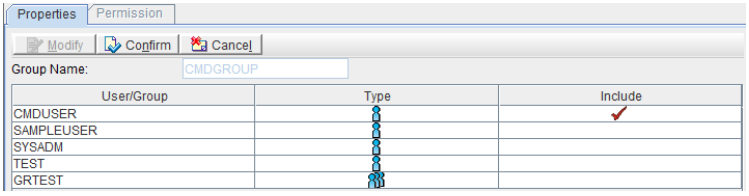
1. Select the **Group** node from the object tree. The **Groups** window will appear displaying all existing groups, the group owner, and the number of members in the group.



©Copyright 1995-2026 CASEMaker Inc.

2. Select a group from the tree or double click the group in the **Groups** table. The group **properties** page will appear. The group **properties** table shows all existing users and groups for the database. If a check mark is present in the include column next to a user or group, then they are a member of the group indicated in the **Group Name** field.

3. Click the **Modify** button from the top of the **Properties** page to include more users or groups or to remove users or groups from the selected group. A check in the **Include** column indicates that the user or group is a member of the selected group.



NOTE *Users may also be added to groups by selecting the user from the user's node, and checking the **Include** column on the group that they are to be a member of.*

4. Click the **Confirm** button to save the changes, or **Cancel** to discard them.

©Copyright 1995-2026 CASEMaker Inc.

23.3 Managing Group Object Privileges

Object privileges control which database objects a user can access and the actions they can perform. There are eight object privileges: SELECT, INSERT, DELETE, UPDATE, INDEX, ALTER, REFERENCE and EXECUTE.

SELECT privilege allows you to select the data in a database object. This privilege applies to the entire object and cannot be granted on specific columns.

INSERT privilege allows you to insert new data into a database object. This privilege can also be restricted to specific columns.

DELETE privilege allows you to delete data from a database object. This privilege applies to the entire object and cannot be granted on specific columns.

UPDATE privilege allows you to update data in a database object. This privilege can also be restricted to specific columns.

INDEX privilege allows you to create an index on a database object. This privilege applies to the entire object and cannot be granted on specific columns.

ALTER privilege allows you to alter the schema of a database object. This privilege applies to the entire object and cannot be granted on specific columns.

REFERENCE privilege allows you to create referential constraints (such as foreign keys) on a database object. This privilege can also be restricted to specific columns.

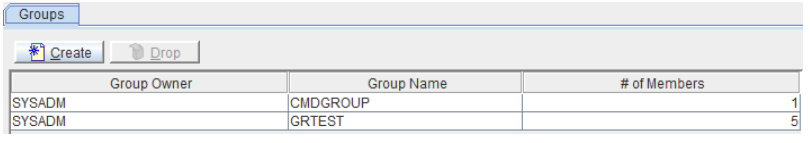
EXECUTE privilege allows the user to use a stored procedure or command.

It is possible to grant object privileges to all users simultaneously by granting the privileges to PUBLIC. All current and future users will then have those privileges on the database object.

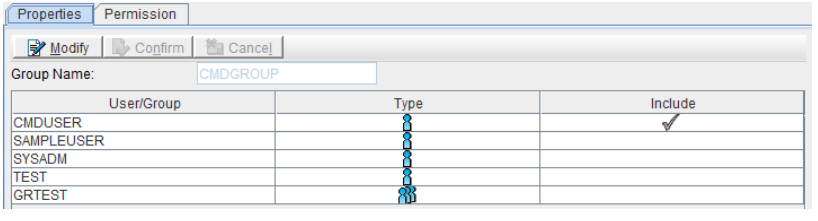
©Copyright 1995-2026 CASEMaker Inc.

➲ Granting and removing group object privileges:

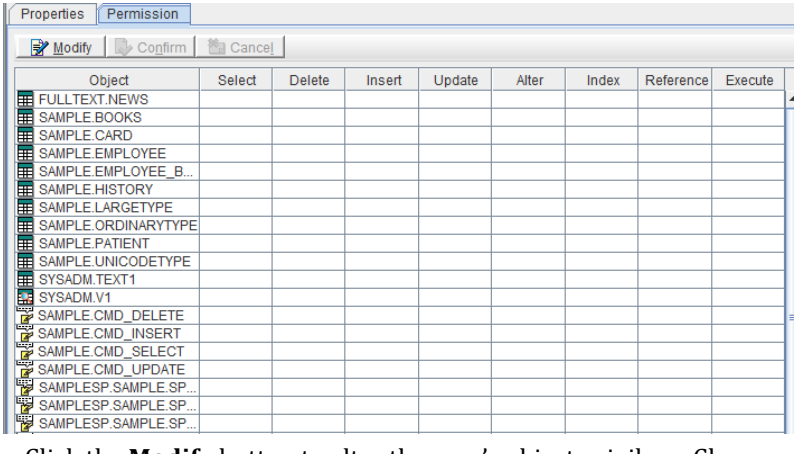
1. Select the **Group** node from the object tree. The **Groups** window will appear displaying all existing groups, the group owner, and the number of members in the group.



2. Double-click the group name in the tree or in the group table. The **Properties** page appears.



3. Select the **Permission** tab at the top of the page. The **Permission** page appears.



4. Click the **Modify** button to alter the user's object privilege. Choose which type of privilege to grant to the user on each object. Click the column and row corresponding to the object privilege to be granted and the object to grant the privilege to. Privileges can be likewise removed.

©Copyright 1995-2026 CASEMaker Inc.

A check mark will appear for each privilege selected.

5. Click the **Confirm** button to save the changes, or **Cancel** to discard them.

©Copyright 1995-2026 CASEMaker Inc.

24 Working with Schema

A schema is essentially a namespace. A schema contains schema objects, also known as named objects. Tables, views, indexes, synonyms, domains, commands, and procedures are all examples of schema objects. Synonyms and domains must have unique names in the database, but other schema object names may duplicate those of other objects existing in other schemas. Schema objects are accessed by qualifying their names. Qualifying a schema object means tagging the object with the schema's name as a prefix.

You can easily add and remove schema for the current database using the JTool interface. The schema name must be unique within the current database.

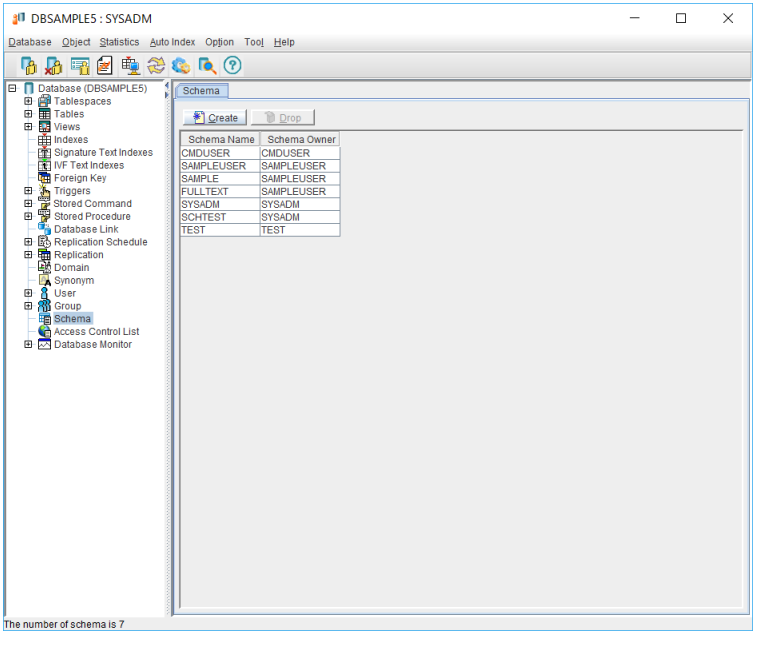
©Copyright 1995-2026 CASEMaker Inc.

24.1 Browsing Schema

You can browse all schema contained in a database by selecting the **Schema** icon from the data structure tree. The schema's name and owner are displayed in the panel to the right of the data structure tree.

➲ Browsing schema:

1. Click the **Schema** icon from the tree. A list of all schemas in the database is displayed.



2. Browse the schema listed in the right panel. It may be necessary to use the scroll bar to see all of the information.

©Copyright 1995-2026 CASEMaker Inc.

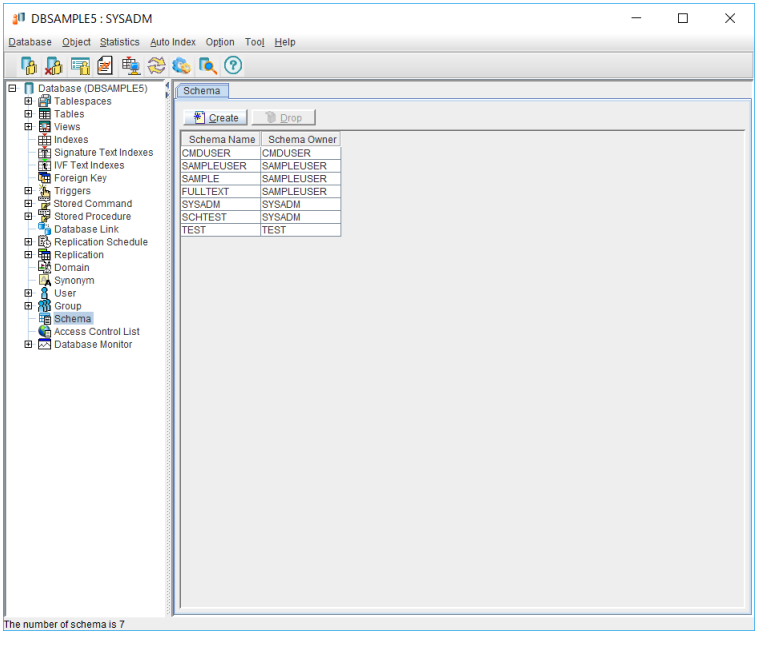
24.2 Creating a Schema

Creating a schema using JDBA Tool is a very intuitive process. You are guided through the process by a user-friendly user interface. After the schema is created, you can create objects within the schema. The schema owner has full control of the privileges granted on the schema including those created by the owner and others assigned by the database administrator (DBA). By default, users with RESOURCE authority or higher have CREATE privilege for the schema. A user only with CONNECT authority may not create any schema.

NOTE *Schema must have a unique name within the database.*

➲ Creating a schema:

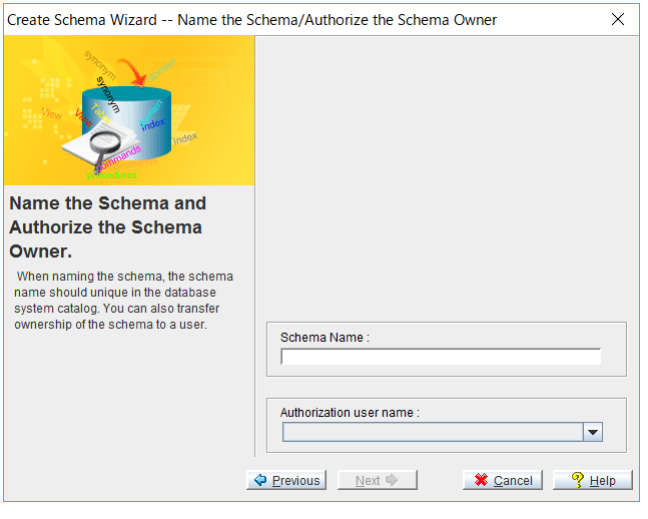
1. Click the **Schema** icon located in the tree. A list of all schemas in the database appears in the panel to the right.



©Copyright 1995-2026 CASEMaker Inc.

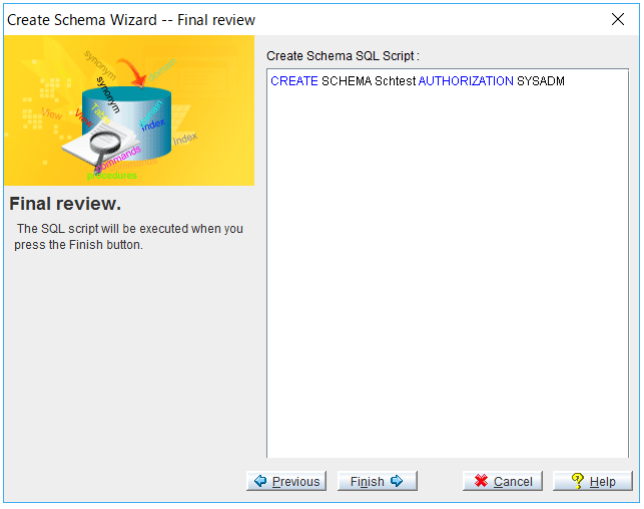
2. Click the **Create** button in the **Schema** window. The **Create Schema Wizard Introduction** window appears.

3. Click the **Next** button in the **Create Schema Wizard Introduction** window. The **Create Schema Wizard Name Schema/Authorize the Schema** window appears.

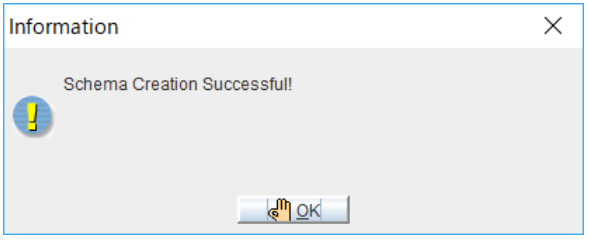


©Copyright 1995-2026 CASEMaker Inc.

4. In the **Schema name** field, enter a name for the schema. Schema names may not contain spaces and every schema name must be unique within the current database.
5. Select the schema owner from the **Authorization user name** drop down list. If no name is selected from the list, the **Create Schema Wizard** assigns ownership to the schema creator.
6. Click the **Next** button. The **Create Schema Wizard Final Review** window appears. The SQL code in the **Create Schema SQL Script** field can be edited.



7. Click the **Finish** button when you are satisfied with the SQL code that will be used for creating the schema. A message appears to confirm a successful schema creation. The new schema now appears in the **Schema** window to the right.



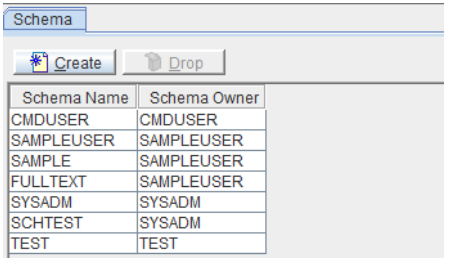
©Copyright 1995-2026 CASEMaker Inc.

24.3 Dropping a Schema

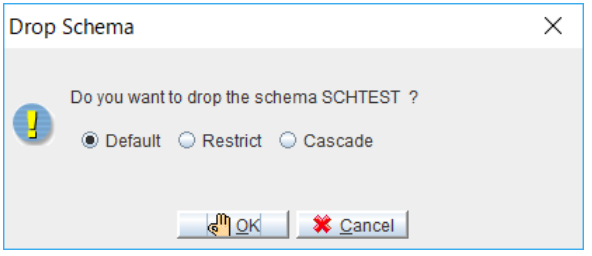
When a schema is no longer required, it can be removed. By default, users with RESOURCE authority or higher have DROP privilege for the schema. A user only with CONNECT authority may not drop any schema.

➲ Removing (i.e., dropping) a schema:

1. Click the **Schema** icon from the tree. A list of all schemas in the database is displayed in the panel to the right.



2. Click the schema you want to drop then click the **Drop** button in the **Schema** window. The **Drop Schema** confirmation dialog box appears.



3. There are three options for dropping schema. The **Default** option deletes the shema without considering dependent object within the schema. Dependent objects become invalid after the schema is dropped. The **Restrict** option considers table, domain, and procedure dependencies. This option does not permit a schema drop when one or more objects depend on the schema. The **Cascade** option drops the schema and all of its dependent objects. Click the radio button to select a schema dropping option.

©Copyright 1995-2026 CASEMaker Inc.

4. Click the **OK** button. The schema is removed from the database or a warning dialog box appears if the schema cannot be dropped.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

25 Managing Access Control List

Access Control List provides an ability to determine whether allow some users from some IPs to connect to the database. DBA can use it to allow or block some users come from any network address. You can easily create, alter or drop the Access Control List with the JTOOL interface. After creating a new database, there are three defaulted tuples in the list. They are (PUBLIC, * , allow), (SYSADM, 127.0.0.1, allow) and (SYSADM, * , allow), respectively. Please notice that the record (SYSADM,127.0.0.1,allow) cannot be removed because the SYSADM is always allowed to connect to database from the local machine.

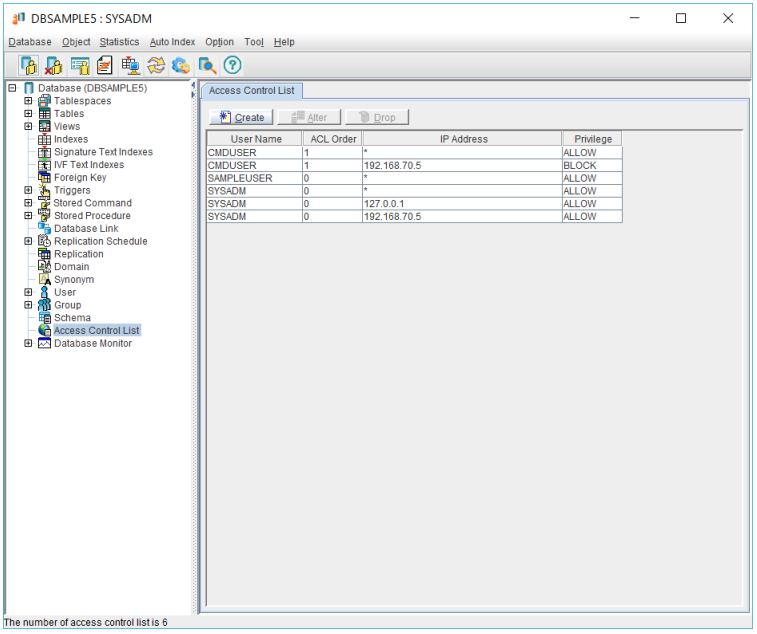
©Copyright 1995-2026 CASEMaker Inc.

25.1 Browsing Access Control List

You can quickly browse the Access Control List (ACL) in a database by selecting the Access Control List icon from the data structure tree. The schema's name and owner appear in the panel to the right of the data structure tree.

➲ Browsing ACLs:

1. Click the **Access Control List** icon from the tree. A list of all users' name and the IP address in the database appears.



2. Browse the listed on the right panel by using the upward and downward arrows.

©Copyright 1995-2026 CASEMaker Inc.

25.2 Creating Access Control List

Creating an ACL using JDBA Tool is a very intuitive process. You are guided through the process by a user-friendly user interface.

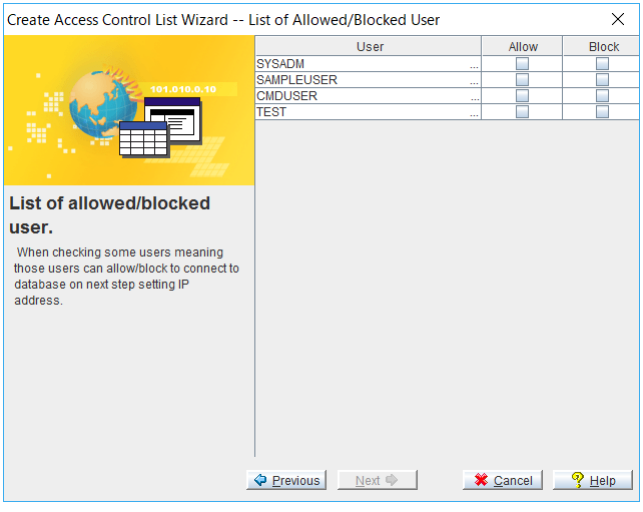
DBMaker uses the new keyword “PUBLIC” to mean that all users that want to connect to database. SYSADM cannot create a user name with “PUBLIC”. By default, a user only with SYSADM or SYSDBA authority have CREATE privilege for the access control list.

➲ Creating an ACL:

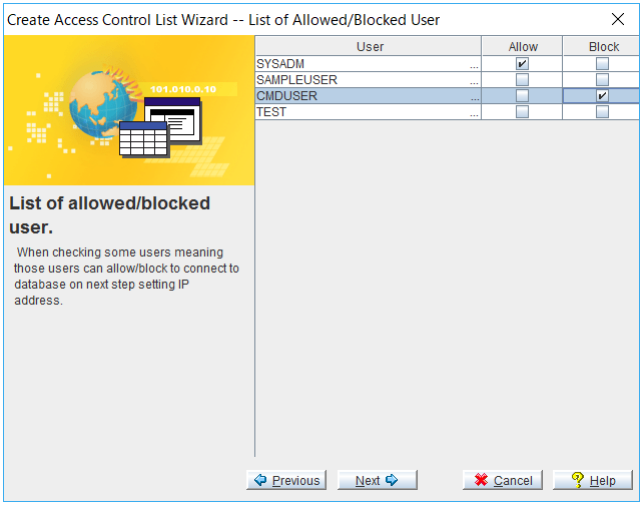
1. Click the Access Control List icon from the tree.The **Access Control List** page appears.
2. Click the **Create** button in the **Access Control List** window. The **Create Access Control List Wizard - Introduction** window appears.

3. Click the **Next** button in the **Create Access Control List Wizard - Introduction** window. The **List of Allowed/Blocked User** window appears.

©Copyright 1995-2026 CASEMaker Inc.



4. Choose the user you want to **Allow** or **Block** field.

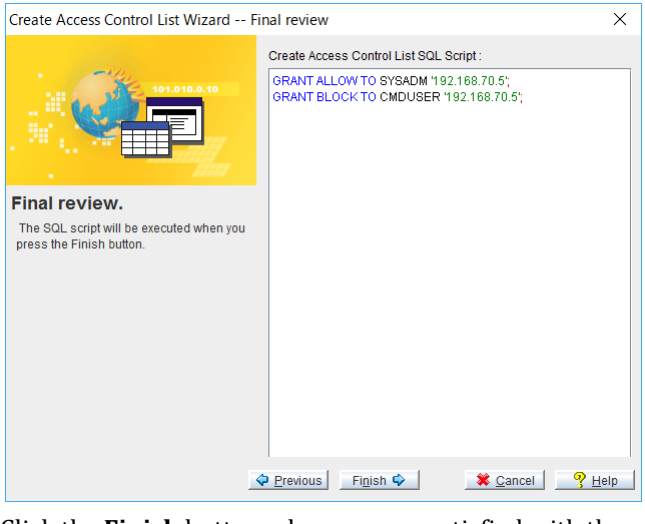


5. Click the **Next** button in the **List of Allowed/Blocked User** window. The **List of Allowed/Blocked IP Address** window appears.

©Copyright 1995-2026 CASEMaker Inc.

6. Input the IP address you want to allow in the **input allowed/blocked IP address** field.

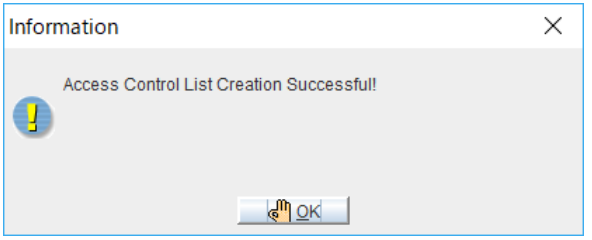
7. Click the **Next** button the **Final review** window appears. From this window the user can edit the SQL code in the **Create Access Control List SQL Script** field.



8. Click the **Finish** button when you are satisfied with the result SQL Script. A conformation message appears stating that the creation of the

©Copyright 1995-2026 CASEMaker Inc.

ACL was successful. The created Access Control List appears in the **Access Control List** window to the right.



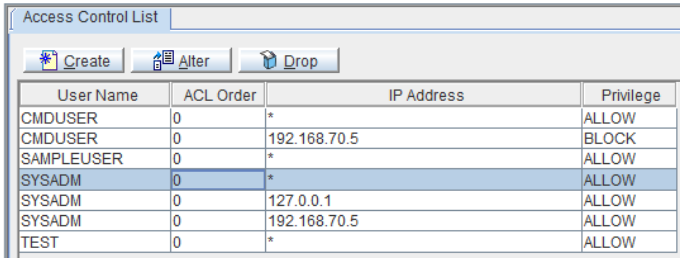
©Copyright 1995-2026 CASEMaker Inc.

25.3 Altering Access Control List

If an ACL order has to be changed, it can be altered by the following procedure. By default, a user only with DBA authority or higher can alter the access control list.

➲ Altering an ACL:

1. Select the object **Access Control List** from the tree and select the ACL from the list.

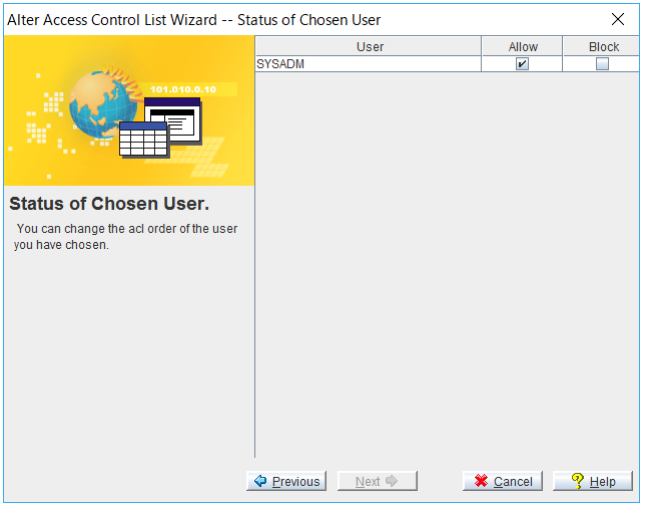


User Name	ACL Order	IP Address	Privilege
CMDUSER	0	*	ALLOW
CMDUSER	0	192.168.70.5	BLOCK
SAMPLEUSER	0	*	ALLOW
SYSADM	0	*	ALLOW
SYSADM	0	127.0.0.1	ALLOW
SYSADM	0	192.168.70.5	ALLOW
TEST	0	*	ALLOW

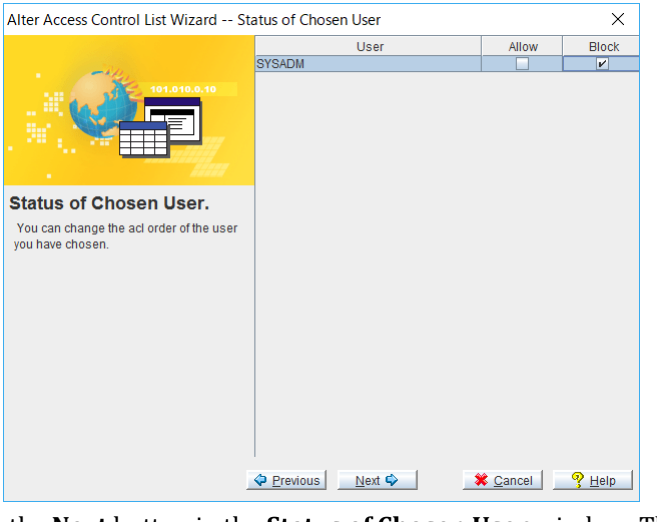
2. Click the **Alter** button in the **Access Control List** window. The **Alter Access Control List Wizard - Introduction** window appears.

©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button in the **Alter Access Control List Wizard - Introduction** window. The **Status of Chosen User** window appears.

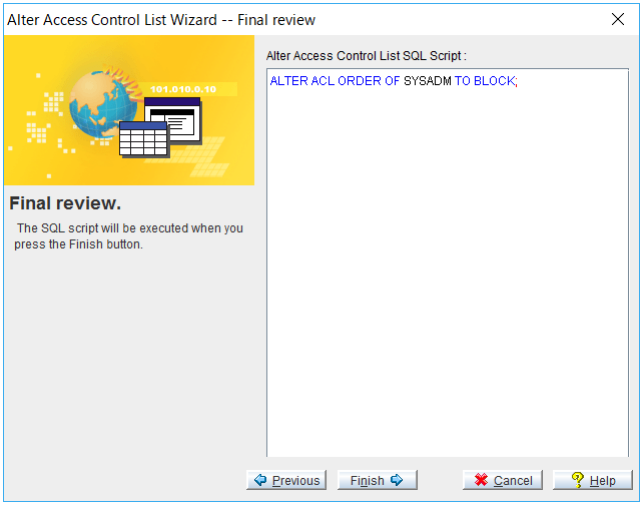


4. Choose the user you want to **Allow** or **Block** field.

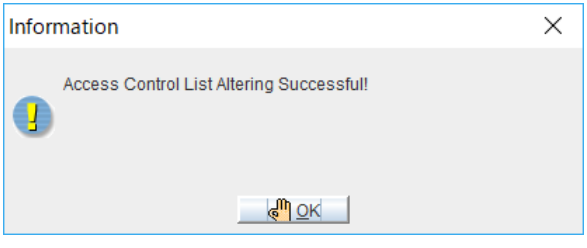


5. Click the **Next** button in the **Status of Chosen User** window. The **Final review** window appears.

©Copyright 1995-2026 CASEMaker Inc.



6. Click the **Finish** button when you are satisfied with the result SQL Script. A conformation message appears stating that the change of the ACL was successful. The altered Access Control List appears in the **Access Control List** window to the right.



7. Click the **OK** button.

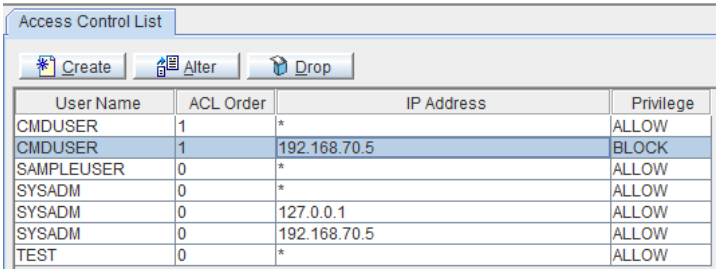
©Copyright 1995-2026 CASEMaker Inc.

25.4 Dropping Access Control List

Consider dropping an ACL if it is no longer required. By default, a user only with DBA authority or higher can drop the access control list.

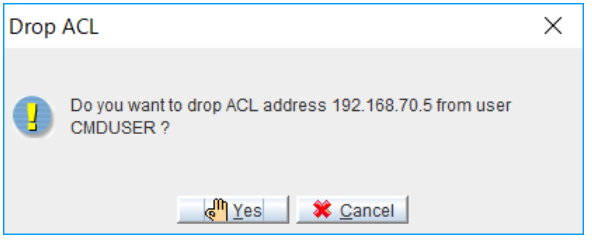
➲ Dropping an ACL:

1. Select the object **Access Control List** from the tree and select the ACL from the list.



User Name	ACL Order	IP Address	Privilege
CMDUSER	1	*	ALLOW
CMDUSER	1	192.168.70.5	BLOCK
SAMPLEUSER	0	*	ALLOW
SYSADM	0	*	ALLOW
SYSADM	0	127.0.0.1	ALLOW
SYSADM	0	192.168.70.5	ALLOW
TEST	0	*	ALLOW

2. Click **Drop**. The **Drop ACL** window will open to confirm if you want to drop the ACL



3. Click the **Yes** button.

©Copyright 1995-2026 CASEMaker Inc.

26 SQL Query

SQL Query is a graphical user interface (GUI) that hides the complexity of the DBMS and query language behind an intuitive. It allows users to quickly manage and manipulate the database without the trouble of entering formal commands using SQL. Users can click the button to call-out the SQL Query interface.

This chapter gives systematic instructions on how to use **SQL Query** tool to run a SQL query, and view query results by table mode or plain text mode. Users should have some SQL language knowledge before using SQL Query tool.

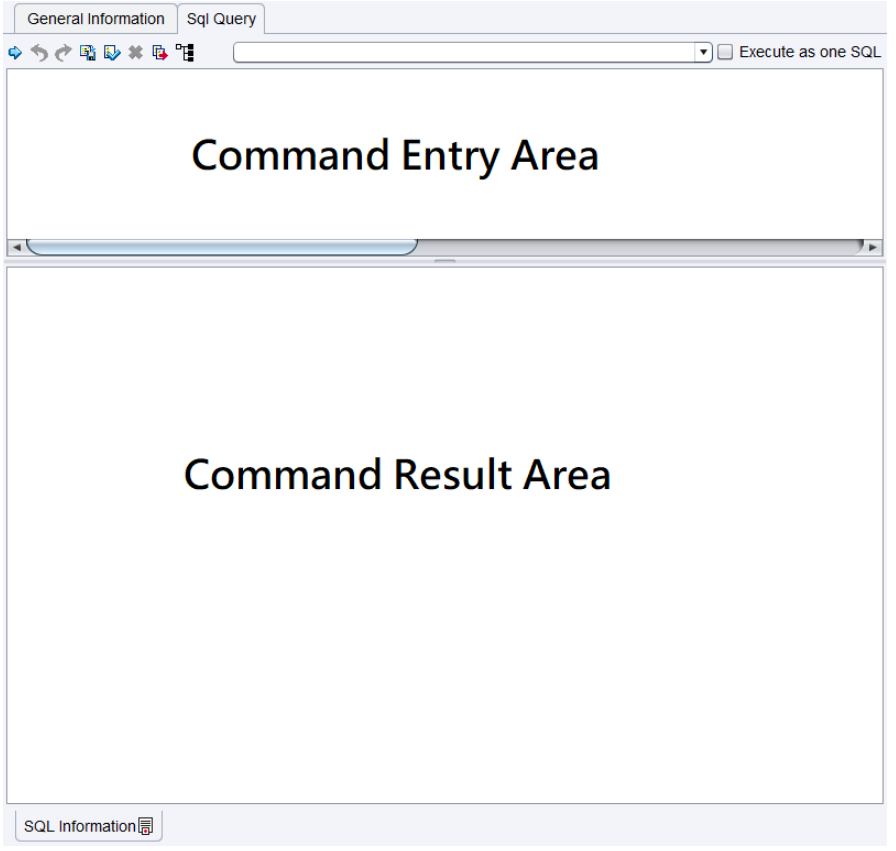
SQL Query tool has a wide variety of features that enable users to effectively manage databases:

- Run multiple query commands simultaneously
- View query results by plain text mode/table mode
- Load/save SQL commands
- Execute by batch file
- Multiple query results viewed for table mode

©Copyright 1995-2026 CASEMaker Inc.

26.1 The main functional components of SQL Query

The following graphic displays the important features of the Workspace. Users can input SQL query commands in the Command Entry Area, and view the results in the Command Result Area. This tool is similar to the dmSQL, but does not support loading/unloading of host variables.



©Copyright 1995-2026 CASEMaker Inc.

Command Entry Area

Users can use the Command Entry Area to enter more than one command to execute but separate each command with a semicolon, input or load from external files to the command window. Then press **F12** or click the **RUN SQL** button to execute SQL commands.

NOTE *Users can only enter SQL statement in the Command Entry Area. If users need to use SQL block, please refer to the following chapter:* ***Executing commands as one SQL****.*

Command Result Area

The results display as a plain text (like *dmSQL* tool), or a table below the command window when execute queries. Whether the command executes successfully or not, the success or error message will display in result area.

Common options

In the SQL Query user interface, users can use the following buttons and checkbox to implement common commands.

: Click this button or press F12 to run the SQL command.

: Click this button to undo the last move.

: Click this button to redo the next move.

: Click this button to save the SQL command that you have been entered.

: Click this button to load the SQL command from the external file.

: Click this button or press F4 to clear the input or loaded SQL command.

: Click this button to execute the batch file.

: Click this button to show the execution plan.

Execute as one SQL: This checkbox can execute every SQL command in the Command Entry Area as on SQL, often used to create function/procedure.

©Copyright 1995-2026 CASEMaker Inc.

26.2 Using SQL Query

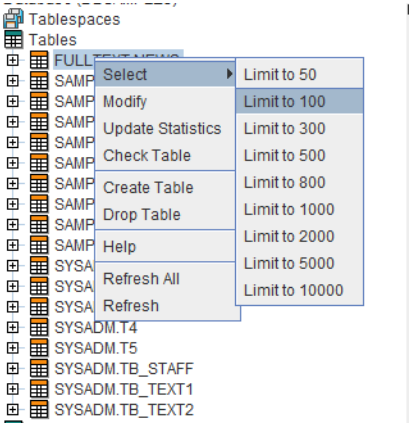
This section mainly introduces the usage of SQL Query tool. It mainly includes the following options: run SQL, save command, load command, clear command and execute batch. You can choose it according to the requirement.

Executing Commands

Users can run commands which have been inputted or loaded. You can aslo double-click the table node of JDBA Tree to select a table, and right-click it to choose SELECT commands.

➲ Run the SQL script command via the table node

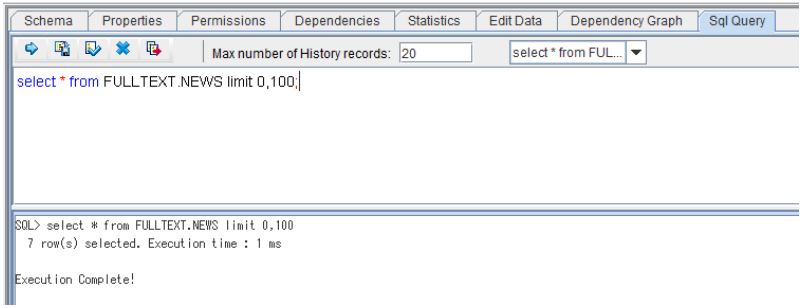
1. Double-click the table node of JDBA Tree to select a table, and right-click it to choose the SELECT commands：

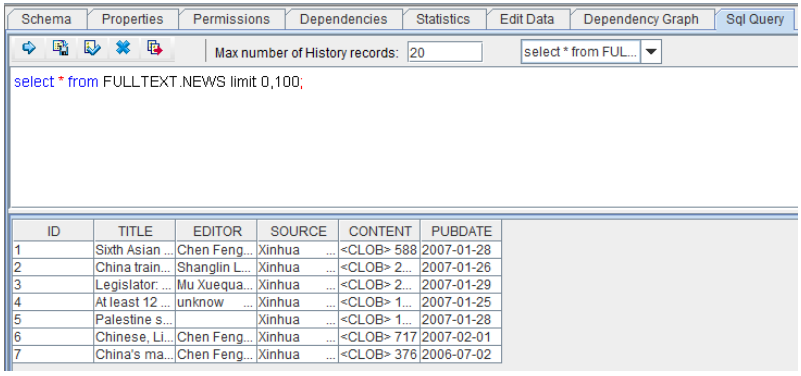


NOTE *The option after the select command indicates the number of records that the user wants to query at most.*

2. Click the number of records you want to query, and the query results will be displayed in the Command Result Area.

©Copyright 1995-2026 CASEMaker Inc.





3. Use the up and down arrows to adjust record's order or view the remaining records in table when viewing a large number of records.

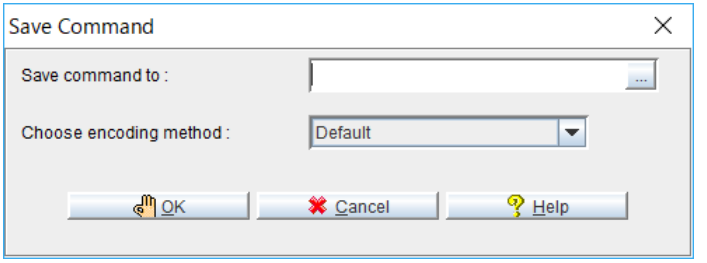
Saving commands

Users can save query command into an external file.

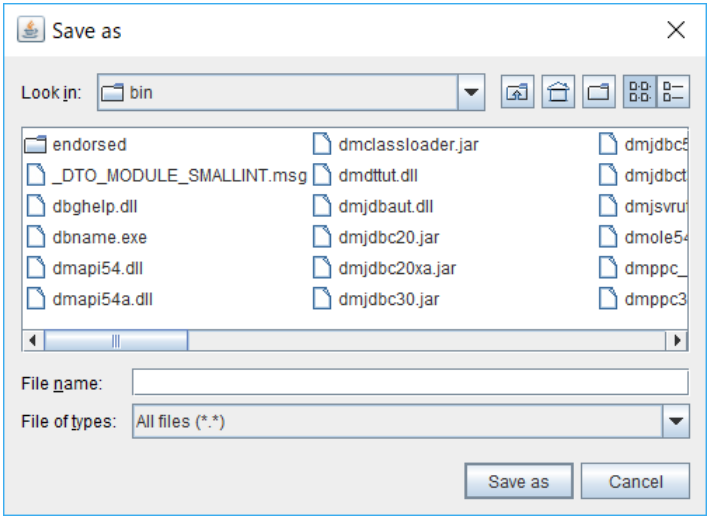
➲ To save SQL commands

1. Select **Save Command** from the menu. The **Save command** dialog box displays.

©Copyright 1995-2026 CASEMaker Inc.



2. The **Save command to** field specifies the path to which command files are placed.
3. You can identify a new location for the command file directory by entering a directory path in the field or by selecting a path using the browse button (...).
4. Selecting the browse button opens the Save as dialog box.



   d) Enter a name and location for the file in the dialog box

   e) Click the **Save as** button.

5. Choose an encoding method from the **Choose encoding method** drop down list. SQL Query tool supports three encoding methods to save the command: **Default**, **Unicode Little**, or **Unicode Big**.
6. Click the **OK** button.

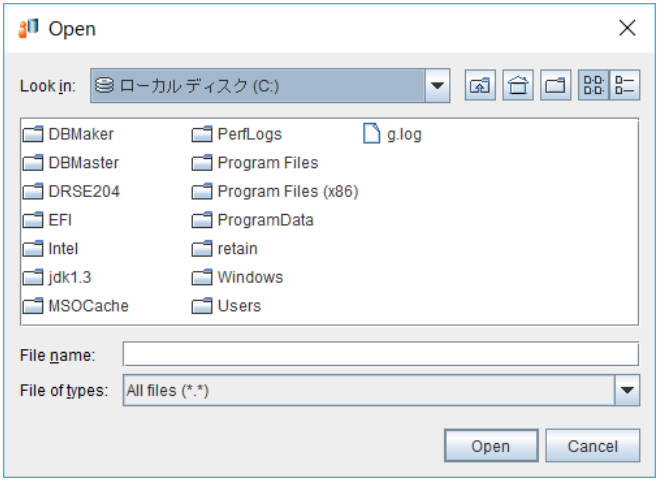
©Copyright 1995-2026 CASEMaker Inc.

Loading Commands

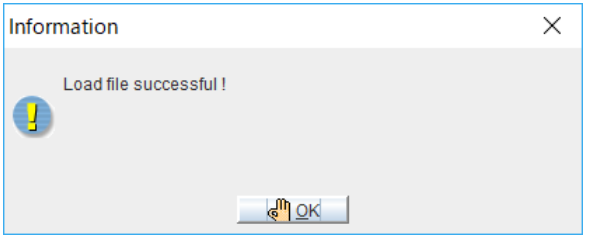
Users can also load SQL commands into the command window from external files.

➲ To load commands

1. Choose **Load command** from the **File** menu. The **Open** dialog appears.



2. Select the location of the file from the **Look in** field.
3. Click the **Open** button. A confirmation dialog box appears.



4. Click the **OK** button. The load command appears in the SQL command entry area.
5. Press **F12** or click the **Run** button ( ) to execute the SQL command.

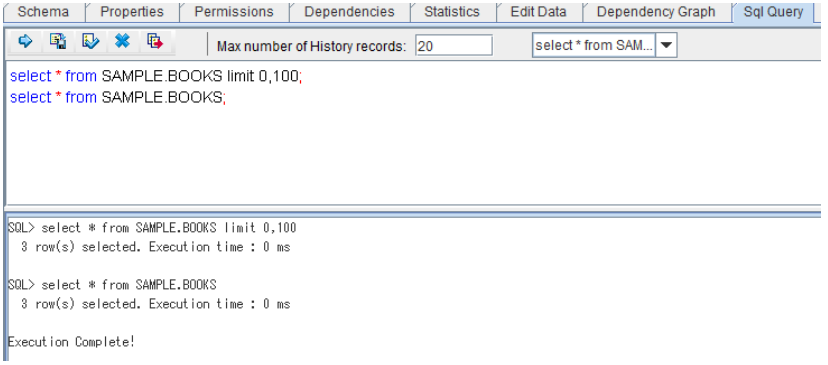
©Copyright 1995-2026 CASEMaker Inc.

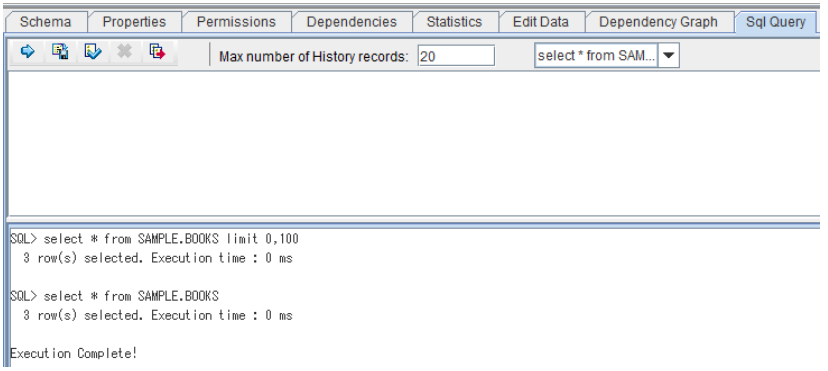
Clearing commands

Users can clear the SQL commands that don't need to be used.

➲ To clear commands

1. click the button or using the shortcut key F4 to clear the command that is in the command entry area.





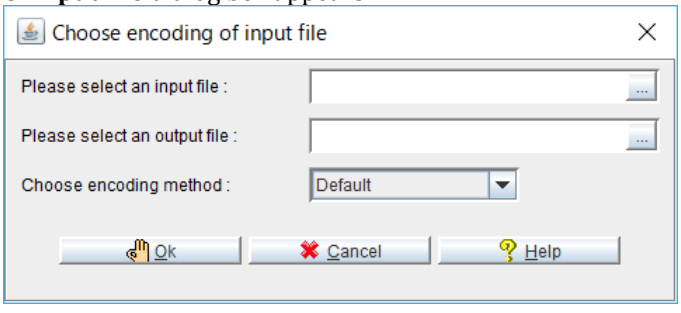
Execute Batch

Users can run a batch file directly without load commands to the command window.

©Copyright 1995-2026 CASEMaker Inc.

➲ To execute batch commands

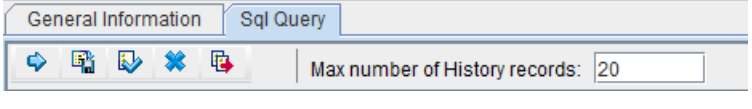
1. Select **Execute Batch** from the common options, the **Choose Encoding of input file** dialog box appears.



2. Enter a directory for the input file in the **Please select an input file** field or by using the accompanying browse button (...).
3. Enter a directory for the result to save in the **Please select an output file** field or by using the browse button (...).
4. Select an encoding method from the **Choose encoding method** drop down list. Four methods are supported for encoding file: **Default**, **Auto**, **Unicode Little**, or **Unicode Big**.
5. Click the **OK** button. The input command saves into the output file using the selected encoding method.

Preference setting

In the SQL Query interface, you can set the number of historical commands saved in the database by setting the **Max number of History records**. You can change the maximum number of historical records to any value. The default value is 20.



©Copyright 1995-2026 CASEMaker Inc.

Execute commands as one SQL

Users can execute the commands in the Command Entry Area as one SQL. Select the **Execute as one SQL** checkbox means semicolon will no longer separate commands, the whole commands in the area will be seen as one SQL commands. This checkbox allows users to create function without setting block delimiter.

➲ Execute commands as one SQL:

1. Enter SQL commands in the Command Entry Area and select **Execute as one SQL** checkbox.
2. Press **F12** or click the **Run SQL** button to execute the SQL command.

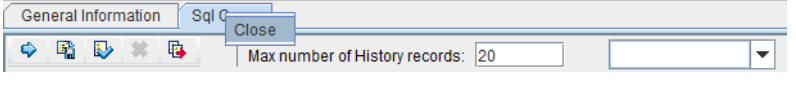
➲ Create function/stored procedure with Execute commands as one SQL checkbox:

1. Enter the CREATE FUNCTION or CREATE PROCEDURE syntax. Following uses function as example

2. Select **Execute as one SQL** checkbox.
3. Press **F12** or click the **Run SQL** button.

26.3 Exit SQL Query

When you want to quit the SQL Query interface, you can right-click the SQL query tab, and select the close button to close it.



©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

27 Data Transfer

The JData Transfer Tool provides a user-friendly interface for transferring data in and out of the database. The tool performs the following functions:

- Import from text
- Import from XML
- Import from JSON
- Import from ODBC
- Export to text
- Export to XML
- Export to JSON

A wizard performs each type of data transformation. Each wizard guides the user through every step in the data transformation process, and gives descriptive information on the purpose of each step and the effect of different choices on the result.

The JData Transfer Tool is a separate application it may be opened from windows start>programs>DBMaker 5.4>JDataTransfer, or opened from JDBA Tool. It consists of a main console and a menu bar, as illustrated in *Figure 27-1*. The main console provides seven options: import from text, import from XML, import from JSON, import from ODBC, export to text, and export to XML and export to JSON. The Menu Bar consists of three menus: the Transfer menu, the Option menu, and the Help menu. The Transfer menu provides the same

©Copyright 1995-2026 CASEMaker Inc.

transfer functions as the main console, with the addition of the batch transfer function. The option menu changes the language that the UI is displayed in. Currently English, Japanese, and Chinese (traditional) are supported. The help menu provides access to the help system for JDBA Tool.

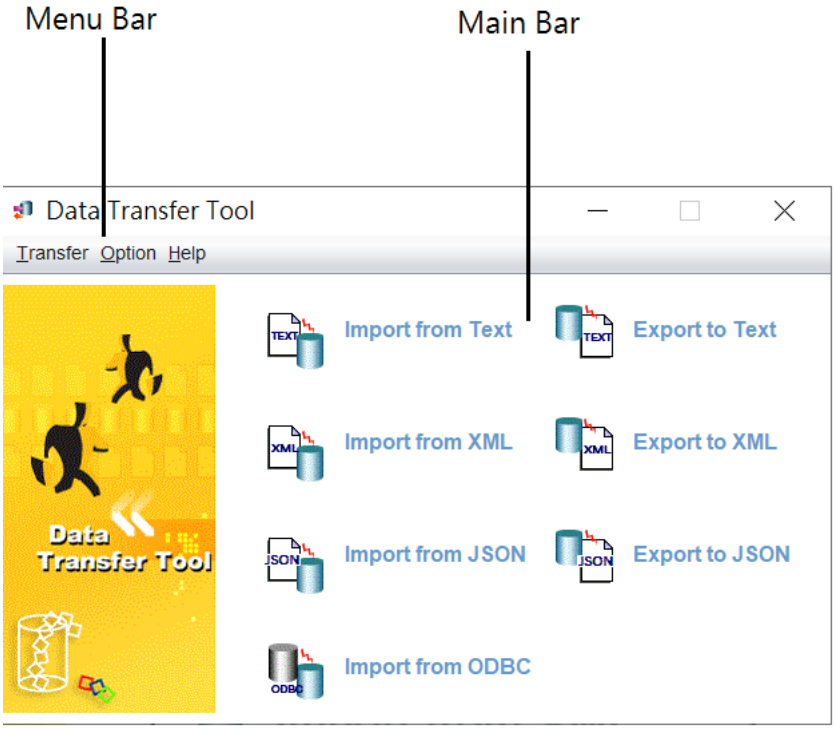


Figure 27-1 The JData Transfer Tool main window

➲ Opening the JData Transfer Tool by JDBA Tool:

1. Start JDBA Tool and connect to the database that data is to be transferred to or from.
2. Select **JData Transfer** from the Tool menu. The JData Transfer tool window appears.

NOTE: You can also right click the tree node of the table and select Import/Export from the popup menu to import/export the selected table.

©Copyright 1995-2026 CASEMaker Inc.

27.1 Importing Data from Text

The ability to import table data from a text file is an important feature in a database, and is made easy with the JData Transfer Tool. Text data must be properly formatted to be acceptable for import. Data may be imported to the database only from a properly formatted text file. This section describes the types of formatting that JData Transfer Tool supports, and then describes how to import text to the database using the tool.

Before attempting to import data from a text file, you should be sure that the file is in a format that will result in coherent structured data within the database. Certain programs output data in a fixed text format, if this is the case, check the format of the output file that you want to import. Some important settings to consider in the format of a text file include the following:

- Row Delimiter: Determines the type of character that signifies a break between the rows of a table. Possible characters: {CR/LF} (Carriage return / line feed. In Windows applications, a new line in the text is normally stored as a pair of CR LF characters. In UNIX applications, a new line is normally stored as a LF character. Some applications use only a CR character to store a new line), {CR}, {semicolon} (;), {comma} (,), {tab}, {vertical bar} (|), (semicolon}{LF}, or {comma}{LF}.
- Column Delimiter: Determines the type of character that signifies a break between columns in each row. Possible column delimiter characters include: semicolon, comma, or vertical bar.
- Text Qualifier: Determines how each tuple of any data type except BINARY, LONGVARBINARY, or numeric data types (integer, smallint, serial, decimal, double, float) is enclosed. Possible values: none, single quote, or double quote.
- Binary Qualifier: Determines how each tuple of BINARY or LONGVARBINARY type data is enclosed. Possible values: none, single quote, or double quote.

©Copyright 1995-2026 CASEMaker Inc.

- Binary Padding: Binary type data may have a character appended to it.
- Fixed Field: Instead of using a row delimiter, the text file may be formatted with fixed fields. This means a number of spaces, or fields, defines each column.
- Include column name: The first line in a text file may be used to define the column names. The format is *"column1"."table name"."owner name";"column2"."table name"."owner name";* etc. In this case the column delimiter is set to semicolon (;).
- Include table schema: The first line in a text file may be used to define the column schema (or the second line if the first line was used to define column names). The format is *data type(scale,precision);data type(scale,precision)* etc. In this case, the column delimiter is set to semicolon (;).
- Use NULL to display null data: Columns that contain no data display "NULL".
- File link name for FILE type data: The file name for system or user file objects is displayed.
- Use escape character "/": This character is used when qualifiers or delimiter characters appear in the data. If the data contains a reserved character, the reserved character will be enclosed by an escape character (/) so that the text import engine recognizes it as data, not a qualifier or delimiter.
- Use temp files to store LONGVARBINARY or LONGVARCHAR type data column content: BLOB data is stored as a separate, linked file (as a file object), and the name of the file containing the BLOB is displayed.

Data may be imported to a new table or an existing table. Three options are available when importing data to an existing table. The destination table may be replaced, removing the schema and data of the original table. The rows of the destination table may be replaced, meaning schema is retained, but data is removed. The last option is to append rows to the destination table, in which

©Copyright 1995-2026 CASEMaker Inc.

case the table's original schema and data is preserved, and the data from the text file is appended as new rows. In the last two cases, be aware that the schema of the destination table must be such that it is able to accept the imported data. For example, a set of integer data can be imported into a column of CHAR type data, but not vice-versa. Refer to the *SQL Command and Function Reference* for more information on acceptable formats for supported data types in DBMaker.

Example:

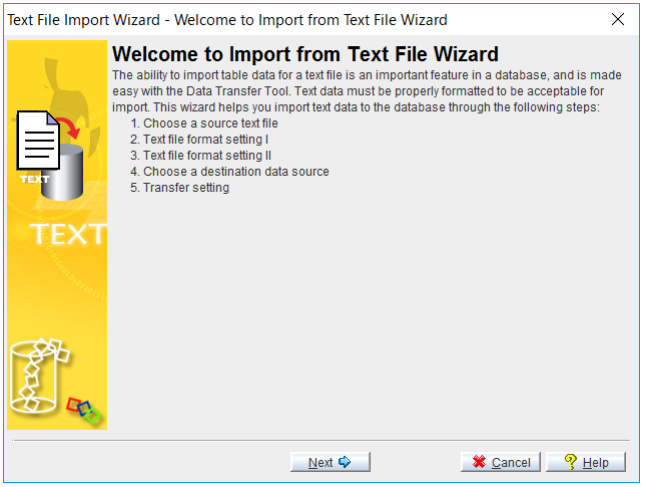
The following text file contains owner, table, and column information in the first line, and the table's schema in the second line. Subsequent lines contain raw data. Semicolons delimit columns. Single quotes qualify non-binary data; double quotes qualify binary data. The LONGVARCHAR and LONGVARBINARY columns display temporary file names (the data has been stored as file objects), and binary data is padded with "9".

```
"SYSADM"."EXPORT"."LOGINID";"SYSADM"."EXPORT"."REQUEST";"SYSADM"."EXPORT"."REQUES
TTIME";"SYSADM"."EXPORT"."ATTACHMENT";"SYSADM"."EXPORT"."BINARY_C";"SYSADM"."EXPO
RT"."DECIMAL_C"
SQL_CHAR(20);SQL_LONGVARCHAR;SQL_TIMESTAMP;SQL_FILE;SQL_BINARY(10);SQL_DECIMAL(10
, 3)
'A_HOWARD            ';'blobtmpdir2\\blbtmpf0.txt';'2001-09-09
12:47:05.100';'C:\\DBMAKER\\5.4\\BIN\\WEBDB\\FO\\ZZ00000B.GIF';"10000000000000000
000"9;10.250
'A_HOWARD            ';'blobtmpdir2\\blbtmpf1.txt';'2001-09-22
10:14:21.000';'C:\\DBMAKER\\5.4\\BIN\\WEBDB\\FO\\ZZ00000C.GIF';"20000000000000000
000"9;13.550
'A_HOWARD            ';'blobtmpdir2\\blbtmpf2.txt';'2001-10-04
16:22:06.000';'C:\\DBMAKER\\5.4\\BIN\\WEBDB\\FO\\ZZ00000D.GIF';"30000000000000000
000"9;27.333
```

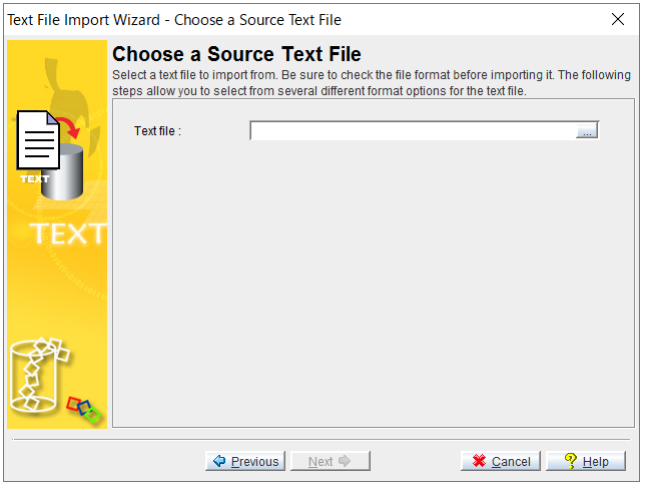
➲ Importing a text file to a database:

1. Open the JData Transfer Tool.
2. Select **Import Text File** from the main console or the **Transfer** menu. The Welcome to Import from Text File Wizard window will open, displaying a summary of the steps to be taken in the wizard

©Copyright 1995-2026 CASEMaker Inc.

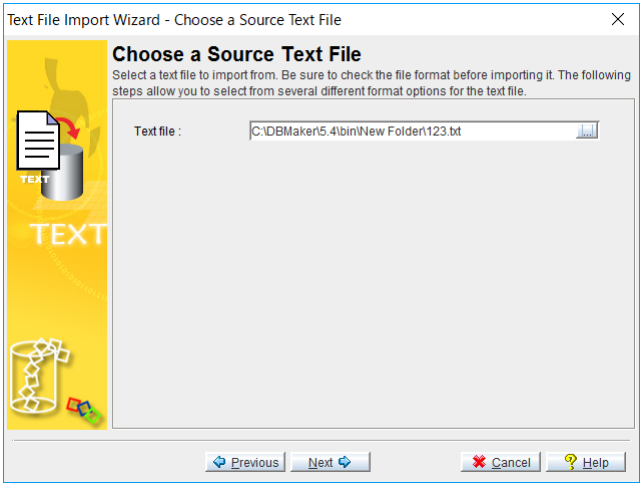


3. Click the **Next** button. The **Choose a Source Text File** window appears.

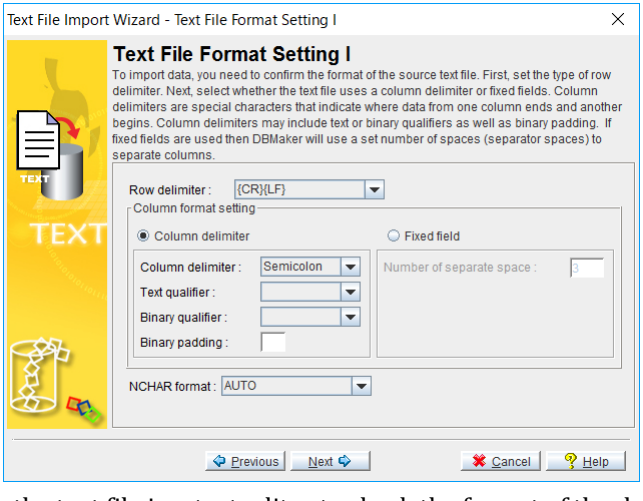


4. Enter the full path of a text file to import or click the **browse** button to search for a text file.

©Copyright 1995-2026 CASEMaker Inc.



5. After you have selected a text file, click the **Next** button, the **Text File Format Setting 1** window appears.

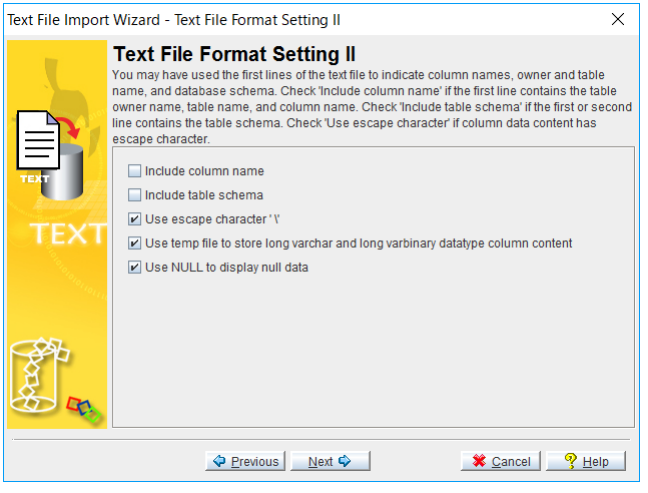


6. Open the text file in a text editor to check the format of the data.
7. Select the appropriate settings for the format of the text file you are importing.
8. Select the **Nchar format**. Users have three options, **AUTO**, **HEX** and **CHAR**. AUTO represents trying to import data as a file in HEX format

©Copyright 1995-2026 CASEMaker Inc.

first, if fails, as characters; HEX represents trying to import data as a file in HEX format. CHAR represents trying to import data as characters.

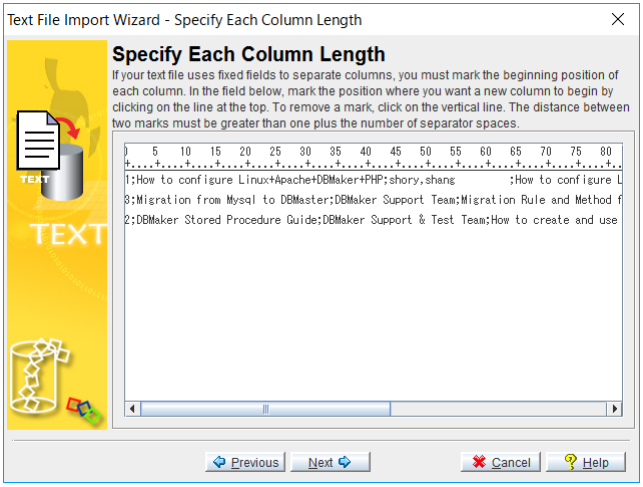
9. Click the **Next** button, the **Text File Format Setting 2** window appears.



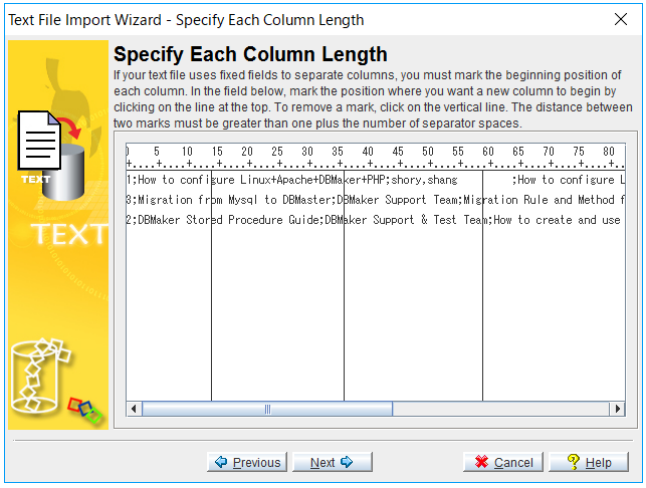
10. Finish selecting the appropriate settings for the format of the text file you are importing. Click the **Next** button.

11. If you selected **Fixed field** from the **Text File Format Setting 1** window, the **Specify Column Length** window will open. If you selected **Column Delimiter**, then proceed to step 12.

©Copyright 1995-2026 CASEMaker Inc.

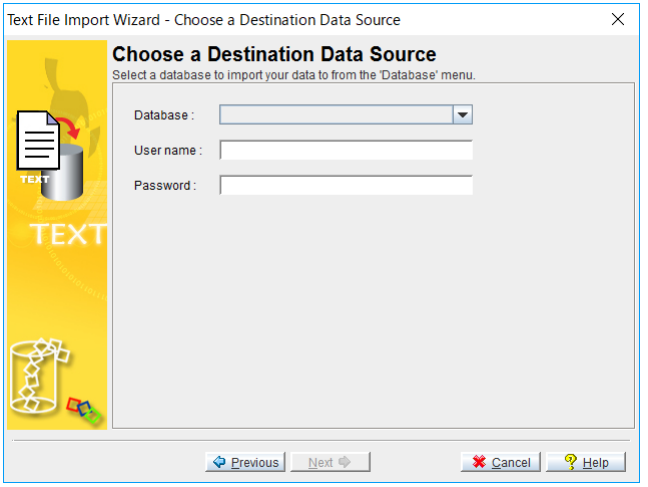


12. Click the horizontal line where you want to indicate the beginning of a column. A vertical line appears to indicate the column break. Use the scroll bar at the bottom to advance to other columns. Click the **Next** button.



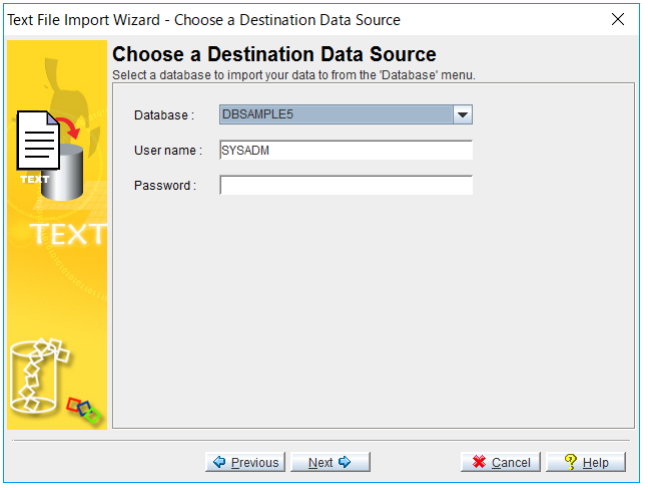
13. The **Choose Destination Data Source** window appears.

©Copyright 1995-2026 CASEMaker Inc.



NOTE *The user must have CONNECT or higher authority to login.*

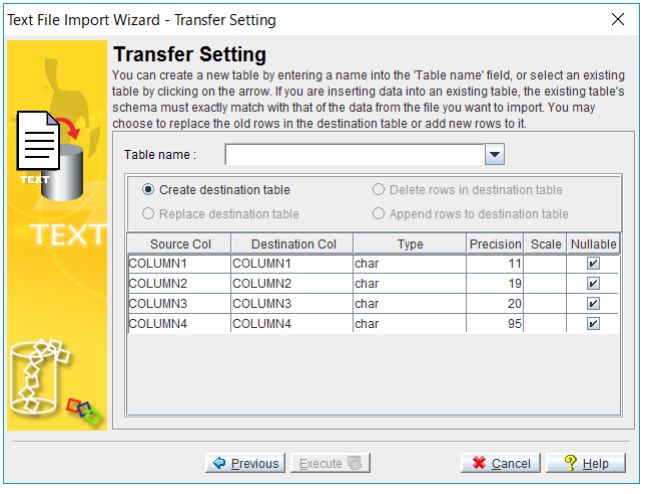
14. Select the destination database to import data to from the **Database** menu.
15. Enter a user name and password into the appropriate fields.



NOTE *A user must have INSERT privilege to import a text file.*

16. Click the **Next** button. The **Transfer Setting** window appears.

©Copyright 1995-2026 CASEMaker Inc.



17. Enter a new table name into the **Table Name** field, or select a table from the menu. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the select privilege is granted to him. The user with CONNECT authority can select tables on which the select privilege is granted to him. Selecting a table from the menu will allow you to choose to replace the destination table, delete rows in the destination table, or append new rows to the destination table. Click **Execute** to import the text file. A confirmation dialog box appears.

18. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

27.2 Importing Data from XML

XML files may also be imported into the database. XML tags may first be defined in a *Document Type Definition* (DTD) file before being imported into the database. Furthermore, the DTD may define the schema in a way that is acceptable to the database.

It is important to consider the structure of the XML file you wish to import. To ensure that the structure of the XML file and associated DTD have compatible structure, examine the structure of XML files produced by the JData Transfer Tool: Export to XML File wizard. Examples may be found in section 27.6, *Exporting Data to XML*. Files produced using the Export to XML wizard always can be imported, however, the extent to which a table's schema is reproduced varies. The setting that influences table schema the most is the Column as Element / Attribute setting.

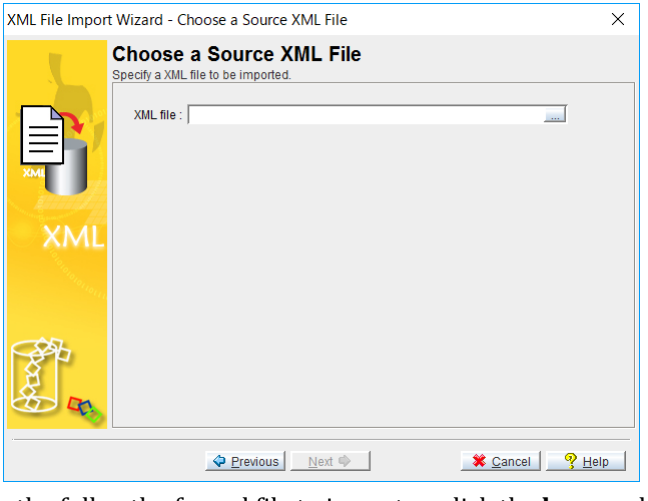
- Column as Element: Stores data items in elements. If table schema information exists as element attributes (data type, column name, length, etc.) in the DTD, then columns will be created with names and of the appropriate data type and length. Columns are child elements, and the table is represented as the parent element. File objects must be referenced as entities in the DTD file if Column as Element is chosen.
- Column as Attribute: Stores data in an attribute of an element. Each element is a record. If column names are represented as attributes of the root element (the table) in the DTD, and each tag in the XML file represents one record, then Column as Attribute should be chosen.

➲ Importing data from an XML file:

1. Open the **JData Transfer Tool**.
2. Select **Import XML File** from the main console or the **Transfer** menu. The **Welcome to Import from XML File Wizard** window starts.

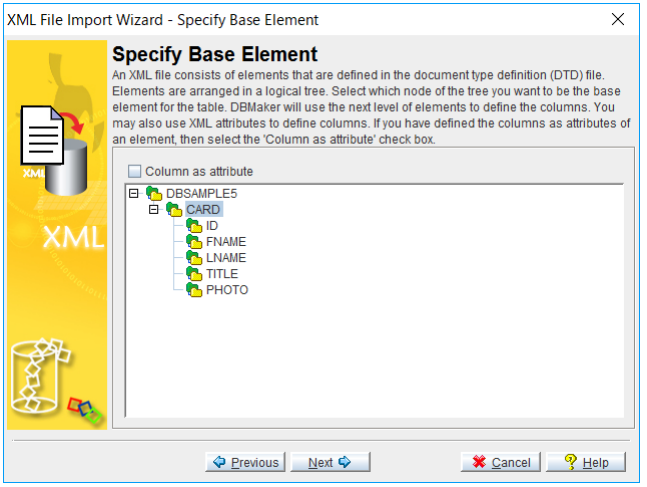
©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Next** button. The **Choose a Source XML File** window appears.

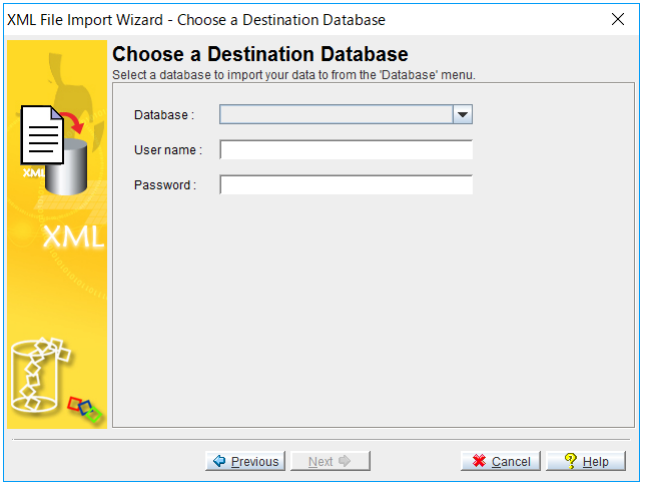


4. Enter the full path of a xml file to import or click the **browse** button to search for a xml file.
5. Click the **Next** button. If the XML file has a structure acceptable to DBMaker's parser, the **Specify Base Element** window appears.

©Copyright 1995-2026 CASEMaker Inc.



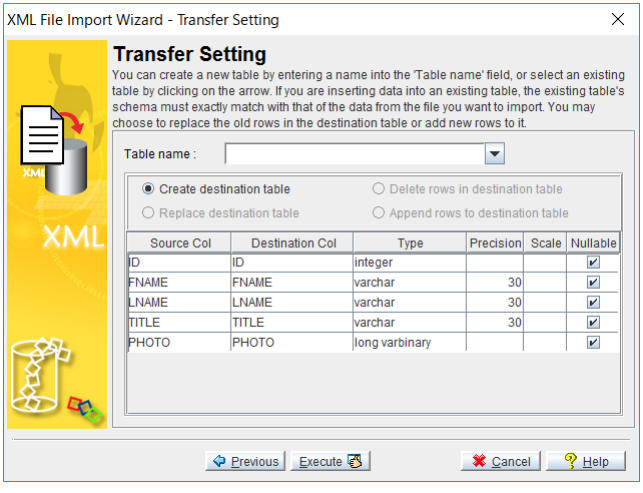
6. The nodes of the tree structure represent the elements in the XML file. Click the nodes on the tree until they are fully expanded. Select a parent element to be the table name. The child elements will become the columns of the table. Check **Column as attribute** if appropriate.
7. Click the **Next** button. The **Choose a Destination Data Source** window appears.



8. Select the destination database to import data to from the **Database** menu.

©Copyright 1995-2026 CASEMaker Inc.

9. Enter a user name and password into the appropriate fields.
10. Click the **Next** button. The **Transfer Setting** window appears.



11. Enter a new table name into the **Table Name** field, or select a table from the menu. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the insert privilege is granted to him. The user with CONNECT authority can select tables on which the insert privilege is granted to him. Selecting a table from the menu will allow you to choose to replace the destination table, delete rows in the destination table, or append new rows to the destination table. Click **Execute** to import the XML file. A confirmation dialog box appears.
12. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

27.3 Importing Data from ODBC

A large number of software developers have developed applications to be Open Database Connectivity (ODBC) compatible. ODBC is an industry standard for sharing data between diverse data sources. DBMaker can import data from any ODBC compliant data source through the Import from ODBC wizard.

Data may be imported using three methods:

- directly from tables
- writing one or more SQL SELECT statements
- through an XML batch file

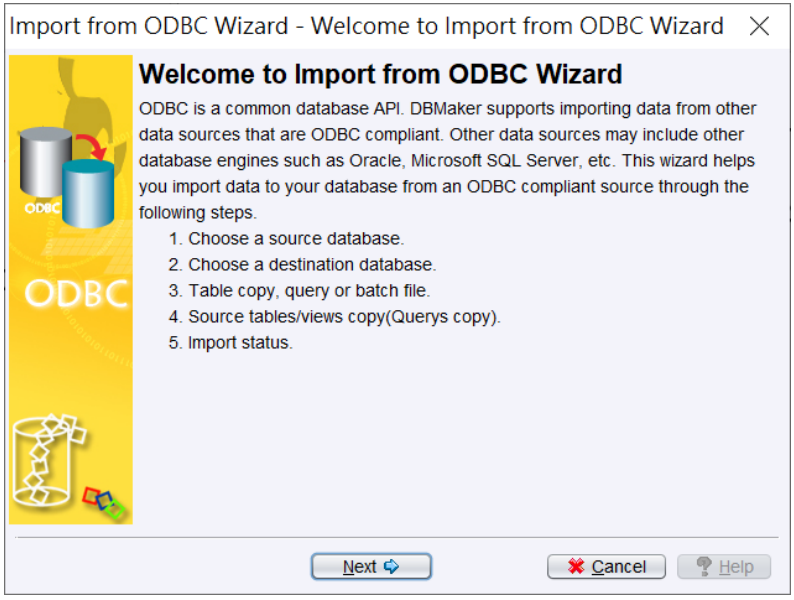
Furthermore, you may specify the mapping of column data through the transformation function. The transformation function supports direct column-to-column mapping or mapping through SQL SELECT and SQL INSERT statements.

When importing data directly from tables or through SQL SELECT statements it is possible to save a 'map' of the data transformation to an XML batch file. The saved XML batch file is a well-formed XML document with a form that can be parsed by the JData Transfer Tool. Batch files may be used to import table schema from a data source to multiple DBMaker databases.

➲ Importing data from an ODBC database:

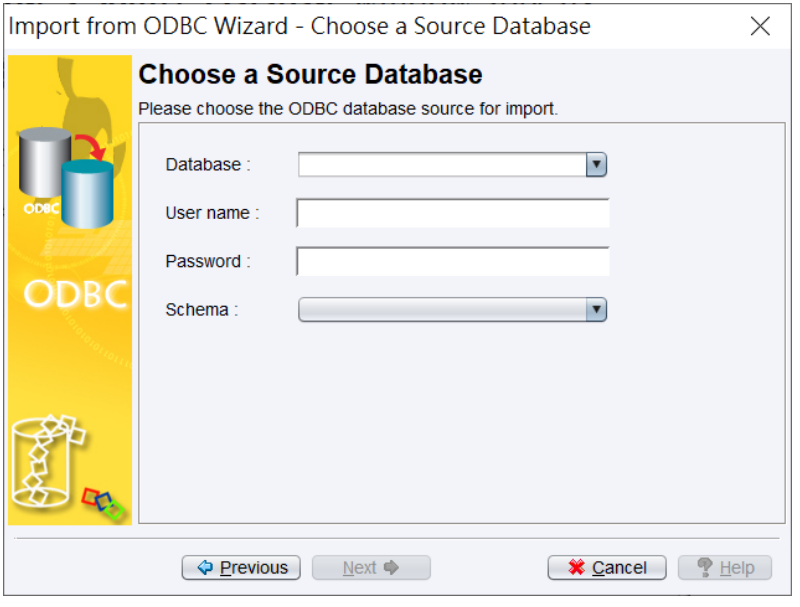
1. Open the **JData Transfer** Tool.
2. Select **Import from ODBC** from the main console or the **Transfer** menu. The **Welcome to Import from ODBC Wizard** window appears. This window displays a summary of the wizard's steps.

©Copyright 1995-2026 CASEMaker Inc.



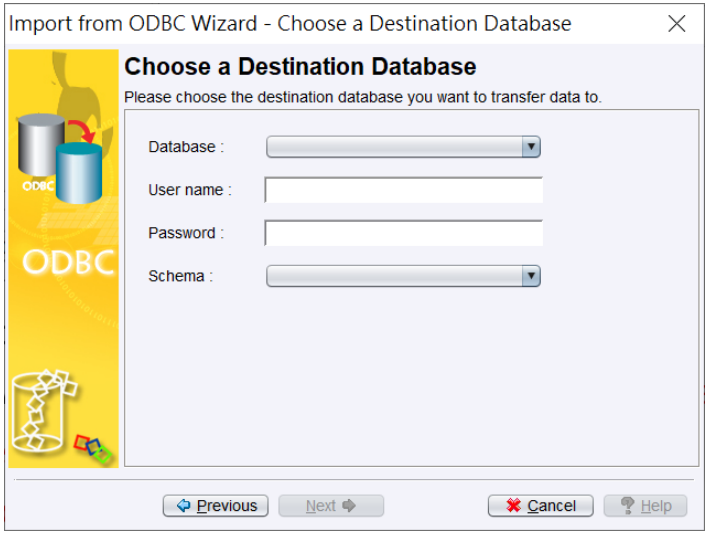
3. Click the **Next** button. The **Choose a Source Database** window appears.

©Copyright 1995-2026 CASEMaker Inc.

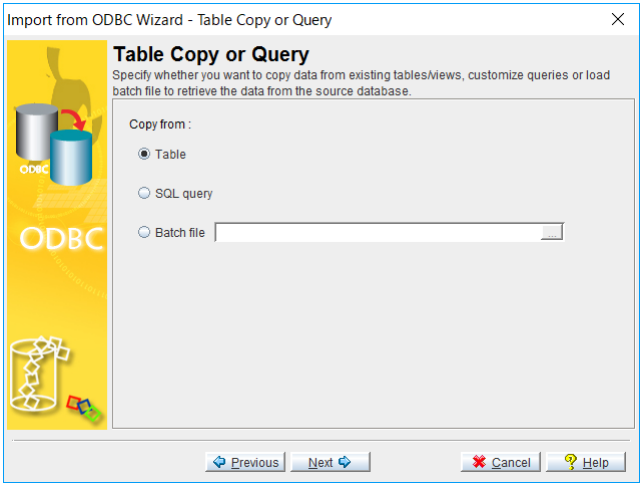


4. Select the source database to export data from in the **Database** menu.
5. Enter a user name and password into the appropriate fields. And select a schema, the schema means the owner of the table. Users can only import tables from this table owner during this importing.
6. Click the **Next** button. The **Choose a Destination Data Source** window appears.

©Copyright 1995-2026 CASEMaker Inc.



7. Select the destination database to import data to from the **Database** menu.
8. Enter a user name and password into the appropriate fields. Select a schema, schema means the user want to create table in the destination database.
9. Click the **Next** button. The **Table Copy or Query** window appears.



©Copyright 1995-2026 CASEMaker Inc.

10. Select from one of the three methods for data transfer:
    - To import data from a list of tables, select **Table**.
    - To import data using a series of SQL SELECT statements, select **SQL query**.
    - To import data through an XML file, select **Batch file**

Importing ODBC Data from Tables

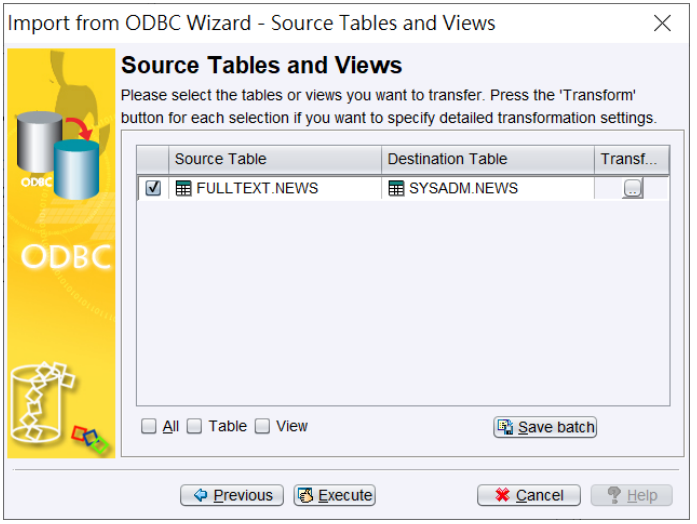
Select tables in the source database to import ODBC data. The **Import from ODBC Wizard** provides an intuitive graphical interface for selecting tables and setting how data is transformed during the import process.

➲ Importing ODBC data from a list of tables:

1. In the Import from ODBC wizard, select a source and destination database, and then select **Table** from the **Table Copy or Query** window.
2. Click the **Next** button. The **Source Tables and Views** window appears.

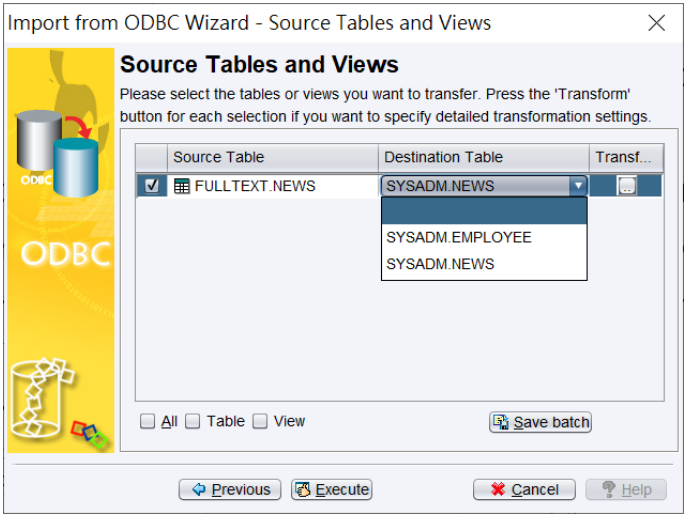
©Copyright 1995-2026 CASEMaker Inc.

3. All tables from the source database belong to the schema (table owner) will appear in the **Source Table** column. Check the box to the left of each table to import.

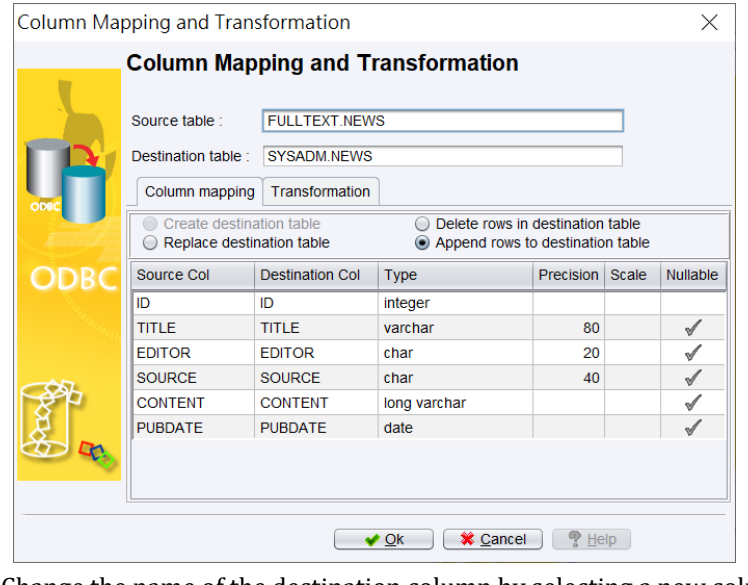


4. For each source table or view selected, click the **Destination Table** field. If desired, change the name of the destination table by selecting a new table from the destination table combo box. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the insert privilege is granted to him. The user with CONNECT authority can select tables on which the insert privilege is granted to him.

©Copyright 1995-2026 CASEMaker Inc.

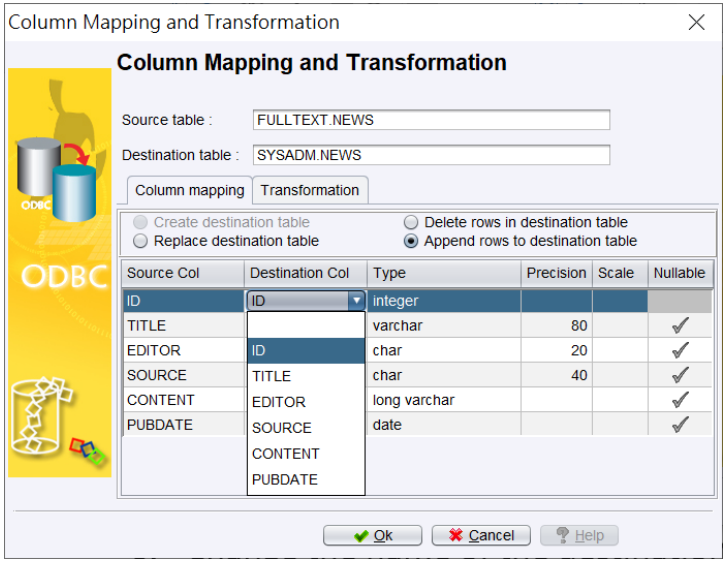


5. You may modify column mapping or the result set to import by clicking on the **Transform** button of the corresponding source and destination table.

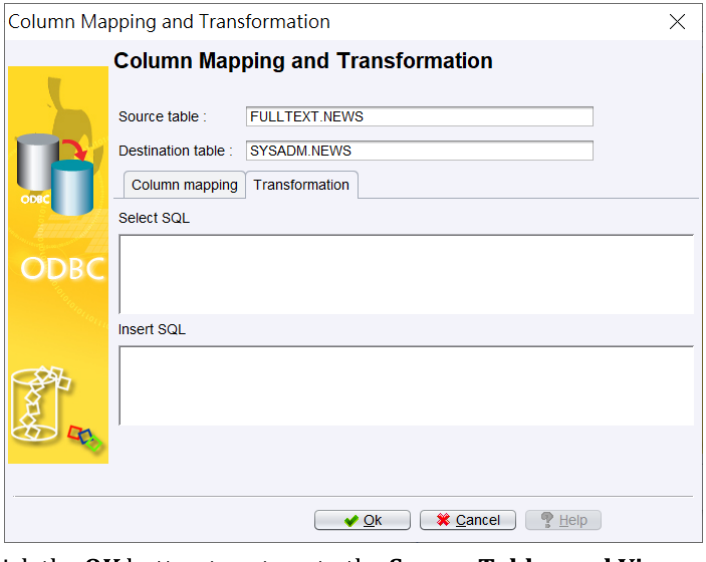


6. Change the name of the destination column by selecting a new column from the menu or entering a new name.

©Copyright 1995-2026 CASEMaker Inc.



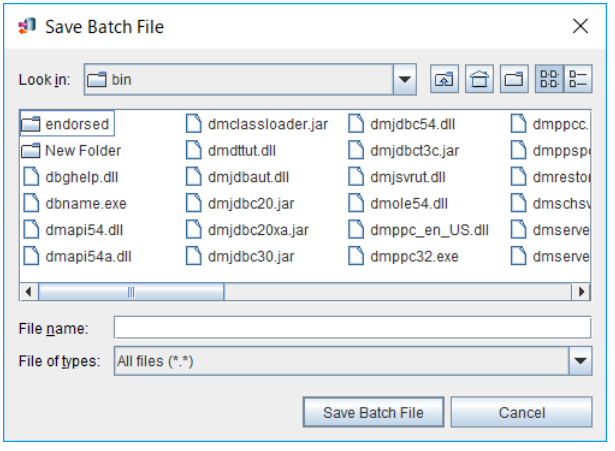
7. Click the **Transformation** tab to specify constraints on the result set. Enter a valid SQL SELECT statement into the **Select SQL** field, and a valid SQL INSERT statement into the **Insert SQL** field.



8. Click the **OK** button to return to the **Source Tables and Views** window.

©Copyright 1995-2026 CASEMaker Inc.

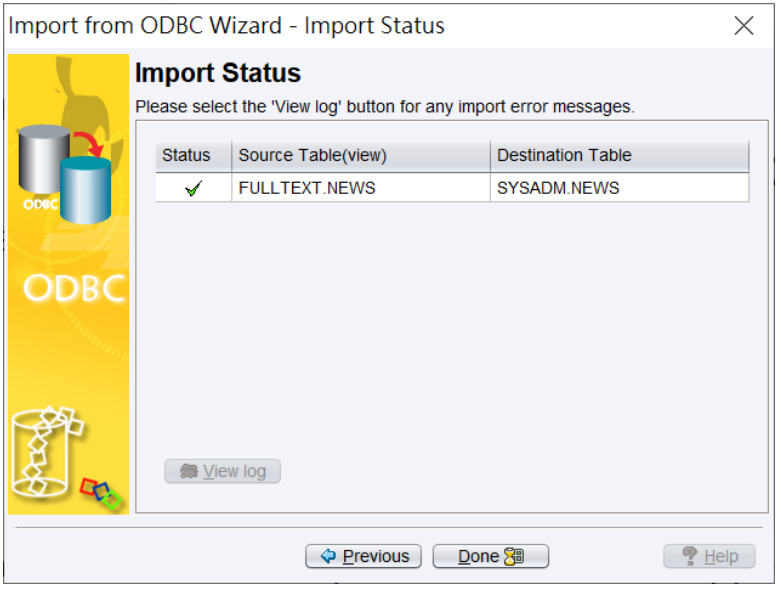
9. You may also choose to save the map of the import ODBC schema to an XML file by clicking **Save batch**. The **Save Batch File** will open.



10. Select or create an XML file to save the imported ODBC map schema to. Click **Save Batch File** to create the XML file. The **Source Tables and Views** window will reappear.

11. Click the **Execute** button to import the source data. The Import Status window appears.

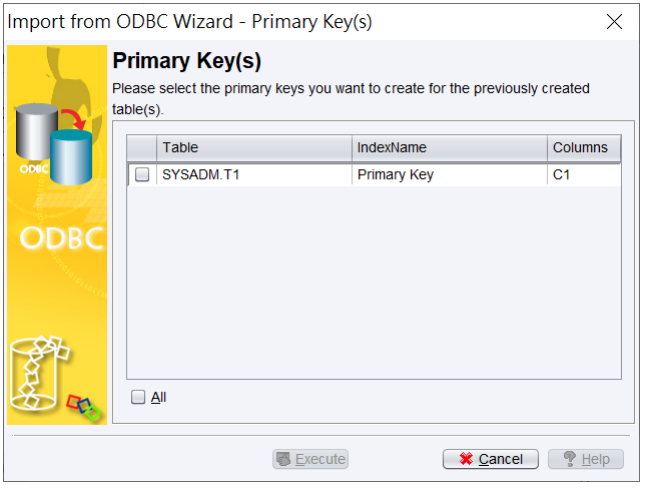
©Copyright 1995-2026 CASEMaker Inc.



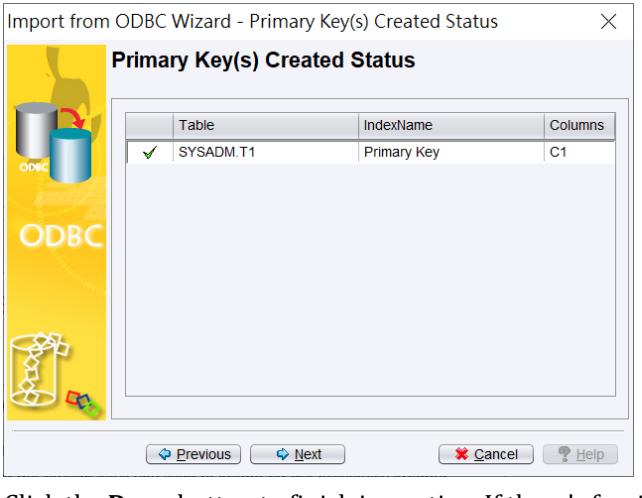
12. If errors appear, click the **View log** button and scroll to the bottom to see the error message. If no errors occurred, click the **Done** button or **Next** button(if the destination table is new created and there's primary/foreign keys on the source table).

13. Click the **Next** button., the **Primary Key(s)** window will appear. Select the primary key to create it on the destination table or click **Cancel** button to finish importing.

©Copyright 1995-2026 CASEMaker Inc.



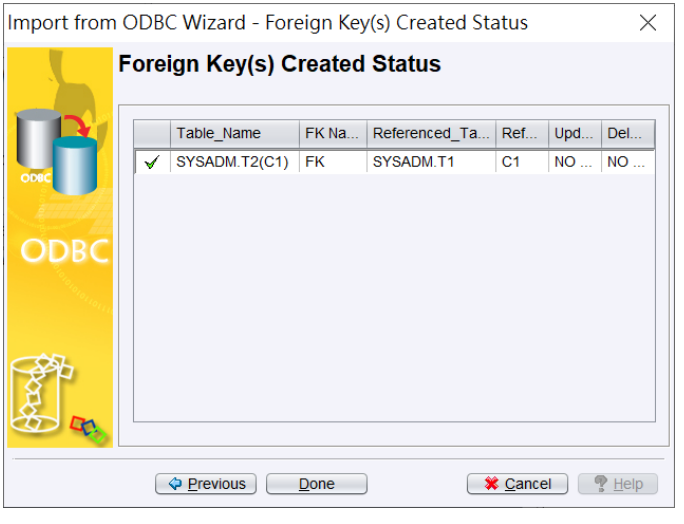
14. Click the **Execute** button., the **Primary Key(s) Created Status** will appear.



15. Click the **Done** button to finish importing. If there's foreign key on the source table, the button **Done** will become button **Next**.
16. Click the **Next** button, the **Foreign Key(s)** window will appear. Select the foreign key to create it on the destination table or click **Cancel** button to finish importing.

©Copyright 1995-2026 CASEMaker Inc.

17. Click the **Execute** button , the **Foreign Key(s)** created status will appear.



18. Click the **Done** button to finish importing.

Importing ODBC Data Using SQL SELECT Statements

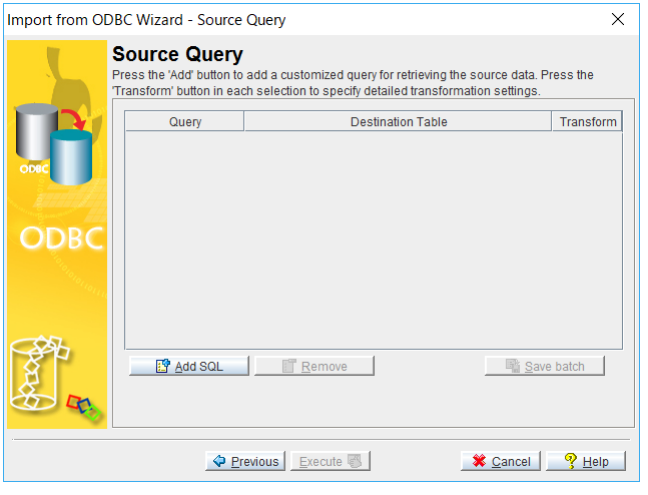
Data may also be imported from ODBC sources by creating a series of SQL

©Copyright 1995-2026 CASEMaker Inc.

SELECT statements. If you have knowledge of the schema of tables you want to import, this may be a faster option.

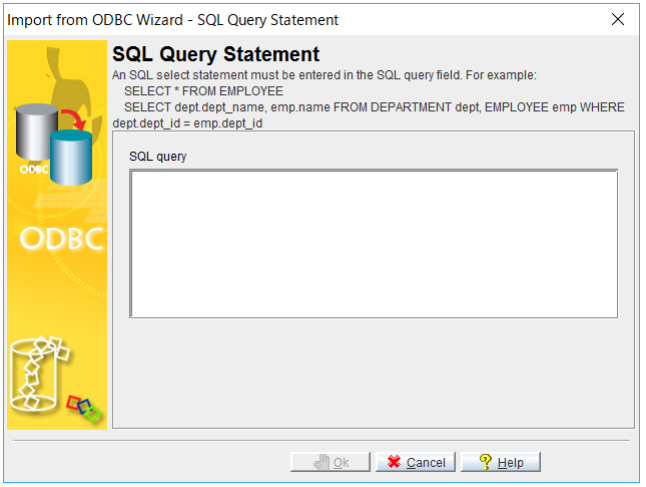
➲ Importing ODBC data using a series of SQL SELECT statements:

1. In the Import from ODBC wizard, select a source and destination database, and then select **Table** from the **Table Copy or Query** window.
2. Click the **Next** button. The **Source Query** window appears.

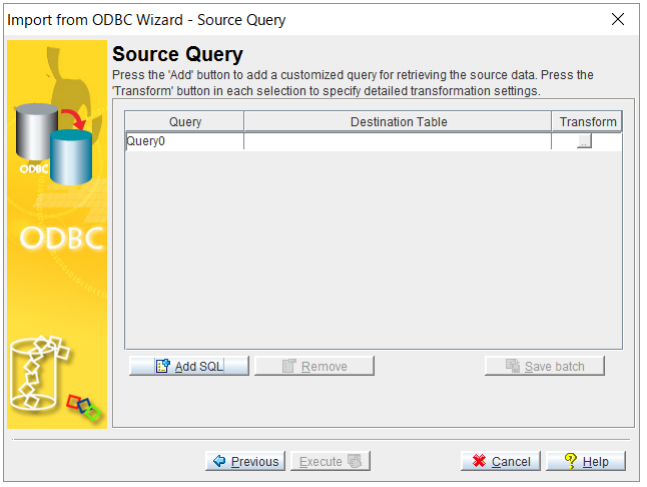


3. Click the **Add SQL** button. The **SQL Query Statement** window appears.

©Copyright 1995-2026 CASEMaker Inc.



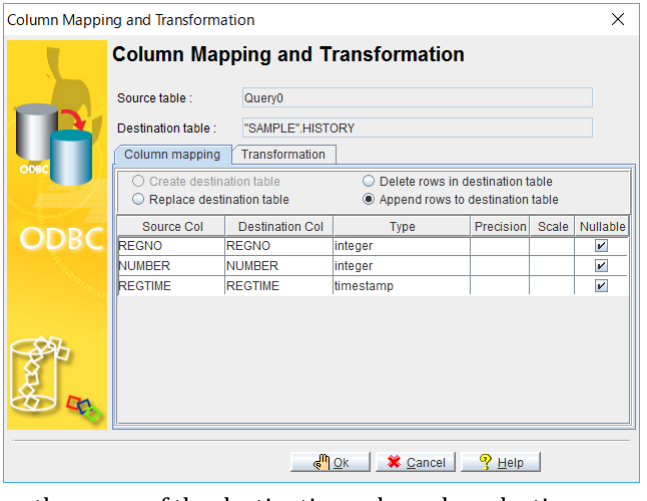
4. Enter a valid SQL SELECT statement into the **SQL Query** field.
5. Click the **OK** button. The **Source Query** window reappears.



6. Select or create a destination table from the **Destination Table** column.The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the insert privilege is granted to him. The user with CONNECT authority can select tables on which the insert privilege is granted to him.

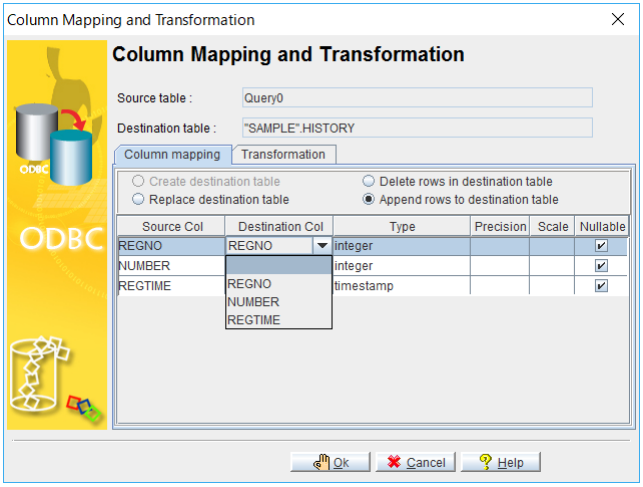
©Copyright 1995-2026 CASEMaker Inc.

7. You may add more SQL query statements by clicking the **Add SQL** button, or modify the mapping of source and destination columns by clicking the **Transform** button.

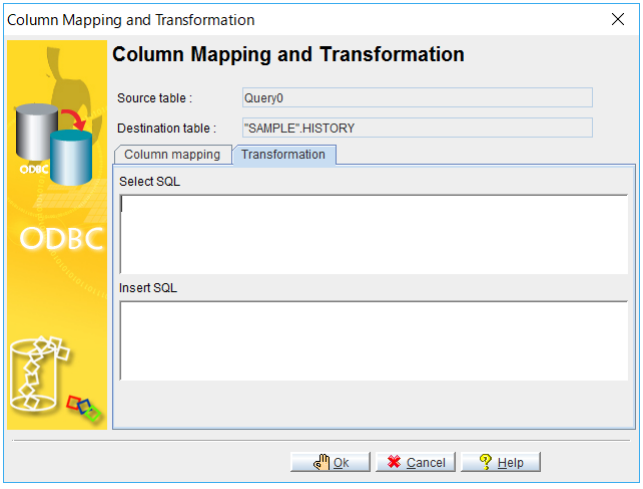


8. Change the name of the destination column by selecting a new column from the menu or entering a new name.

©Copyright 1995-2026 CASEMaker Inc.

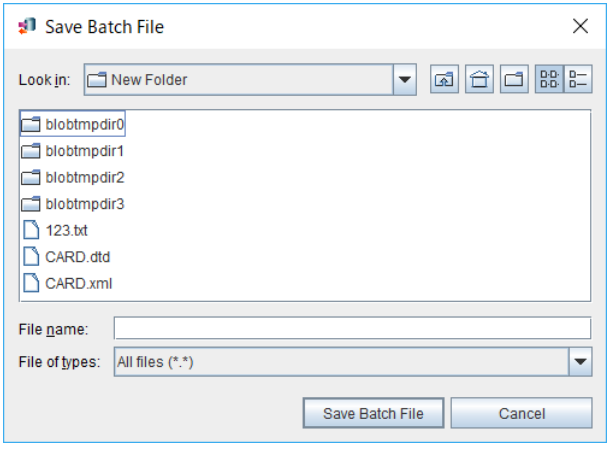


9. Click the **Transformation** tab to specify constraints on the result set. Enter a valid SQL SELECT statement into the **Select SQL** field, and a valid SQL INSERT statement into the **Insert SQL** field.

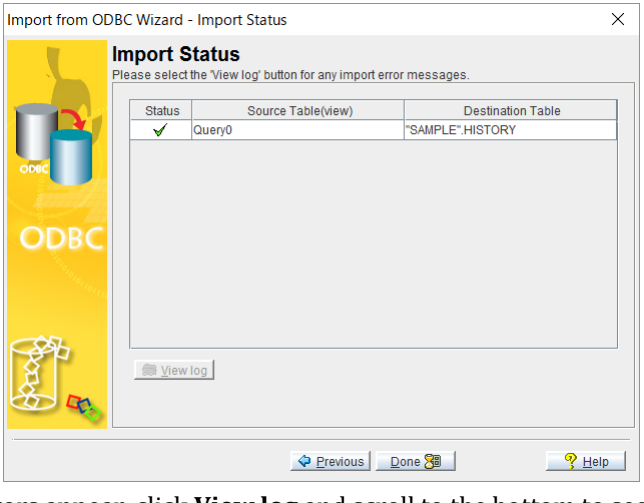


10. Click the **OK** button to return to the **Source Tables and Views** window.
11. You may also choose to save the map of the imported ODBC schema to an XML file by clicking the **Save batch** button. The **Save Batch File** will open.

©Copyright 1995-2026 CASEMaker Inc.



12. Select or create an XML file to save the import ODBC map schema to. Click the **Save Batch File** button to create the XML file. The **Source Tables and Views** window reappears.
13. Click the **Execute** button to import the source data. The Import Status window appears.



14. If errors appear, click **View log** and scroll to the bottom to see the error message. If no errors occurred, click the **Done** button.

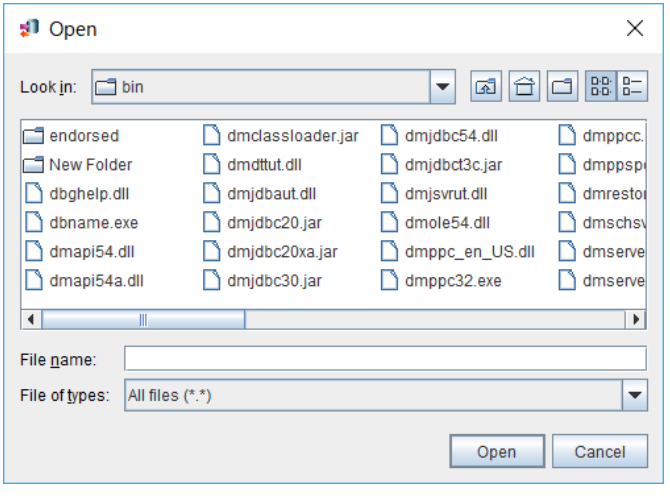
©Copyright 1995-2026 CASEMaker Inc.

Importing ODBC Data through an XML Batch File

It is also possible to use an XML Batch file to specify which tables are to be imported. Users have the option to create an XML batch file when importing ODBC data from tables or with SQL Select statements. Batch files may be used to import table schema from a data source to multiple DBMaker databases

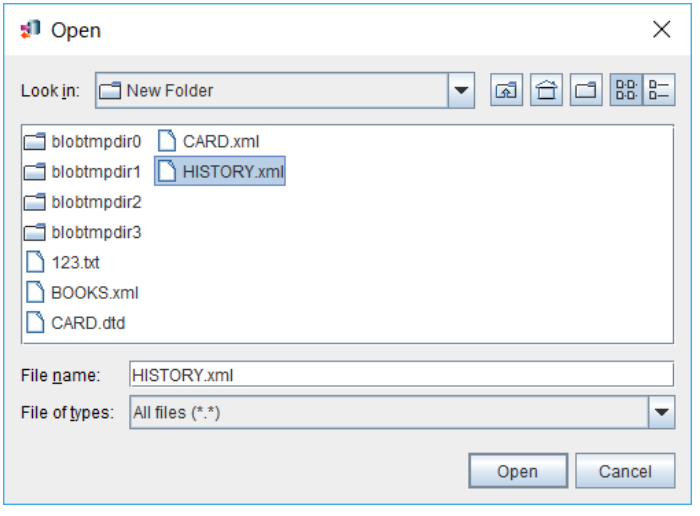
➲ Importing data through an XML batch file:

1. In the Import from ODBC wizard, select a source and destination database, and then select **Batch file** from the **Table Copy or Query** window. The **Open** window appears.

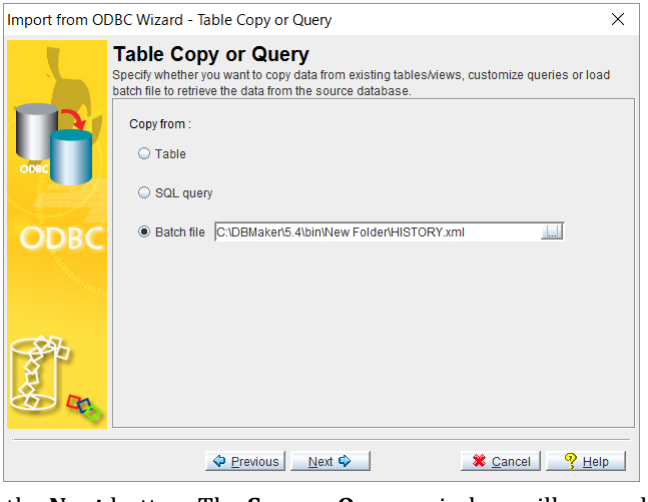


2. Select an XML file from which to import the ODBC map schema.

©Copyright 1995-2026 CASEMaker Inc.



3. Click the **Open** button. The **Table Copy or Query** window reappears.

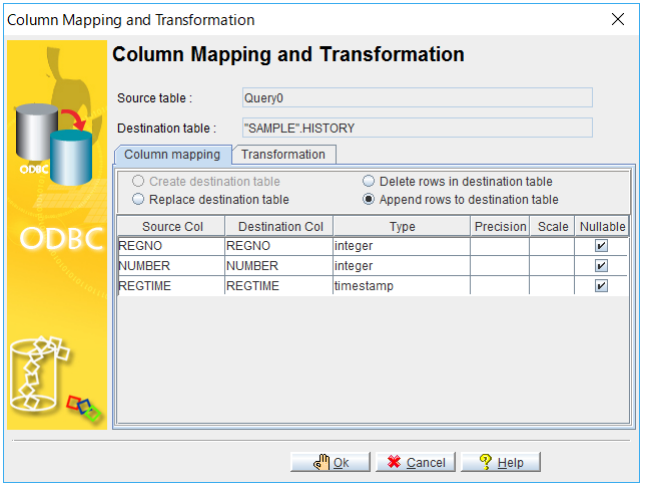


4. Click the **Next** button. The **Source Query** window will open, displaying a mapping schema according to the XML file.

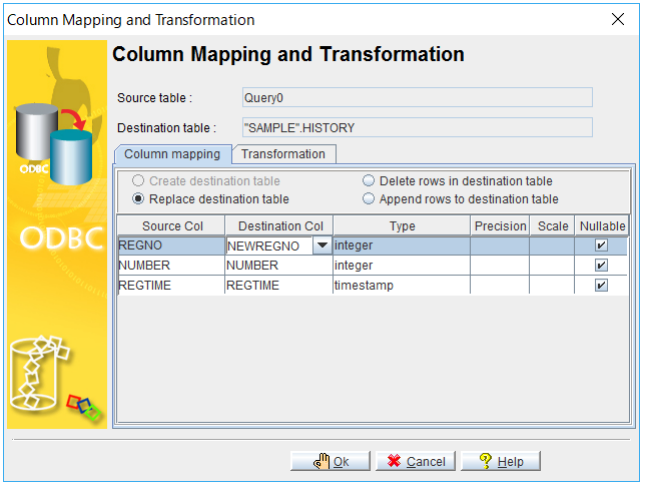
©Copyright 1995-2026 CASEMaker Inc.

5. Click the **Destination Table** field. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the insert privilege is granted to him. The user with CONNECT authority can select tables on which the insert privilege is granted to him.If desired, change the name of the destination table by selecting a new table from the combo box or entering a new name.
6. You may modify column mapping or the result set to import by clicking on the **Transform** button of the corresponding source and destination table.

©Copyright 1995-2026 CASEMaker Inc.

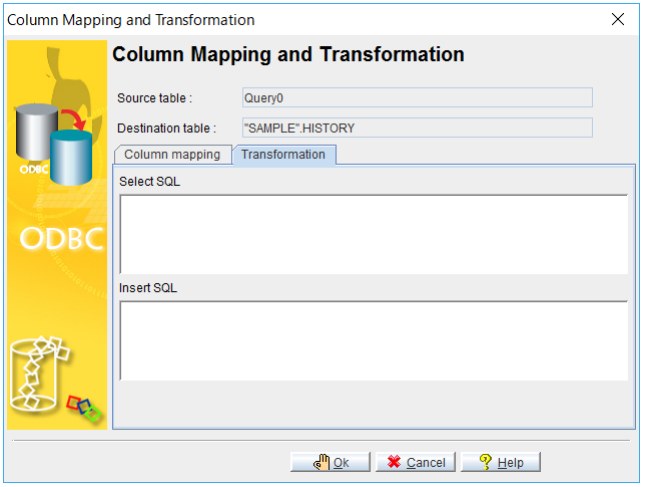


7. Change the name of the destination column by selecting a new column from the menu or entering a new name.



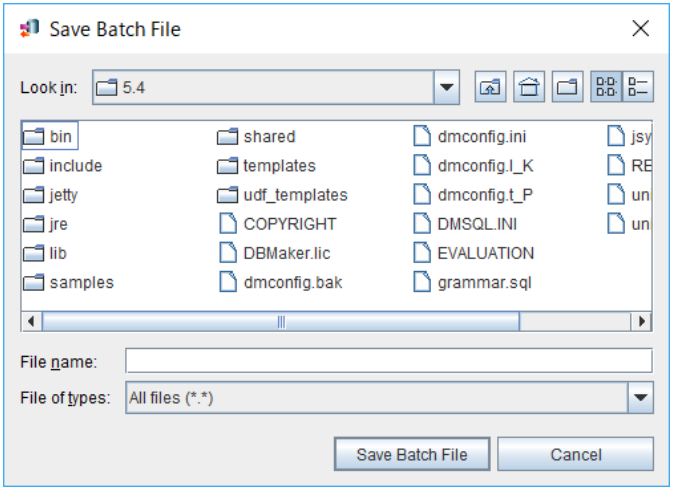
8. Click the **Transformation** tab to specify constraints on the result set. Enter a valid SQL SELECT statement into the **Select SQL** field, and a valid SQL INSERT statement into the **Insert SQL** field.

©Copyright 1995-2026 CASEMaker Inc.



9. Click the **OK** button to return to the **Source Tables and Views** window.

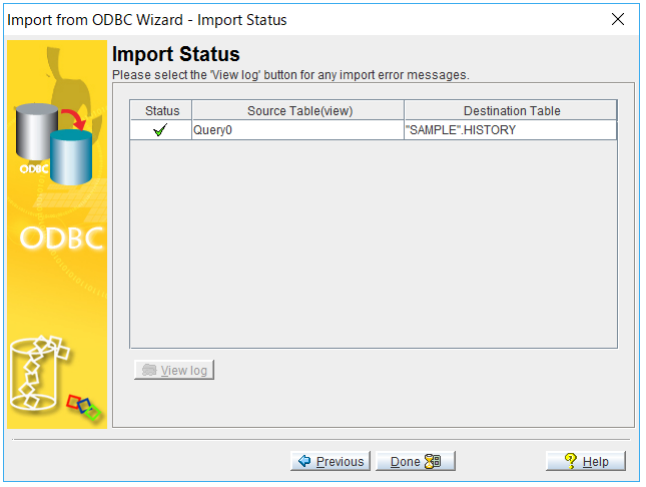
10. You may also choose to save the map of the import ODBC schema to an XML file by clicking the **Save batch** button. The **Save Batch File** will open.



11. Select or create an XML file to save the imported ODBC map schema to. Click the **Save Batch File** button to create the XML file. The **Source Tables and Views** window will reappear.

©Copyright 1995-2026 CASEMaker Inc.

12. Click the **Execute** button to import the source data. The Import Status window will appear.



13. If errors appear, click the **View log** button and scroll to the bottom to see the error message. If no errors occurred, click the **Done** button.

©Copyright 1995-2026 CASEMaker Inc.

27.4 Importing Data from JSON

JSON files may also be imported into the database. Json is an open standard file format and data interchange format that uses readable text to store and transmit data objects consisting of attribute-value pairs and arrays.

➲ Importing data from a JSON file:

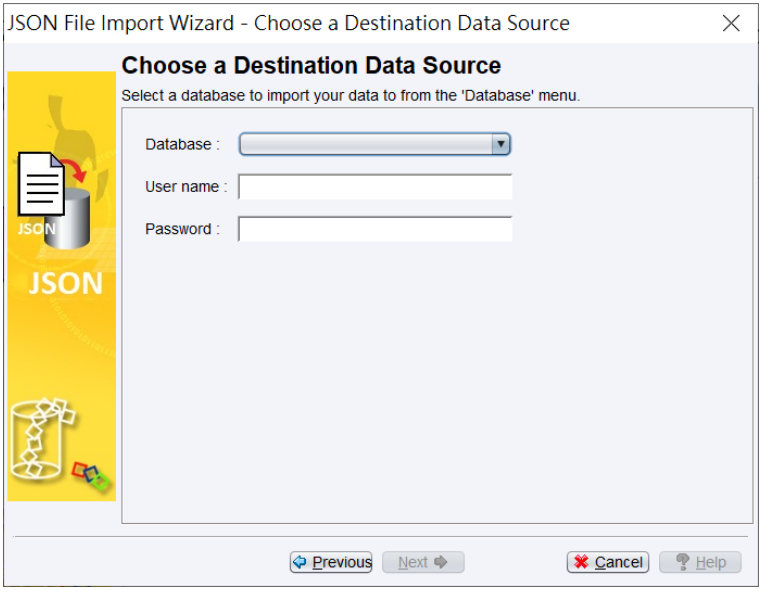
1. Open the **JData Transfer** Tool.
2. Select **Import JSON File** from the main console or the **Transfer** menu. The **Welcome to Import from JSON File** Wizard window appears.

3. Click **Next**. The **Choose a Source JSON** File window appears.

©Copyright 1995-2026 CASEMaker Inc.

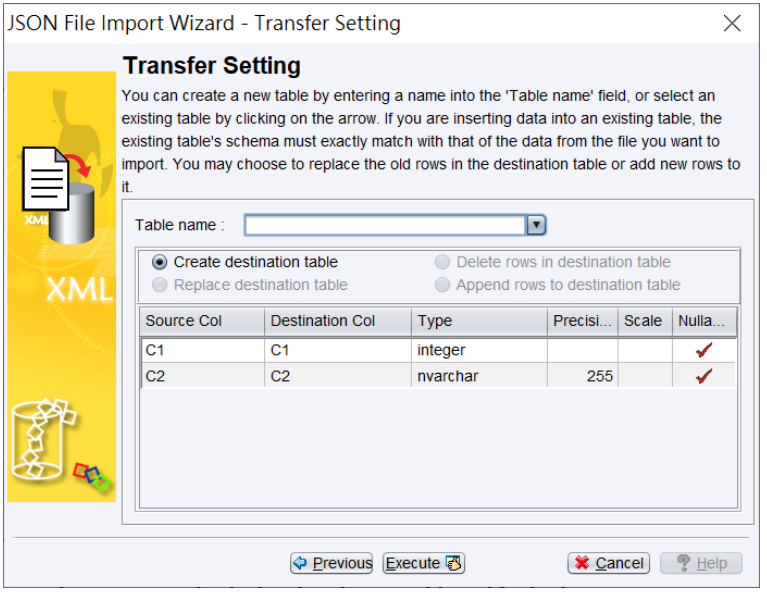
4. Enter the full path of a json file to import or click the browse button to search for a json file.
5. Click **Next**. The **Choose a Destination Data Source** window appears.

©Copyright 1995-2026 CASEMaker Inc.



6. Select the destination database to import data to from the **Database** menu.
7. Enter a user name and password into the appropriate fields.
8. Click **Next**. The **Transfer Setting** window appears.

©Copyright 1995-2026 CASEMaker Inc.



9. Enter a new table name into the Table Name field, or select a table from the menu. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the insert privilege is granted to him. The user with CONNECT authority can select tables on which the insert privilege is granted to him. Selecting a table from the menu will allow you to choose to replace the destination table, delete rows in the destination table, or append new rows to the destination table. Click Execute to import the JSON file. A confirmation dialog box appears.

10. Click **OK**.

©Copyright 1995-2026 CASEMaker Inc.

27.5 Exporting Data to Text

Data may be exported from the database to form a structured text file. This section describes the different text file formats that the JData Transfer tool can produce. Consider the following settings affecting the format of the text file before exporting data.

- Row Delimiter: Determines the type of character that signifies a break between the rows of a table. Possible characters: {CR/LF} (Carriage return / line feed. In Windows applications, a new line in the text is normally stored as a pair of CR LF characters. In UNIX applications, a new line is normally stored as a LF character. Some applications use only a CR character to store a new line), {CR}, {semicolon} (;), {comma} (,),{tab}, {vertical bar} (|), (semicolon}{LF}, or {comma}{LF}.
- Column Delimiter: Determines the type of character that signifies a break between columns in each row. Possible characters: semicolon, comma, or vertical bar.
- Text Qualifier: Determines how each tuple of any data type except BINARY, LONGVARBINARY, or numeric data types (integer, smallint, serial, decimal, double, float) is enclosed. Possible values: none, single quote, or double quote.
- Binary Qualifier: Determines how each tuple of BINARY or LONGVARBINARY type data is enclosed. Possible values: none, single quote, or double quote.
- Binary Padding: A character appended to each tuple containing binary data.
- Fixed Field: Instead of using a row delimiter, the text file may be formatted with fixed fields. This means a number of spaces, or fields, defines each column.

©Copyright 1995-2026 CASEMaker Inc.

- Include column name: The first line in a text file may be used to define the column names. The format is *"column1"."table name"."owner name";"column2"."table name"."owner name";* etc. In this case the column delimiter is set to semicolon (;).
- Include table schema: The first line in a text file may be used to define the column schema (or the second line if the first line was used to define column names). The format is *data type(scale,precision);data type(scale,precision)* etc. In this case the column delimiter is set to semicolon (;).
- Use NULL to display null data: Columns that contain no data output "NULL".
- File link name for FILE type data: The file name for system or user file objects is displayed.
- Use escape character "/": This character is used when qualifiers or delimiter characters appear in the data. If the data contains a reserved character, the reserved character will be enclosed by an escape character (/).
- Use temp files to store LONGVARBINARY or LONGVARCHAR type data column content: BLOB data is stored as a separate, linked file (as a file object), and the name of the file containing the BLOB is displayed.

Data can be exported by selecting individual columns from a table, or through a valid SQL SELECT statement. For more information on SQL syntax, refer to the *SQL Command and Function Reference*.

Example:

The following is text produced by the JData Transfer Tool Export to Text wizard. The table's name and schema will be displayed on the first two lines. The data has been exported with fixed fields to delimit columns and CR to delimit rows. File link names appear where LONGVARCHAR data once stored as BLOBs in the table have been exported as file objects.

©Copyright 1995-2026 CASEMaker Inc.

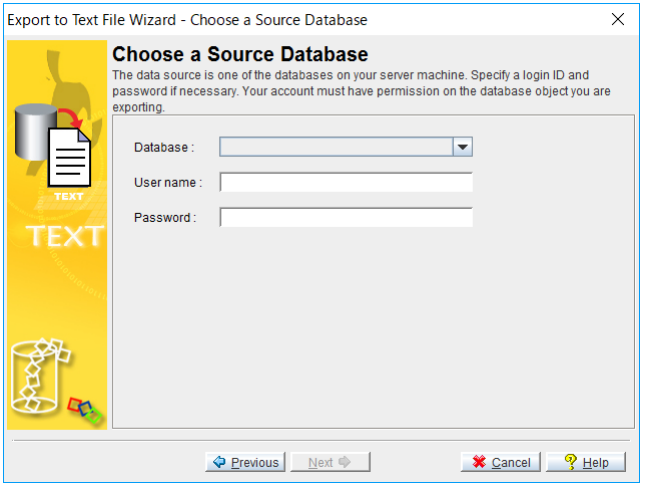
```
"SYSADM"."SUPPORTQUERIES"."LOGINID"   "SYSADM"."SUPPORTQUERIES"."REQUESTTIME"
"SYSADM"."SUPPORTQUERIES"."ATTACHMENT"   "SYSADM"."SUPPORTQUERIES"."DECIMAL_C"
SQL_CHAR(10)  SQL_TIMESTAMP                SQL_LONGVARCHAR    SQL_DECIMAL(10, 3)
A_HOWARD      2001-09-09 12:47:05.000      C:\WEBDB\FO\ZZ000000.GIF       10.250
A_HOWARD      2001-09-22 10:14:21.000      C:\WEBDB\FO\ZZ000001.GIF       13.550
A_HOWARD      2001-10-04 16:22:06.000      C:\WEBDB\FO\ZZ000002.GIF       27.333
A_HOWARD      2001-10-09 17:44:56.000      C:\WEBDB\FO\ZZ000003.GIF       16.140
A_HOWARD      2001-10-12 09:12:38.000      C:\WEBDB\FO\ZZ000004.GIF       88.847
A_HOWARD      2001-10-31 23:16:11.000      C:\WEBDB\FO\ZZ000005.JPG       841.336
```

➲ Exporting a table to a text file

1. Open the JData Transfer Tool.
2. Select **Export to Text** from the main console or the **Transfer** menu. The **Welcome to the Export toText File Wizard** window will appear, displaying a summary of the steps to be taken in the wizard.

3. Click the **Next** button. The **Choose a Source Database** window will open.

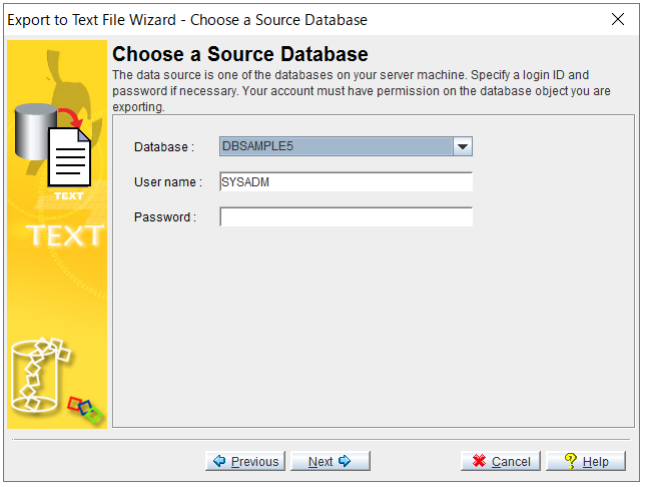
©Copyright 1995-2026 CASEMaker Inc.



NOTE *The user must have CONNECT or higher authority to login.*

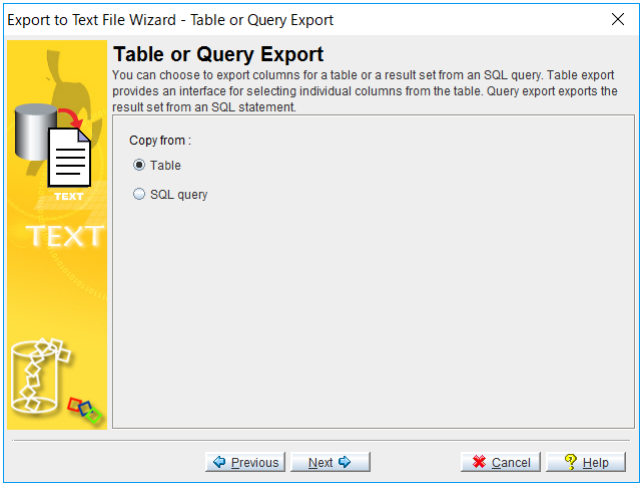
4. Select the database to export data from in the **Database** menu.
5. Enter a user name and password into the appropriate fields.

NOTE *A user must have SELECT privilege to export a text file.*



6. After you have selected a database, click the **Next** button, the **Table or Query Export** window will open.

©Copyright 1995-2026 CASEMaker Inc.



7. Select Table to export data from a table. Select SQL query to export data from the result set of a SQL SELECT statement.

8. If you selected Table from the **Table or Query Export** window, the **Table Export** window will open. If you selected SQL query, then proceed to step 13.

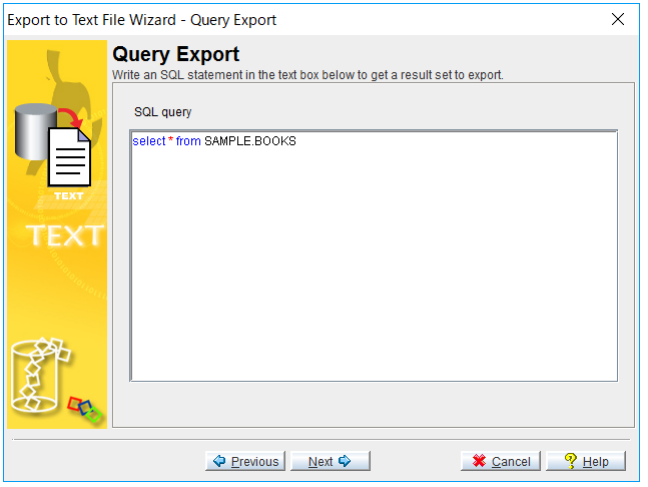
9. Click the **Next** button, the **Table Export** window will open.

10. Select a table to export from the **Table name** menu. The user with DBA

©Copyright 1995-2026 CASEMaker Inc.

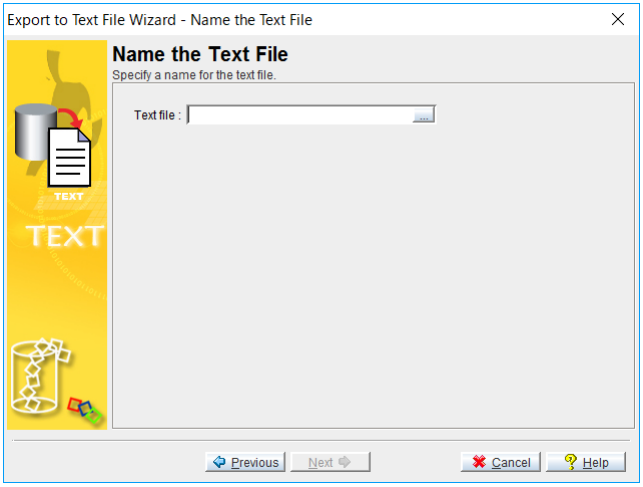
authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the select privilege is granted to him. The user with CONNECT authority can select tables on which the select privilege is granted to him. A list of columns in the table will appear in the Select columns to export field.

11. Select columns by clicking on the column name and clicking the **Add** button, or select all columns by clicking the **Add All** button. Selected column names will appear in the right hand field.
12. Click the **Next** button. The **Name the Text File** window will appear (proceed to step 16).
13. If you selected SQL query from the **Table or Query Export** window, the **Query Export** window will open.
14. Enter a valid SQL select statement into the **SQL query** field.

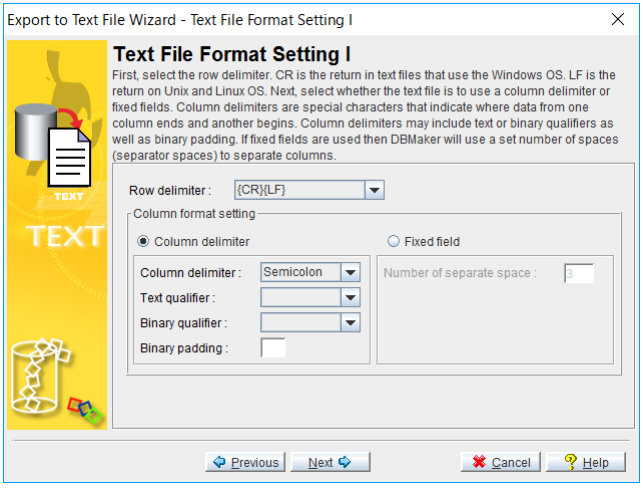


15. Click the **Next** button. The **Name the Text File** window appears.

©Copyright 1995-2026 CASEMaker Inc.

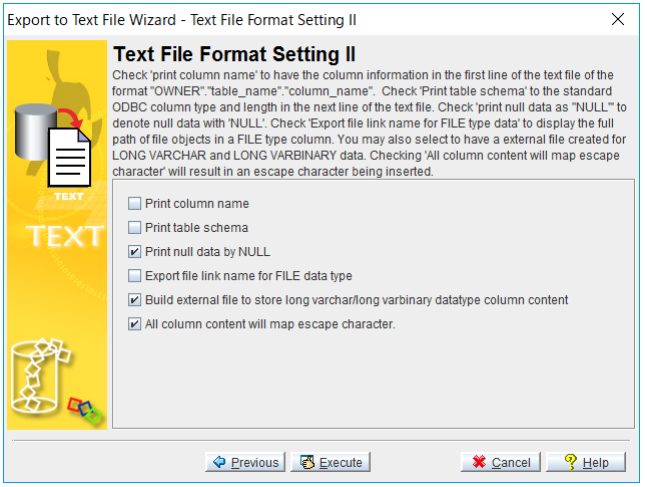


16. Enter the full path of a text file in the Text file field, or select one using the browse button.
17. Click the **Next** button. The **Text File Format Setting I** window appears.



18. Select the appropriate settings for the format of the new text file.
19. Click the **Next** button. The **Text File Format Setting II** window appears.

©Copyright 1995-2026 CASEMaker Inc.



20. Finish selecting the appropriate settings for the format of the text file you are creating. Click the **Execute** button to export the data to the text file. A confirmation dialog box appears.
21. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

27.6 Exporting Data to XML

DBMaker supports the export of data from a table to an XML file. Columns may be stored as individual elements, or as attributes of the table element. When an XML file is created, an associated DTD file is created. The DTD contains information necessary for defining the elements and attributes of the XML file. The structure of both the DTD and XML file will vary depending on whether the columns are stored as attributes or elements.

Consider how the following settings affect the XML file produced by the Export to XML wizard.

- Column as Element: If columns are represented as elements in the resultant files, then schema information will be retained as element attributes (data type, column name, length, etc.) in the DTD. Columns are child elements, and the table is represented as the parent element. If the XML file is later imported back into the database, then the table's structure will be exactly replicated. File objects are referenced as entities in the DTD file if Column as Element is chosen.
- Column as Attribute: Columns are represented as attributes of the table element in the DTD. There is no record of the table's schema. An element in the XML file represents each record.
- Export file link name for FILE type data: Select this original for the full path to reference system and user file objects, otherwise, when unselected file type data is treated as Long Varbinary.
- Translate all tag names to uppercase: All tag names are converted to uppercase characters.
- Build temp file to store LONGVARCHAR and LONGVARBINARY data type column constant: If this option is chosen, BLOB data will be stored in a temporary directory under the directory the XML file resides in. If this option is not selected, BLOB data is stored directly in the XML file.

©Copyright 1995-2026 CASEMaker Inc.

- XML file cannot include DTD file reference: if this option is selected, no DTD is created. No information about the elements will be preserved in the DTD if this option is selected.

Example 1:

Assume the table 'supportqueries' with columns 'LOGINID' CHAR(10); 'REQUEST' SQL_LONGVARCHAR; 'REQUESTTIME' SQL_TIMESTAMP; 'ATTACHMENT' 'SQL_FILE; 'BINARY_C' SQL_BINARY(10); DECIMAL_C SQL_DECIMAL(10, 3). The table has two records. The entire table is exported to an XML file with columns as elements. File link names are exported, temp files are built to store BLOB data, and the DTD is included. The resulting XML file is:

```
<?xml version="1.0" encoding="BIG5"?>
<!DOCTYPE WEBDB SYSTEM "Support.dtd">
<WEBDB>
  <SUPPORTQUERIES>
    <LOGINID>A_HOWARD            </LOGINID>
    <REQUEST>&BLBTMP_TXT0;</REQUEST>
    <REQUESTTIME>2001-09-09 12:47:05.000</REQUESTTIME>
    <ATTACHMENT>&DBMAKER_FO_0;</ATTACHMENT>
    <BINARY_C>10000000000000000000</BINARY_C>
    <DECIMAL_C>10.250</DECIMAL_C>
  </SUPPORTQUERIES>
  <SUPPORTQUERIES>
    <LOGINID>A_HOWARD            </LOGINID>
    <REQUEST>&BLBTMP_TXT1;</REQUEST>
    <REQUESTTIME>2001-09-22 10:14:21.000</REQUESTTIME>
    <ATTACHMENT>&DBMAKER_FO_1;</ATTACHMENT>
    <BINARY_C>20000000000000000000</BINARY_C>
    <DECIMAL_C>13.550</DECIMAL_C>
  </SUPPORTQUERIES>
</WEBDB>
```

The associated DTD is:

```
<!ELEMENT SUPPORTQUERIES (LOGINID, REQUEST, REQUESTTIME, ATTACHMENT, BINARY_C,
DECIMAL_C)>
<!ELEMENT LOGINID (#PCDATA)>
  <!ATTLIST LOGINID
```


©Copyright 1995-2026 CASEMaker Inc.

```
      TYPE CDATA #FIXED "SQL_CHAR"
      NAME CDATA #FIXED "LOGINID"
      LENGTH CDATA #FIXED "20"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ELEMENT REQUEST (#PCDATA)>
  <!ATTLIST REQUEST
      TYPE CDATA #FIXED "SQL_LONGVARCHAR"
      NAME CDATA #FIXED "REQUEST"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ELEMENT REQUESTTIME (#PCDATA)>
  <!ATTLIST REQUESTTIME
      TYPE CDATA #FIXED "SQL_TIMESTAMP"
      NAME CDATA #FIXED "REQUESTTIME"
      STORAGE CDATA #FIXED "29"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ELEMENT ATTACHMENT (#PCDATA)>
  <!ATTLIST ATTACHMENT
      TYPE CDATA #FIXED "SQL_FILE"
      NAME CDATA #FIXED "ATTACHMENT"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ELEMENT BINARY_C (#PCDATA)>
  <!ATTLIST BINARY_C
      TYPE CDATA #FIXED "SQL BINARY"
      NAME CDATA #FIXED "BINARY C"
      LENGTH CDATA #FIXED "10"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ELEMENT DECIMAL_C (#PCDATA)>
  <!ATTLIST DECIMAL_C
      TYPE CDATA #FIXED "SQL_DECIMAL"
```

©Copyright 1995-2026 CASEMaker Inc.

```
      NAME CDATA #FIXED "DECIMAL_C"
      LENGTH CDATA #FIXED "(10, 3)"
      ISNULL (true|false) 'true'
      xml:space (default|preserve) 'preserve'
  >
<!ENTITY BLBTMP_TXT0 SYSTEM "blobtmpdir0\blbtmpf0.txt">
<!ENTITY DBMAKER_FO_0 SYSTEM "C:\DBMAKER\5.4\BIN\WEBDB\FO\ZZ000000.GIF">
<!ENTITY BLBTMP_TXT1 SYSTEM "blobtmpdir0\blbtmpf1.txt">
<!ENTITY DBMAKER_FO_1 SYSTEM "C:\DBMAKER\5.4\BIN\WEBDB\FO\ZZ000001.GIF">
<!ENTITY BLBTMP_TXT2 SYSTEM "blobtmpdir0\blbtmpf2.txt">
<!ELEMENT WEBDB (SUPPORTQUERIES*)>
```

Example 2:

Given the same table as example 1, but with the entire table exported with columns as attributes. The resulting XML file is:

```
<?xml version="1.0" encoding="BIG5"?>
<!DOCTYPE WEBDB SYSTEM "Support.dtd">
<WEBDB>
  <SUPPORTQUERIES
LOGINID="A_HOWARD&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x2
0;&#x20;" REQUESTTIME="2001-09-09 12:47:05.000"
ATTACHMENT="C:\DBMAKER\5.4\BIN\WEBDB\FO\ZZ000000.GIF"
BINARY_C="10000000000000000000" DECIMAL_C="10.250"/>
  <SUPPORTQUERIES
LOGINID="A_HOWARD&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x20;&#x2
0;&#x20;" REQUESTTIME="2001-09-22 10:14:21.000"
ATTACHMENT="C:\DBMAKER\5.4\BIN\WEBDB\FO\ZZ000001.GIF"
BINARY_C="20000000000000000000" DECIMAL_C="13.550"/>
```

The associated DTD is:

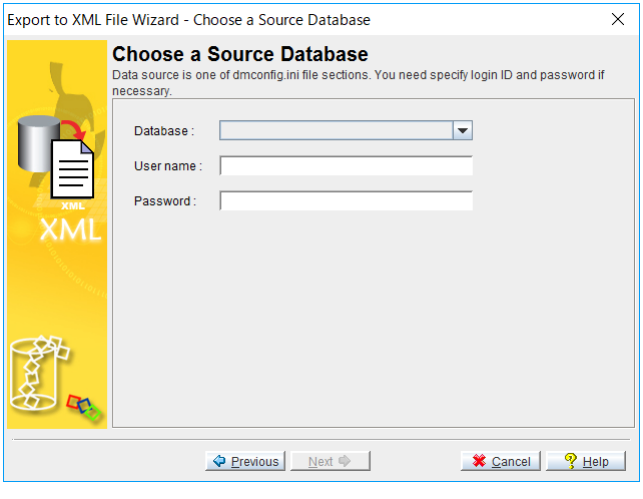
```
<!ELEMENT SUPPORTQUERIES EMPTY>
  <!ATTLIST SUPPORTQUERIES
              LOGINID     CDATA #IMPLIED
          REQUESTTIME     CDATA #IMPLIED
           ATTACHMENT     ENTITY #IMPLIED
             BINARY_C     CDATA #IMPLIED
            DECIMAL_C     CDATA #IMPLIED
  >
<!ELEMENT WEBDB (SUPPORTQUERIES*)>
```


©Copyright 1995-2026 CASEMaker Inc.

➲ Exporting a table to an XML file:

1. Open the JData Transfer Tool.
2. Select **Export to XML** from the main console or the **Transfer** menu. The **Welcome to Export to XML File Wizard** window appears.

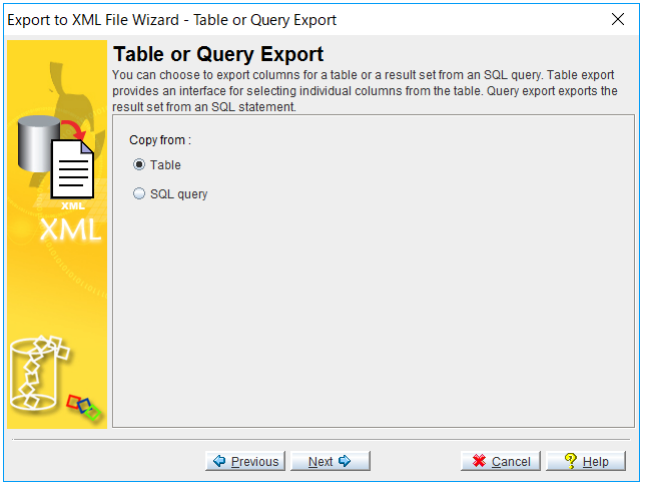
3. Click the **Next** button. The **Choose a Source Database** window appears.



©Copyright 1995-2026 CASEMaker Inc.

NOTE *The user must have CONNECT or higher authority to login.*

4. Select a database from the **Database** menu. Enter a user name and password into the appropriate fields.
5. Click the **Next** button. The **Table or Query Export** window appears.



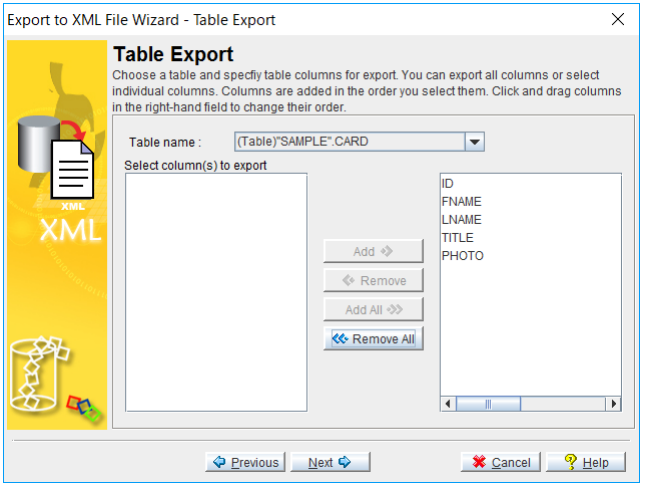
6. If you selected **Table** from the **Table or Query Export** window, the **Table Export** window will open. If you selected **SQL query**, then proceed to step 13.
7. Click the **Next** button, the **Table Export** window appears.

©Copyright 1995-2026 CASEMaker Inc.

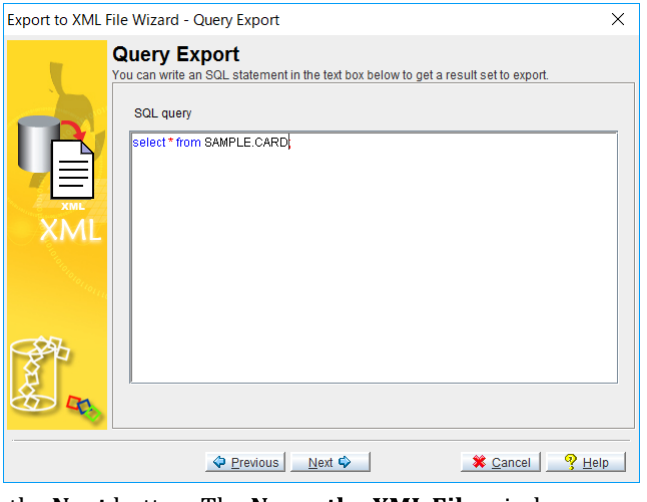
8. Select a table to export from the **Table name** menu. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the select privilege is granted to him. The user with CONNECT authority can select tables on which the select privilege is granted to him. A list of columns in the table will appear in the **Select columns to export** field.

9. Select columns by clicking on the column name and clicking the **Add** button, or select all columns by clicking the **Add All** button. Selected column names will appear in the right hand field.

©Copyright 1995-2026 CASEMaker Inc.

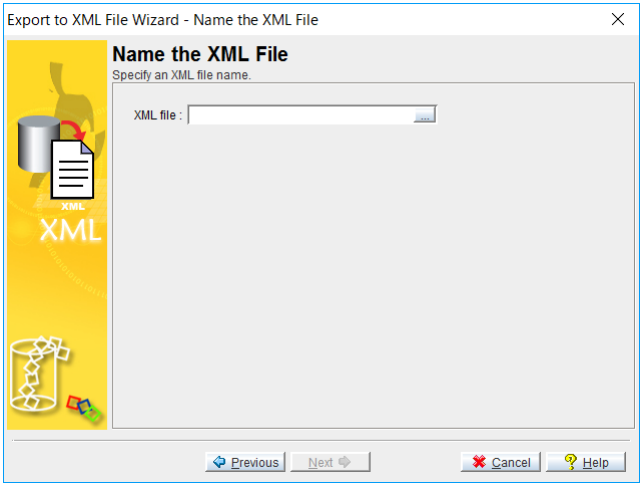


10. Click the **Next** button. The **Name the XML File** window appears. Please proceed to step 16.
11. If you selected **SQL query** from the **Table or Query Export** window, the **Query Export** window will open.
12. Enter a valid SQL select statement into the **SQL query** field.

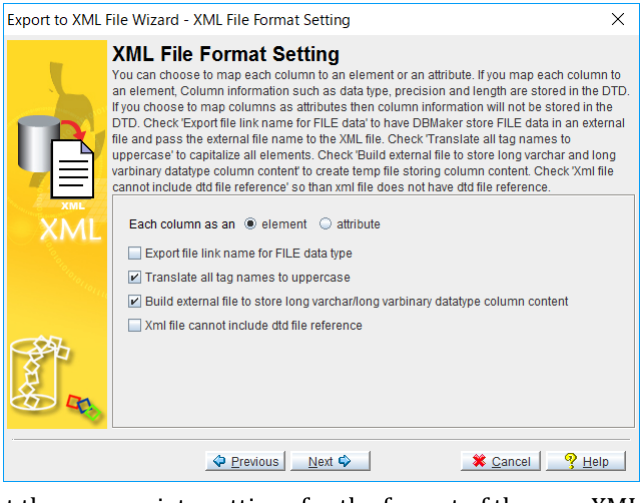


13. Click the **Next** button. The **Name the XML File** window appears.

©Copyright 1995-2026 CASEMaker Inc.

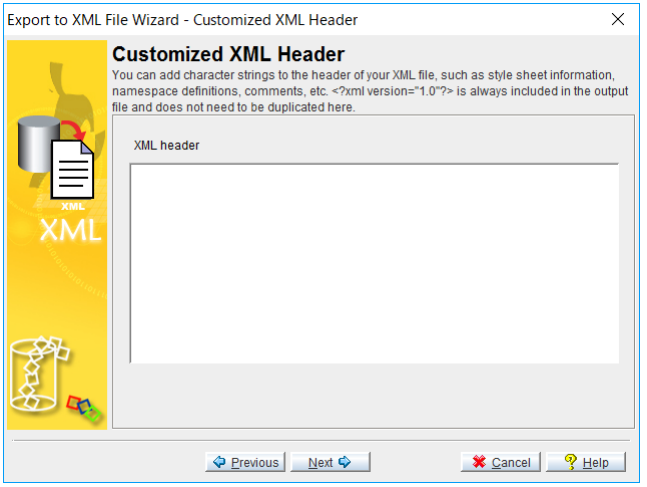


14. Enter the full path of an XML file to export to, or select one by using the browse button.
15. Click the **Next** button. The **XML File Format Setting** window appears.

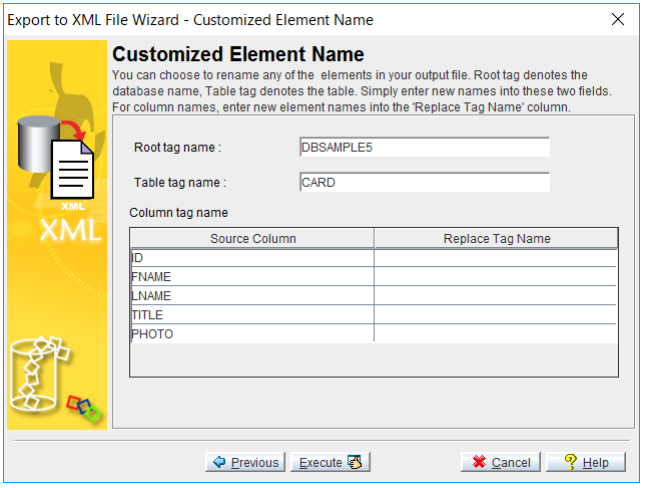


16. Select the appropriate settings for the format of the new XML file.
17. Click the **Next** button. The **Customized XML Header** window appears.

©Copyright 1995-2026 CASEMaker Inc.



18. Enter appropriate information, such as namespace and style sheet definitions, if relevant.
19. Click the **Next** button. The **Customized Element Name** window appears.



20. It is possible to modify the tag definitions. Enter new tag definitions into the **Replace Tag Name** column. The name of the corresponding column will be changed in the resulting XML file.
21. Click the **Execute** button to export the table to the XML file. A

©Copyright 1995-2026 CASEMaker Inc.

confirmation window will appear.

22. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.

27.7 Exporting Data to JSON

Data may be exported from the database to form a structured json file. This section describes how to transfer tables in the database to JSON file.

➲ Exporting a table to json file:

1. Open the **JData Transfer** Tool.
2. Select **Export to JSON** from the main console or the **Transfer** menu. The **Welcome to the Export to JSON File Wizard** window will appear, displaying a summary of the steps to be taken in the wizard.

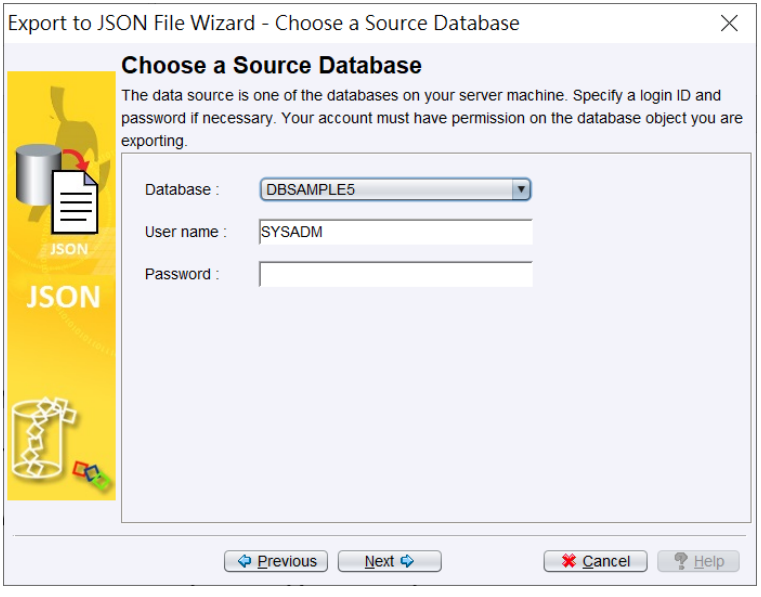
3. Click **Next**. The **Choose a Source Database** will appear.

©Copyright 1995-2026 CASEMaker Inc.

NOTE: The user must have CONNECT or higher authority to login.

4. Select the database to export data from in the **Database** menu.
5. Enter a user name and password into the appropriate fields.

©Copyright 1995-2026 CASEMaker Inc.



NOTE: User must have SELECT privilege to export a text file

6. Click **Next**. The Table or Query Export window will appear.

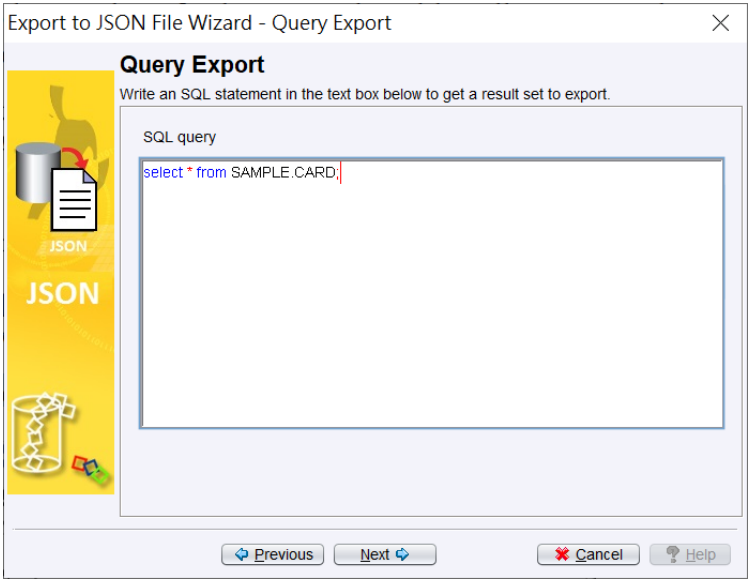
©Copyright 1995-2026 CASEMaker Inc.

7. Select Table to export data from a table. Select SQL query to export data from the result set of a SQL SELECT statement.
8. If you selected Table from the Table or Query Export window, the Table Export window will open. If you selected SQL query, then proceed to step 13.
9. Click **Next**, the Table Export window will open.

10. Select a table to export from the Table name menu. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the select privilege is granted to him. The user with CONNECT authority can select tables on which the select privilege is granted to him. A list of columns in the table will appear in the Select columns to export field.
11. Select columns by clicking on the column name and clicking Add, or select all columns by clicking Add All. Selected column names will appear in the right hand field.
12. Click **Next**. The Name the JSON File window will appear (proceed to step 16).

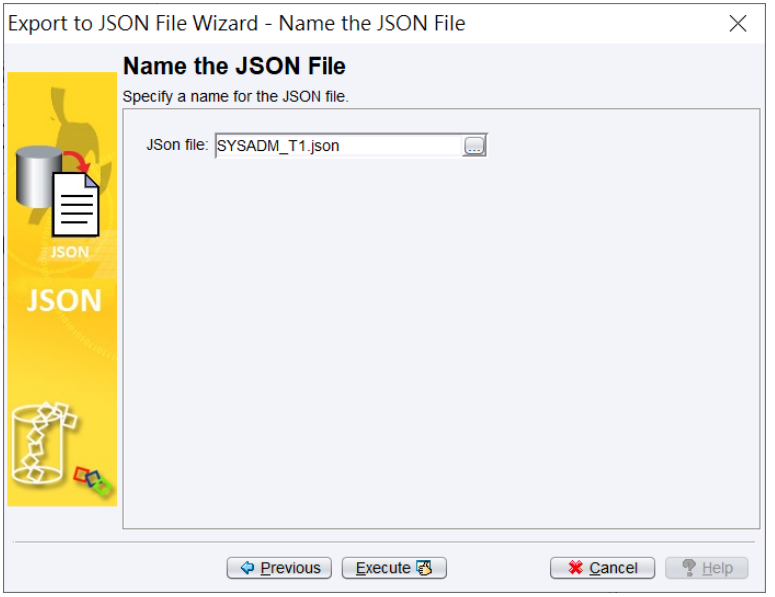
©Copyright 1995-2026 CASEMaker Inc.

13. If you selected SQL query from the Table or Query Export window, the Query Export window will open.
14. Enter a valid SQL select statement into the SQL query field.



15. Click **Next**. The **Name the JSON File** window will appear.

©Copyright 1995-2026 CASEMaker Inc.



16. Enter a JSON file name in the JSON file field, the default directory will be in the directory of DB_DBDIR. You can also enter a full path or select one using the browse button.
17. Click **Execute**. The Information message box will appear.
18. Click **OK**.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

28 Monitoring a Database

DBMaker provides two interfaces for monitoring the status of a database: the Database Monitor Tool and the Database Monitor tree object.

The Database Monitor Tool provides a graphical representation of various aspects of the database over time. It is ideally suited for monitoring overall database performance.

The Database Monitor tree object should be familiar to users who have experience with previous versions of JDBA Tool. It provides a series of tables that allow the viewer to monitor the status of individual connections.

©Copyright 1995-2026 CASEMaker Inc.

28.1 Database Monitor Tool

The Database Monitor Tool is a separate user interface that may be accessed through the Tool menu of the JDBA Tool or through the Windows Start menu. Opening the Database Monitor Tool allows performance related statistics to be viewed on a graph that plots a statistical value on the Y-axis and time on the X-axis. Statistical parameters (or 'objects') to display on the graph are selected from a menu at the bottom of the screen. Each statistical parameter is sampled for a period of time, and the values are plotted on the x-axis. The scale of the graph determines the sample period.

The Database Monitor Tool provides an interface to customize aspects of the view, including the following:

- Items: track up to 34 statistical parameters
- Customize the line color, line width, and scale of each item
- Select from sample periods of 5 seconds, 2 minutes, 1 hour, and 1 day
- Specify the grid color and background color of the chart
- Specify the scale of the chart's Y-axis

The Database Monitor Tool can track 34 performance-related statistical parameters. Table 20-1 summarizes the statistical parameters.

PARAMETER	DESCRIPTION
Number of Transactions Started	The number of transactions started within the sample period.
Number of Transactions Committed	The number of transactions committed within the sample period.
Number of Transactions Aborted	The number of transactions aborted during the sample period.
Number of Checkpoints	The number of checkpoints that occurred within the sample period.

©Copyright 1995-2026 CASEMaker Inc.

Number of Lock Requests	The total number of locks requested during the sample period.
Number of Deadlocks Detected	The number of deadlocks occurring within the sample period.
Number of Journal Block Reads	The number of journal blocks read from file within the sample period.
Number of Journal Block Writes	The number of journal blocks written to file within the sample period.
Number of Journal Record Writes	The number of journal records generated during the sample period.
Number of Journal Forced Writes	The number events within the sample period that flush dirty journal buffer to disk.
Number of Commit Waits	The number of transactions within the sample period waiting for group commit transactions.
Number of Logical Reads	The number of page/frame read events from the data buffer within the sample period.
Number of Logical Writes	The number of page/frame write events from the data buffer within the sample period.
Number of Physical Reads	The number of page/frame read events from disk within the sample period.
Number of Physical Writes	The number of page/frame write events from disk within the sample period.
Number of Index Page Splits Occurring	The number of index pages split within the sample period.
Number of Data Pages Compressed	The number of data pages reorganized within the sample period.
Number of Row to Page Lock Escalations	The number of events within the sample period causing escalation from row locks to page locks.
Number of Page to Table Lock Escalations	The number of events within the sample period causing escalation from page locks to table locks.
Number of Failed Events due to Lock Timeout	The number of events within the sample period that failed due to a lock time-out.
Number of Events Waiting for Locks	The number of events within the sample period that were required to wait for the

©Copyright 1995-2026 CASEMaker Inc.

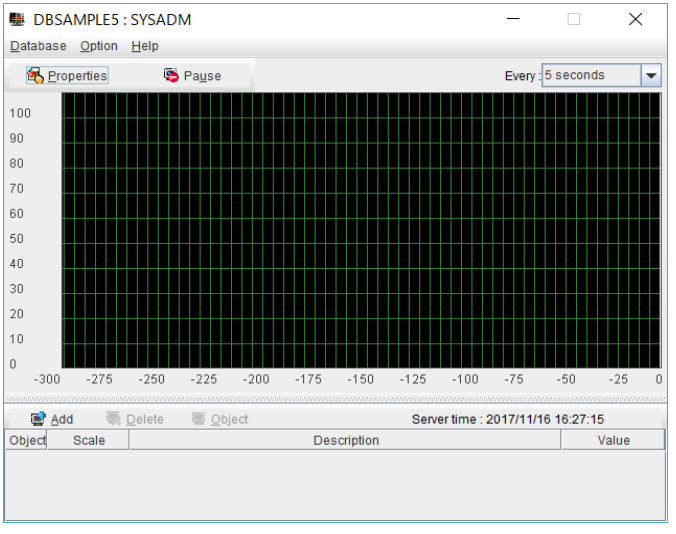
	release of a lock on an object.
Number of SELECT Operations	The number of SELECT statements within the sample period.
Number of INSERT Operations	The number of INSERT statements within the sample period.
Number of UPDATE Operations	The number of UPDATE statements within the sample period.
Number of DELETE Operations	The number of DELETE statements within the sample period.
Number of SQLPrepare() Calls	The number of SQLPrepare() calls within the sample period.
Number of SQLExecute() Calls	The number of SQLExecute() calls within the sample period.
Number of SQLExecDirect() Calls	The number of SQLExecDirect() calls within the sample period.
Number of Fetches	The number of fetch operations within the sample period.
CPU Usage*	The average percentage of total CPU processing capacity experienced within the time period.
Peak Number of Connections	The maximum number of simultaneous connections occurring within the sample period.
Free Physical Memory (KB)	The average amount of free physical memory occurring within the sample period.
Total Swap Space (Bytes)	The total operating system memory swap space allocated on disk.
Free Swap Space (Bytes)	The total operating system memory swap space available on disk.

Table 28-1 Database Monitor Tool Statistical Parameters

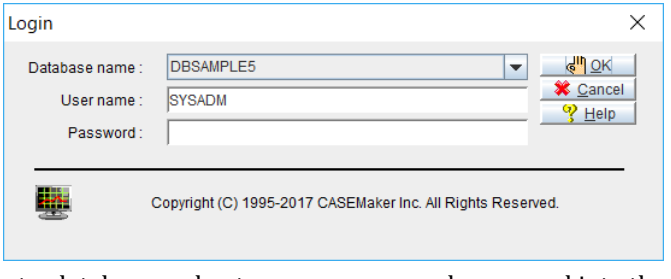
©Copyright 1995-2026 CASEMaker Inc.

➲ Viewing statistical parameters with the Database Monitor Tool:

1. Open the **Database Monitor Tool** from the Windows **Start** menu or from the Tool menu in the JDBA Tool. The **Database Monitor Tool** will start.



2. If the **Database Monitor Tool** is started from the Tool menu in JDBA Tool it is automatically connected to the database. If the Database Monitor Tool is started from the Windows Start menu, connecting to the database must be performed manually. Connect to a database by selecting **Connect Database** from the **Database** menu. The **Login** window appears.

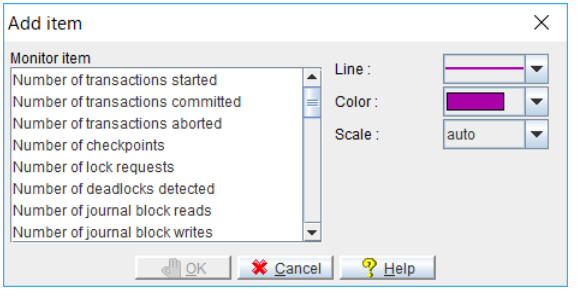


3. Select a database and enter a username and password into the appropriate fields. You will return to the Database Monitor Tool.

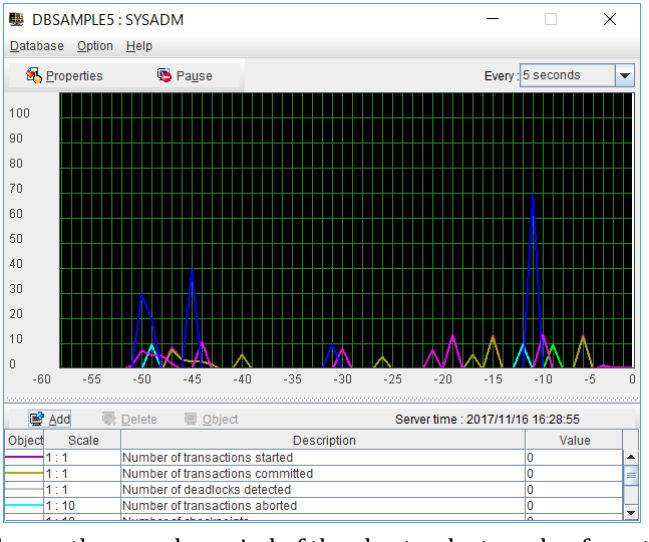
©Copyright 1995-2026 CASEMaker Inc.

NOTE *The user must have DBA authority or higher to login.*

4. Click the **Add** button to add a statistical parameter to the chart. The **Add Item** dialog box will open.



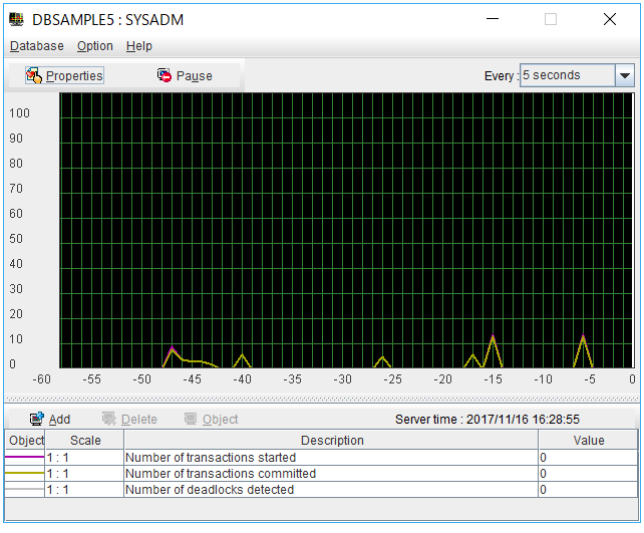
5. Select the item to monitor.
6. Select the line type, color, and scale.
7. Click the **OK** button. The **Add Item** dialog box will disappear and the monitor item will appear on the graph. Monitoring begins the moment you click the **OK** button. The newly plotted data will not appear until after the first sample period has elapsed.



8. To change the sample period of the chart, select a value from the **Every** menu. The values of items will change in proportion to the new sample

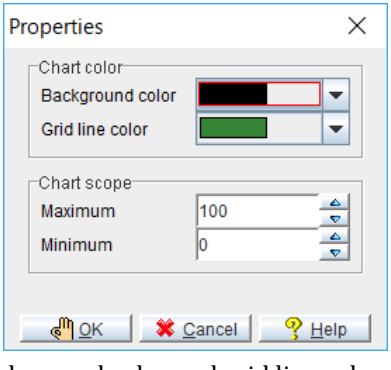
©Copyright 1995-2026 CASEMaker Inc.

period.



9. To change the properties of the chart:

   a) Click the **Properties** button. The **Properties** dialog appears.



   b) Select a background color and grid line color.

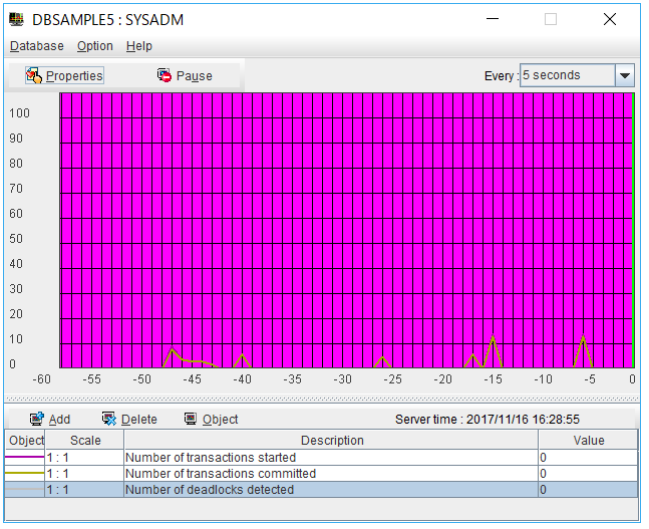
   c) Enter a chart scope; minimum value will reflect the bottom of the chart, maximum value will reflect the top of the chart.

   NOTE *The maximum value must be greater than the minimum value by at least 10, and the difference between minimum and maximum values must be a factor of 10. Maximum*

©Copyright 1995-2026 CASEMaker Inc.

range (10 – 9,000,000). Minimum range (0 – 8,999,990)

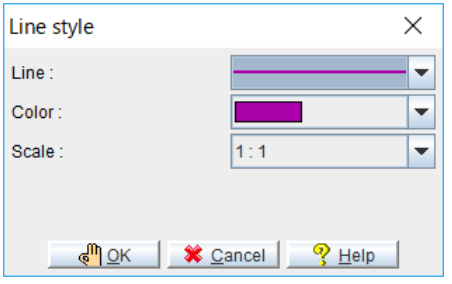
d) Click the **OK** button. The chart changes to reflect in the changes.



10. To change the line type of an item:

a) Select one of the items.

b) Click the **Object** button. The **Line Style** window will open.



c) Select a new line type, color, and scale.

d) Click the **OK** button.

11. To delete an item:

a) Select one of the items.

b) Click the **Delete** button. The item will be removed from the chart.

©Copyright 1995-2026 CASEMaker Inc.

12. Click the **Pause** button to stop the progress of the chart.

©Copyright 1995-2026 CASEMaker Inc.

28.2 Export to File Options

The export to file feature allows DBA to look at log files created by Jmonitor over an extended period of time. The DBA can then analyise the data and then adjust his database for optimal performance.

You have several options when setting the view of the created log files.

- Display the log data in column or row format
- Set the caption break for headings of monitored items by row or day
- Set the number of log files that will be created
- Set the size of the log files that be created

When setting the output of the log file to display in a column or row format. In a column format, the log file will list the monitored items in columns. For example:

```
                            [Free swap space (KB)]     [Total swap space (KB)]
==============================================================================
2004/06/21 19:45:20               459216                      633044
2004/06/21 19:45:25               459408                      633044
```

In a row format, the monitored items in the log file will be listed in a row. For example:

```
 Item Name                            Data                 Time
================================================================================
[Free swap space (KB)]               457176          2004/06/21 19:59:25
[Total swap space (KB)]              633044          2004/06/21 19:59:25
```

You can set the caption break for headings of monitored items, like Date, Data and Item Names, to appear in the log file after a specified number of rows or days. When specifying the caption break for a number of rows the valid range is between 10 and 200. The default value is ***25***. When specifiying caption

©Copyright 1995-2026 CASEMaker Inc.

break value in days the valid range is from 1 to 30, with the default value set at 2.

The JMonitor export monitor item to external file function supports multi file exporting. You can set the number of log files for exporting. The exporting file function will create a new file and rename the previous exporting file with another file name. The current monitor item information will write into the new log file. The file count follows an ascending value. The valid range is between 10 and 20, with the default value set at 10.

Example:

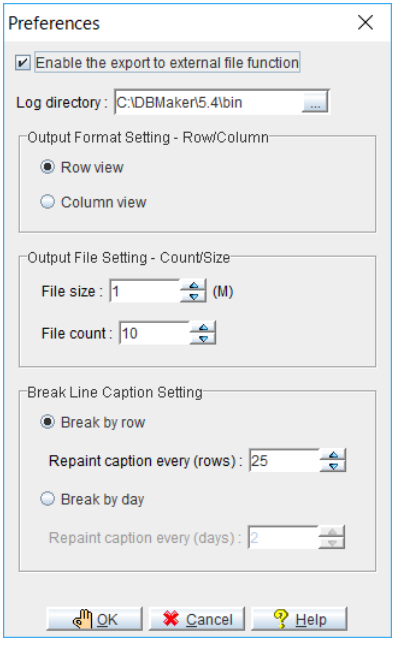
If we set the file count as 10, the first log created will be **Message.log**. After **Message.log** reaches its maximum size (which is set in the File Size field), the system will create a new log **Message0.log**. The contents of **Message.log** are transferred to **Message0.log**. The most current data is stored in **Message.log**. Once the log **Message.log** reaches its specified maximum size **Message1.log** is created. The contents of Message.log are transferred to **Message0.log**, and the contents of **Message0.log** are transferred to **Message1.log**. This process continues.

The file size option allows you to limit the size of the logs to a manageable size. The valid range is between 1MB to 10MB, with the default value set at 1 MB.

➲ Accessing the export to file options:

1. Select **Option** from the menu bar. A menu list appears.
2. Select **Preferences** from the menu list. The **Preferences** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



3. Set the **Enable the export to external file function** check box. The fields in the **Preferences** dialog box become active. This check box is checked by default. To disable the feature, clear the check box.
4. Enter a file path for the log files in the **Log directory** field. Or select a path by clicking the **Browse** button. The default path is *...\DBMaker\5.4\bin*.
5. Select an output format from the **Output Format Setting – Row\Column** portion of the dialog box.
6. Set the out put file size number of files to be created in the **Output File Settings – Count\Size** portion of the dialog box.
7. Select the caption break for the log files in the **Break Line Caption Setting** portion of the dialog box.
8. Click the **OK** button after setting all options. You are returned to the **JMonitor** screen.

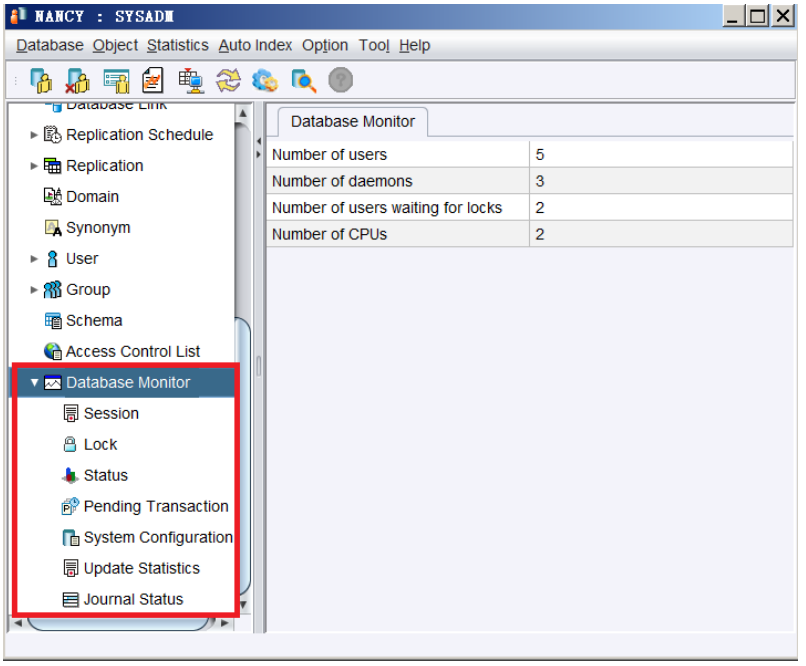
©Copyright 1995-2026 CASEMaker Inc.

28.3 Database Monitor (Tree)

JDBA Tool provides monitoring functions that allow you to view the status of individual user connections, transactions, journals and upper memory handling. The statistics provided by the monitoring functions can be useful for database performance evaluation and tuning. There are seven database-monitoring functions provided by JDBA Tool: Session information, Lock status, Database status, Pending Transaction status, System Configuration, Update Statistics and Journal Status.

- Session information shows all connections with the database (users, backup demon, IO daemon…), and information related to the connection. Sessions can be terminated by the DBA with the Kill connection function.
- Lock status is for viewing database objects that have locks on them, and the type of lock on the object.
- Database status allows the DBA to view statistics relating to journal files and buffers, as well as view the current database setup.
- Pending Transaction status provides information on distributed transactions that have been prepared to commit but have not been committed yet.
- System Configuration information allows the DBA to view all server configuration setting.
- Update Statistics information allows DBA to monitor database update statistics progress.
- Journal Status information shows information about journal. Users can check journal status and journal user. Disconnect the stuck connection if necessary.

©Copyright 1995-2026 CASEMaker Inc.



Monitoring Database Sessions

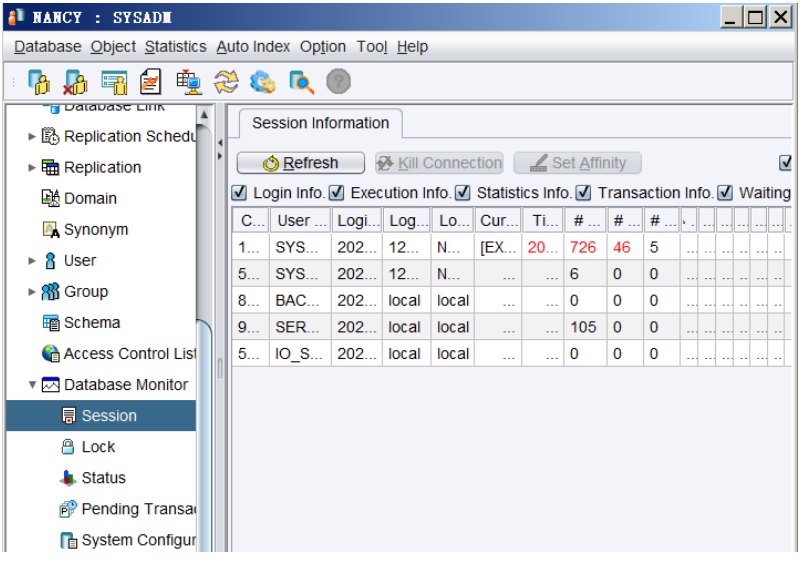
Session information allows you to view information that relates to users connected to the database and resources used by each user. Session Information is organized by six categories:

- Login Information
- Execution Information
- Statistical Information
- Transaction Information
- Waiting Information
- Lock Graph

➲ Viewing session information:

©Copyright 1995-2026 CASEMaker Inc.

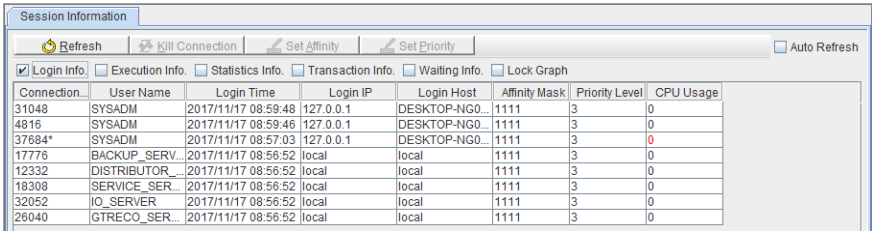
1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Session** node. The following window appears.



2. You can view session information by sliding the scroll bar at the bottom of the page, or by selecting the appropriate check boxes next to each category.
3. After a refresh, newly updated information appears in red. Information that remains the same appears in black.

LOGIN INFORMATION

This category contains the user's name, information relevant to the connection and information about the computer their using.



Connection...	User Name	Login Time	Login IP	Login Host	Affinity Mask	Priority Level	CPU Usage
31048	SYSADM	2017/11/17 08:59:48	127.0.0.1	DESKTOP-NG0...	1111	3	0
4816	SYSADM	2017/11/17 08:59:46	127.0.0.1	DESKTOP-NG0...	1111	3	0
37684*	SYSADM	2017/11/17 08:57:03	127.0.0.1	DESKTOP-NG0...	1111	3	0
17776	BACKUP_SERV...	2017/11/17 08:56:52	local	local	1111	3	0
12332	DISTRIBUTOR_...	2017/11/17 08:56:52	local	local	1111	3	0
18308	SERVICE_SER...	2017/11/17 08:56:52	local	local	1111	3	0
32052	IO_SERVER	2017/11/17 08:56:52	local	local	1111	3	0
26040	GTRECO_SER...	2017/11/17 08:56:52	local	local	1111	3	0

©Copyright 1995-2026 CASEMaker Inc.

Figure 28-1 Session Information: Login Information

COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session
User_Name	The name of the user using the session
Login_Time	The time that the user connected to the database
Login_IP	The IP address of the user using the session
Login_Host	The host computer using the session
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

Table 28-2 Login Information

EXECUTION INFORMATION

This category describes the type and status of SQL commands being executed by each user.

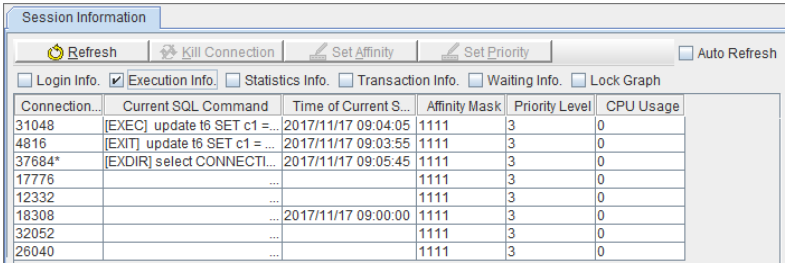


Figure 28-2 Session Information: Execution Information

COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session

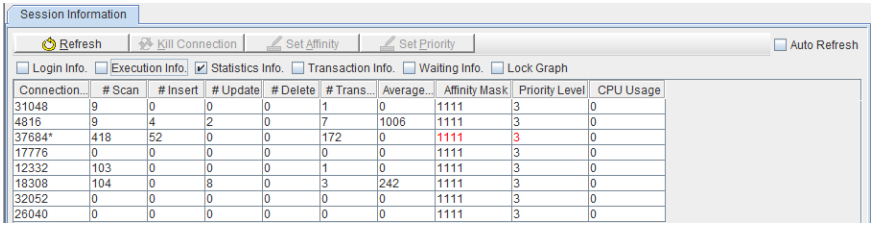
©Copyright 1995-2026 CASEMaker Inc.

COLUMN NAME	DESCRIPTION
Current SQL Command	The most recently executed SQL command and the command status. The command status can be the following: [PRE] - The SQL command is being prepared. [EXEC] - The SQL command is being executed from SQLExecute call. [EXDIR] - The SQL command is being executed from SQLExecDirect call. [FETCH] - The operation is in the fetch data phase. [EXIT] - The SQL command is finished with the prepare, execute or fetch operation.
Time of Current SQL Command	The time when the most recently used SQL command was executed
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

Table 28-3 Execution Information

STATISTICAL INFORMATION

This category displays statistics relevant to the data.



Connection...	# Scan	# Insert	# Update	# Delete	# Trans...	Average...	Affinity Mask	Priority Level	CPU Usage
31048	9	0	0	0	1	0	1111	3	0
4816	9	4	2	0	7	1006	1111	3	0
37684*	418	52	0	0	172	0	1111	3	0
17776	0	0	0	0	0	0	1111	3	0
12332	103	0	0	0	1	0	1111	3	0
18308	104	0	8	0	3	242	1111	3	0
32052	0	0	0	0	0	0	1111	3	0
26040	0	0	0	0	0	0	1111	3	0

Figure 28-3 Session Information: Statistical Information

©Copyright 1995-2026 CASEMaker Inc.

COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session
#Scan	The number of SELECT operations in the session
#Insert	The number of INSERT operations in the session
#Update	The number of UPDATE operations in the session
#Delete	The number of DELETE operations in the session
#Transaction	The number of transactions in the session
Average Journal Bytes Per Transaction	The average number of journal bytes used per transaction in the session
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

Table 28-4 Statistical Information

TRANSACTION INFORMATION

Displays journal file activity, specifically the start and end points of transactions within the journal.

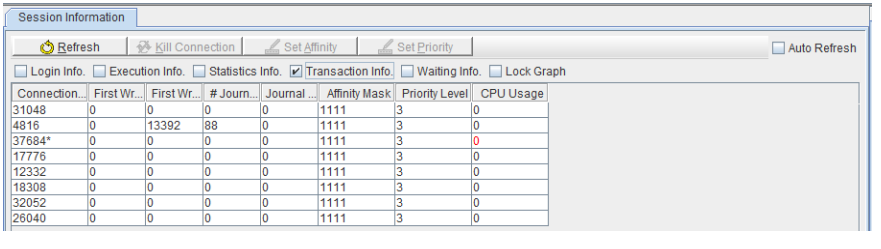


Figure 28-4 Session Information: Transaction Information

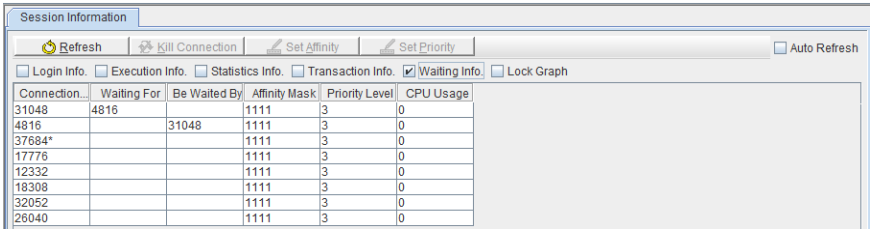
©Copyright 1995-2026 CASEMaker Inc.

COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session
First Written Journal file of Current Transaction	The first journal file that is fully written to in the current transaction
First Written Journal block of Current Transaction	The first journal block that is fully written to in the current transaction
# Journal Bytes of Current Transaction	The total amount of journal data used in the current transaction in the session
Journal Blocks Distance of Current Transaction	The distance of the current journal block from the first written journal block in the current transaction
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

Table 28-5 Transaction Information

WAITING INFORMATION

Displays information on which users are waiting for locks on objects and which users have the locks on the objects being waited for.



Connection...	Waiting For	Be Waited By	Affinity Mask	Priority Level	CPU Usage
31048	4816		1111	3	0
4816		31048	1111	3	0
37684*			1111	3	0
17776			1111	3	0
12332			1111	3	0
18308			1111	3	0
32052			1111	3	0
26040			1111	3	0

©Copyright 1995-2026 CASEMaker Inc.

Figure 28-5 Session Information: Waiting Information

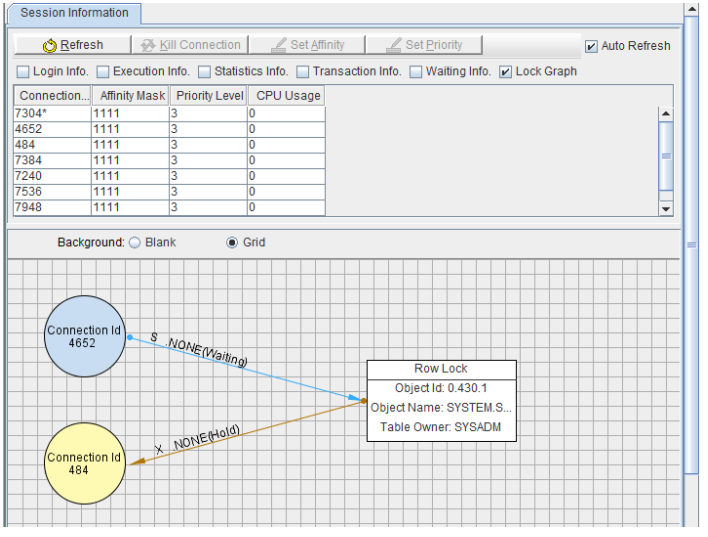
COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session
Waiting For	The connection ID number of a user that has a lock on an object that the current user's transaction is waiting for
Be Waited By	The connection ID of a user that is waiting for release of a lock on an object held by the current user
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

Table 28-6 Waiting Information

LOCK GRAPH

Display all the connections in the current database with graph. Users can see which connections are waiting via the graph. The blue circle indicates that the current connection is waiting for some resources and the yellow circle indicates that the current connection takes up some resources. Users also can right-click **Kill connection** option at the yellow circle to disconnect some useless connections.

©Copyright 1995-2026 CASEMaker Inc.



NOTE *When you choose the kill connection option, please note that not to disconnect important connections that are currently in use.*

COLUMN NAME	DESCRIPTION
Connection ID	The ID of the session
Affinity Mask	The CPU affinity of the session
Priority Level	The Priority Level of the session
CPU Usage	The CPU usage of the session

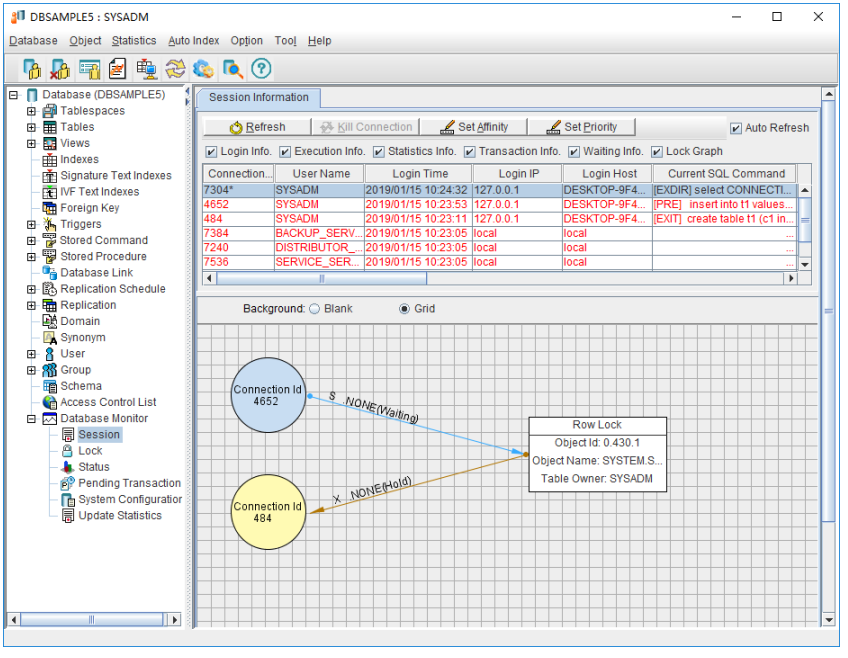
KILLING CONNECTIONS

You can choose to terminate an active connection.

➲ Killing a connection:

1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Session** node. The **Session Information** page will appear.
2. Select a connection from the list in the **Session Information** table. It will highlight blue and the **Kill Connection** field will be active.

©Copyright 1995-2026 CASEMaker Inc.



3. Clicking the **Kill Connection** button will terminate the connection.

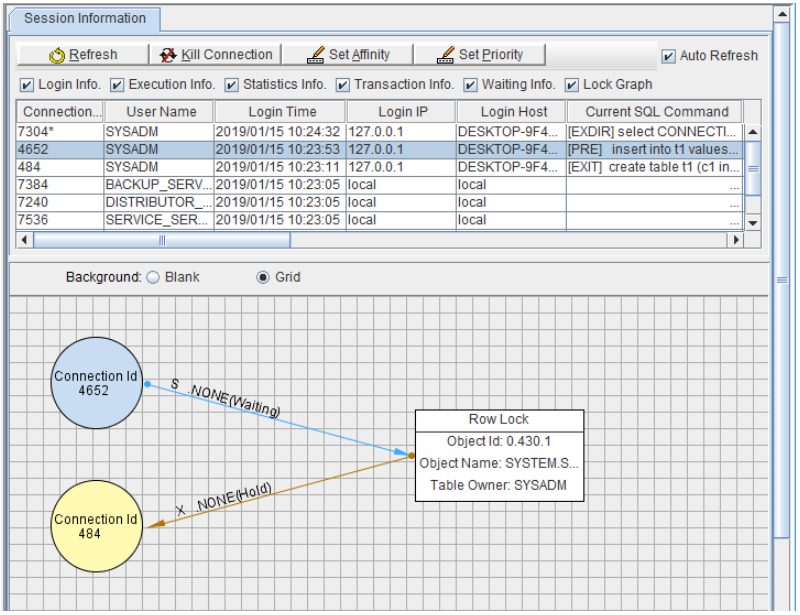
SET AFFINITY

You can set one connection's affinity to improve the performance of the whole system.

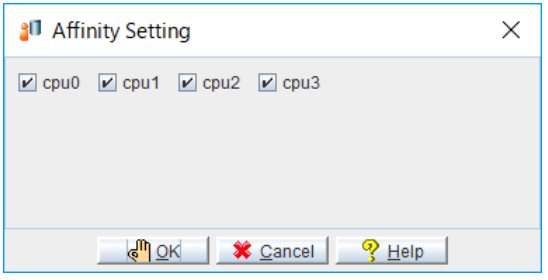
➲ Set affinity:

1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Session** node. The **Session Information** page will appear.
2. Select a connection from the list in the **Session Information** table. It will highlight blue and the **Set Affinity** field will be active.

©Copyright 1995-2026 CASEMaker Inc.



3. Click the **Set Affinity** button. The **Affinity Setting** page will appear.



4. Slect a check box and click the **OK** button. The selected connection CPU affinity value will be set.

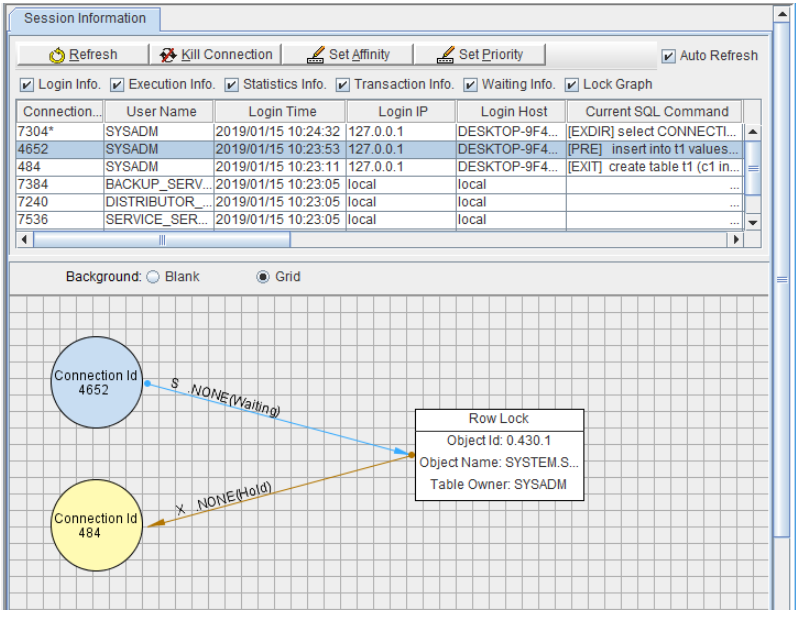
SET PRIORITY

You can set one connection's priority to improve the performance of the whole system.

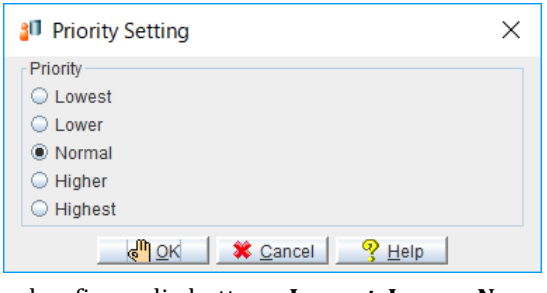
➲ Set priority:

©Copyright 1995-2026 CASEMaker Inc.

1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Session** node. The **Session Information** page will appear.
2. Select a connection from the list in the **Session Information** table. It will highlight blue and the **Set Priority** field will be active. If there is only one CPU on user's computer, disable "Set Priority" button.



3. Click the **Set Priority** button.The **Priority Setting** page will appear.



4. This dialog has five radio buttons: **Lowest**, **Lower**, **Normal**, **Higher**, **Highest**. The default value of CPU priority is Normal. Choose a priority and click the **OK** button at the bottom, the selected connection CPU

©Copyright 1995-2026 CASEMaker Inc.

priority value will be set.

Monitoring Database Locks

You can monitor details of the locks used in the database. This information includes the lock table, lock granularity, lock status, and lock modes for all data objects.

Shows a graph displays information about lock details. Please refer to Monitoring Database Locks-VIEWING DATABASE LOCK GRAPH for more information.

The following table summarizes the items on the lock page:

ITEM	DESCRIPTION
Object Name	The object name for a locked object, or an object that a lock is requested for
Level	The current lock level. Four lock levels exist for an object: System, Table, Page, and Row.
Object ID	An identifying number unique to each table, page, or row in the database
Login Name	Name of the user that has the lock on the object
Connection ID	The ID of the session
Lock Mode	The current lock mode of an object. The shown lock modes are either S, IS, IX, X and SIX. IS, SIX, and IX denote that the object has an intention lock; part of the lock on the object is specified at a lower granularity.
Waiting Mode	The lock mode that the user is requesting. It can be “none”, indicating that the user does not desire a new lock on the object, or “S”, “IS”, “IX”, “X” or “SIX”, indicating that the user has requested the corresponding lock type on the object and is waiting for the lock.

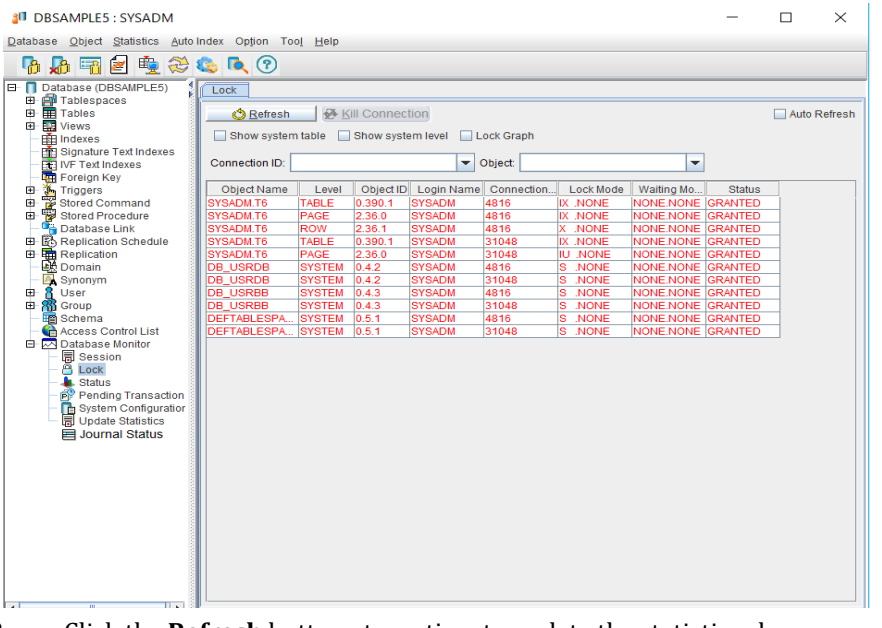
©Copyright 1995-2026 CASEMaker Inc.

ITEM	DESCRIPTION
Status	Status indicates the condition of the Waiting Mode. "Granted" indicates that the user has been granted the requested lock and should correspond to a waiting mode of "none". "Waiting" indicates that the user has not yet been granted the requested object lock and should correspond to a Waiting mode of "S", "IS", "IX", "X" or "SIX". "Convert" indicates that the user has requested a change in lock level (for example, from "S" to "X") and the user is waiting for the new lock level on the object.

Table 28-7 Database Lock items

➲ Viewing database lock status:

1. Select the **Lock** node from the **Database Monitor** node in the tree. The following window appears.



2. Click the **Refresh** button at any time to update the statistics shown on this page. You may also choose what information to display on the **Lock**

©Copyright 1995-2026 CASEMaker Inc.

page.

- Selecting **Show System Table** allows you to see any locks placed on the system table.
- Selecting **Show System Level** allows the DBA to see locks placed on objects in the system level.
- Selecting **Lock Graph** allows you to see the lock details by graph. User can specify the connection ID, lock level and what type of lock.
- You can choose to view locks only on certain objects. Select an object name from the Object menu.
- You can choose to view locks on all objects associated with a user. Select a connection ID from the Connection ID menu.

3. User can select a connection and click **Kill Connection** to kill the connection.

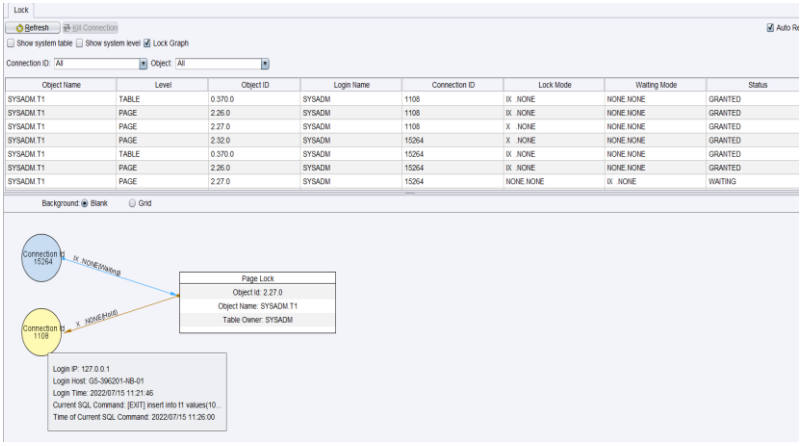
VIEWING DATABASE LOCK GRAPH

Lock graph can show a graph displays information about lock details. Users can specify the waiting and waited connection, lock level, type of lock and sql statement.

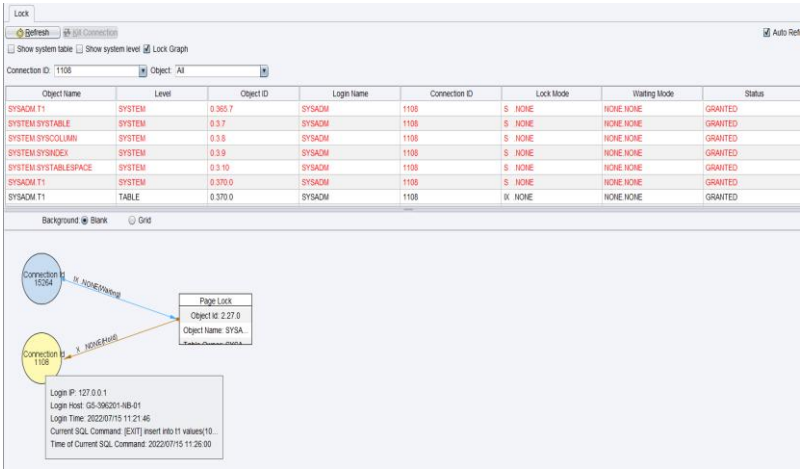
➲ Viewing Lock Graph

1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the Lock node. The Lock information page will appear. (Users can also select Session node, there's a Lock Graph button in the Session information page).
2. Click Lock Graph. The Lock Graph tab will appear below.

©Copyright 1995-2026 CASEMaker Inc.



3. Users can move the mouse on the connection to see connection details. Click the connection icon will mark the related information above in red. Users can specify the connection cause the lock and click Kill Connection to disconnect it.



Monitoring Database Status

The database status node of the database monitor tree item gives access to database statistics stored in the SYSINFO system table. Database status items

©Copyright 1995-2026 CASEMaker Inc.

are grouped into eight categories of related items. Each category is grouped together on a tabbed page. The categories include:

- Locks
- Connection
- Data Operation
- Database Info
- System Info
- Page and I/O
- Journal
- Transaction

VIEWING LOCK STATUS

Lock status items are statistics related to the cumulative number of locks, lock escalations, deadlocks, and failed events due to lock timeouts. The lock status page displays lock information statistics from the SYSINFO table. Table 28-8 summarizes the lock status items and their corresponding SYSINFO.INFO values.

LOCK STATUS ITEM	DESCRIPTION
Number of Page Locks Escalation	The accumulated number of row locks that have been escalated to page locks. (NUM_ROW_LOCK_UPG)
Number of Table Locks Escalation	The accumulated number of page locks that have been escalated to table locks. (NUM_PAGE_LOCK_UPG)
Number of fail locks because of timeout	The accumulated number of actions failing due to a timeout while waiting to acquire a lock on an object. (NUM_LOCK_TIMEOUT)
Number of Locks Waited	The accumulated number of events waiting to place locks on objects. (NUM_LOCK_WAIT)

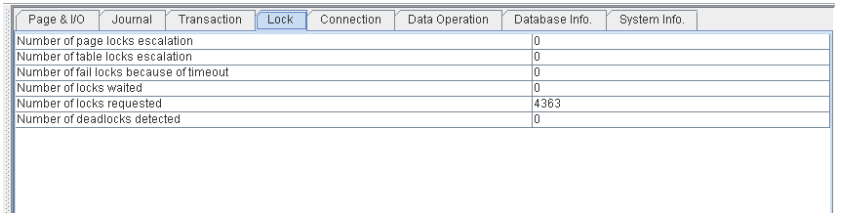
©Copyright 1995-2026 CASEMaker Inc.

LOCK STATUS ITEM	DESCRIPTION
Number of Locks Requested	The accumulated number of locks requested on data objects. (NUM_LOCK_REQUEST)
Number of deadlocks detected	The accumulated number of deadlocks detected. (NUM_DEADLOCK)

Table 28-8 Lock Status Items

➲ Viewing lock status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Lock** tab. The following page appears.



VIEWING CONNECTION STATUS

The connection status page displays connection information statistics from the SYSINFO table. Table 28-9 summarizes the connection status items and their corresponding SYSINFO.INFO values.

CONNECTION STATUS ITEM	DESCRIPTION
Maximum number of allowed connections for a database	The hard limitation of connections. This value is determined by the DB_MaxCo keyword when the database is created or started with a new journal. (NUM_MAX_HARD_CONNECT)
Maximum number of allowed connections at a time	The soft limitation of connections, which is less than or equal to the number of hard connections. (NUM_MAX_SOFT_CONNECT)

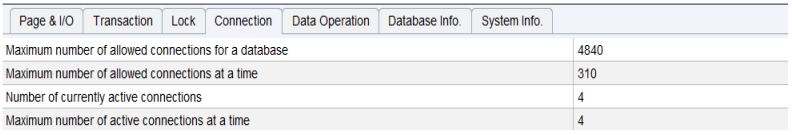
©Copyright 1995-2026 CASEMaker Inc.

Connection Status Item	Description
Number of currently active connections	The current number of connections to the database. (NUM_CONNECT)
Maximum number of active connections at a time	Records the largest number of active connections experienced by the database since the database was started. (NUM_PEAK_CONNECT)

Table 28-9 Connection Status items

➲ Viewing connection status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Connection** tab. The following page appears.



Viewing Data Operation Status

The data operation status page displays data operation statistics from the SYSINFO table. Table 28-10 summarizes the data operation status items and their corresponding SYSINFO.INFO values.

Data Operation Status Item	Description
Number of SELECT Operation	The total number of SELECT operations performed since the database was started. (NUM_SQL_SELECT)
Number of INSERT (including INSERT INTO) Operation	The total number of SELECT operations performed since the database was started. (NUM_SQL_ INSERT)
Number of UPDATE Operation	The total number of UPDATE operations performed since the database was started. (NUM_SQL_UPDATE)

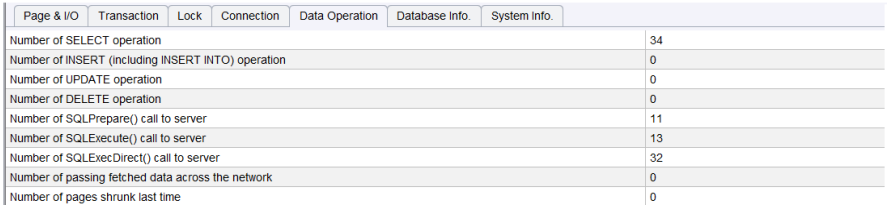
©Copyright 1995-2026 CASEMaker Inc.

DATA OPERATION STATUS ITEM	DESCRIPTION
Number of DELETE Operation	The total number of DELETE operations performed since the database was started. (NUM_SQL_DELETE)
Number of SQLPrepare() call to server	The total number of SQLPrepare() calls to the database since the database was started. (NUM_SQL_PREPARE)
Number of SQLExecute() call to server	The total number of SQLExecute()calls to the database since the database was started. (NUM_SQL_EXECUTE)
Number of SQLExecDirect() call to server	The total number of SQLExecDirect()calls to the database since the database was started. (NUM_SQL_EXEDIRECT)
Number of passing fetched data across the network	The total number of fetch operations performed by clients since the database was started. (NUM_SQL_FETCH)
Number of pages shrunk last time	The total number of pages was shrunk last time (NUM_PAGE_SHRINK)

Table 28-10 Data Operations Status items

➲ Viewing data operation status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Data Operation** tab. The following page appears.



VIEWING DATABASE INFORMATION

The database information page displays database information statistics from the SYSINFO table. Table 24-11 summarizes the database information items and their corresponding SYSINFO.INFO values.

©Copyright 1995-2026 CASEMaker Inc.

DATABASE INFORMATION ITEMS	DESCRIPTION
Time of the counter of SYSINFO being restarted	The last time and date that data in the SYSINFO table was reset. The SYSINFO table gets reset when one of the counters overflows, or the SET SYSINFO CLEAR command is used. (SYSINFO_RESET_TIME)
Total size of DCCA (byte)	Total memory size of the Database Communication and Control Area (DCCA). Refer to the *Database Administrators Guide* for detailed descriptions of the DCCA. (DCCA_SIZE)
Available size of DCCA (byte)	Amount of memory available in the DCCA. (FREE_DCCA_SIZE)
Distributed database mode	Distributed Database Mode. (DDB_MODE)
Backup mode	Indicates whether incremental backups can be performed. It also indicates whether the incremental backup will include BLOB files, 1 means to backup the data file only, and 2 means to backup both data and BLOB files. (BACKUP_MODE)
User file object mode	Shows if user file objects have been enabled. A value of 1 means user file objects are enabled. (USER_FO_MODE)
Read-only mode	Shows if the database has been started in read-only mode. A value of 1 means he database has been started in read-only mode. (READ_ONLY_MODE)
BLOB frame size	The size of each BLOB frame in bytes. (FRAME_SIZE)
Time of database creation	The date and time the database was created. (CREATE_DB_TIME)
Time of database start	The date and time the database was last started. (START_DB_TIME)
DBMaker version	The version number of the DBMaker engine. (VERSION)

©Copyright 1995-2026 CASEMaker Inc.

DATABASE INFORMATION ITEMS	DESCRIPTION
Database file version	The version number of the database (FILE_VERSION)
Product information	The product information of DBMaker, include platform and operating system (PRODUCT_INFO)
Build version	The build version of DBMaker(BUILD_VERSION)
Time of the database startup with force new journal	The date and time the database was last started in new journal mode. (FORCE_NEW_JNL_TIME)
Time of turning the journal off	The most recent date and time the journal was stopped. (START_NO_JNL_TIME)
Time of turning the journal on	The most recent date and time the journal was started. (END_NO_JNL_TIME)
Max allowed memory size allocated for ITT	The max size of allowed memory for internal temporary tables. (in bytes) (MAX_ITT_SIZE)
Current memory size allocated for ITT	The current size of internal temporary tables. (in bytes) (CURRENT_ITT_SIZE)
Full backup cost (second)	The last full backup time cost. (in seconds) (FULL_BACKUP_COST)
Differential backup cost (second)	The last differential backup time cost. (in seconds) (DIFF_BACKUP_COST)
Differential backup PCT	(The last diff backup size) / (Database size) * 100% (DIFF_BACKUP_PCT)
Transparent data encryption status	The status of column encryption. OPEN means encrypted column can be accessed, CLOSE means encrypted column cannot be accessed (TDE_STATUS)

Table 28-11 Database Information Items

➲ Viewing Database Information:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Database Info** tab. The following page appears.

©Copyright 1995-2026 CASEMaker Inc.

Page & I/O | Transaction | Lock | Connection | Data Operation | Database Info. | System Info.

Time of the counter of SYSINFO being restarted	2024/08/01 10:43:17
Total size of DCCA (byte)	33062912
Available size of DCCA (byte)	12261120
Distributed database mode	OFF
Backup mode	NON-BACKUP
User file object mode	OFF
Read-only mode	OFF
BLOB frame size	32768
Time of database creation	2024/08/01 10:41:13
Time of database start	2024/08/01 10:43:17
DBMaker version	5.46
Database file version	5.41
Product information	dbmaker-WIN64
Build version	5.4.6 (#31539, 20240506)
Time of the database startup with force new journal	
Time of turning the journal off	
Time of turning the journal on	
Max allowed memory size allocated for ITT	5412945920
Current memory size allocated for ITT	0
Full backup cost (second)	0
Differential backup cost (second)	0
Differential backup PCT	0
Transparent data encryption status	CLOSE

VIEWING SYSTEM INFORMATION

The system information page displays operating system statistics from the SYSINFO table. Table 28-11 summarizes the system information items and their corresponding SYSINFO.INFO values.

JMonitor tool get sytem history information via sysinfohistory2(). However, sysinfohistory() is still used to get system history information.

SYSTEM INFORMATION ITEM	DESCRIPTION
The average CPU load during a short period (about 5 seconds)	The maximum percentage of total CPU processing capacity experienced within a time period. (CPU_USAGE)
Total physical memory	The total physical memory available to the system (TOTAL_MEMORY)
The current free physical memory (KB)	The amount of free physical memory. (TOTAL_FREE_MEMORY)
Total swap space (Bytes)	The total operating system memory swap space allocated on disk. (TOTAL_SWAP_SPACE)

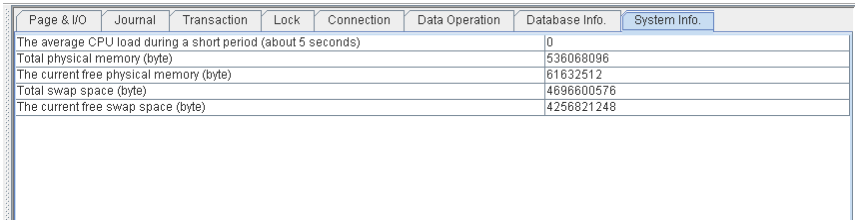
©Copyright 1995-2026 CASEMaker Inc.

The current free swap space (Bytes)	The total operating system memory swap space available on disk. (TOTAL_FREE_SWAP_SPACE)

Table 28-12 System Information items

➲ Viewing system information:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **System Info** tab. The following page appears.



VIEWING PAGE AND I/O STATUS

The page and I/O status page displays operating system statistics from the SYSINFO table. Table 28-12 summarizes the page and I/O status items and their corresponding SYSINFO.INFO values.

PAGE AND I/O STATUS ITEMS	DESCRIPTION
Number of Index Page Splits Occurring	The accumulated number of index pages split in the current run time. (NUM_IDX_SPLIT)
Number of Data Pages Compressed	The accumulated number of data pages reorganized in the current run time. (NUM_PAGE_COMPRESS)
Number of Physical Reads	The accumulated page number of page/frame disk reads in the current run time. (NUM_PHYSICAL_READ)
Number of Physical Writes	The accumulated number of page/frame disk writes in the current run time. (NUM_PHYSICAL_WRITE)

©Copyright 1995-2026 CASEMaker Inc.

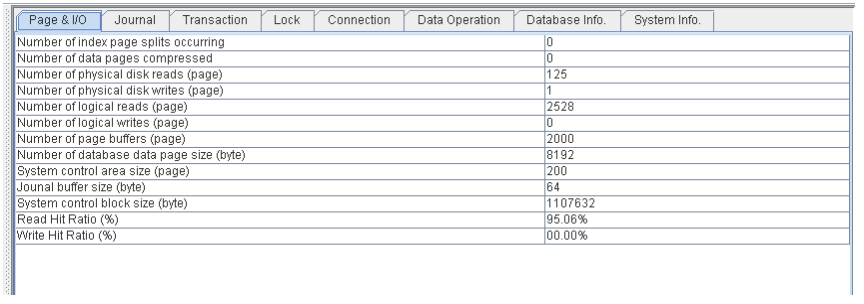
PAGE AND I/O STATUS ITEMS	DESCRIPTION
Number of Logical Reads	The accumulated page number of logical reads (reading data from page buffer cache) in the current run time. (NUM_LOGICAL_READ)
Number of Logical Writes	The accumulated number of logical writes (writing data into page buffer cache) in the current run time. (NUM_LOGICAL_WRITE)
Number of Page Buffers	The number of page buffers currently used by the database. (NUM_PAGE_BUF)
Number of database data page size (byte)	Page size of the database (byte) (DB_PAGE_SIZE)
System control area size (page)	The size of the System Control Area in the shared memory (page) (DB_SCA_SIZE)
Jounal buffer size (page)	The number of the journal buffers in the shared memory (page) (NUM_JOURNAL_BUF)
System control block size (byte)	The size of internal system control block in the shared memory.(byte) (DB_SYSCB_SIZE)
Read Hit Ratio (%)	The page buffer read hit ratio. (%) (READ_HIT_RATIO)
Write Hit Ratio (%)	The page buffer write hit ratio. (%) (WRITE_HIT_RATIO)

Table 28-13 Page and I/O Status items

➲ Viewing page and I/O status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Page & I/O** tab. The following page appears.

©Copyright 1995-2026 CASEMaker Inc.



VIEWING JOURNAL STATUS

The journal status page displays operating system statistics from the SYSINFO table. Table 28-13 summarizes the journal status items and their corresponding SYSINFO.INFO values.

JOURNAL STATUS ITEMS	DESCRIPTION
Number of journal blocks read from journal files	The accumulated number of journal block reads in the current run time. (NUM_JNL_BLK_READ)
Number of journal blocks written to journal files	The accumulated number of journal block writes in the current run time. (NUM_JNL_BLK_WRITE)
Number of journal records generated	The accumulated number of journal record writes in the current run time. (NUM_JNL_REC_WRITE)
Number of journal forced writes	The accumulated number of journal force writes in the current run time. (NUM_JNL_FRC_WRITE)
Number of journal files	The number of blocks set in a journal file. (NUM_JOURNAL_FILE)
Number of journal blocks in a file	The number of journal blocks in each journal file. (NUM_JOURNAL_BLOCKS)
Number of free journal blocks	The current number of free journal blocks in all journal files. (NUM_JNR_BLOCK_FREE)

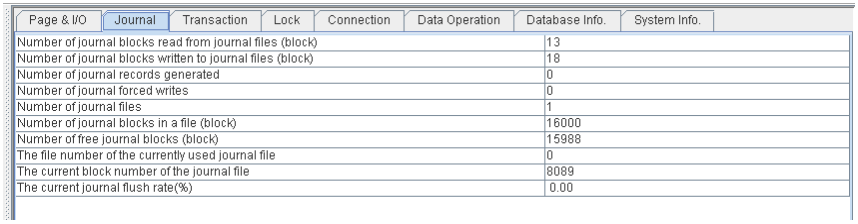
©Copyright 1995-2026 CASEMaker Inc.

JOURNAL STATUS ITEMS	DESCRIPTION
The file number of the currently used journal file	For databases with multiple journal files, the number of the current journal file. (CURRENT_JOURNAL_FN)
The current block number of the journal file	The number of the currently used journal block. (CURRENT_JOURNAL_BN)
The current journal flush rate (%)	(the number of journal file blocks written by DMServer/the times of flush) the number of journal files in buffer. (%) (JOURNAL_FLUSH_RATE)

Table 28-14 Journal Status items

➲ Viewing journal status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Journal** tab. The following page appears.



VIEWING TRANSACTION STATUS

The transaction status page displays operating system statistics from the SYSINFO table. Table 28-14 summarizes the transaction status items and their corresponding SYSINFO.INFO values.

TRANSACTION STATUS ITEMS	DESCRIPTION
Number of started transactions	The cumulative number of transactions started during the current runtime. (NUM_STARTED_TRANX)

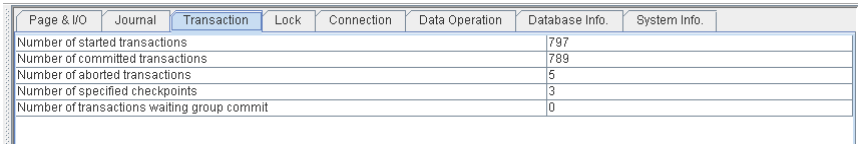
©Copyright 1995-2026 CASEMaker Inc.

TRANSACTION STATUS ITEMS	DESCRIPTION
Number of committed transactions	The cumulative number of committed transactions during the current runtime. (NUM_COMMITED_TRANX)
Number of aborted transactions	The cumulative number of transactions aborted in the current runtime. (NUM_ABORTED_TRANX)
Number of specified checkpoints	The cumulative number of checkpoints made during the current runtime. (NUM_CHECKPOINTS)
Number of transactions waiting group commit	The cumulative number of transactions waiting for a group commit action during the current runtime. (NUM_COMMIT_WAITER)

Table 28-15 Transaction Status items

➲ Viewing transaction status:

1. Select the **Status** node from the **Database Monitor** node in the tree.
2. Select the **Transaction** tab. The following page will be displayed.



Monitoring Pending Transactions

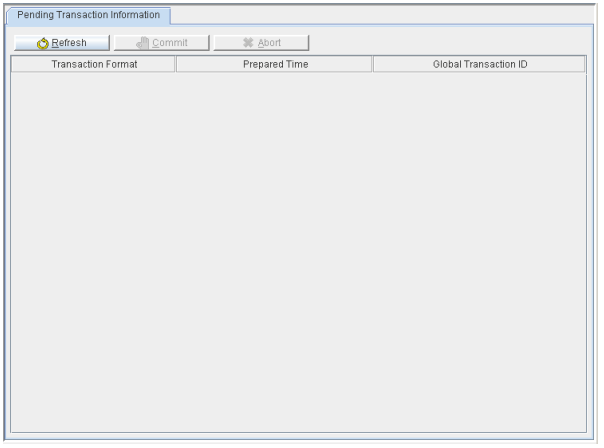
In a distributed database environment, transactions can be made across multiple databases. DBMaker effectively manage this sort of transaction using a two-phase commit method. The user inputs a command affecting the remote databases at the coordinator database. First, the coordinator database sends the command to the appropriate remote databases where they prepare to commit the transaction. After confirmation that all remote databases are prepared, the coordinator database sends a COMMIT transaction command to the remote databases, where the transaction is completed. It is possible that a

©Copyright 1995-2026 CASEMaker Inc.

network failure can occur between these two phases. The global transaction recovery daemon is responsible for logging all such errors and holding them as pending transactions until they can be committed at a later time. Pending transactions can be viewed using the JDBA Tool and can also be committed or aborted.

➲ Viewing pending transactions:

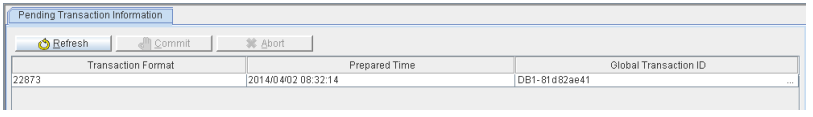
1. Select Pending Transaction from the Database Monitor node of the tree
2. A list of current pending transactions will appear.



3. Select the **Refresh** button to view the current pending transaction status.

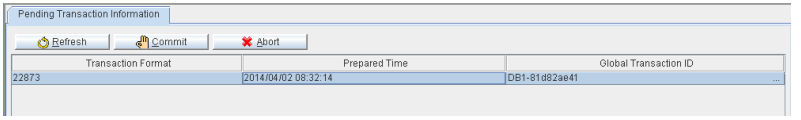
➲ Commiting pending transactions:

1. Select **Pending Transaction** from the **Database Monitor** node of the tree
2. A list of current pending transactions will appear.

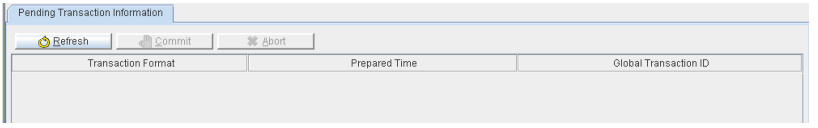


3. Select the transaction to commit from the list in the main console. It will highlight blue.

©Copyright 1995-2026 CASEMaker Inc.

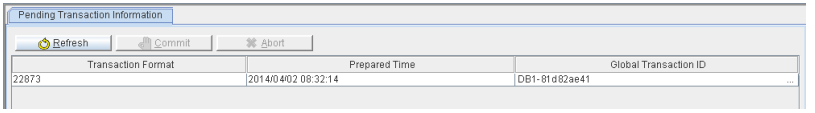


4. Click the **Commit** button at the top of the page. The transaction will commit and will no longer appear in the list. The next transaction will highlight blue.

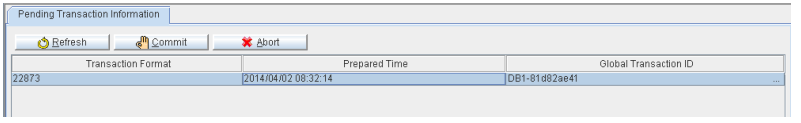


➲ Aborting pending transactions:

1. Select **Pending Transaction** from the **Database Monitor** node of the tree
2. A list of current pending transactions will appear.



3. Select the transaction to abort from the list in the main console. It will highlight blue.



4. Click the **Abort** button at the top of the page. The transaction will abort and will no longer appear in the list. The next transaction will highlight blue.

Monitoring System Configurations

During runtime, database settings may be altered. This makes it even more important for users, on the client site, to know the current server settings. Users can do this with either the JDBA tool or the SYSCONFIG system table.

The following table summarizes items found on the system configuration

©Copyright 1995-2026 CASEMaker Inc.

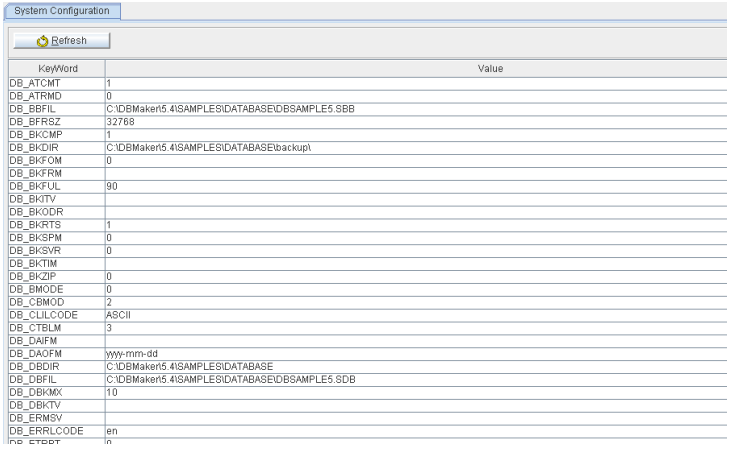
page:

ITEM	DESCRIPTION
Keyword	The keyword used in dmconfig.ini
Value	Current value of the setting

Table 28-16 System Configuration Items

➲ Viewing system configuration information:

1. From the **Database Monitor** node select **System Configuration**.
2. A list of current server system settings appears.



KeyWord	Value
DB_ATCMT	1
DB_ATRMD	0
DB_BBFIL	C:\DBMaker\5.4\SAMPLES\DATABASE\DBSAMPLE5.SBB
DB_BFRSZ	32768
DB_BKCMP	1
DB_BKDIR	C:\DBMaker\5.4\SAMPLES\DATABASE\backup\
DB_BKFOM	0
DB_BKFRM	
DB_BKFUL	90
DB_BKITV	
DB_BKODR	
DB_BKRTS	1
DB_BKSPM	0
DB_BKSVR	0
DB_BKTIM	
DB_BKZIP	0
DB_BMODE	0
DB_CBMOD	2
DB_CLILCODE	ASCII
DB_CTBLM	3
DB_DAIFM	
DB_DAOFM	yyyy-mm-dd
DB_DBDIR	C:\DBMaker\5.4\SAMPLES\DATABASE
DB_DBFIL	C:\DBMaker\5.4\SAMPLES\DATABASE\DBSAMPLE5.SDB
DB_DBKMX	10
DB_DBKTV	
DB_ERMSV	
DB_ERRLCODE	en

3. Click the **Auto Refresh** button to refresh the system configuration information.

Monitoring Update Statistics

The Update Statistics status node of the database monitor tree item gives access to database update statistics information stored in the **SQL_CMD** column of the SYSUSER system table. You can monitor details of the Update Statistics status used in a database's update statistics progress. This information includes the three parts: total, table and index. Total means all update statistics objects. Table means one table statistics, includes indexes

©Copyright 1995-2026 CASEMaker Inc.

belong to it and data page. Index means index statistics or data page statistics.

The following table summarizes the items on the Update Statistics page:

ITEM	DESCRIPTION
Connection ID	The objects ID of the Update Statistics
Total	All update statistics objects
Start Time	The start times of all update statistics objects
Execute Time	The execute time of all update statistics objects
Remain Time	The remain time of all update statistics objects
Complete Percent	The complete percent of all update statistics objects
Table Name	The name of one table statistics, includes indexes belong to it and data page
Start Time	The start times of one table statistics
Execute Time	The execute times of one table statistics
Remain Time	The remain times of one table statistics
Complete Percent	The complete percent of one table update statistics
Index Name	The names of one index statistics or data page statistics
Start Time	The start times of one index statistics
Execute Time	The execute times of one index statistics
Remain Time	The remain times of one index statistics
Complete Percent	The complete percent of one index update statistics

Table 28-17 Update Statistics Items

➲ Viewing Update Statistics information:

1. From the **Database Monitor** node select **Update Statistics**.
2. A list of current Update Statistics status information appears.

©Copyright 1995-2026 CASEMaker Inc.

3. Click the **Auto Refresh** button to refresh the Update Statistics information.

➲ Stopping the update statistics process:

1. From the **Database Monitor** node select **Update Statistics**.
2. A list of current Update Statistics status information appears.
3. Select a single session of list and click the **Abort** button.
4. A confirm dialog should be shown and ask if user wants to stop the update statistics process.

Monitoring Journal Status

The journal status node of the database tree item gives access to journal statistics stored in the SYSINFO system table. Journal status items are grouped into three categories of related items. Each category is grouped together on a tabbed page. The categories include:

- Journal Status
- Journal User
- Journal Graph

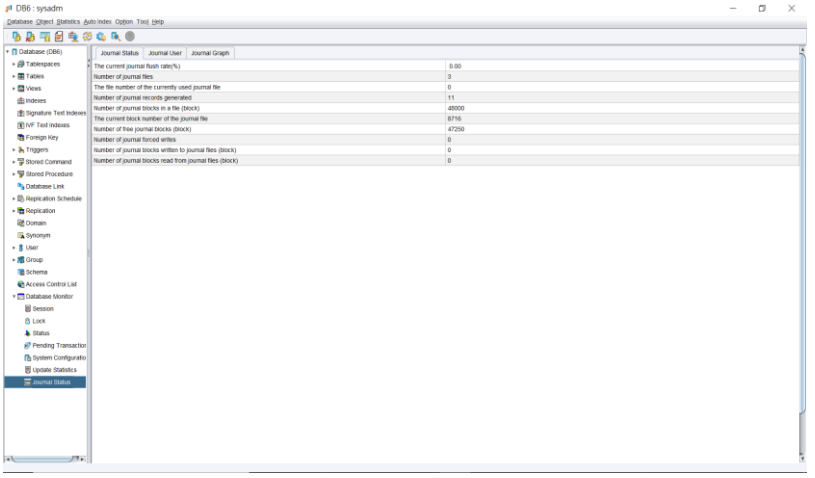
VIEWING JOURNAL STATUS

The journal status page displays operating system statistics from the SYSINFO table. Refer to Monitoring Database Status-VIEWING JOURNAL STATUS and Table 23-14 Journal Status items for more information.

➲ Viewing Journal Status:

1. Select the Database Monitor node from the tree. Seven nodes will appear, select the Journal Status node.
2. Select the Journal Status tab. The following page appears.

©Copyright 1995-2026 CASEMaker Inc.



VIEWING JOURNAL USER

You can monitor details of the users occupying journal in the database. The information includes user name, file number and block number. Table 23-18 summarizes the items on the journal user page.

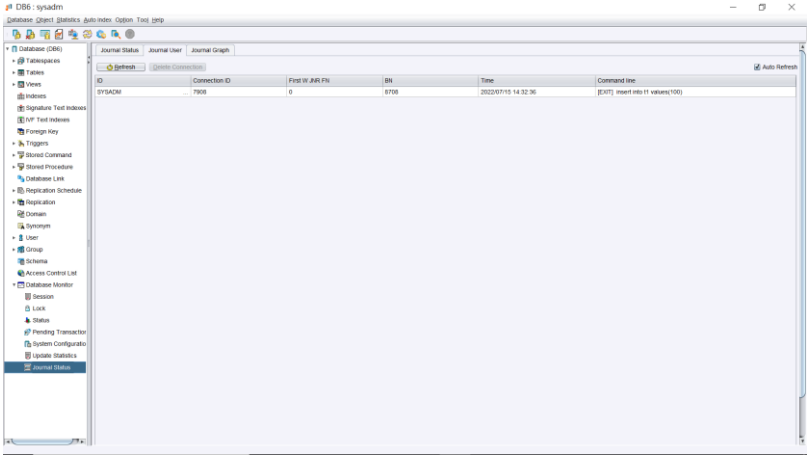
ITEM	DESCRIPTION
ID	Name of the user who's using journal
Connection ID	ID of the user
First W JNR FN	The first written journal number
BN	The first written block number
Time	The last command executed time
Blocks of journal used	Every user's journal used
Command line	The last command user execute

Table 24-18 Journal User items

➲ Viewing Journal User:

©Copyright 1995-2026 CASEMaker Inc.

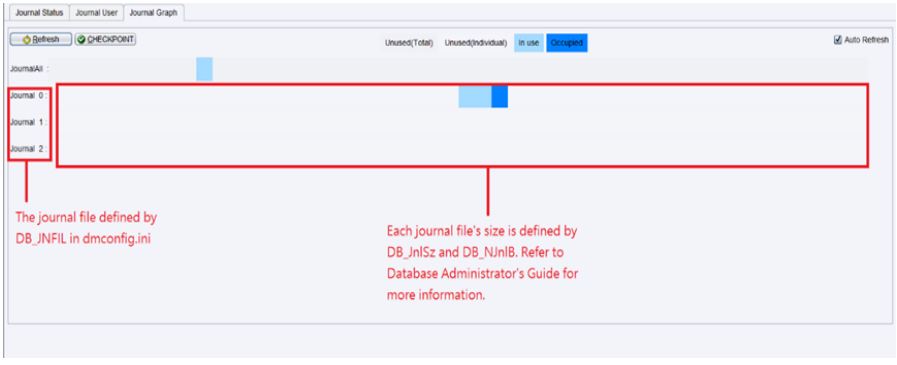
1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Journal Status** node.
2. Select the **Journal User** tab. The following page appears.



3. Click **Refresh** at any time to update the statistics shown on this page.
4. Click **delete connection** to delete the selected connection.

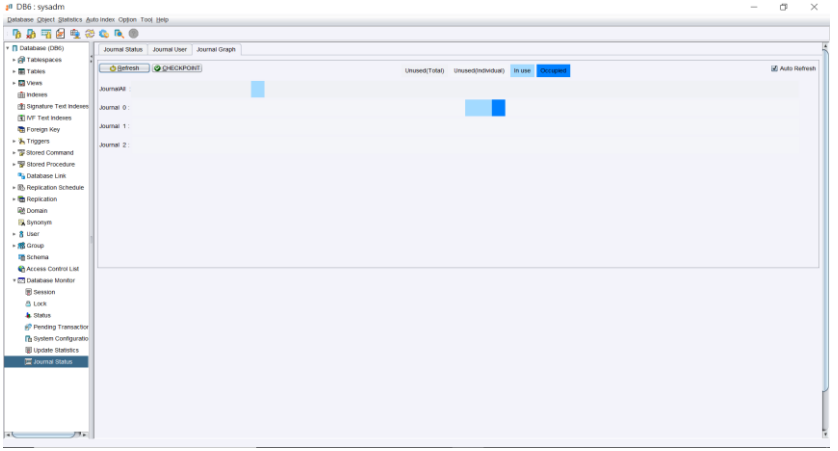
VIEWING JOURNAL GRAPH

Journal Graph displays a graph of the journal usage. This makes users can specify the occupied connection and delete it more easily.



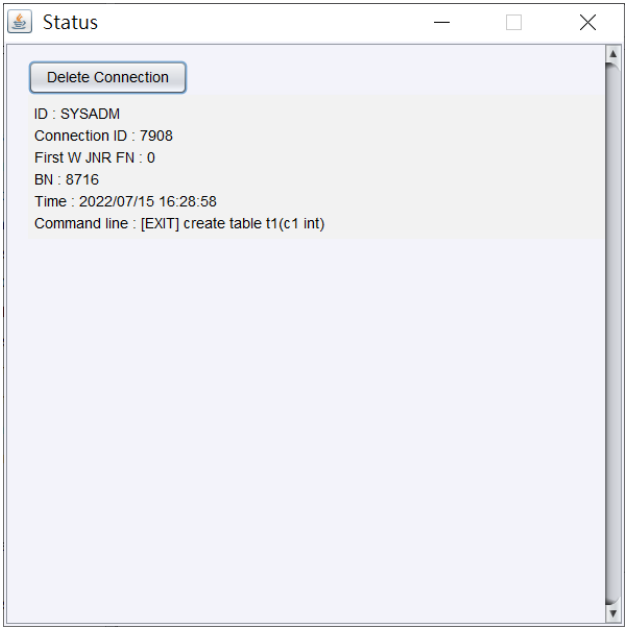
➲ Viewing Journal Graph:

1. Select the **Database Monitor** node from the tree. Seven nodes will appear, select the **Journal Status** node.
2. Select the **Journal Graph** tab. The following page appears.



3. Click **Refresh** at any time to update the statistics shown on this page.
4. Click **CHECKPOINT** to execute the checkpoint command. Refer to the SQL Command and Function Reference for more information.
5. Click the occupied connection (The dark blue bar). The following window will appear.

©Copyright 1995-2026 CASEMaker Inc.



6. Click **Delete Connection** to delete the occupied connection.

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

29 Setting JDBA Options

The JDBA Options menu gives access to the following menu items:

- Preferences: Provides a method for adjusting the way certain information is presented in the user interface. Preferences include language setting, auto-refresh, table, and SQL command display options.
- Run Time settings: The Run Time Settings window provides an interface for changing database configuration parameters while the database is running. Settings include Backup settings, file object settings, system control settings (including distributed database mode and autoextend tablespace options), and connection settings.
- Refresh: The Refresh option gets current database information to display on the JDBA window, and renews the screen.
- Query by Example Interface: The query by example (QBE) interface provides a streamline, user-friendly, and efficient interface for creating SQL query strings.
- Manage Transparent Data Encryption (TDE): The Manage Transparent Data Encryption (TDE) options provides an interface for altering TDE's password and set TDE status.

©Copyright 1995-2026 CASEMaker Inc.

29.1 Altering Run Time Settings

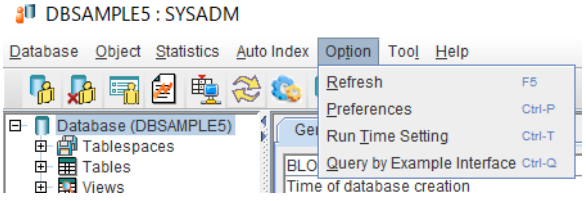
Certain database configuration parameters may be changed while the database is on-line. JDBA Tool provides an interface for changing these settings quickly. These settings include:

- **Backup settings**: Allows you to modify automatic full backup frequency and location, set file object backup mode, or disable the backup server daemon.
- **Differential and Incremental settings**: Allows you to modify automatic differential and incremental backup frequency.
- **File object settings**: Allows you to enable or disable user file objects and modify the location of system file objects.
- **System control settings**: Allows you to enable distributed database mode for synchronous table replication and set the number of pages or frames to extend autoextend tablespaces.
- **Log system settings**: Allows you to enable and set options for the Log System.
- **Character Set settings**: Allows client side users to select the language code for the database and error messages.
- **Statistics settings**: Allows you enable and set options for the Update Statistics.
- **Auto Index settings**: Allows you enable and set options for the Auto Index.
- **Schedule Daemon settings**: Only allows you set the schedule daemon interval and the number of tasks.
- **Connection settings**: Provides the option to block new connections.

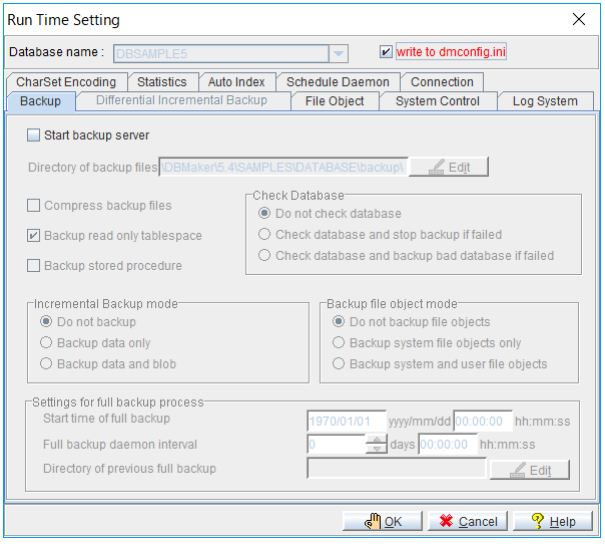
©Copyright 1995-2026 CASEMaker Inc.

➲ Accessing the Run Time settings window:

1. Select **Run Time Settings** from the **Option** menu on the menu bar.



2. The **Run Time Setting** dialog appears.



3. To use the updated settings in the next session, make sure that the **write to dmconfig.ini** check box is enabled.
4. To allow the updated settings to apply to the current session only, clear the checkmark in the **dmconfig.ini** check box.

NOTE *The **write to dmconfig.ini** option is only available on the server side.*

5. Set the following options:
   - Backup Settings
   - Differential and Incremental Settings

©Copyright 1995-2026 CASEMaker Inc.

- File Object Settings
- System Control
- Log System
- Character Set
- Statistics
- Auto Index
- Schedule Daemon
- Connection

Changing Backup Settings

The following backup settings may be altered during run-time: Start backup server, Directory of backup files, Compress backup files, Backup read only tablespace, Backup stored procedure, Check Database, Incremental Backup Mode, Backup file object mode, Settings for backup process. During run-time, backup modes can be changed but only to a state that backs up less data than does the current state. In the case of file objects, this exception does not apply. For example, this may be useful to free up resources by changing the backup mode from "backup data and BLOB" to "backup data".

©Copyright 1995-2026 CASEMaker Inc.

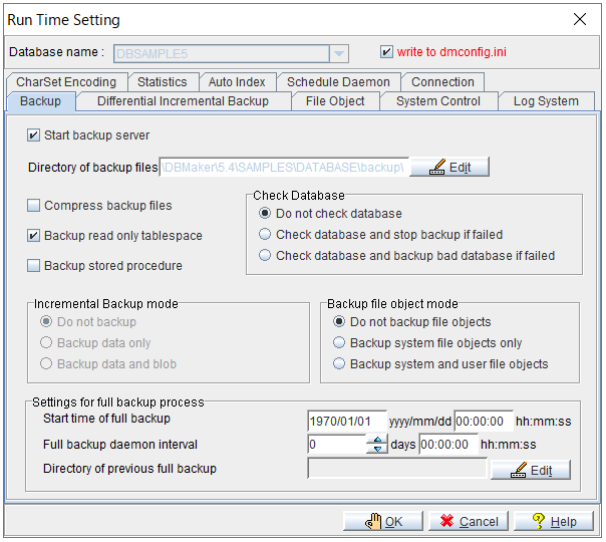


Figure 29-1 The Backup page of the Run Time Settings dialog

START BACKUP SERVER

Enabling the **Start Backup Server** check box activates the Backup Server daemon. Some settings can be changed during run-time. Do this from the **Backup** tab and the **Differential Incremental Backup** tab. Please note, settings for **Incremental Backup Mode** cannot be adjusted during run-time. All other items found in the **Backup** and **Differential Incremental Backup** tabs are disabled.

COMPRESS BACKUP FILES

Selecting **Compress Backup files** activates backup file compression. When activated, the Backup Server compresses backup files during full backups.

BACKUP READ-ONLY TABLESPACE

Selecting this check box tells Backup Server to backup read-only tablespace files when performing a full backup. Deselecting this check box tells Backup Server to skip backup of read-only tablespace files. Please note, a full backup

©Copyright 1995-2026 CASEMaker Inc.

must be performed after setting a tablespace to read-only. If you don't backup the latest files, serious errors may result when restoring, from a full backup, databases containing a read-only tablespace.

Backup Stored Procedures

Selecting this check box tells Backup Server to backup stored procedures when performing a full backup. Deselecting this check box tells Backup Server to skip backup of stored procedures.

Directory of Backup Files

The Backup Server stores back up files such as journal files in the backup directory. The backup directory sand the database files should not be located on the same disk. The default path for backup files is the database directory */backup* and DBMaker automatically creates it. The total length of the backup directory path must not exceed 255 characters.

Check Database

You can choose to check databases before full or differential backups. The following table describes how the check modes operate.

Check Mode	Description
Do not check database	Backup Server does not check databases before full backup or differential backups.
Check database and stop backup if failed	Backup Server checks databases before full and differential backups. Error messages are logged and the backup is terminated when a database is damaged.
Check database and backup bad database if failed	Backup Server checks databases before full and differential backups. Error message are logged and when a damaged database is first detected, the backup continues with the damaged files copied to the BKDIR/BADDB directory. During subsequent back up operations, Backup

©Copyright 1995-2026 CASEMaker Inc.

CHECK MODE	DESCRIPTION
	Server logs error messages but does not copy damaged databases. Damaged databases that subsequently report no damage are backed up normally and Backup Server removes the damaged copy of the database from the BKDIR/BADDB directory.

Table 29-1 Check Database Modes

Incremental backup files are always generated follow full and differential backups. Backup Server places incremental backup files associated with a damaged database into BKDIR/BADDB. The incremental backup files are moved back to BKDIR when the associated database is no longer damaged.

INCREMENTAL BACKUP MODE

You can choose to turn off the incremental backup mode, or turn off just backup BLOB mode. Run time settings only allow for deactivation of the incremental backup mode, either deactivating backup BLOB, or deactivating backup data.

To change the backup mode to Backup Data or Backup Data and BLOB you must shut down the database, start the database, select the **Setup** button, and make the changes to the **Backup** page of the **Start Database Advanced Settings** window.

The following backup modes are available.

BACKUP MODE	DESCRIPTION
Do not backup	Disables the incremental backup daemon. In this mode, journal files are not backed up.
Backup data only	Sets the incremental backup daemon to backup all data journal pages.
Backup data and blob	BLOB data is written to the journal and the incremental backup daemon backs

©Copyright 1995-2026 CASEMaker Inc.

BACKUP MODE	DESCRIPTION
	up all journal files.

Table 29-2 Incremental backup Modes

FILE OBJECT BACKUP MODE

You can choose what types of file objects to backup during incremental backups by changing the file object backup mode. The following file object backup modes are available.

FO BACKUP MODE	DESCRIPTION
Do not backup file objects	No file objects are backed up during full or incremental backups
Backup system file objects only	System file objects are backed up during any full or incremental backup.
Backup system and user file objects	System and user file objects are backed up during any full or incremental backup.

Table 29-3 File Object Backup Modes

START TIME OF FULL BACKUP

The date and time values that you set in the **yyyy/mm/dd** and **hh:mm:ss** fields specifie when the database's first full backup will begin. Please careful input this two values.

FULL BACKUP DAEMON INTERVAL

This value specifies, in days, the full backup time interval. The adjacent field accepts a value, in hours, minutes, and seconds, and specifies this portion of the full backup time interval. The total time interval between full backups is determined by adding the two values. For example, inputting 1 for the **days** field and inputting 12:00:00 for the **hh:mm:ss** field specifies a full backup execution every day and a half.

DIRECTORY OF PREVIOUS FULL BACKUP

©Copyright 1995-2026 CASEMaker Inc.

This Directory of previous full backup specifies the directory where backup server saved previous versions of full backup files.

➲ Changing backup settings:

1. Select the **Start backup server** check box.
2. Select the **Compress backup files** check box to enable backup compression.
3. Select the **Backup read-only tablespace** check box to include read-only tablespace files when performing a full backup.
4. Select the **Backup stored procedure** check box to backup stored procedures when performing a full backup.
5. Enter a path into the **Directory of backup files** field or click the adjacent **Edit** button Edit to specify a destination for the backup files.
6. Select one of the check modes from the **Check Database** box.
   - Select **Do not check database** to skip checking databases before full and differential backups.
   - Select **Check database and stop backup if failed** to check database before full and differential backups. Additionally, with this option, Backup Server logs error messages, and upon discovering a damaged database, terminates the backup operation.
   - Select **Check database and backup bad database if failed** to check database before full and differential backups. Additionally, Backup Server logs error messages but continues to back up damaged databases, placing them in the *BKDIR/BADDB* directory.
7. Select a backup mode from the **incremental backup** mode box.
   - To change the backup mode to backup data only, select the **Backup data only** option button.
   - To change the backup mode to backup data and blob, select the **Backup data and blob** option button.
   - To turn off the backup mode, select the **Do not backup** option

©Copyright 1995-2026 CASEMaker Inc.

button.

8. Select a file object backup mode from the **Backup file object mode** box.
   - Select **Do not backup file objects** to exclude file objects from the backup.
   - Select **Backup system file objects only** to backup system file objects.
   - Select **Backup system and user file objects** to backup system and user file objects.
9. Establishing settings for a full backup process from the **Settings for full backup process** area.
   - Use the **Start time of full backup** fields to specify a date and a time when full backups will begin.
   - Enter the number of days, hours, minutes, and seconds between each successive full backup in the **Full backup daemon interval** fields.
   - Enter a path in the **Directory of previous full backup** field or select the adjacent **Edit** button to specify a destination for Backup Server to put previous version of full backup files.
10. Select one of the other tabs at the top (to change other runtime settings) or click the **OK** button at the bottom of the Run Time Settings window to save your changes.

Changing Differential and Incremental Backup Settings

Differential and incremental backup settings page allows users to specify details for their differential and incremental backup processes.

©Copyright 1995-2026 CASEMaker Inc.

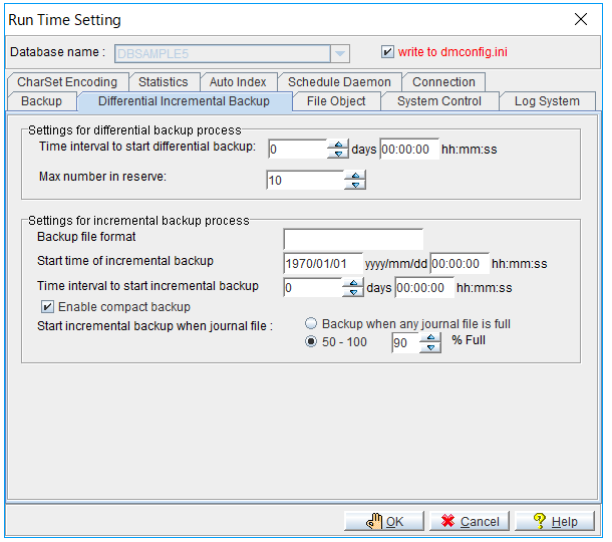


Figure 29-2 The Differential and Incremental Backup page of the Run Time Settings dialog

Time Interval to Start Differential Backup

This number specifies, in days, the interval at which differential backups occur. Beside the combo box is a field to input an hours, minutes and seconds time value. The sum of these three values determines the total time interval between differential backups. For example, entering 1 in the **days** field and entering 12:00:00 in the **hh:mm:ss** field will cause Backup Server to execute a differential backup every day and a half.

Max Number in Reserve

The value specifies the maximum number of differential backups, after a full backup (i.e., the differential base), that Backup Server retains. When the number of saved differential backups exceeds this value, Backup Server removes the oldest differential backup. The default value is ***10***. Set this value to 0 to save an unlimited number of differential backups.

©Copyright 1995-2026 CASEMaker Inc.

Backup File Format

This value specifies the naming format Backup Server uses for incremental backup journal files. The backup filename format may include both text constants and format sequences (e.g., escape sequences), that represent special character strings.

Start Time of Incremental Backup

This value specifies when the database's first incremental backup begins. Enter your desired date in the **yyyy/mm/dd** field and time in the **hh:mm:ss** field.

Time Interval to Start Incremental Backup

This number specifies, in days, the interval at which incremental backups occur. Beside the combo box is a field to input an hours, minutes and seconds time value. The sum of these three values determines the total time interval between incremental backups. For example, entering 1 in the **days** field and entering 12:00:00 in the **hh:mm:ss** field will cause Backup Server to execute a incremental backup every day and a half.

Enable Compact Backup

Enable Compact Backup Mode ensures that Backup Server only copies needed journal files when performing incremental backups. As a result, less storage space is required at the backup location, however, restoring a database may take more time.

Start Incremental Backup when Journal File

DBMaker can initiate an incremental backup before the journal file is full. Backup Server performs an online incremental backup when a journal file fills to the percent value specified by the journal trigger. Journal trigger values between 50 and 100 are valid. You can combine a journal trigger value with the backup schedule to ensure a regular schedule of database backups.

©Copyright 1995-2026 CASEMaker Inc.

➲ Changing differential and incremental backup settings:

1. Setting up a differential backup process from the **Settings for differential backup process** area.
   - Enter the number of days, hours, minutes and seconds between each successive full backup in the **Time interval to start differential backup** time fields.
   - Enter a maximum value in the **Max number in reserve** fields.
2. Setting up an incremental backup process from the **Settings for incremental backup process** area.
   - Enter a format the Backup Server will use to name incremental backup journal files.
   - Indicate a date and a time in the **Start time of incremental backup** fields to specify when incremental backups will begin.
   - Enter the number of days, hours, minutes and seconds between each successive full backup in the **Time interval to start incremental backup** time fields.
3. To enable compact backups, select the **Enable Compact Backup** check box.
4. Incremental backups can automatically execute when journal files have filled to a predetermined percent. Next to **Start incremental backup when journal file**:
   - Select the **Backup when any journal file is full** option button to set incremental backups to execute when any journal file is full.
   - Enter a value from 50 to 100 in the **50 – 100 %** field to set incremental backups to execute when any journal file has filled to the value entered.

Changing File Object Settings

Selecting the **File Object** tab in the **Run Time Settings** window allows the user to enable external user file objects and change the storage location of

©Copyright 1995-2026 CASEMaker Inc.

system file objects. DBMaker can automatically create subdirectories within the FO directory. Each subdirectory is filled with new file objects up to a threshold value. When the threshold is reached, DBMaker creates a new FO subdirectory.

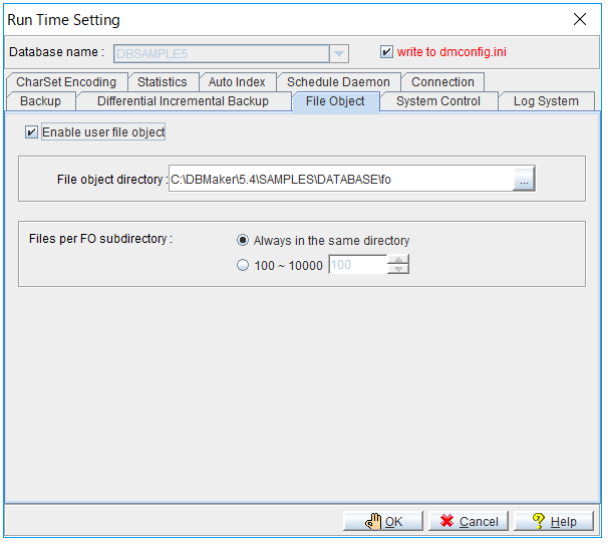


Figure 29-3 The File Object page of the Run Time Settings dialog

➲ Changing file object settings:

1. Select the **File Object** tab from the **Run Time Setting** window.
2. To enable the database to use external file objects, click on the **Enable User File Object** check box.
3. Enter a path into or select the **browse** button .. next to the **File Object Directory** field to indicate the location of the System File Object Directory.
4. Select from **Files per FO Subdirectory**:

   - To have DBMaker always store system file objects in the system file object directory, select always in the same directory.
   - To have DBMaker create subdirectories with a set number of file objects in each subdirectory, select the option button next to 100

©Copyright 1995-2026 CASEMaker Inc.

~10000 and enter the threshold number of file objects (between 100 and 10000) at which a subdirectory is declared full and a new one created.

5. Select one of the other tabs at the top (to change other runtime settings) or select **OK** from the bottom of the **Run Time Settings** window.

Changing System Control Settings

The system control page allows settings to be changed that directly affect database performance. It also allows distributed mode to be enabled during the runtime and allows for the number of pages to extend a file when an autoextend tablespace is full.

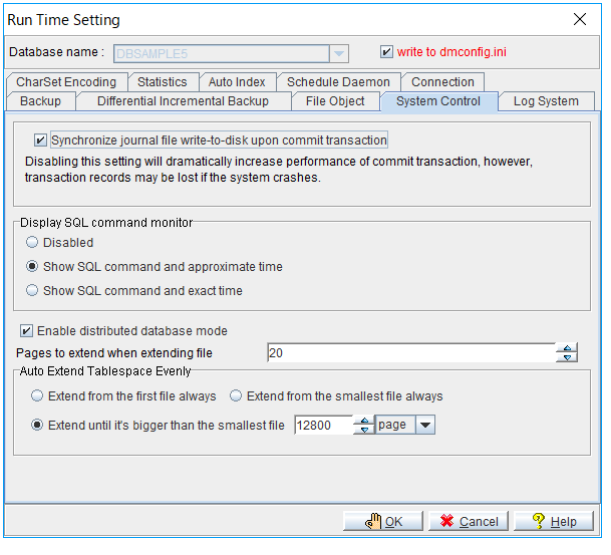


Figure 29-4 The System Control page of the Run Time Setting dialog

JOURNAL FILE SYNCHRONIZATION

Under normal operating conditions, when a transaction is committed, DBMaker initiates a two-step process with regards to the journal blocks involved in the transaction. The first step is to flush the relevant journal blocks

©Copyright 1995-2026 CASEMaker Inc.

into the file system of the operating system, or the operating system's cache. The second step is to force the operating system to write these journal blocks to disk, so that the transaction is preserved in the event of a system crash.

Omitting the second step may greatly increase the performance of the commit transaction process. The journal blocks in this case remain in the operating system's cache until the operating system writes them to disk. However, if the system crashes at this point, the records of any transactions that were still in the operating system's cache will be lost.

Display Mode of SQL Command Monitor

Display Mode affects the display content of the **SQL_CMD** and **TIME_OF_SQL_CMD** columns in the **SYSUSER** system table. JDBA Tool Users can view information about users accessing the database with the Database Monitor function. **Session Information** displays users currently connected to the database. The columns *Current SQL command* and *Time of Current SQL Command* display the most recent SQL transaction committed by the user and the time of execution, respectively. **Display Mode of SQL Command Monitor** sets the configuration for how these attributes is displayed. No SQL commands are shown if **Disabled** is clicked. The most recent SQL command executed by the user and its approximate time of execution are shown if **Show SQL command and approximate time** is clicked. The most recent SQL command executed by the user and its exact time of execution are shown if **Show SQL command and exact time** is clicked. Displaying the exact time of execution uses more CPU resources and slows down the database. The default mode is ***Show SQL command and approximate time*.**

Enable Distributed Database Mode

This setting enables use of the database in distributed mode. It must be enabled for synchronous table replication to work. Synchronous table replication is set using the JDBA Tool. For more information on distributed data, synchronous table replication, or coordinator and participant databases, refer to the *JDBA Tool User's Guide*, or the *Database Administrator's Guide*.

©Copyright 1995-2026 CASEMaker Inc.

PAGES TO EXTEND WHEN EXTENDING A FILE

When all pages in a data file or BLOB file are full, DBMaker can automatically extend the number of pages or frames in the file to allow the database to grow. The **Pages to extend when extending file** setting tells DBMaker how many pages or frames to add to the full file in the event that it is full. Select a greater number, to reduce the frequency of file appendages, if the database administrator expects that the database will grow very quickly. One page can be 4 KB, 8 KB, 16 KB, or 32 KB.

AUTO EXTEND TABLESPACE EVENLY

When a tablespace is full, DBMaker automatically extends the number of pages or frames in the file to allow the tablespace to grow. The **Extend from the first file always** setting tells DBMaker automatically expand the tablespace from the first file always, it can obtain good performance, but all files in the tablespace are not be expanded evenly. The **Extend from the smallest file always** setting keeps all files in a tablespace in even expansion, but performance may be bad because all rows of a table may be scattered into all files in turn. The **Extend until it's bigger than the smallest file** setting expands from the smallest file firstly, after that, continue to expand this file and does not switch to the second smallest file until the current file's size is bigger than the sum of the second smallest file's size and the value of **DB_ExtHd**. In this way, both performance and balance of file size can be taken into account.

➲ Changing system settings:

1. To disable synchronized journal block writing to disk, remove the check mark from the **Synchronize journal file write-to-disk upon commit transaction** checkbox.
2. Select a setting for the SQL command monitor:
   - Select **Disabled** to not display any SQL commands.
   - Select **Show SQL command and approximate time** to show the most recent SQL command executed by the user and its approximate time of execution.

©Copyright 1995-2026 CASEMaker Inc.

- Select **Show SQL command and exact time** to show the most recent SQL command executed by the user and its exact time of execution.

3. To enable distributed database mode, check the **Enable Distributed Database Mode** box.
4. Enter the number of pages to extend a file in the **Pages to extend when extending file** field .
5. Select a setting for the **Auto Extend Tablespace Evenly**.

   - Select **Extend from the first file always** to expand the tablespace from the first file always.
   - Select **Extend from the smallest file always** to expand the tablespace from the smallest file always.
   - Select **Extend until it's bigger than the smallest file** field to expand a tableapce from the smallest file firstly, after that, continue to expand this file and does not switch to the second smallest file until the current file's size is bigger than the sum of the second smallest file's size and the specified value. The setting can be altered by inputting a value in the field or by using the arrows to the right of the field. There are three unit options for page size: Page, M (megabytes), and G (gigabytes).
6. Select one of the other tabs at the top (to change other runtime settings) or select **OK** from the bottom of the **Run Time Setting** window.

Changing Log System Settings

Selecting the **Log System** tab in the **Run Time Settings** window allows the user to set the log correlative setting. The options on this page apply only to the server.

©Copyright 1995-2026 CASEMaker Inc.

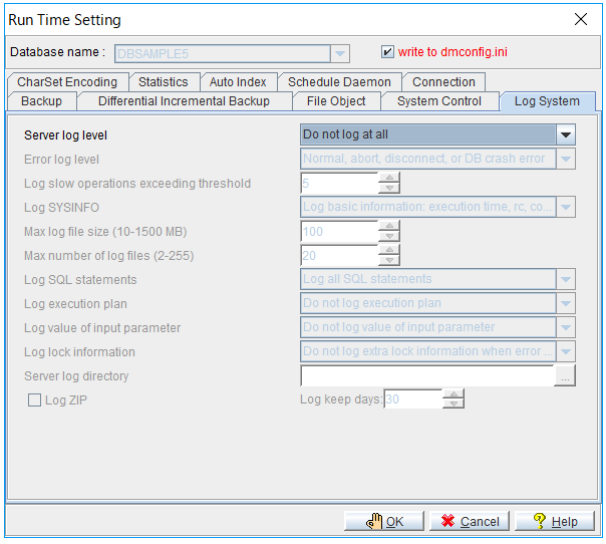


Figure 29-5 The Log System page of the Run Time Setting dialog

SERVER LOG LEVEL

Users specify when and what operations are logged according to set the Server log level. DBMaker provides seven log levels as follows:

- **Do not log at all**: disables the log system
- **Log error, see DB_LGERR for details**: logs errors and the default errors according to the logging level specified in **Error log level**
- **Log slow operation, see DB_LGSTM for details**: logs operations that exceed the threshold specified by **Log slow stmt over the number of seconds**
- **Log error and slow operation**: logs both errors (see **Log error, see DB_LGERR for details** above) and slow operations (see **Log slow operation, see DB_LGSTM for details** above)
- **Log SQL, see DB_LGSQL for details**: logs connects, disconnects, commits, rollbacks, SQL command executions, errors, and slow

©Copyright 1995-2026 CASEMaker Inc.

operations

- **Log all operation on exit**: logs all operations during the exit process
- **Log all operation on startup/exit**: logs all operations during the enter process and the exit process.

Error Log Level

Users can specify the level of detail that will be recorded when server logging is enabled as specified by the **Server log level** setting including log error. The **Error log level** indicates what kind of errors will be logged according to the five levels described here:

- **Core dumped or DB crash error:** logs all errors that occur having error codes > 30000
- **Disconnect error or DB crash error:** logs all errors that occur having error codes > 20000
- **Abort, disconnect or DB crash error:** logs all errors that occur having error codes > 10000
- **Normal, abort, disconnect or DB crash error:** logs all errors that occur having error codes > 100
- **Warning or other error:** logs all errors (i.e., all errors that occur having error codes > 0)

This setting corresponds to the **DB_LgErr** keyword in the **dmconfig.ini** file. The default level is **Normal, abort, disconnect or DB crash error**.

Log Slow Stmt Over The Number Of Seconds

Users can specify the threshold value in seconds to record statements that execute too slowly (i.e., execution time exceeds the specified threshold) when server logging is enabled as specified by the **Server log level** setting including log slow statement. Users can enter a value into the field or adjust the field's value using the up and down arrows located to the right of the field.

©Copyright 1995-2026 CASEMaker Inc.

LOG SYSINFO

Users can specify what system information will be logged using **Log SYSINFO** when server logging is enabled. There are three SYSINFO logging settings for recording different levels of information detail:

- **Log basic information: execution time, rc, connection id, etc.**
- **Log above information plus SYSUSER and SYSINFO information**
- **Log above information plus SYSTEM memory information if detectable**

MAXIMUM LOG FILE SIZE

This setting specifies, in megabytes, the maximum file size for log and text files. DBMaker stores the log as DBNAME_1.LOG. When the log file size reaches the specified value, subsequent logs are generated as follows: DBNAME_2.LOG, DBNAME_3.LOG, … , DBNAME_n.LOG, where n is the value specified by **DB_LgFNo** or when **DB_LgFNo** is unspecified the default value of 20 is used. Log files are over written starting from DBNAME_1.LOG after the final log file reaches the maximum file size. Users can enter a value into the field or adjust the value using the up and down arrows located to the right of the field.

MAXIMUM NUMBER OF LOG FILES

This setting specifies the number of the log files that are generated before the log rolls-over (i.e., the first log file is overwritten with new log information). DBMaker stores the log as DBNAME_1.LOG. When the log file size reaches the default value of 100 MB or the size specified by **DB_LgFSz**, the log information is logged to a subsequent log file as follows: DBNAME_2.LOG, DBNAME_3.LOG, … , DBNAME_n.LOG where n is the number of log files. The log information will be written to next log file until either the default of 20 log files have been generated or the number of log files generated is equal to the value specified by this setting. At this point, the log rolls-over. In other words, log files are erased and recreated with the new log information beginning with log file DBNAME_1.LOG. Users can enter a value into the field or adjust the

©Copyright 1995-2026 CASEMaker Inc.

existing value using the up and down arrows located to the right of the field.

LOG SQL STATEMENT

Users can specify what system information will be logged using **Log SQL** when server logging is enabled for **Log SQL, detail in DB_LgSQL(DB_LgSvr= 4)**. There are three SQL logging settings for recording different levels of information detail:

- **Do not log SQL statements**
- **Log non-select SQL statements**
- **Log all SQL statements**

LOG EXECUTION PLAN

This setting specifies whether to log the execution plan for select, update, and delete statements. There are two logging options:

- **Do not log execution plan**
- **Log execution plan**

LOG INPUT PARAMETER'S VALUE

User can specify whether to log the input parameter's value. DBMaker provides five logging options. Each offers a different amount of information detail:

- **Do not log value of input parameter**
- **Log value of input parameter**
- **Log value of input parameter and SQL commands executed by stored procedures**
- **Log trigger's SQL statement and parameter**
- **Log both stored procedure and trigger's SQL statement and parameter**

©Copyright 1995-2026 CASEMaker Inc.

LOG LOCK INFORMATION

User can specify whether to log lock timeout and deadlock events when the log server is enabled for error logging. DBMaker provides two logging options each offering a different amount of information detail:

- **Do not log extra lock information when error code is lock time out or deadlock**
- **Log extra lock time out information**

SERVER LOG DIRECTORY

The directory path of the log files can be specified according to a user's needs. It may be entered manually or selected by using the **browse** button ( ). Selecting the **browse** button ( ) opens a dialog box.

LOG ZIP

Consider that the number of log files might grow, so packing/zipping the earlier closed log files would be necessary in order to save some storage. DBMaker provides the log zip function to compress the earlier log files.

LOG KEEP DAYS

Server log can include current date in the log filename, so the log filename would be unique and will not be overwritten. User can specify the number of days that the log files will be kept available. The expired log file would be removed by the daemon service.

➲ Changing the Log System Settings:

1. Select a Level for Log Server:
   - Select **Do not log at all** to shut down the Log System function
   - Select **Log errors, see DB_LGERR for details** to log the error informations
   - Select **Log slow operations, see DB_LGSTM for details** to log the operations those runs too slowly

©Copyright 1995-2026 CASEMaker Inc.

- Select **Log errors and slow operations** to log both the errors and slow operations
- Select **Log SQL, see DB_LGSQL for details** to log the SQL command, connection, disconnections, commit,rollback, errors and slow operations
- Select **Log all operations on exit** to log all operations when exit from database
- Select **Log all operations on startup/exit** to log all operations when both connection and disconnect the server

2. Select a error log level for logging errors:
   - Select **Core dumped or DB crash error** to log the error which code > 30000
   - Select **Disconnect error or DB crash error** to log the error which code > 20000
   - Select **Abort, disconnect, or DB crash error** to log the error which code > 10000
   - Select **Normal, abort, disconnect, or DB crash error** to log the error which code > 100
   - Select **Warning or other error** to log all errors which code > 0
3. Specify the threshold for the slow operations. The unit is second.
4. Select Log SYSINFO options:
   - Select **Log basic information: execution time, rc, connection id, etc.** to log the basic system information.
   - select **Log above information plus SYSUSER and SYSINFO information** to log SYSUSER and SYSINFO as well as basic information
   - Select **Log above information plus SYSTEM memory information if detectable** to log both the above and the SYSTEM

©Copyright 1995-2026 CASEMaker Inc.

memory information.

5. Specify the maximum size of the Log file and the unit is MB.
6. Specify the maximum numbers of the Log file.
7. Select the Log SQL action:
   - Select **Do not log SQL statements** to do not log the SQL statement
   - Select **Log non-select SQL statements** to log the SQL statement except the SELET statement
   - Select **Log all SQL statements** to log all kinds of SQL statement
8. Specify whether to log the execution plan.
9. Specify whether to log the input parameter's value:
   - Do not want to log the input parameter's value, select **Do not log value of input parameter**
   - Select **Log value of input parameter** to log the input parameter's value
   - Select **Log value of input parameter and SQL commands executed by stored procedures** to log both input parameter's value and the stored procedure's SQL command
   - Select **Log trigger's SQL statement and parameter** to log both input parameter's value and the trigger's SQL command
   - Select **Log both stored procedure and trigger's SQL statement and parameter** to log both input parameter's value and the stored procedure and the trigger's SQL command
10. Specify whether to log the lock information when time out or deadlock.
11. Specify the log directory in the Server Log Directory filed or click the **borwse** button (...) to select a path.
12. Specify whether to zip the earlier log file to save the storage. Select the check box to start the zip function.
13. Input how many days the log file can be kept.

©Copyright 1995-2026 CASEMaker Inc.

Changing Character Set Encoding Settings

The **CharSet Encoding** tab found on the **Run Time Settings** window allows client-side users to set a database's language code and error message code.

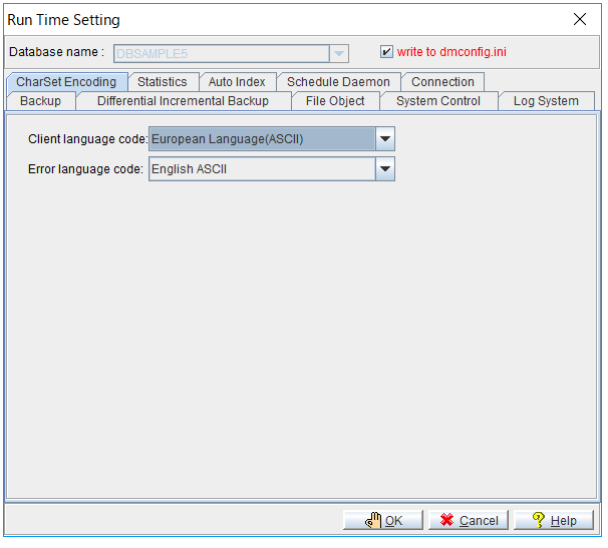


Figure 29-6 The CharSet Encoding page of the Run Time Setting dialog

CLIENT LCODE

Client-side users can choose a language code for connecting to UTF-8 Database Servers (i.e., the Database Server's Lcode is set to **10**) when using multilingual database. The client-side language codes are:

- DEFAULT
- European Language (ASCII)
- European Language (ISO-8859-1)
- European Language (ISO-8859-2)
- European Language (ISO-8859-3)

©Copyright 1995-2026 CASEMaker Inc.

- European Language (ISO-8859-4)
- European Language (ISO-8859-5)
- European Language (ISO-8859-7)
- European Language (ISO-8859-9)
- European Language (ISO-8859-10)
- European Language (ISO-8859-13)
- European Language (ISO-8859-14)
- European Language (ISO-8859-15)
- European Language (ISO-8859-16)
- European Language (KO18-R)
- European Language (KO18-U)
- European Language (KO18-RU)
- European Language (CP1250)
- European Language (CP1251)
- European Language (CP1252)
- European Language (CP1253)
- European Language (CP1254)
- European Language (CP1257)
- European Language (CP850)
- European Language (CP866)
- European Language (MACROMAN)
- European Language (MACCENTRALEUROPE)
- European Language (MACICELAND)
- European Language (MACCROATIAN)

©Copyright 1995-2026 CASEMaker Inc.

- European Language (MACROMANIA)
- European Language (MACCYRILLIC)
- European Language (MACUKRAINE)
- European Language (MACGREEK)
- European Language (MACTURKISH)
- European Language (MACINTOSH)
- Semitic Language (ISO-8859-6)
- Semitic Language (ISO-8859-8)
- Semitic Language (CP1255)
- Semitic Language (CP1256)
- Semitic Language (CP862)
- Semitic Language (MACHEBREW)
- Semitic Language (MACARABIC)
- Japanese (SHIFT-JIS)
- Japanese (SHIFT-JP)
- Japanese (CP932)
- Japanese (ISO-2022-JP)
- Japanese (ISO-2022-JP-1)
- Japanese (ISO-2022-JP-2)
- Chinese (GBK)
- Chinese (GB18030)
- Chinese (BIG5)
- Chinese (EUC-CN)
- Chinese (EUC-TW)

©Copyright 1995-2026 CASEMaker Inc.

- Chinese (CP936)
- Chinese (CP950)
- Korean (EUC-KR)
- Korean (CP949)
- Korean (JOHAB)
- Georgian (GEORGIAN-ACADEMY)
- Georgian (GEORGIAN-PS)
- Tajik (KO18-T)
- Kazakh (PT154)
- Thai (TIS-620)
- Thai (CP874)
- Thai (MACTHAI)
- Laotian (MULELAO-1)
- Laotian (CP1133)
- Vietnamese (VISCII)
- Vietnamese (TCVN)
- Vietnamese (CP1258)
- Unicode (UTF-8)

ERROR LCODE

In multilingual database, client side can set their own output locale codes of error message. They can be:

- DEFAULT
- English ASCII
- English ISO-8859-1

©Copyright 1995-2026 CASEMaker Inc.

- English ISO-8859-2
- English ISO-8859-5
- English ISO-8859-7
- English UTF-8
- Japanese Shift-JIS
- Japanese EUC
- Japanese UTF-8
- Simple chinese GBK
- Simple chinese GB18030
- Simple chinese UTF-8
- Traditional chinese Big5
- Traditional chinese UTF-8

➲ Changing the Character Set setting:

1. Select the **Client language code**.
2. Select the **Error language code** for the client side.

Changing Statistics Settings

Selecting the **Statistics** tab in the **Run Time Settings** window allows the user to enable update statistics server daemon and set the process of statistics.User also can set the update statistics data page sample in the **Sample ratio** field.

©Copyright 1995-2026 CASEMaker Inc.

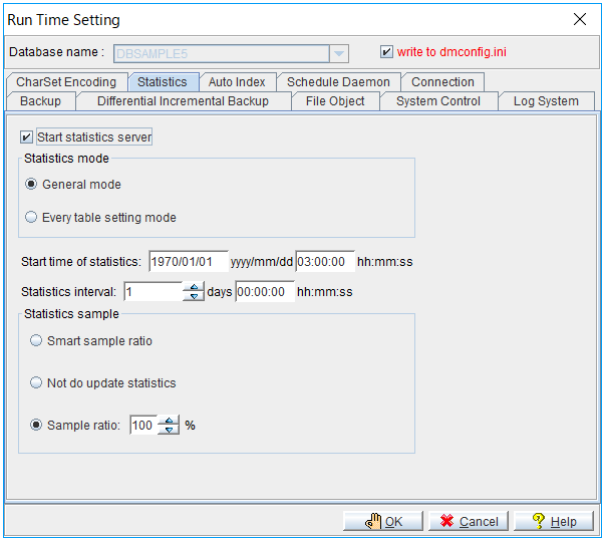


Figure 29-7 The Statistics page of the Run Time Setting dialog

START STATISTICS SERVER

DBMaker keeps statistical information about database objects in the system tables. Periodically the database needs to be scanned for recalculating the statistics. The updated statistics are then written to the system tables to ensure efficient database operation. Enabling **Update statistics** ensures that DBMaker automatically and periodically recalculates database statistics according to an internal scheduling daemon.

Selecting the **Start statistics server** check box enables the automatically update statistics server daemon. This keyword corresponds to the **DB_StSvr** keyword in the **dmconfig.ini** file. The default setting is ***enabled***.

STATISTICS MODE

The settings in the **Statistics** Mode field specify the Updated Statistics daemon mode for a database. Selecting **general mode** enables startup update statistics daemon in general mode and the sample ratio will be decided by value of the keyword **DB_StsSp** in **dmconfig.ini**. Selecting **every table setting mode**

©Copyright 1995-2026 CASEMaker Inc.

enables startup update statistics daemon in every table setting mode, that is to say, the sample ratio will be decided by every table's mode and sample ratio. Users can set every table's mode and sample ratio with the command "UPDATE STATISTICS SET".

START TIME OF STATISTICS

Specify when the first update statistics will begin for the database by entering a date in the **yyyy/mm/dd** field and a time in the **hh:mm:ss** field. This setting corresponds to the **DB_StsTm** keyword in the **dmconfig.ini** file. Default value is ***1970-01-01 03:00:00***.

STATISTICS INTERVAL

The number in the combo box specifies, in days, the time interval at which the Update Statistics Daemon is activated. Next to the combo box is a field for time input, which specifies the time interval in hours, minutes, and seconds. The total time interval is determined by adding the two values together. For example, inputting 1 in days and inputting 12:00:00 in **hh:mm:ss** causes the Update Statistics Daemon to be activated every day and a half. The number of days can be manually entered into the combo box, or increased or decreased by clicking the arrows to the right; the number of hours, minutes, and seconds are manually entered. This setting corresponds to the **DB_StsTv** keyword in the **dmconfig.ini** file.

SAMPLE RATIO

Users can set update statistics sample ratio in **Statistics sample** field of **Statistics** page. This setting corresponds to the **DB_StsSp** keyword in the **dmconfig.ini** file. **Smart sample ratio** indicates the database intelligently obtain sample ratio; **Not do update statistics** indicates the database doesn't update statistics value. Additional, users can set sample ratio to a number between 1 and 100.

➲ Changing statistics settings:

1. Select the **Start statistics server** check box to enable the auto update

©Copyright 1995-2026 CASEMaker Inc.

statistics server daemon.

1. Select an update statistics mode from the **Statistics mode** box.
   - To change the update statistics mode to startup update statistics daemon in general mode, select the **General mode** option button.
   - To change the update statistics mode to startup update statistics daemon in every table setting mode, select the **Every table setting mode** option button.
2. Use the **Start time of statistics** fields to specify a date and a time when the first update statistics will begin.
3. Enter the number of days, hours, minutes, and seconds between each update statistics in the **Statistics interval** fields.
4. Select an update statistics sample ratio from the **Statistics sample** box.
   - Select the **Smart sample ratio** option button to enable the database intelligently obtain sample ratio.
   - Select the **Not do update statistics** option button to not update statistics database value.
   - Enter a value from 1 to 100 in the **Sample Ratio** field to set sample ratio value.

Changing Auto Index Settings

Selecting the **Auto Index** tab in the **Run Time Settings** window allows the user to set the **Auto Index** setting option.

©Copyright 1995-2026 CASEMaker Inc.

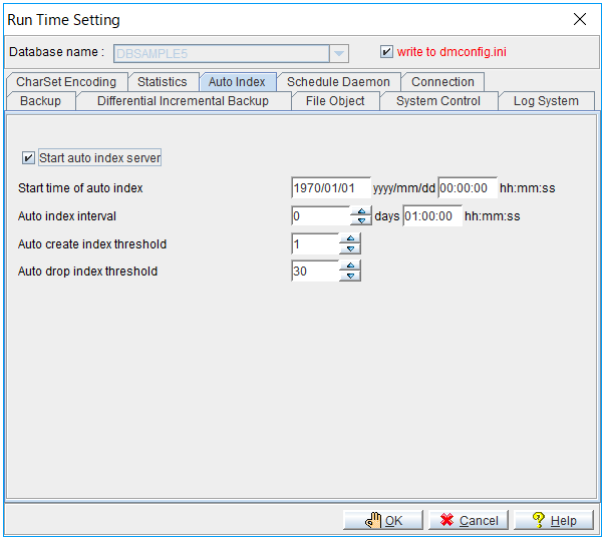


Figure 29-8 The Auto Index page of the Run Time Setting dialog

START AUTO INDEX SERVER

Auto indexes can be created automatically by the auto index daemon. Meanwhile, auto indexes can be dropped by the auto index daemon automatically if the index can merged with other indexes or it is not be used exceed the number of specified dropped days. The maximum column number of an auto index is 16. Creating an auto index requires **U** lock when set *COMMIT ON*.

DBMaker also supports "*SET LOADAUTOINDEX ON|OFF* " syntax and ODBC function *SQLSetConnectOption* for user to decide whether load auto index or not. For more information on ODBC functions, please refer to the *ODBC Programmer's Guide*.

Selecting the **Start auto index server** check box enables the auto index daemon. This setting corresponds to the **DB_IdxSv** keyword in the **dmconfig.ini** file. The default setting is ***enabled***.

START TIME OF AUTO INDEX

©Copyright 1995-2026 CASEMaker Inc.

Specify the first time of the auto index daemon to perform auto index by entering a date in the **yyyy/mm/dd** field and a time in the **hh:mm:ss** field. Reenter the numbers if they appear incorrectly the first time. This setting corresponds to the **DB_IdxTm** keyword in the **dmconfig.ini** file. Default value is ***1970-01-01 00:00:00***.

SETTING AUTO INDEX INTERVAL

The number in the combo box specifies, in days, the time interval at which the auto index daemon is activated. Next to the combo box is a field for time input, which specifies the time interval in hours, minutes, and seconds. The total time interval is determined by adding the two values together. For example, inputting 1 in days and inputting 12:00:00 in **hh:mm:ss** causes the auto index daemon to be activated every day and a half. The number of days can be manually entered into the combo box, or increased or decreased by clicking the arrows to the right; the number of hours, minutes, and seconds are manually entered. This setting corresponds to the **DB_IdxTv** keyword in the **dmconfig.ini** file.

SETTING AUTO CREATE INDEX THRESHOLD

This number specifies the threshold value for auto creating an auto index by auto index daemon. If the number of scan log (these logs with the same table id, table version and the column id list) reaches or exceeds this value**,** auto index daemon will create an auto index according to the serial log. This number can be manually entered into the combo box, or increased or decreased by clicking the arrows to the right. This setting corresponds to the **DB_IdxLn** keyword in the **dmconfig.ini** file. The default value is ***1***.

SETTING AUTO DROP INDEX THRESHOLD

This number specifies auto drop index threshold for auto index daemon, if an index not be used reaches or exceeds *n* days will be dropped by auto index daemon. This number can be manually entered into the combo box, or increased or decreased by clicking the arrows to the right. This setting corresponds to the **DB_IdxDp** keyword in the **dmconfig.ini** file. The default

©Copyright 1995-2026 CASEMaker Inc.

value is ***30***.

➲ Changing auto index settings:

1. Select the **Start auto index server** check box to activate the auto index daemon.
2. Use the **Start time of auto index** fields to specify a date and a time when auto indexes will begin.
3. Enter the number of days, hours, minutes, and seconds between each auto index in the **Auto index interval** fields.
4. Enter the number in the **Auto create index threshold** field to specific the threshold value for auto creating an auto index by auto index daemon.
5. Enter the number in the **Auto drop index threshold** field to specific the threshold value for auto dropping an auto index by auto index daemon.

Changing Schedule Daemon Settings

The **Schedule Daemon** page of the **Run Time Setting** dialog allows the user to set the **dmschsvr** server daemon setting option. The options on this page apply only to the server.

©Copyright 1995-2026 CASEMaker Inc.

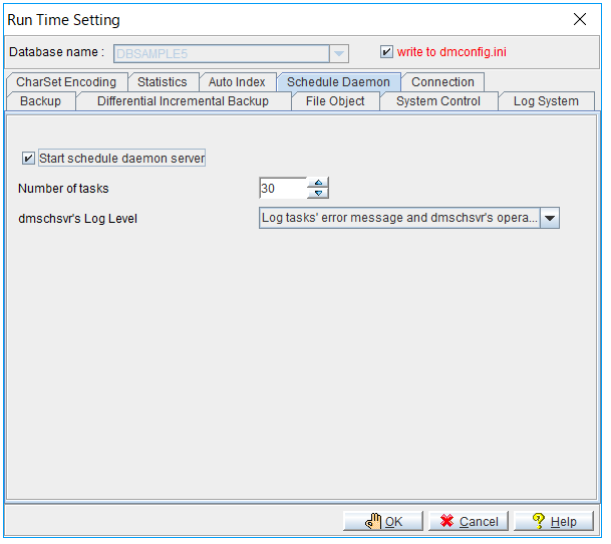


Figure 29-9 The Schedule Daemon page of the Run Time Setting dialog

START SCHEDULE DAEMON SERVER

The **Start schedule daemon server** option is only applied to the server, so user must first start the **dmschsvr** daemon server by JConfigration Tool, and then restart the database server, the **dmschsvr** daemon server will be activated. This setting corresponds to the **DB_SchSv** keyword in the **dmconfig.ini** file. Please refer to the *JConfiguration Tool Reference* for more information on Schedule Daemon settings.

NUMBER OF TASKS

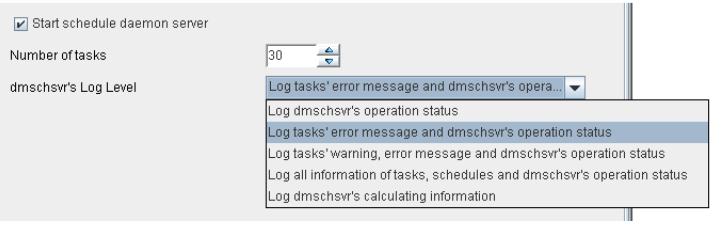
The **Number of tasks** field value is used to specify the number of task can be aroused by **dmschsvr** at the same time. This setting corresponds to the **DB_TskNo** keyword in the **dmconfig.ini** file. The default value is ***30*.**

DMSCHSVR LOG LEVEL

Users can specify what schedule daemon log information will be output according to the following five output levels. This setting corresponds to the

©Copyright 1995-2026 CASEMaker Inc.

DB_SchLgLev keyword in the **dmconfig.ini** file. The default value is ***1.*** DBMaker provides five dmschsvr's log levels as follows:



- **Log dmschsvr's operation status:** Value of **DB_SchLgLev** should be set to 0, and only log **dmschsvr**'s operation status
- **Log task's error message and dmschsvr's operation status:** Value of **DB_SchLgLev** should be set to 1, and log tasks' error message and **dmschsvr**'s operation status
- **Log tasks' warning, error message and dmschsvr's operation status:** Value of **DB_SchLgLev** should be set to 2, and log tasks' warning, error message and **dmschsvr**'s operation status
- **Log all information of tasks, schedules and dmschsvr's operation status:** Value of **DB_SchLgLev** should be set to 3, and log all information of tasks, schedules and **dmschsvr**'s operation status
- **Log dmschsvr's calculating information:** Value of **DB_SchLgLev** should be set to 4, and log dmschsvr's calculating information, so that a user can monitor whether **dmschsvr** is running normally

➲ Changing schedule daemon settings:

1. Select the **Start backup server** check box by JConfigration Tool to activate the **dmschsvr** server daemon.
2. Restart the database server.
3. Select the **Schedule Daemon** tab from the **Run Time Setting** window.
4. Enter the number of task can be aroused by **dmschsvr** at the same time in the **Number of tasks** field.
5. Select a Level for **dmschsvr** log:

©Copyright 1995-2026 CASEMaker Inc.

- Select **Log dmschsvr's operation status** to set the **DB_SchLgLev** value is 0, and only log **dmschsvr**'s operation status
- Select **Log task's error message and dmschsvr's operation status** to set the **DB_SchLgLev** value is 1, and log tasks' error message and **dmschsvr**'s operation status
- Select **Log tasks' warning, error message and dmschsvr's operation status** to set the **DB_SchLgLev** value is 2, and log tasks' warning, error message and **dmschsvr**'s operation status
- Select **Log all information of tasks, schedules and dmschsvr's operation status** to set the **DB_SchLgLev** value is 3, and log all information of tasks, schedules and **dmschsvr**'s operation status
- Select **Log dmschsvr's calculating information** to set the **DB_SchLgLev** value is 4, and log dmschsvr's calculating information, so that a user can monitor whether **dmschsvr** is running normally

Changing Connection Settings

The Connection page of the Run Time Setting dialog allows you to prevent new connections handles from being made. This is useful if the database is experiencing performance problems and needs to be evaluated, or to allow more processor resources to be used for critical database operations. Database administrators may enable or disable this function at any time from the DBMaker server.

©Copyright 1995-2026 CASEMaker Inc.

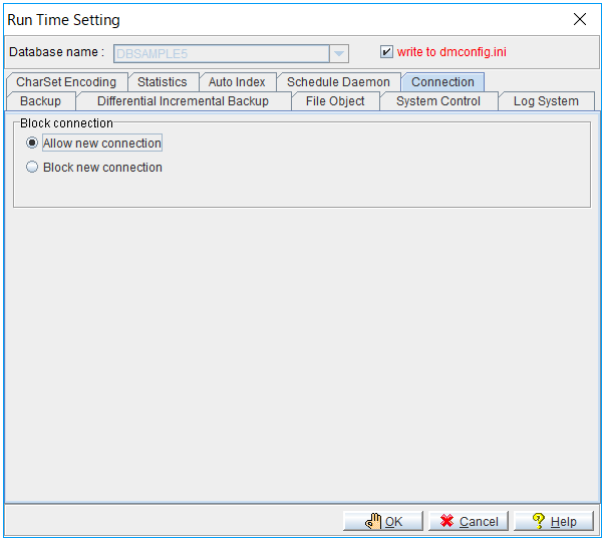


Figure 29-10 The Connection page of the Run Time Setting dialog

➲ Changing connection settings:

1. To allow new connections to be made to the database, select **Allow new connection**.
2. To prevent new connections from being made to the database, select **Block new connection**.

©Copyright 1995-2026 CASEMaker Inc.

29.2 Setting Preferences

JDBA Tool allows users to set preferences for the GUI. The following settings are available in the user preference window:

General:

Language Choice: JDBA Tool supports English, Simplified Chinese, traditional Chinese, and Japanese interfaces.

Auto Refresh for Database Monitor: Determines the rate at which database monitor information is updated to the screen.

Auto Refresh Time: Shows the time interval for auto refresh.

Setting store procedure recursive depth: Shows the SQL store procedure's max recursive executive counts.

Dark theme: Determine to activate dark theme or not.

Table:

Show System Table: Shows system objects in the tree view.

Show SQL Script: Allows the DBA to view and edit the SQL script that corresponds to functions carried out in JDBA Tool.

Records retrieved: Shows the number of records displayed per page.

JSQL:

Save history commands: Allows users to save their history commands into outer file, the directory is the database directory, the file name is jsyscom_dbmaker.xml.

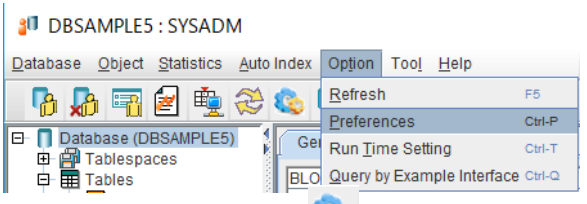
Max number of history commands: Defines the maximum number of commands which will be saved.

Auto completion: Activates the auto completion when using SQL Query.

©Copyright 1995-2026 CASEMaker Inc.

➲ Accessing the Preferences window:

1. Select **Preferences** from the **Option** menu on the menu bar.



NOTE *Users can also click the button from the Tool Bar to set the preferences.*

2. The **User Preference** window will appear with the **General** page displayed.

Setting General Options

Items included on the General Options page of the **User Preference** window are language selection, auto refresh and store procedure recursive depth options. Language selection toggles the GUI language; currently English and Chinese are supported by the user interface.

The **Auto Refresh for Database Monitor** check box activates or deactivates the automatic refresh function. All statistics shown in the Database Monitor are periodically automatically updated by the automatic refresh function. The time interval is indicated in the **Auto Refresh Time** field. The **Setting store procedure recursive depth** option shows the SQL store procedure's max recursive executive counts, and the default setting is **8**.

©Copyright 1995-2026 CASEMaker Inc.

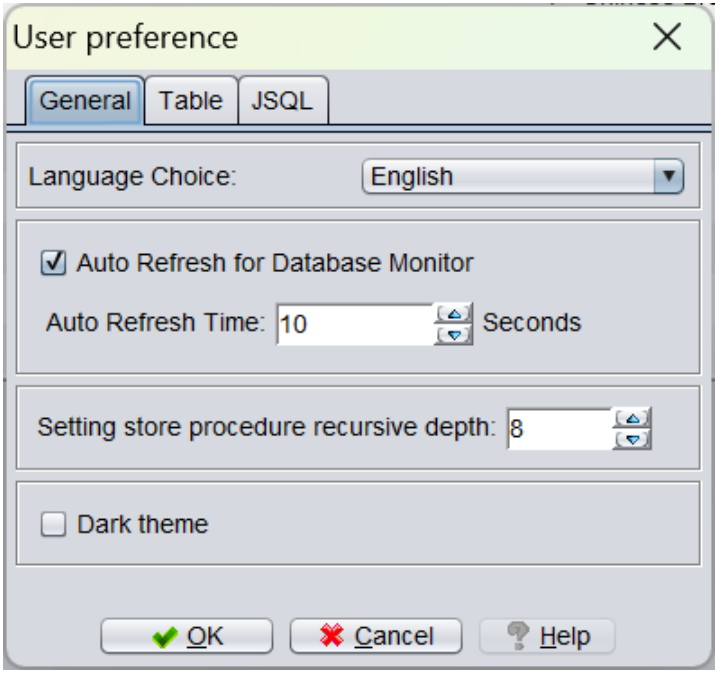


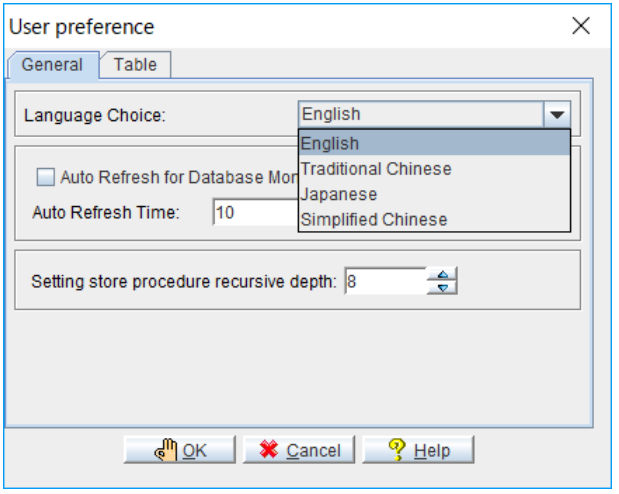
Figure 29-11 The General page of the User Preference Window

➲ Changing the GUI language setting:

1. Select the language for the GUI from the **Language Choice** drop down menu.

NOTE *To view Chinese your operating system must support Big 5 Chinese character coding.*

©Copyright 1995-2026 CASEMaker Inc.



➲ To Change the Auto Refresh for Database Monitor settings:

1. Selecting the check box toggles the automatic refresh function
2. To change the **Auto Refresh Time**, click the field and enter a new value.

NOTE *Setting a low **Auto Refresh Time** will use more processor resources and can substantially slow down the database.*

Setting Table Options

The **Table** page of the **User Preference** window contains the following settings:

- **Show System Table**: Shows system objects in the tree view.
- **Show SQL Script**: Allows the DBA to view and edit the SQL script that corresponds to functions carried out in JDBA Tool.
- **Records Retrieved**: sets the number of records to retrieve from a table at one time when viewing a large number of records.

©Copyright 1995-2026 CASEMaker Inc.

You can choose to disable this feature by changing the setting in the user preference window, or by enabling **Do not show SQL commands** at the bottom of the **Alter SQL Command** window.

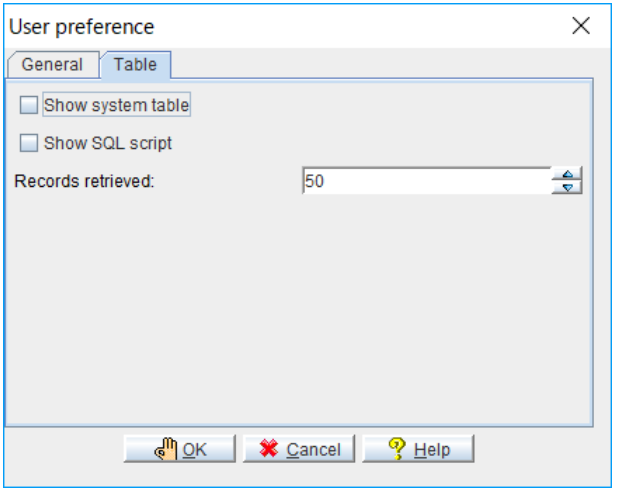


Figure 29-12 The Table page of the User Preference Window.

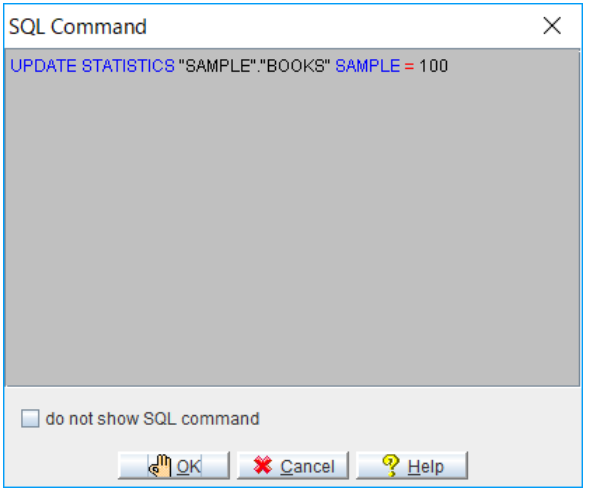
➲ Viewing system tables in the tree view:

1. Select the check box next to **Show system table**. A check mark in the box indicates that system tables are shown.

➲ Viewing SQL scripts or Altering SQL Scripts before execution:

1. Select the check box next to **Show SQL script** on the **Table Page** of the **User Preferences** window. A check mark in the box indicates that the **Alter SQL Command** window will open before execution.
2. Subsequent actions in JDBA that have a corresponding SQL Command will open a window as shown the following figure.

©Copyright 1995-2026 CASEMaker Inc.



3. Click the **OK** button at the end of the procedure displays the **Alter SQL Command** window. You can edit the SQL command in the text field as needed.
4. Click the **OK** button to execute the command. Click the **Cancel** button to abort the command.

Setting JSQL Options

The **JSQL** page of the **User Preference** window contains the following settings:

- **Save history commands**: Saves history commands into database directory, the file name is jsyscom_dbmaker.xml.
- **Max number of history commands**: Defines the maximum number of commands to save.
- **Auto completion**: Activates the auto completion when using SQL Query.

©Copyright 1995-2026 CASEMaker Inc.

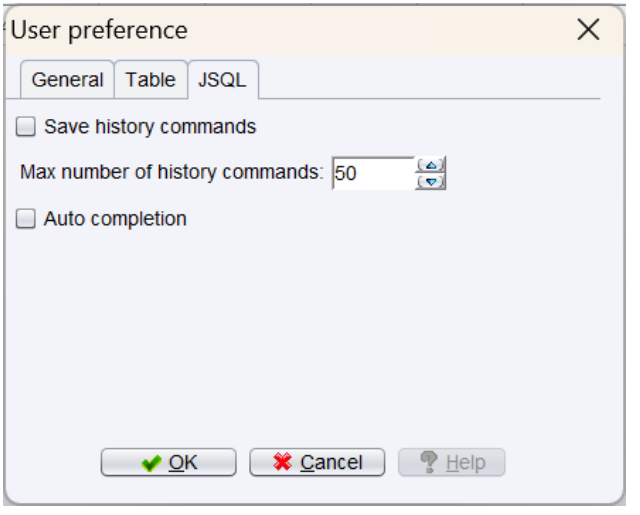


Figure 29-13 The JSQL page of the User Preference Window.

➲ Saving SQL commands to the outer file when using SQL Query:

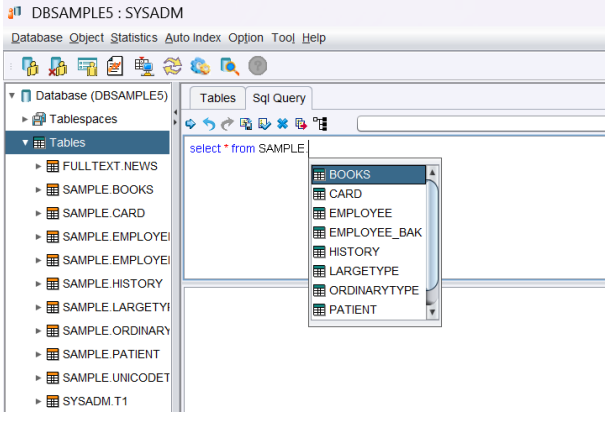
1. Select the check box next to **Save history commands**. A check mark in the box indicates that SQL commands will be saved.
2. Users can select the **Max number of history commands** value to define the maximum commands to save. The default value is 50.
3. Click the **OK** button.

The file will look like follows, the red square is the history commands

➲ Activate auto completion when using SQL Query:

1. Select the check box next to **Auto completion**. A check mark in the box indicates that auto completion is activated.
2. Click the **OK** button.

The following shows the auto completion when using SQL Query:



©Copyright 1995-2026 CASEMaker Inc.

29.3 Query by Example (QBE)

QBE is a query language included with DBMaker. The query by example (QBE) interface provides a streamline, user-friendly, and efficient interface for creating SQL query strings. The QBE interface guides you through the creation of SQL query strings. SQL programming knowledge is not required. Both advanced users and beginners will find the interface makes the creation of SQL query strings a quick and easy process by offering these functions:

- Single and multi table queries
- Table/Column alias names
- WHERE CLAUSE improvement: LIKE, MATCH, CONTAIN, IN/NOT IN, IS NULL/IS NOT NULL, BETWEEN
- Wildcards
- DISTINCT function for removing duplicate records
- Projection column operations: +, -, ×, /, ||
- Projection column aggregate/UDF functions: AVG, COUNT, MAX, MIN, SUM
- GROUP BY/HAVING clause
- ORDER BY CLAUSE
- Manual SQL query editing

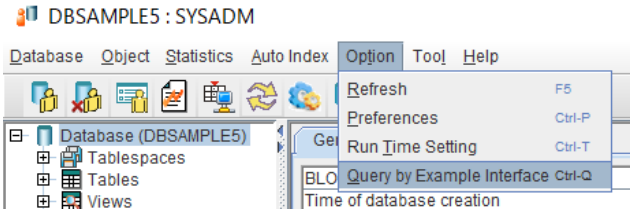
The QBE interface guides you through each step while creating a SQL string. Base the SQL string on a single table or multiple tables. The interface provides a complete list of all tables and views in the database.

In subsequent steps, columns are selected in tables that are affected by the query string. You can set an alias name for the tables or the columns. The GROUP BY clause will produce summary data within a group. The table and column order can also be set.

©Copyright 1995-2026 CASEMaker Inc.

➲ Creating a SQL string using QBE:

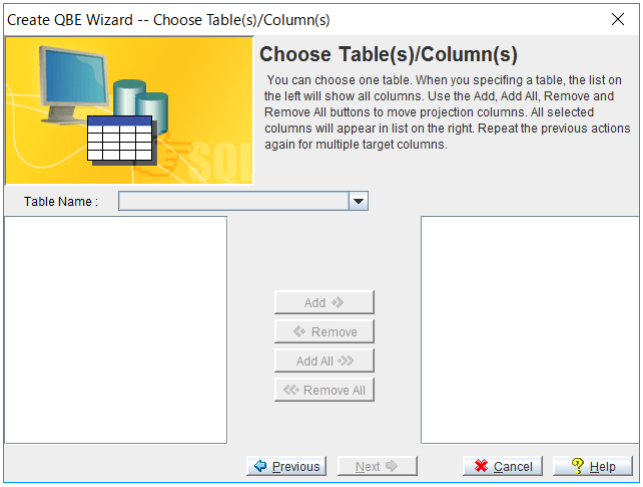
1. Select **Query by Example Interface** from the **Option** menu.



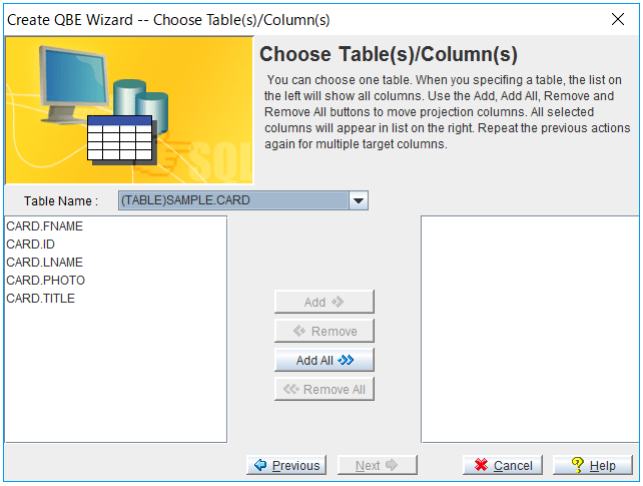
2. The **Create QBE Wizard Introduction** dialog box appears.

3. Click the **Next** button. The **Choose Table(s)/Column(s)** window appears.

©Copyright 1995-2026 CASEMaker Inc.

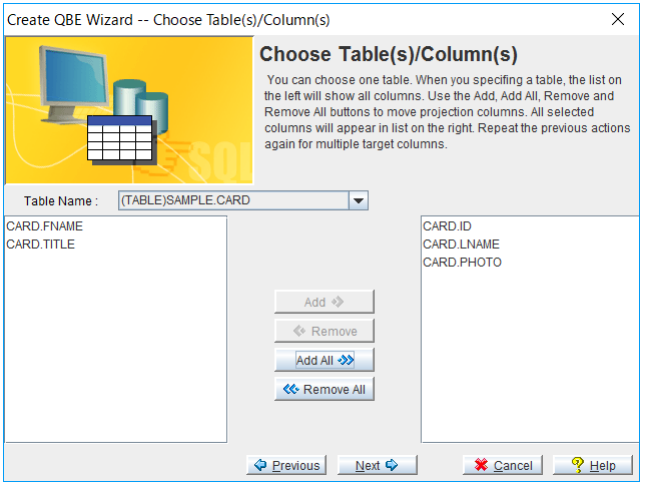


4. Choose one table/view or multiple tables/views from the **Table Name** drop down list. The user with DBA authority or higher can select all tables. The user with RESOURCE authority can select tables owned to him or on which the select privilege is granted to him. The user with CONNECT authority can select tables on which the select privilege is granted to him. When specifying a table or view, the columns contained in the selected table(s)/view(s) appear on the left side of the window.



©Copyright 1995-2026 CASEMaker Inc.

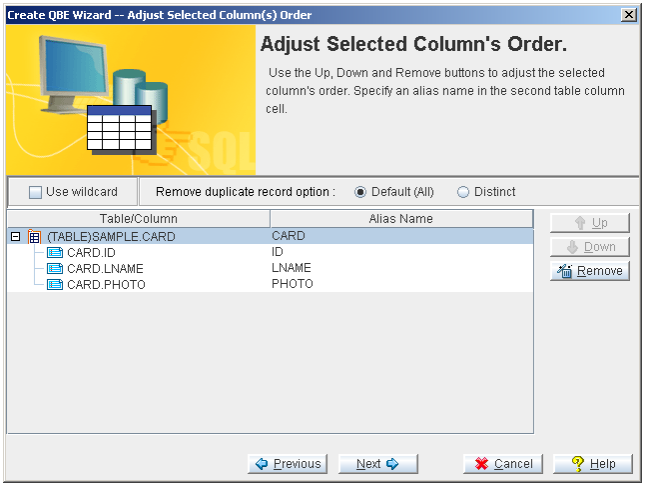
5. Double-clicking a column name selects the column. Alternatively, click once on a column then click the **Add** button. The **Add all** button adds all columns to the replication. All the selected columns appear in the list box to the right. Likewise, columns are removed from the replication by double-clicking the column name or selecting the column and clicking the **Remove** button.



NOTE *To select multiple columns press and hold the **Ctrl** key and select the column with the mouse.*

6. Click the **Next** button. The **Adjust Selected Column's Order** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



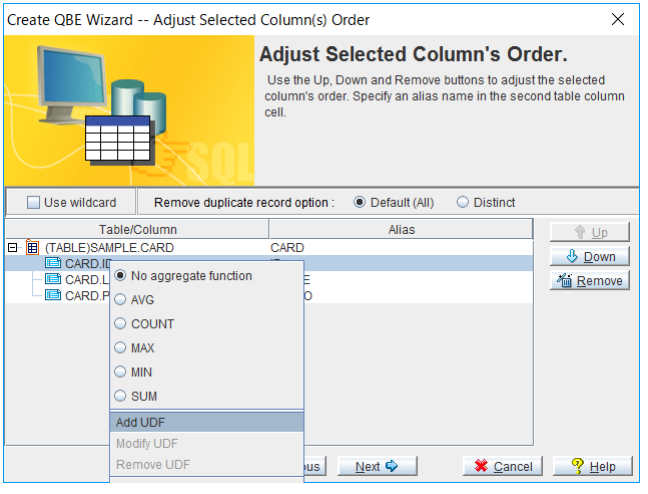
7. Enabling the **Use wildcard** setting is helpful for fetching all the table's records using wildcards. Optionally click the **Use wildcard** checkbox to enable this feature.

8. Select a **Remove duplicate record** option. The **Default (ALL)** radio button fetches all records. In other words, duplicate records are not exclused from the results. The **Distinct** radio button option removes duplicate data records from the results of a query (i.e., only unique data record are included in the results).

9. To expand the table to view to include all of the available columns double-click the table or click the expanding icon ( ) at the side of table name.

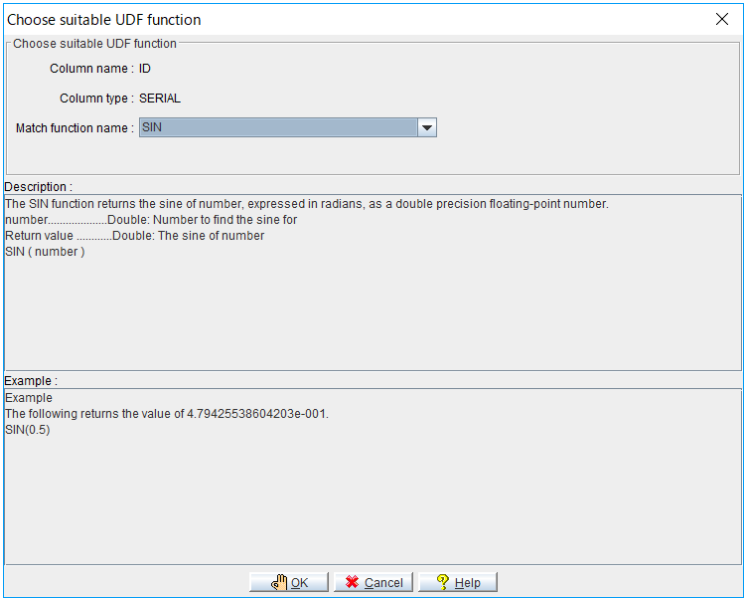
10. Select a column from the column list. The selected column highlights. Right-click the column to set an aggregate function or remove it from the projected column using a pop-up menu. QBE supports five aggregate functions: AVG, COUNT, MAX, MIN and SUM.

©Copyright 1995-2026 CASEMaker Inc.



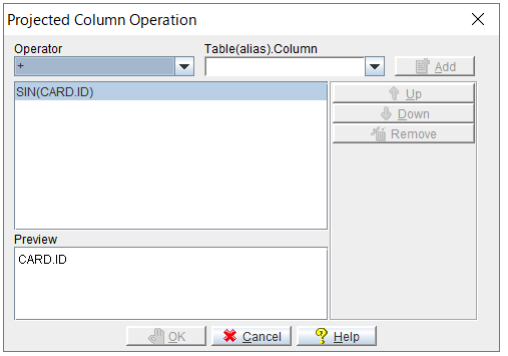
11. QBE can apply UDF functions into projected columns. Click the **Add UDF** button and the following window appears. Column Name and Column Type fields show the selected column's name and type. Select a UDF from the **Match Function Name** drop down menu, the selected UDF appears in the **Description** area. To help illustrate the UDF, examples appear in the **Example** area. Click the **OK** button.

©Copyright 1995-2026 CASEMaker Inc.



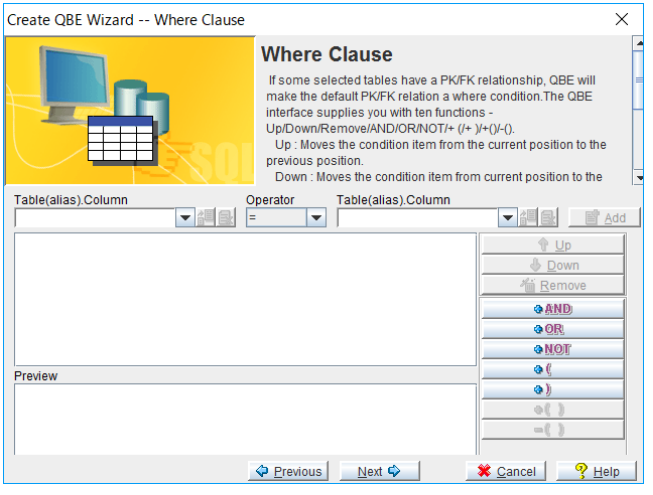
12. Click the **Modify UDF/Remove UDF** buttons to modify or remove the selected UDF function.

13. Click the **Setting** button. The **Projected Column Operation** window appears as shown below. You can choose one operator from the **Operator** drop down list. Choose one column from the **Table(alias).Column** drop down list. Click the **Add** button. The operation appears in the Preview area below the window. Use the **Up**, **Down** and **Remove** buttons to adjust the selected column's order. Click the **OK** button.



©Copyright 1995-2026 CASEMaker Inc.

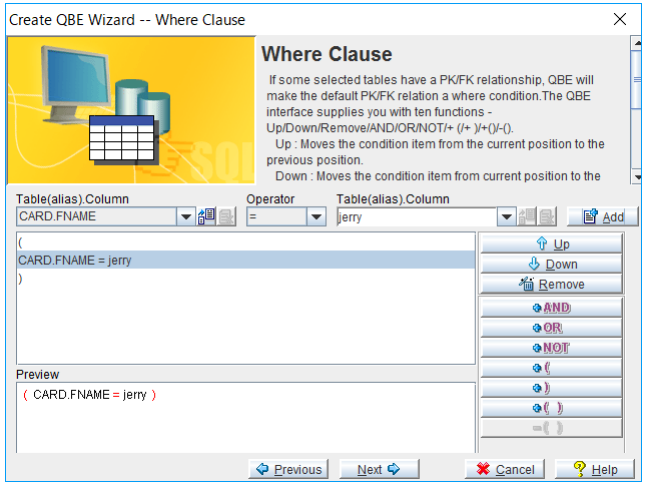
14. Double-click the **Alias Name** field for a column to specify an alias name.
15. Select a column from the **Table/Column** field using the mouse. The selected columns highlight. Click the **Up**, **Down,** or **Remove** buttons to adjust or drop the selected column.
16. Click the **Next** button. The **Where Clause** window appears.



17. Select a column from the **Table(alias).Column(alias)** drop down list on the left.

    - Select an operator from the Operator drop down list. QBE supports following operators: =, >, <, >=, <=, <>, +, -, *, /, ||, between, is null/is not null, in/not in, like/not like, match/not match, and contain/not contain.
    - Enter a value in the **Table(alias).Column(alias)** field on the right or select a column from the drop down list.
    - Click the **Add** button. The where condition appears in the panel below the window.

©Copyright 1995-2026 CASEMaker Inc.



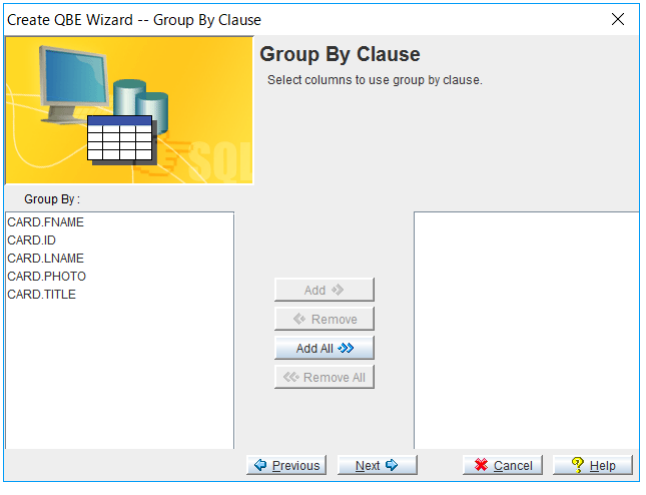
NOTE *If some selected tables have a PK/FK relationship, QBE makes the default PK/FK relation a where condition.*

18. Optionally, add a UDF function by clicking the **Import** button ( ) on the right of the **Table(alias).Column(alias)** field to add a UDF function.
19. Click the **Export** button ( ) to cancel the selected UDF function.
20. The QBE interface supports ten functions:
    - Up: Moves the condition item from the current position to the previous position
    - Down: Moves the condition item from current position to the next position
    - Remove: Removes the selected condition item
    - AND : Adds an AND operator into the next position
    - OR: Adds an OR operator into the next position
    - NOT: Adds a NOT operator into the previous position
    - +(: Adds a left bracket into the previous position

©Copyright 1995-2026 CASEMaker Inc.

- +): Adds a right bracket into the next position
- +(): Adds brackets into the selected condition item
- -(): Removes brackets from the selected condition item

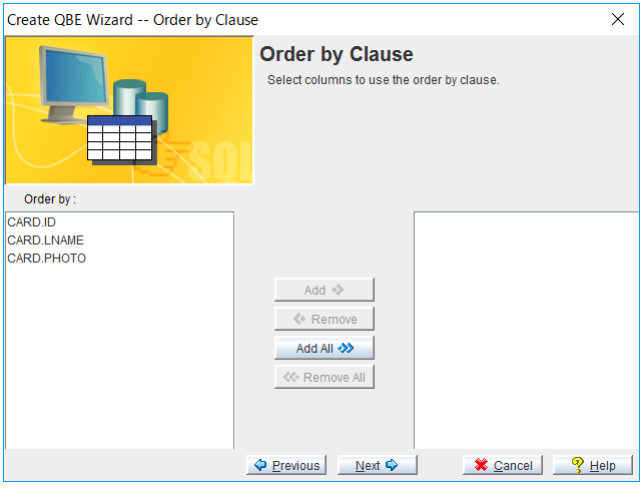
21. Click the **Next** button. The **Group by Clause** window appears.



22. Select columns to use **Group by Clause**. A group is a set of rows having the same values of group by columns. A single row of aggregate results is generated for each group.
23. Click the **Next** button. The **Having Clause** window appears.

©Copyright 1995-2026 CASEMaker Inc.

24. Click the **Next** button when you are satisfied with the clauses. The **Order by Clause** window appears.

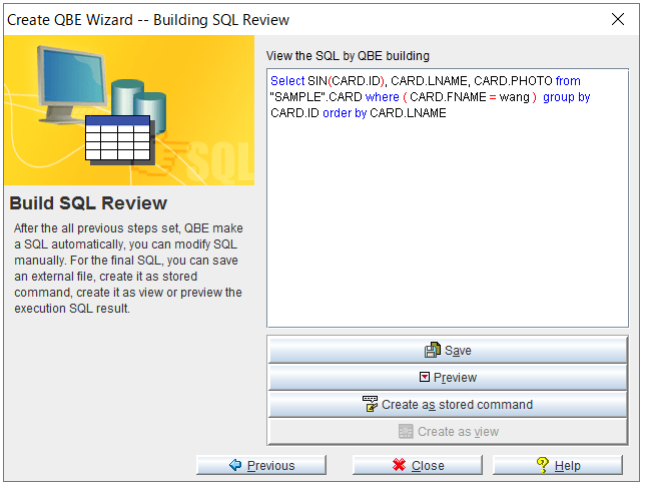


25. Select the sort order for the columns using the **Order by Clause** dialog box. The order by clause can sort query results by the values contained in selected columns.

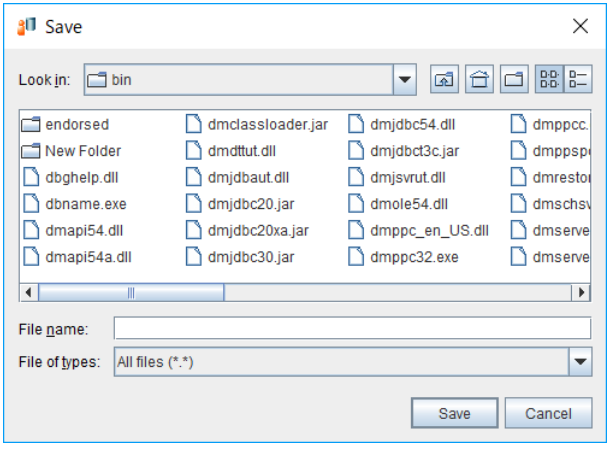
26. Click the **Next** button. The **Building SQL Review** dialog box appears. QBE make a SQL script automatically, you can modify it manually. For

©Copyright 1995-2026 CASEMaker Inc.

the final SQL, you can save an external file, create it as stored command, create it as view, or preview the execution SQL result.



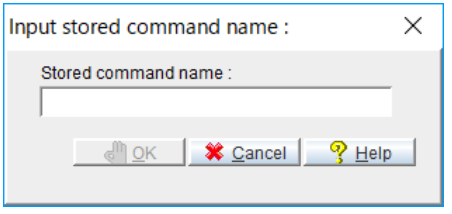
27. Click the **Save** button to save the query. The **Save** dialog box appears.



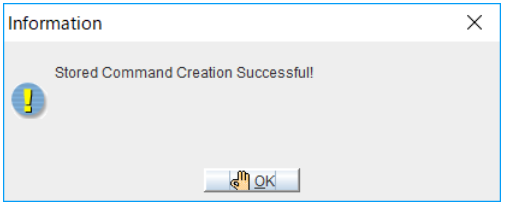
- Enter a name and location for the query in the dialog box and click the **Save** button.

28. To save the query as a stored command, click the **Save as stored command** button. The **Input stored command name** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



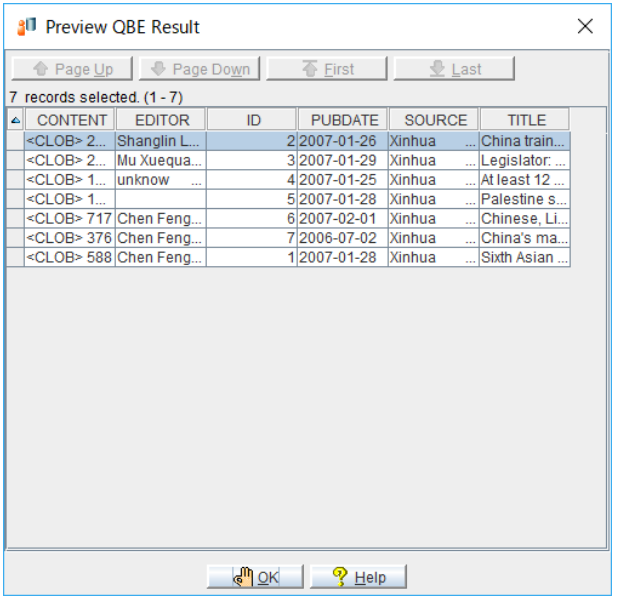
- Enter a name in the **Stored command name** field.
- Click the **OK** button. A message saying the stored command is created successfully is displayed.



- Click the **OK** button. The created SQL command saves as stored command appears in the stored command list.

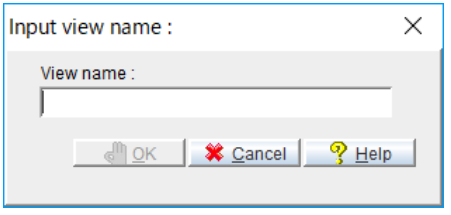
29. To preview the query, click the **Preview** button. The **Preview QBE Result** dialog box appears.

©Copyright 1995-2026 CASEMaker Inc.



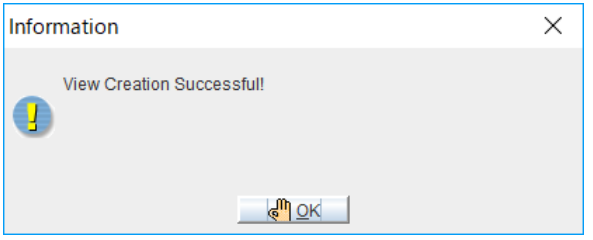
- Browse the results using the scroll bars. For large results use the **Page Up**, **Page Down**, **First** and **Last** buttons.

30. To create a view on the query, click the **Create as view** button. The **Input view name** dialog box appears.



- Enter a name for the view in the **View name** field.
- Click the **OK** button. A message confirming successful creation appears.

©Copyright 1995-2026 CASEMaker Inc.



- Click the **OK** button.

31. Click the **Previous** button to return to prior windows or click the **Close** button when you are satisfied with the QBE.

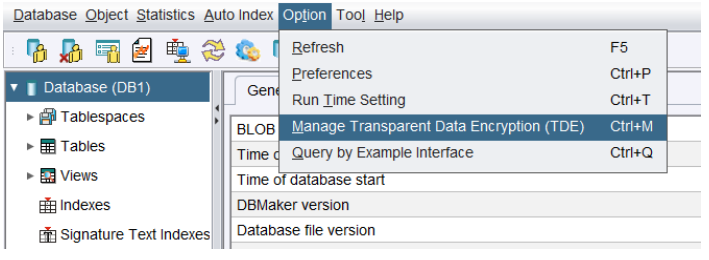
©Copyright 1995-2026 CASEMaker Inc.

29.4 Manage Transparent Data Encryption (TDE)

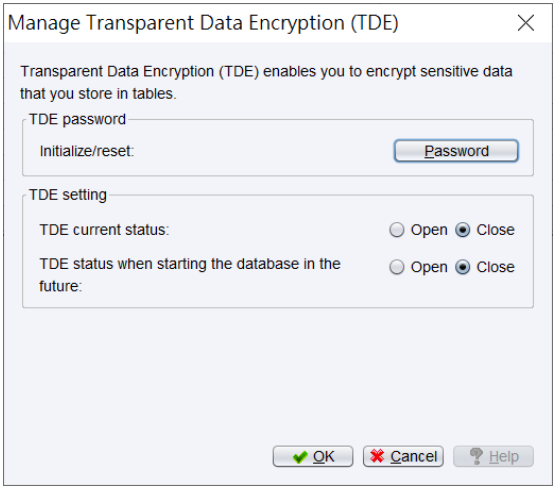
JDBA Tool allows SYSDADM and SYSDBA to set TDE password and open/close it. Only SYSADM and SYSDBA can see this option.

➲ Accessing the Manage Transparent Data Encryption (TDE) window:

1. Select **Manage Transparent Data Encryption (TDE)** from the **Option** menu on the menu bar.



2. The **Manage Transparent Data Encryption (TDE)** window will appear.



©Copyright 1995-2026 CASEMaker Inc.

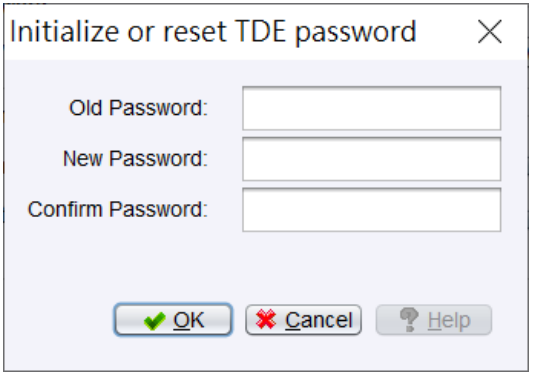
Setting TDE options

The Manage **Transparent Data Encryption (TDE)** window contains following settings:

- **TDE password initialize/reset:** Set password of column encryption.
- **TDE current status:** The current status of database's column encryption, **Open** means column encryption is opened, users can create and access encrypted columns; **Close** means column encryption is closed, users cannot create and access encrypted columns.
- **TDE status when starting the database in the future:** Defines column encryption will open/close automatically when database is started next time.

➲ Altering/Setting TDE password:

1. Press the button **Password** in the TDE password blank, a window as shown in the following figure will open.



2. Enter the **Old Password**, **New Password** and **Confirm Password**, click the **OK** button to alter password from old one to new one.

NOTE *SYSADM privilege can alter password without Old Password.*

©Copyright 1995-2026 CASEMaker Inc.

Glossary

Asynchronous Table Replication
A set of columns selected to form a projection that is replicated to the destination tables of another database. Asynchronous table replication occurs according to a schedule. Changes made to the source table are saved to a log file that is used to periodically update the destination tables.
BLOB
Acronym for Binary Large Objects. A table that consists of one column of LONGVARCHAR or LONGVARBINARY type data.
BLOB Frames
Measurement unit used to allocate disk space for BLOB type data.
Concurrency Control
A system of locks placed on objects to prevent multiple users from simultaneously manipulating the same data sets.
Coordinator Database
In a distributed database environment, the database that the client connects to. If the client accesses data from another database then that database is a participant database.
Daemon
A routine that automatically executes at a set time interval.
Data Pages
Data measurement unit for allocation of disk space.
Destination Database
The database that receives data for table replication. A database containing destination tables for a synchronous or asynchronous replication.

©Copyright 1995-2026 CASEMaker Inc.

Destination Tables
Tables that receive replicated data from a source table. The table on the destination database that the data is replicated to.
Distributed Database Environment
A system of networked remote databases that allow for any table in the system to be accessed by a client on any of the participant (remote) databases.
Exclusive Lock (X Lock)
An access block placed on a database object that prevents other users from accessing the object
Foreign Key
A column or set of columns whose rows contain the same values as the set of columns in the primary key or unique index of another table.
Fragment
Also called a horizontal partition, a fragment is the replication of a given range of data tuples.
Journal Blocks
Internal data measurement unit (514 bytes) that DBMaker uses to manage journal data.
Journal Buffer
Upper memory where current journal blocks are stored before writing them to disk.
Journal Pages
Measurement unit for allocation of disk space.
Lock
Locks allow only one user update and delete permission on the locked object at the time the object is locked.
Page Buffer
Upper memory allocated for data pages accessed by a user.
Participant Database
In a distributed database environment, a database that is accessed by the client through a coordinator database.
Primary Database
The source database for database replication.
Primary key
A column or set of columns in a table that contain values that uniquely identify the rows in the table.
Projection
The selected columns from a base table chosen for replication.

©Copyright 1995-2026 CASEMaker Inc.

Publication
A data set on the source table available for the replication.
Remote Database
A database located on another server from that being accessed by the client.
Remote Tables
Tables on a database located on a server that is different from the one the client is connected to.
Replication Domain
The replication fragment (horizontal partition) and projection (vertical partition) together are called a replication domain. It is the range of a table's data that is replicated.
Select Lock (S Lock)
An access block placed on a database object that allows other users to browse the object but not to update or delete any part of it.
Slave Database
A read-only database that receives data from a database replication
Source Database
The database that contains source tables used to replicate.
Source Table
The table on the source database that the replicated data is from.
Subscription
The data set on the destination table to receive a publication.
Synchronous Table Replication
A set of columns selected to form a projection that is replicated to the destination tables of another database. Synchronous table replication occurs simultaneously – changes made to the source table are simultaneously made to the destination table(s).

©Copyright 1995-2026 CASEMaker Inc.

©Copyright 1995-2026 CASEMaker Inc.

Index

Action Time, 11-6
Adding Files to Tablespaces, 4-10–4-11
  How to, 4-11–4-13
Additional Resources, 1-3
Advanced Table Options, 5-26
  How to, 5-27–5-28
Altering Tablespaces, 4-10
Asynchronous Table Replication, 16-20
  Adding Subscribers, 16-31
    How to, 16-31–16-35
  Creating, 16-21
  Dropping, 16-38
    How to, 16-39
  How to Create, 16-22–16-31
Authority Level, 20-8–20-9
  How to Change, 20-9
Auto Refresh for Database Monitor, 25-24
Autoextend Tablespaces, 4-3, 4-5
  Changing to Regular, 4-17–4-18
    How to, 4-17–4-18
  Creating, 4-8
  Pages to Extend a File, Run Time, 25-14
Backup Mode
  Run Time Settings, 25-5
Backup Options
  Run Time Settings, 25-4
Backup Options, Run Time
  How to Set, 25-8–25-9
Batch File, 23-17
Browsing Tablespaces, 4-2
  How to, 4-3–4-4
Cache, 5-40
Changing Tablespace Mode, 4-17
  How to, 4-17–4-18
Command Monitor, 25-13
Commands
  Stored. *See* "Stored Commands"
Compact Backup Mode, 25-7
Connecting to a Database, 2-7
  How to, 2-7–2-9
Connection Settings, 25-21–25-22

©Copyright 1995-2026 CASEMaker Inc.

Connections, 2-7
Execution Information, 24-17–24-18
Lock Waiting Information, 24-20–24-21
Login Information, 24-16–24-17
Statistical Information, 24-18–24-19
Status, 24-27
Termination, 24-21
Transaction Information, 24-19–24-20
Constraints
Column, 5-11, 5-37
Table, 5-11
Creating Database Links, 14-3
How to, 14-4–14-5
Creating Domains, 17-2
How to, 17-3–17-5
Creating Foreign Keys, 10-2
Adding, 10-2
How to, 10-3–10-6
Setting Options
How to, 10-7–10-8
Setting Options, 10-6–10-7
Creating Groups, 21-2
How to, 21-2–21-4
Creating Indexes, 7-2
How to, 7-2–7-6
Creating IVF Text Indexes, 9-3
How to, 9-3–9-8
Creating Replication Schedules, 15-3
How to, 15-3–15-8
Creating Signature Text Indexes, 8-2
How to, 8-2–8-6
Creating Stored Commands, 12-2
How to, 12-3–12-4
Creating Stored Procedures, 13-1–13-2
How to, 13-3–13-7
Creating Synonyms, 18-2
How to, 18-2–18-3
Creating Tables, 5-3
Adding a Foreign Key, 5-18
How to, 5-20–5-25
Adding a UNIQUE Constraint, 5-14
How to, 5-15–5-17
Completing Table Creation, 5-28
How to, 5-29
Setting Advanced Table Options, 5-26
How to, 5-27–5-28
Setting Column and Table Constraints, 5-11
How to, 5-12–5-13
Setting Column Options, 5-8–5-9
How to, 5-9–5-10
Setting Table, Tablespace, and Schema Options, 5-6
How to, 5-7–5-8
Starting the Wizard, 5-4
How to, 5-5–5-6
Creating Tablespaces, 4-5
How to, 4-6–4-9
Creating Triggers, 11-2

©Copyright 1995-2026 CASEMaker Inc.

Action Settings, 11-6–11-7
How to, 11-8–11-9
Indicating the Reference Clause, 11-9
How to, 11-10
Naming, 11-2
How to, 11-4–11-6
Trigger Action, 11-13
How to, 11-13–11-15
WHEN Condition Clause, 11-11
How to, 11-12
Creating User Accounts, 20-2
How to, 20-3–20-7
Creating Views, 6-2
How to, 6-2–6-7
Data
Editing, 5-49
How to, 5-49–5-51
Data Operations, 24-27–24-28
JData Transfer, 22-1–23-2
Exporting to Text, 23-37–23-39
How to, 23-39–23-44
Exporting to XML, 23-46–23-49
How to, 23-50–23-56
How to Open the JData Transfer Tool, 23-3
Importing from ODBC, 23-17
How to, 23-18–23-20
SQL Statements, 23-25
Tables, 23-21
XML Batch, 23-31
Importing from Text, 23-4–23-6
How to:, 23-7–23-12
Importing from XML, 23-13
How to, 23-13–23-16
Database Links, 14-1
Creating, 14-3
How to, 14-4–14-5
Dropping, 14-6
How to, 14-6
Deleting User Accounts, 20-16
How to, 20-16–20-17
Dependencies
Tables, 5-45
How to, 5-45–5-46
Directory of Backup Files
Run Time Settings, 25-6
Disconnecting from a Database, 2-11
Distributed Database Options
Enabling, Runtime, 25-14
Document Conventions, 1-5
Domains, 17-1
Creating, 17-2
How to, 17-3–17-5
Dropping, 17-7
How to, 17-7–17-8
Stored Commands
How to, 12-8–12-9
Dropping Subscribers, 16-36
How to, 16-36–16-37
Dropping Database Links, 14-6
How to, 14-6

©Copyright 1995-2026 CASEMaker Inc.

Dropping Domains, 17-7
How to, 17-7–17-8
Dropping Foreign Keys, 10-10
How to, 10-10
Dropping Indexes, 7-7
How to, 7-7
Dropping IVF Text Indexes, 9-10
How to, 9-10–9-11
Dropping Replication Schedules, 15-9
How to, 15-9
Dropping Signature Text Indexes, 8-8
How to, 8-8
Dropping Stored Commands, 12-8
How to, 12-8–12-9
Dropping Stored Procedures, 13-11
How to, 13-11–13-12
Dropping Synonyms, 18-4
How to, 18-4
Dropping Table Replication, 16-38
How to, 16-39
Dropping Tables, 5-54
How to, 5-54
Dropping Tablespaces, 4-21–4-22
How to, 4-21–4-22
Dropping Triggers, 11-16
How to, 11-17
Dropping Views, 6-12–6-14
How to, 6-14–6-15
Enabling Distributed Database Mode
Run Time Settings, 25-14
Executing Stored Commands, 12-5
How to, 12-6–12-7
Executing Stored Procedures, 13-9
How to, 13-9–13-10
Exporting Data. *See* "Data Transfer"
Extending a File
Run Time Settings, 25-14
Features, 2-2–2-3
File Objects
Run Time Settings, 25-10
Run Time Settings Backup Mode, 25-6
Fill Factor, 5-40
Foreign Keys, 9-15–10-1
Adding While Creating a Table
How to, 5-20–5-25
Adding While Creating a Table, 5-18
Creating, 10-2
Adding, 10-2
How to, 10-3–10-6
Setting Options, 10-6–10-7
How to, 10-7–10-8
Dropping, 10-10
How to, 10-10
Groups, 21-1
Creating, 21-2
How to, 21-2–21-4
Members, 21-6

©Copyright 1995-2026 CASEMaker Inc.

How to Add and Remove, 21-7–21-8
Privileges, 21-9
How to Manage, 21-10–21-11
How to
Add a Foreign Key, 10-3–10-6
Add Files to a Tablespace, 4-11–4-13
Browse Tablespaces, 4-3–4-4
Change System Control Settings:, 25-15
Change Tablespace Mode, 4-17–4-18
Check Dependencies, 5-45–5-46
Check IVF Text Index Statistics, 9-14
Check Signature Text Index Statistics, 8-12
Check Table Statistics, 5-47
Check View Properties, 6-8–6-9
Connect to a Database, 2-7–2-9
Create a Database Link, 14-4–14-5
Create a Domain, 17-3–17-5
Create a Group, 21-2–21-4
Create a Replication Schedule, 15-3–15-8
Create a Signature Text Index, 8-2–8-6
Create a Stored Command, 12-3–12-4
Create a Stored Procedure, 13-3–13-7
Create a Synonym, 18-2–18-3
Create a Tablespace, 4-6–4-9
Create a Trigger Action, 11-13–11-15
Create a User Account, 20-3–20-7
Create a View, 6-2–6-7
Create an Index, 7-2–7-6
Create an IVF Text Index, 9-3–9-8
Delete a User Account, 20-16–20-17
Drop Subscribers, 16-36–16-37
Drop a Database Link, 14-6
Drop a Domain, 17-7–17-8
Drop a Foreign Key, 10-10
Drop a Replication Schedule, 15-9
Drop a Signature Text Index, 8-8
Drop a Stored Command, 12-8–12-9
Drop a Stored Procedure, 13-11–13-12
Drop a Synonym, 18-4
Drop a Table, 5-54
Drop a Tablespaces, 4-21–4-22
Drop a Trigger, 11-17
Drop a View, 6-14–6-15
Drop an Index, 7-7
Drop an IVF Text Index, 9-10–9-11
Edit Data, 5-49–5-51
Execute a Stored Command, 12-6–12-7
Execute a Stored Procedure, 13-9–13-10
Export a Text File, 23-39–23-44
Export Data to XML, 23-50–23-56
Grant Privileges on Stored Procedures, 13-13–13-14
Grant Privileges on Views, 6-10
Grant Stored Command Privilege, 12-10–12-11
Import data from ODBC, 23-18–23-20
Import Data from XML, 23-13–23-16
Import Text Data, 23-7–23-12

©Copyright 1995-2026 CASEMaker Inc.

Indicate the Trigger Referencing Clause, 11-10
Manae e Group Privileges, 20-11–20-13
Manage Table Privileges, 5-43–5-44
Modify a File in a Tablespace, 4-16
Modify a Replication Schedule, 15-11–15-12
Modify a Trigger, 11-20
Name a Trigger, 11-4–11-6
Open the JData Transfer Tool, 23-3
Rebuild a Signature Text Index, 8-10–8-11
Rebuild a Text Index, 9-12
Rebuild an Index, 7-8
Rename a Table, 5-52–5-53
Run Time File Object Settings, 25-11
Set Advanced Table Options, 5-27–5-28
Set Column and Table Constraints, 5-12–5-13
Set Column Options, 5-9–5-10
Set File Object Run Time Settings, 25-11
Set Foreign Key Options, 10-7–10-8
Set Run Time Backup Settings, 25-8–25-9
Set Table Properties, 5-41
Shrink and Compress Tablespaces, 4-14–4-15
Specify a Trigger WHEN Condition Clause, 11-12
Specify Trigger Action Settings, 11-8–11-9
Update Tablespace Statistics, 4-20
I/O Status, 24-32–24-33
Import Data. *See* "Data Transfer
Incremental Backup
Begin Time, 25-8
Interval Time, 25-8
Journal Trigger Value, 25-7
Run Time Settings, 25-7, 25-8
Runtime Settings, 25-8
Indexes, 7-1
Creating, 7-2
How to, 7-2–7-6
Dropping, 7-7
How to, 7-7
Rebuilding, 7-8
How to, 7-8
Statistics, 7-10–7-11
Text
Inverted file, 9-1
Signature, 8-1
Introduction, 1-1
IVF Text Indexes, 9-1
Creating, 9-3
How to, 9-3–9-8
Dropping, 9-10
How to, 9-10–9-11
Rebuilding, 9-12
Statistics, 9-14

©Copyright 1995-2026 CASEMaker Inc.

How to Check, 9-14
Java Run Time Environment, 2-1
JDBA Tool, 2-1
Workspace, 2-4–2-5
Journal
Status, 24-33–24-34
Journal File Synchronization, 25-12–25-13
Journal Trigger Value, 25-7
Kill Connection, 24-21
Language Choice, 25-24
Language Preference, 25-23
Links
Database. *See* "Database Links"
Lock Mode, 5-40
Locks
Monitoring, 24-22–24-23
Status, 24-26
log system
error level, 25-17–25-18
, 25-20
, 25-20
, 25-21
, 25-21
, 25-19
, 25-19
log level, 25-16–25-17
, 25-18
, 25-20
, 25-18
, 25-16
Menu Bar, 2-5
Modifying a File in a Tablespace, 4-15
How to, 4-16
Modifying Replication Schedules, 15-11
How to, 15-11–15-12
Modifying Tables, 5-31
Adding Columns, 5-37
How to, 5-38
Column Constraints, 5-37
How to, 5-37
Column Data Type, 5-33
How to, 5-34
Column Default Values, 5-36
How to, 5-36
Column Name, 5-33
How to, 5-33
Column Nullable, 5-35
How to, 5-35
Column Order, 5-32
How to, 5-32–5-33
Column Precision and Scale
How to, 5-34
Column Precision and Scale, 5-34
Column Primary Key, 5-35
How to, 5-36
Deleting Columns, 5-38–5-39
How to, 5-38–5-39
How to, 5-31–5-32
Modifying Triggers, 11-18

©Copyright 1995-2026 CASEMaker Inc.

How to, 11-19
Monitor, 24-1
Tool, 24-2–24-4
How to Use, 24-5–24-9
Tree, 24-14
No-cache, 5-40
ODBC
Importing, 23-17
How to, 23-18–23-20
SQL Statements, 23-25
Tables, 23-21
XML Batch, 23-31
Options, 25-1
General, 25-24
Table, 25-25–25-26
Page Status, 24-32–24-33
Passwords, 20-14
How to Change, 20-14
Pending Transactions, 24-36
Primary Key, 5-31, 5-35
Privileges, 20-10–20-11
Groups, 21-9
How to Manage, 21-10–21-11
How to Change, 20-11–20-13
Stored Commands, 12-10
How to:, 12-10–12-11
Stored Procedures, 13-13
How to, 13-13–13-14
Tables, 5-42
How to Manage, 5-43–5-44
Views, 6-10
How to, 6-10
Procedures
Stored. , *See* "Stored Procedures", *See* "Stored Procedures"
Properties of Views, 6-8
How to Check, 6-8–6-9
Rebuilding Indexes, 7-8
How to, 7-8
Rebuilding IVF Text Indexes, 9-12
Rebuilding Signature Text Indexes, 8-10
How to, 8-10–8-11
Rebuilding Text Indexes
How to, 9-12
Referencing Clause, 11-7
Renaming Tables, 5-52
How to, 5-52–5-53
Replication
Table. *See* "Table Replication"
Replication Schedules, 15-1
Creating, 15-3
How to, 15-3–15-8
Dropping, 15-9
How to, 15-9
Modifying, 15-11
How to, 15-11–15-12
Run Time Settings, 25-2
Backup Mode, 25-5
Backup Options, 25-4

©Copyright 1995-2026 CASEMaker Inc.

Command Monitor, 25-13
Compact Backup Mode, 25-7
Connection Settings, 25-21–25-22
Directory of Backup Files, 25-6
Enabling Distributed Database Mode, 25-14
File Object Backup Mode, 25-6
File Objects, 25-10
How to Change File Object Settings, 25-11
How to Change System Control Settings, 25-15
How to Select a Database, 25-2–25-4
How to Set Backup Options, 25-8–25-9
Incremental Backup Begin Time, 25-8
Incremental Backup Interval Time, 25-8
Journal File Synchronization, 25-12–25-13
Journal Trigger Value, 25-7
Log execution plan, 25-20
Log file number, 25-19
Log Input parameter's value, 25-20
Log lock information, 25-21
Log SQL statement, 25-19–25-20
Log System, 25-16
Server Log directory, 25-21
SQL Command Monitor, 25-13
SQL Display Mode, 25-13
System Control Settings, 25-11–25-12
Scale, 8-5
Schedules
Replication, 15-1, *See* "Replication Schedule"
Security Level
Authority Level, 20-8–20-9
Users, 20-8
Serial Number, 5-40
Resetting, 5-40
Sessions
Execution Information, 24-17–24-18
Lock Waiting Information, 24-20–24-21
Login Information, 24-16–24-17
Monitoring, 24-14–24-15
Statistical Information, 24-18–24-19
Transaction Information, 24-19–24-20
Show SQL Script, 25-25
Shrinking and Compressing Tablespaces, 4-13
How to, 4-14–4-15
Signature Text Indexes
Creating, 8-2
How to, 8-2–8-6
Dropping, 8-8
How to, 8-8
Rebuilding, 8-10
How to, 8-10–8-11
Statistics, 8-12
How to Check, 8-12
SQL Command Monitor
Display Mode, 25-13

©Copyright 1995-2026 CASEMaker Inc.

Statistics
Database, 24-29–24-30
Indexes, 7-10–7-11
IVF Text Indexes, 9-14
How to Check, 9-14
Signature Text Indexes, 8-12
How to Check, 8-12
Tables, 5-47
Viewing, 24-2–24-4, 24-14
How to, 24-5–24-9
Status
Database, 24-25
Storage Options
Extending a File , Run Time, 25-14
Stored Commands, 12-1
Creating, 12-2
How to, 12-3–12-4
Dropping, 12-8
Executing, 12-5
How to, 12-6–12-7
Stored Procedures, 13-1
Creating, 13-1–13-2
How to, 13-3–13-7
Dropping, 13-11
How to, 13-11–13-12
Executing, 13-9
How to, 13-9–13-10
Privileges, 13-13
How to, 13-13–13-14
User Privileges, 12-10
How to, 12-10–12-11
Subscribers
Adding to Asynchronous Table Replication, 16-31
How to, 16-31–16-35
Adding to Synchronous Table Replication, 16-14
How to, 16-15–16-19
Dropping, 16-36
How to, 16-36–16-37
Summary of Features, 2-2–2-3
Synchronous Table Replication, 16-4
Adding Subscribers, 16-14
How to, 16-15–16-19
Creating, 16-4
Dropping, 16-38
How to, 16-39
How to Create, 16-5–16-14
Synonyms, 18-1
Creating, 18-2
How to, 18-2–18-3
Dropping, 18-4
How to, 18-4
System Control Settings, 25-11–25-12
How to Change, 25-15
System Tables
Displaying, 25-25–25-26
SystemInformation, 24-31
Table Replication, 16-1

©Copyright 1995-2026 CASEMaker Inc.

Asynchronous, 16-20, *See* "Asynchronous Table Replication
Dropping, 16-38
How to, 16-39
How to View, 16-2–16-3
Synchronous, 16-4, *See* "Synchronous Table Replication
Tables, 5-1
Creating, 5-3
Adding a Foreign Key, 5-18
How to, 5-20–5-25
Adding a UNIQUE Constraint, 5-14
How to, 5-15–5-17
Completing Table Creation, 5-28
How to, 5-29
Setting Advanced Table Options, 5-26
How to, 5-27–5-28
Setting Column and Table Constraints, 5-11
How to, 5-12–5-13
Setting Column Options, 5-8–5-9
How to, 5-9–5-10, 5-9–5-10
Setting Table, Tablespace, and Schema Options, 5-6
How to, 5-7–5-8
Starting the Wizard, 5-4
How to, 5-5–5-6
Dependencies, 5-45
How to, 5-45–5-46
Dropping, 5-54
How to, 5-54
Editing Data, 5-49
How to, 5-49–5-51
Modifying, 5-31
Adding Columns, 5-37
How to, 5-38
Column Constraints, 5-37
How to, 5-37
Column Data Type, 5-33
How to, 5-34
Column Default Values, 5-36
How to, 5-36
Column Name, 5-33
How to, 5-33
Column Nullable, 5-35
How to, 5-35
Column Order, 5-32
How to, 5-32–5-33
Column Precision and Scale, 5-34
How to, 5-34
Column Primary Key, 5-35
How to, 5-36
Deleting Columns, 5-38–5-39
How to, 5-38–5-39
How to, 5-31–5-32
Privileges, 5-42
How to Manage, 5-43–5-44
Properties, 5-40
How to, 5-41
Renaming, 5-52
How to, 5-52–5-53
Statistics, 5-45, 5-47
How to, 5-47
Tablespaces, 3-1, 3-1, 3-1
Adding Files, 4-10–4-11
How to, 4-11–4-13
Altering, 4-10

©Copyright 1995-2026 CASEMaker Inc.

Autoextend, 4-8
Browsing, 4-2
How to, 4-3–4-4
Changing the Mode, 4-17
How to, 4-17–4-18
Creating, 4-5
How to, 4-6–4-9
Dropping, 4-21–4-22
How to, 4-21–4-22
Modifying a File, 4-15
How to, 4-16
Shrinking and Compressing, 4-13
How to, 4-14–4-15
Updating Statistics, 4-20
How to, 4-20
Technical Support, 1-4
Text
Exporting, 23-37–23-39
How to, 23-39–23-44
Importing, 23-4–23-6
How to:, 23-7–23-12
Text Indexes, 8-1
IVF, 9-1
Rebuilding
How to, 9-12
Total Text Size, 8-5, 9-7
Transactions
Pending, 24-36
Status, 24-35
Transfer
Data. *See* "Data Transfer"
Tree, 2-6
Trigger Event, 11-7
Trigger Type, 11-7
Triggers, 11-1
Creating
Naming, 11-2
How to, 11-4–11-6
Creating, 11-2
Creating
Action Settings, 11-6–11-7
Creating
Action Settings
How to, 11-8–11-9
Creating
Indicating the Reference Clause, 11-9
Creating
Indicating the Reference Clause
How to, 11-10
Creating
WHEN Condition Clause, 11-11
Creating
WHEN Condition Clause
How to, 11-12
Creating
Trigger Action, 11-13
Creating
Trigger Action
How to, 11-13–11-15
Dropping, 11-16
How to, 11-17

©Copyright 1995-2026 CASEMaker Inc.

Modifying, 11-18
How to, 11-19
UNIQUE Constraint, 5-14
How to, 5-15–5-17
Updating Statistics
Tablespaces, 4-20
How to, 4-20
User Accounts, 20-1
Creating, 20-2
How to, 20-3–20-7
Deleting, 20-16
How to, 20-16–20-17
User Preferences, 25-23
Users, 20-1
Authority Level, 20-8–20-9
How to Change, 20-9
Creating Accounts, 20-2
How to, 20-3–20-7
Deleting Accounts, 20-16
How to, 20-16–20-17
Group Membership, 21-6
How to Add and Remove, 21-7–21-8
Passwords, 20-14
How to Change, 20-14
Privileges, 20-10–20-11
How to Change, 20-11–20-13
Security Level, 20-8
View SQL Command, 25-26
Views, 6-1
Creating, 6-2
How to, 6-2–6-7
Dropping, 6-12–6-14
How to, 6-14–6-15
Privileges, 6-10
How to, 6-10
Properties, 6-8
How to Check, 6-8–6-9
Workspace, 2-4–2-5
XML
Batch File, 23-17
Exporting, 23-46–23-49
How to, 23-50–23-56
Importing, 23-13
How to, 23-13–23-16

©Copyright 1995-2026 CASEMaker Inc.