SQL Studio (GUI)

HELP.BCSDBSQL

Release 730

Copyright

Icons in Body Text

Icon	Meaning
⚠	Caution
❖	Example
💡	Note
⬆	Recommendation
<>	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help* → *General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
Example text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example text`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, F2 or ENTER.

SQL Studio (GUI)

Purpose

⚠

We recommend that you use the database tool Database Studio [Extern] instead of SQL Studio. Database Studio has the same functions as SQL Studio and can be used on all platforms supported by MaxDB.

You can use the SQL Studio database tool to access the data in database instances using SQL statements.

SQL Studio has a graphical user interface and you can use this tool in Windows only. However you can use SQL Studio to access a remote computer running a different operating system.

Features

SQL Studio provides the following functions:

- Creating and managing database tables, indexes, and sequences, and managing stored SQL statements, database users, and database procedures with Catalog Manager [Seite 11]
- Creating and executing SQL statements in the SQL dialog [Seite 24]
- Selecting and editing data records as forms with the Form dialog [Seite 26]
- Formulating database queries with the help of visual tools in the Visual Query dialog [Seite 29]

The SQLCLI and Web SQL Studio database tools have similar features to SQL Studio but with a different user interface.

Starting SQL Studio

Procedure

To start SQL Studio, select *Start* → *Programs* → *MaxDB* → *SQL Studio*.

Starting from the Command Line

You can also start SQL Studio from the command line. In this case, you can specify options.

1. Switch to the directory in which the program sqlsto.exe is stored.
2. Enter the following command:

```
sqlsto [<options>]
```

You have the following options:

`-?`	Displays all command line options
`-version`	Displays information on the software version
`-n <database_computer>`	Name of the Database Computer

`-d <database_name>`	Name of the database instance
`-u <database_user>[,<database_user_password>]`	Name and password of the database user
`-c <sql_mode>`	SQL Mode Possible values: `INTERNAL | ORACLE | ANSI | DB2` If you log on to a database instance (version 7.6 or higher) with SQL Studio, you can only choose the SQL modes INTERNAL or ORACLE. The SQL modes DB2 and ANSI are no longer supported. After you start SQL Studio, you can also change the SQL mode in the User Settings [Seite 9].
`-t <trace_file>`	Activates the ODBC trace

See also:

Opening a Database Session [Seite 7]

Opening a database session

Use

To be able to execute SQL statements in a database instance, you have to open a database session.

Procedure

1. If you do not want to have a timeout for the database session you are opening, then select *No Timeout*.
2. Choose *Session* → *Connect*.
3. Enter the name of the database computer and the database instance.
4. Enter the name and password for the database user.

   ⚠

   The database system converts all letters to uppercase. If you want to use upper and lower case letters and/or special characters, place the user name or the password in double quotation marks.
5. Choose *OK*.

Result

The database system opens a database session.

If you want to open an additional database session in a **new** window, choose *Session* → *New SQL Studio*.

See also:

Recent Sessions List [Seite 8]

Ending the Database Session [Seite 34]

Recent Sessions List

Use

The recent sessions list displays database sessions that have recently been opened. The list can contain up to 10 entries.

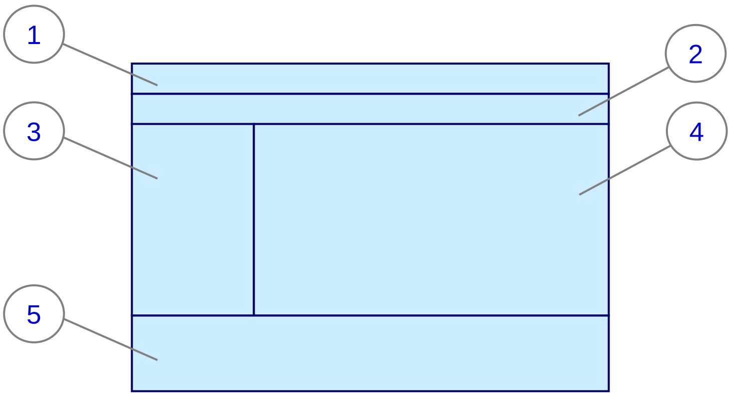
Activities

- You can save a user name and password for database sessions; see User Settings [Seite 9].
- To delete the recent session list, select *Session* → *Clear Recent Sessions*.

  This function is only available as long as you have not opened a database session in SQL Studio.

SQL Studio Screen Areas

The SQL Studio screen is divided into five areas. You can modify the size of the individual screen areas.



Content of the Screen Areas

No.	Name of Area	Description
1	Menu bar	Commands for the current database session
2	Toolbar	Icons for the main commands
3	Catalog Manager [Seite 11]	Available database objects
4	Work area	The work area can include the following windows: • SQL Dialog [Seite 24] • Form Dialog [Seite 26] • Visual Query Dialog [Seite 29]

		• Result Tables [Seite 33]
5	Status Window [Seite 34]	Information and error messages

User Settings

Use

You can personalize the settings in SQL Studio while a session is running. These settings are stored when you exit SQL Studio.

Procedure

To change your settings, choose *Session* → *Settings*.

You can make the following settings:

General

Store Password	The system stores the password that you used at logon for the next logon.
Confirmation for saving SQL Studio Objects	When the dialog window closes, the system asks whether you want to save the newly created SQL Studio object [Seite 11].
Open Form Dialog read only	Form dialogs [Seite 26] are opened in read-only mode first.
Remove failed connection attempts from Recent Sessions list	The system deletes the failed attempts from the recent sessions list [Seite 8].
Identifiers in Create Dialogs uppercase	When you create a new SQL Studio object, the system automatically converts the entries to uppercase.

Result Window

Limit for LOB columns (Bytes)	Maximum number of bytes that can be loaded from the database for a LOB column.
Limit for binary zoom (Bytes)	Maximum number of bytes that can be displayed when zooming into the LOB column of a result table.
Representation of null value	How the NULL value is displayed.
Clipped result view (Internal Mode)	Result set displayed in segments.
Replace non chars in LOB columns	ASCII codes < 32 are replaced by a blank character when LOB columns are displayed.
Copy columns tab separated to clipboard	When results are temporarily stored on the clipboard, the system separates the columns with tabs.
Copy Result with Column Header	When the results are copied, the system also copies the column headers.
Append Exported Data to File	When you export a result table [Seite 34] to a file, the content of this result table is attached to the existing file content.

Display Char Byte Column as Hex Values	Data with the data type CHAR BYTE is displayed as hexadecimal digits. If you do not choose this option, the data is displayed in binary format.

Catalog Manager

Selected table types	Specifies which table types are displayed in the Catalog Manager. [Seite 11]
Path for local folder	Path to be used to save SQL Studio Objects in a local folder.
Restore catalog state	When you log on to the database again, the last Catalog Manager view is restored.
Convert input for catalog filter to uppercase	The system automatically converts filter entries in Catalog Manager to upper case.

Query Dialog

Isolation Level	Isolation level for the creation of SQL statements.
SQL Mode	SQL mode for the creation of SQL statements in the SQL dialog. The INTERNAL SQL mode is always used for the Form dialog and the Visual Query dialog. If you log on to a database instance (version 7.6 or higher) with SQL Studio, you can only choose the SQL modes INTERNAL or ORACLE. The SQL modes DB2 and ANSI are no longer supported.
Execute Statements Asynchronously	SQL statements are executed asynchronously. You can specify a period of time (in milliseconds) during which SQL Studio is to be inactive while SQL statements are executed.
AutoCommit Mode	Default value for autocommit mode.

Fonts

SQL Studio Font	Font type and font size for all text fields, labels, and tree views.
SQL Dialog Font	Font type and font size for the SQL dialog.
Result Window Font	Font type and font size for the result view.
Status Window Font	Font type and font size for the status window.

⚠

If you have the EXCLUSIVE user attribute, you cannot change the standard settings for the isolation level or SQL mode. In this case, SQL Studio automatically works with the COMMITTED isolation level and in the INTERNALSQL mode.

For an explanation of the user attributes, see the *SQL Reference Manual.*

Unicode Support

Use

SQL Studio only supports Unicode to a limited extent.

Prerequisites

In the regional settings for your operating system, you have configured the language in question for non-Unicode programs and selected all the necessary code page conversion tables.

Procedure

Configure the font that you require in the user settings [Seite 9] under *Fonts.*

Catalog Manager

Catalog Manager is located on the left-hand area of the SQL Studio window. It displays all the directories and database objects for which you have at least read access.

SQL Studio Objects [Seite 11]	Stored SQL statements and draft database tables
Tables [Seite 13]	Database and system tables, view tables, and synonyms arranged according to user
Indexes [Seite 20]	Indexes that have been created for tables, arranged according to table and owner
Number series [Seite 21]	Generated number series, arranged by owner
User [Seite 23]	All users for which the user who is logged on to the database has owner rights
Database procedures [Seite 23]	Database procedures that the user created in the database
Trigger [Seite 23]	Triggers that the user created in the database

If you want Catalog Manager to show you certain objects only, specify a filter for the object names. Enter the first few letters of the name that you want to use as a filter, and choose *Catalog Manager → Refresh*. However, the filter does not apply to your saved SQL statements and drafts.

To show or hide Catalog Manager, choose *View → Catalog Manager*.

SQL Studio Objects

You can store SQL statements and drafts of database tables as SQL Studio objects in the database on locally in the *Local Folder* subdirectory. This means that you can use the objects at a later date.

Creating SQL Studio Objects

- To create a new folder for SQL Studio objects in the database using the Catalog Manager [Seite 11], choose *Catalog Manager → New → Folder.*
- To create SQL statements and store them as SQL Studio objects, use the SQL dialog [Seite 24], the Form dialog [Seite 26] or the Visual Query dialog [Seite 29].
- To store drafts of database tables as SQL Studio objects, create the table definition [Seite 13] with the Catalog Manager.

You can enter the path for the *Local Folder* in the user profile settings [Seite 9].

Editing and Executing SQL Studio Objects

- To edit a stored SQL Studio object, select the object and choose *Catalog Manager → Open Object Definition*.
- To use a stored SQL Studio object in dialogs or a table definition, select the object. Choose *Catalog Manager → Send Object To* and then the relevant dialog type.
- To execute the SQL statements of a stored SQL Studio object directly, select the object and choose *Catalog Manager → Execute*. The system displays the result table in a separate window. If an error occurred, an error message is displayed in the status window [Seite 34] of the SQL Studio. Depending on your settings, additional information such as the time it took to execute your SQL statements is also displayed there.

Renaming and Deleting SQL Studio Objects

- To rename a stored SQL Studio object, select the object and choose *Catalog Manager → Rename Object*.
- To edit a stored SQL Studio object, select the object and choose *Catalog Manager → Drop Object*.

Exporting a Database Object Definition

Use

You export the definition of a database object, for example, the definition of a table or index.

Procedure

1. In Catalog Manager, select the database object whose definition you want to export.
2. Choose *Catalog Manager → Open Object Definition.*
3. Choose the *Definition → Export to SQL Dialog* table.
4. From the context menu for the SQL dialog, choose *Export File* and enter the name and path of the file to which you want to export the definition.

Result

The database object definition is exported as an SQL statement.

Example

If a user called MONA exports the definition for the CUSTOMER table, the following SQL statement is returned:

```
CREATE TABLE "MONA"."CUSTOMER"
(
   "CNO"               Fixed (4,0)    NOT NULL,
   "TITLE"             Char (7) ASCII,
   "FIRSTNAME"         Char (10) ASCII,
   "NAME"              Char (10) ASCII    NOT NULL,
   "ZIP"               Char (5) ASCII,
   "ADDRESS"           Char (25) ASCII    NOT NULL,
   PRIMARY KEY ("CNO"),
   CONSTRAINT "CNO_CONS" CHECK   "CNO " > 0,
   CONSTRAINT "TITLE_CONS" CHECK "TITLE " IN ('Mr', 'Mrs',
'Company'),
   CONSTRAINT "ZIP_CONS" CHECK   "SUBSTR(ZIP,1,1)" BETWEEN '0' AND
'9'     AND
               "SUBSTR(ZIP,2,1)" BETWEEN '0' AND '9'     AND
               "SUBSTR(ZIP,3,1)" BETWEEN '0' AND '9'     AND
               "SUBSTR(ZIP,4,1)" BETWEEN '0' AND '9'     AND
               "SUBSTR(ZIP,5,1)" BETWEEN '0' AND '9',
   FOREIGN KEY "CUSTOMER_ZIP_IN_CITY"  ("ZIP") REFERENCES
"MONA"."CITY" ("ZIP") ON DELETE  RESTRICT
)
```

Tables

Under *Tables*, the Catalog Manager [Seite 11] displays the database tables, view tables and synonyms that you have access to in accordance with your user settings [Seite 9]. The entries are arranged by owner.

You can create and edit database tables. You can create, rename, and delete view tables and synonyms.

- Creating a database table [Seite 13]
- Creating a view table [Seite 17]
- Creating a synonym [Seite 18]
- Displaying the definition of a table [Seite 18]
- Changing the definition of a database table [Seite 18]
- Displaying the content of a table [Seite 19]
- Using a Table in Dialogs or Definitions [Seite 19]
- Renaming a table [Seite 19]
- Deleting all rows of a table [Seite 20]
- Deleting a table [Seite 20]

Creating a Database Table

Use

You can use SQL Studio to create new database tables in the database where you are currently logged on. You can use existing database tables or stored SQL Studio objects [Seite 11] as a template for the new database table.

Prerequisites

You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

1. Choose *Catalog Manager* → *New* → *Table Definition*.
A window for defining tables opens.
2. To use an existing database table or a saved SQL Studio object as a template for your new database table, choose *Table Definition* → *Get Table*. Select the relevant table or the SQL Studio object, and click *OK* to confirm.
3. Enter a table name.
4. Perform the following steps:

   Defining Columns [Seite 14] (*Columns*)

   Defining Column Constraints [Seite 15] (*Constraints*)

   Defining Foreign Key Dependencies [Seite 16] (*Foreign Keys*)

   Defining Miscellaneous Information [Seite 16] (*Misc*)
5. To re-use or edit your table draft later, save it as an SQL Studio object [Seite 11] by choosing *Table Definition* → Save As.
6. Choose *Table Definition* → *Create Table*. The system creates the new table in the database.
7. To export the SQL statement for creating the database table to the SQL dialog, choose *Table Definition* → *Export to SQL Dialog*.

Defining Columns

Use

This section shows you how to define a table column when you create a database table [Seite 13] or change a table definition [Seite 18].

Procedure

1. In the *Name* field, enter a name for the column, and confirm it with `Enter`.
2. In the *Datatype* field, select the relevant data type.
3. Depending on the data type, you can specify additional properties:
   - Dimension *(Dim)*
   - Code type *(Code)*
   - Column is key column
   - Columns may not contain NULL values *(Not NULL)*
   - Default value for column value *(Default)*
   - Uniqueness of column values *(Unique)*
   - Comment

- To insert a new column, choose *New Column*. The new column is added to the end of the table.

- To insert a new column and at the same time copy the definition of existing columns in the same table, select the required columns, and choose *Clone Column*. The new column is added to the end of the table.

- To insert new columns and copy the definition of existing columns from existing tables, choose *Import Column*. A window opens, containing all columns defined by the current user. Select one or more columns and choose *OK*. The new column is added to the end of the table.

- In order to remove a column from the table, select the column and choose *Remove Column*.

  You can change the columns sequence by selecting one or more columns and moving them upwards or downwards using the arrows ⇑ ⇓.

Defining Constraints

Use

This section shows you how to define constraints for table columns when you create a database table [Seite 13] or change a table definition [Seite 18]. A constraint for the values of a column is composed of one or more conditions that you formulate line-by-line and can link together.

Prerequisites

You have already defined the table columns [Seite 14].

You can only formulate constraints for columns with valid column definitions.

Procedure

1. In the *Name* field, enter a name for the constraint (optional).
2. In the *Column* field, select the column that you want to formulate the condition for.
3. In the *Predicate* field, select a predicate for the condition. To negate the condition, use the NOT operator `!`. Negation is not possible with relational operators.
4. In the *Expression* field, enter the relational expression for the condition, and confirm it with `Enter`.
   When you enter the relational expression, you can choose between the following options:

Option	**Meaning**
Single Value	The expression consists of a single value.
ALL	The condition must be fulfilled with all expressions in the list (*Expression List)*.
ANY	The condition must be fulfilled with at least one expression in the list (*Expression List*).
DEFAULT	The expression consists of the default value of the column.

5. Enter how the formulated condition is to be linked with the next condition (`AND`/`OR`).

If you retrospectively reset the relation type to <*none*>, all subsequently formulated conditions are deleted immediately.

6. To insert a new constraint, choose *New Constraint*.
7. To insert new constraints and at the same time copy the definition of existing constraints in the same table, select the required constraints, and choose *Copy Constraint*.
8. To insert new constraints and copy the definition of existing constraints from existing tables, choose *Import Constraint*. A window opens, containing all constraints defined by the current user. Select one or more constraints and choose *OK* to confirm.
9. In order to delete a constraint, select the constraint and choose *Remove Constraint*.

Defining Foreign Key Dependencies

Use

This section shows you how to define foreign key dependencies (foreign keys, referential CONSTRAINT definitions) to other database tables when you create a database table [Seite 13] or change the table definition [Seite 18]. Foreign key dependencies describe the dependencies between the data of two tables.

Prerequisites

You have already defined the table columns [Seite 14].

Procedure

1. In the field *Foreign Key Name*, enter the name for the foreign key (optional).
2. In the field *Referencing Columns*, choose the referencing column, and confirm with `Enter`.

Copy the referenced database table from the Catalog Manager [Seite 11] into *Reference* by using drag and drop. Select the column that is referenced and confirm with `Enter`.

3. In the field *On Delete*, select how you want the referenced table to behave if a data record in the referencing table is deleted (DELETE rule).
4. In the *Comment* field, enter a comment (optional).
5. To add a new link, choose *New Foreign Key*.
6. To insert new links and at the same time copy the definition of existing links of the same table, select the required links, and choose *Copy Foreign Key*.
7. To insert new links and copy the definition of existing links from existing tables, choose *Import Foreign Key*. A window opens, containing all links defined by the current user. Select one or more links and choose *OK*.
8. To delete a link, select it and choose *Remove Foreign Key*.

For information on the syntax of referential CONSTRAINT definitions, see the *SQL Reference Manual*.

Defining Miscellaneous Information

When you create database tables [Seite 13] or change table definitions [Seite 18], you can make the following additional entries in the *Misc* field:

Specification	Meaning
Comment	Notes
Duplicates	You can use the DUPLICATES clause to control behavior if key collisions occur.
Statistics	*Sample Value* specifies how large the portion of data is that is included for optimization of the database strategy.
Temporary Table *Ignore Rollback*	The table is a temporary table and is saved locally under *Temp*. For temporary tables, you can specify the option *Ignore Rollback*.
Identifiers Uppercase	Automatic conversion of table names, column headers and other names into uppercase. ⚠ A change to this setting affects the entire table definition immediately. If you use a database table as a template for the creation of another database table, the user setting is copied from the template.

Creating a View Table

Use

You can use SQL Studio to create view tables.

Prerequisites

- If you want the view table to be based on a database query, you first need to save this query as an SQL Studio object [Seite 11].
- You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

1. Choose *Catalog Manager*→ *New* → *View Definition*.
   A window for defining the view table is opened.
2. To use a database table as a basis for the view table, choose *View Definition* → *Get Table*. Choose the relevant database table, and click *OK* to confirm.
3. To use the result set structure of a stored database query as the basis for the view table, choose *View Definition* → *Get Stored Query.* Choose the relevant database query, and click *OK* to confirm.
4. Enter a name for the view table.
5. If you want to rename the columns of your view table, enter the new column names under *Alias* in *View*.
6. Choose the required options:

Replace Existing View	Replaces the existing view table
Check on Insert or Update	Checks the view conditions when changing values

Identifiers Uppercase	Automatically converts table names and column headers into uppercase.

7. Choose *View Definition* → *Create View*.

   To export the SQL statement for creating the view table to the SQL dialog, choose *View Definition* → *Export to SQL Dialog*.

Creating a Synonym

Use

You can use SQL Studio to create synonyms for database tables.

Prerequisites

You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

In the Catalog Manager [Seite 11], select the database table where you want to create a synonym, and choose *Catalog Manager* → *New* → *Synonym Definition.*
The system opens a window for defining the synonym.

1. Enter a synonym name and choose whether the synonym is to be private or *Public*.
2. Choose *Synonym Definition* → *Create Synonym*.

   To export the SQL statement for creating the synonym to the SQL dialog, choose *Synonym Definition* → *Export to SQL Dialog*.

Displaying the Definition of a Table

Use

You can use SQL Studio to display the definition of a database table, view table, or synonym.

Prerequisite

You need to have the required user authorizations.

Procedure

1. In Catalog Manager [Seite 11], select the table whose definition you want to view.
2. Choose *Catalog Manager* → *Open Object Definition.*

Changing the Definition of a Database Table

Prerequisites

You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

1. Select the table and choose *Catalog Manager* → *Alter Table*.
   A window for defining tables opens.
2. Make the required changes:

   Defining Columns [Seite 14] (*Columns*)

   Defining Column Constraints [Seite 15] (*Constraints*)

   Defining Foreign Key Dependencies [Seite 16] (*Foreign Keys*)

   Defining Miscellaneous Information [Seite 16](*Misc*)
3. Choose *Table Definition* → *Alter Table*.

   To export the SQL statement for changing the database table to the SQL dialog, choose *Table Definition* → *Export to SQL Dialog*.

Displaying the Content of a Table

Prerequisites

You need to have the required user authorizations.

Procedure

1. In Catalog Manager [Seite 11], select the table whose content you want to display.
2. Choose *Catalog* Manager → *Show Content*.

Using a Table in Dialogs or Definitions

Use

You can copy tables directly from the Catalog Manager [Seite 11] into an SQL dialog [Seite 24], a form dialog [Seite 26], or a Visual Query dialog [Seite 29]. You can also use tables as the basis for a table definition or view definition.

Procedure

Select the table in the Catalog Manager. Choose *Catalog Manager* → *Send Object To* and then the relevant dialog type.

Renaming a Table

Prerequisites

You need to have the required user authorizations.

Procedure

In Catalog Manager [Seite 11], select the table that you want to rename.

1. Choose *Catalog Manager → Rename Object* and enter the new name.

Deleting all Rows in a Table

Prerequisites

You need to have the required user authorizations.

Procedure

In Catalog Manager [Seite 11], select the table whose rows you want to delete.

3. Choose *Catalog Manager → Delete All Rows*

⚠

If you delete the rows of a synonym, you also delete the rows of the original database table.

Deleting a Table

Prerequisites

You need to have the required user authorizations.

Procedure

In Catalog Manager [Seite 11], select the table that you want to delete from the database.

4. Choose *Catalog Manager → Drop Object*.

Indexes

Under *Indexes*, the Catalog Manager [Seite 11] displays all the tables that indexes have been created for. The following options are available:

- Creating an index [Seite 20]
- Displaying the definition of an index [Seite 21]
- Deleting an index [Seite 21]

Creating an Index

Use

You can use SQL Studio to create new indexes for database tables.

Prerequisites

You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

1. In the Catalog Manager [Seite 11], select the database table where you want to create an index, and choose Catalog Manager → New → Index Definition.
2. In the *Index Name* field, enter a name for the index.
3. In the *In Index* field, choose *Yes* for the columns whose values you want to index.
4. To sort the index values in ascending or descending order, choose the *ASC* or *DESC* option in the *Sort* field.
5. To change the sequence of the table columns, select a table column and use the ⇑ ⇓, arrows to move it up or down.
6. Select the required options:

Unique	Ensures that the indexed columns are unique.
Index Name Uppercase	Automatically converts table names and column headers into uppercase.

7. Choose *Table Definition* → *Create Index*.
8. To export the SQL statement for creating the index to the SQL dialog, choose *Index Definition* → *Export to SQL Dialog*.

Displaying the Definition of an Index

Prerequisites

You need to have the required user authorizations.

Procedure

1. In Catalog Manager [Seite 11], select the index whose definition you want to view.
2. Choose *Catalog Manager* → *Open Object Definition.*

Deleting an Index

Prerequisites

You need to have the required user authorizations.

Procedure

In Catalog Manager [Seite 11], select the index that you want to delete.

3. Choose *Catalog Manager* → *Drop Object*.

Number Sequences

Under *Sequences* , the Catalog Manager [Seite 11] displays all the sequences that were created in the database. The following options are available:

- Creating a sequence [Seite 22]
- Displaying the definition of a sequence [Seite 22]

- Deleting a sequence [Seite 22]

Creating a Sequence

Use

You can use SQL Studio to create new sequences in the database.

Prerequisites

You need to have the necessary user authorizations and the NOT EXCLUSIVE user attribute.

Procedure

1. Choose *Catalog Manager* → *New* → *Sequence Definition*.
2. Enter a name for the sequence, and define the required values.
3. Choose *Sequence Definition* → *Create Sequence*.
4. To export the SQL statement for creating the sequence to the SQL dialog, choose *Sequence Definition* → *Export to SQL Dialog*.

Displaying the Definition of a Sequence

Use

You can use SQL Studio to display the definition of a sequence and the most recently generated value of this sequence.

Prerequisite

You need to have the required user authorizations.

Procedure

4. In Catalog Manager [Seite 11], select the sequence whose definition you want to display.
5. Choose *Catalog Manager →Open Object Definition.*

Deleting a Sequence

Prerequisites

You need to have the required user authorizations.

Procedure

In Catalog Manager [Seite 11], select the sequence that you want to delete.

6. Choose *Catalog Manager* →*Drop Object*.

User

Definition

Under *Owned Users*, the Catalog Manager [Seite 11] displays the users for which you have owner rights. SQL Studio allows you to view information about these users, and rename and delete users.

Prerequisites

You need to have the required user authorizations.

Procedure

- To display information about a user, select the user name, choose *Catalog Manager → Open Object Definition.*
- To change a user name, select it and choose *Catalog Manager → Rename Object.*
- To delete a user, select the user name, and choose *Catalog Manager → Drop Object*.

Database Procedures

Definition

Under *Procedures*, the Catalog Manager [Seite 11] displays all the database procedures that were created in the database. You can use SQL Studio to display the definitions of these database procedures, and delete the database procedures.

Prerequisites

You need to have the required user authorizations.

Procedure

- To display a database procedure, choose Catalog *Manager → Open Object Definition*.
- To delete a database procedure, select it and choose *Catalog Manager → Drop Object.*

Triggers

Definition

Under *Triggers*, the Catalog Manager [Seite 11] displays all the triggers that were created in the database. You can use SQL Studio to display the definitions of these triggers, and delete the triggers themselves.

Prerequisites

You need to have the required user authorizations.

Procedure

- To display the definition of a trigger, select it and choose *Catalog Manager → Open Object Definition*.

- To delete a trigger, select it and choose *Catalog Manager* → *Drop Object*.

SQL Dialog

If you want to create SQL statements directly in SQL, you use the SQL dialog.

The following options are available in the SQL dialog:

- Creating and Executing SQL Statements [Seite 24]
- Setting Parameters for SQL Statements [Seite 25]
- Importing and Exporting SQL Statements [Seite 26]

Creating and Executing SQL Statements

Use

In the SQL dialog [Seite 24], you can create and save SQL statements in a number of different modes.

Procedure

5. Choose *View* → *SQL Dialog*. The system opens a window for the SQL dialog.
6. The default values that you set in the user settings [Seite 9] are used for the SQL mode and the isolation level. If you want to use different values, select a different SQL mode and isolation level in the lower part of the dialog box.

   If you log on to a database instance (version 7.6 or higher) with SQL Studio, you can only choose the SQL modes INTERNAL or ORACLE. The SQL modes DB2 and ANSI are no longer supported.
7. You can also change the commit mode in the lower part of the dialog box.

   If you have the user attribute EXCLUSIVE, you can select the mode Auto Commit: ON only.
8. Enter your SQL statements.

   You can add SQL statements that were already executed during the current database session. Choose *SQL Dialog* → *Recent Statements.* Select the required statement, and choose *Get.*
9. To comment out a row, use // or –– at the start of the row.
10. To separate several SQL statements from one another, enter a commented-out separator line.
11. To scroll through the statement history of the database session, choose *SQL Dialog* → *Previous Statement* or *SQL Dialog* → *Next Statement*
12. You can set parameters [Seite 25] for SQL statements.
13. To use or edit your statements again later, save them as SQL Studio objects [Seite 11] by choosing *SQL Dialog* → *Save As*.

14. To execute the SQL statements, choose *SQL Dialog* → *Execute*. If you only want to execute some of the displayed SQL statements, select the section that you want to execute and choose *SQL Dialog* → *Execute*.

Result

The status line of the statement window indicates whether your SQL statements were executed successfully. If an error occurred, an error message is displayed in the status window [Seite 34] of the SQL Studio. Depending on your settings, additional information such as the time it took to execute your SQL statements is also displayed there.

If your SQL statement returns a result table, this is displayed in a separate window.

Keyword List

Definition

The keyword list contains reserved terms.

Use

The list is available on the SQL dialog [Seite 24] and provides list support for creating SQL statements [Seite 24].

To access the list of keywords, choose *SQL Dialog* → *Show Keyword List*.

Setting Parameters for SQL Statements

Use

In the SQL dialog [Seite 24], you can set parameters for SQL statements. When you execute the SQL statement for which a parameter has been set, a dialog window opens in which you must enter a value for the parameter.

Procedure

1. When you create an SQL statement [Seite 24], insert square brackets in all the places you want a variable entry to be possible.
2. If you want to formulate a prompt for the dialog window, enter this prompt within the square brackets.

   If you want to set parameters for queries for the columns `Article No.` and `Unit Price` in the `Article` table, you can set the following parameters among the selection criteria:

   ```
   SELECT  "Article No.", "Unit Price" FROM "Article"
   WHERE  "Article No." > [article number (from)]
   AND  "Article No." < [article number (to)]
   AND  "Unit Price" <= [upper price limit]
   ```

   By setting parameters you can also replace entire sections of an SQL statement.

   You can formulate a SELECT statement and add a WHERE condition using a parameter:

```
SELECT * FROM USERS [Your WHERE condition]
```

If you specify a comparison value for the data type CHAR, VARCHAR, DATE or TIME, you must enclose the comparison value in single quotation marks.

Importing and Exporting SQL Statements

Use

In the SQL dialog [Seite 24], you can import and export SQL statements as ASCII format text files.

Procedure

Choose *Import File* or *Export File* in the context menu of the SQL dialog.

Form Dialog

You use the *Form dialog* if you want to display, select, and edit data records from tables in a screen display.

The following options are available in the *Form dialog*:

- Executing a Form dialog [Seite 26]
- Selecting data records using search conditions [Seite 27]
- Changing data records [Seite 27]
- Inserting data records [Seite 28]
- Deleting a data record [Seite 28]
- Importing and exporting LONG columns [Seite 29]

If you want to create statements directly in SQL or want to use database options other than the editing of data records, use the SQL dialog [Seite 24].

Executing a Form Dialog

Use

You can use the Form dialog [Seite 26] to execute statements to edit data records at different isolation levels.

Procedure

1. Choose *View* → *Form Dialog*. The system opens a window for the Form dialog.
2. Depending on your user settings, at first you are in read only or in change mode. If you want to change the dialog mode, activate or deactivate the *Read Only* selection in the lower part of the dialog box.

3. The default value that you set in the user settings is entered for the isolation level. If you want to use a different isolation level, select the relevant isolation level in the lower part of the dialog box.
4. Choose *Form Dialog* → *Get Table*. Select the table that you want to edit, and choose *OK* to confirm.
   The first data record is displayed in the left half of the window. You can use the arrows **< >** to scroll through the data records. In change mode, you can change [Seite 27], insert [Seite 28], or delete [Seite 28] data records. In the right half of the window, you can enter search conditions for selecting specific records [Seite 27].

   To hide the window area for entering the search conditions, deselect the *Filter* flag.
5. To view the data records as a list, choose *Report*. To switch back to the screen display, choose *Record*.
6. To use or edit the dialog again later, save it as an SQL Studio object [Seite 11] by choosing *Form Dialog* → Save As.
7. To export the SQL statement of your Form dialog to the SQL dialog, choose *Dialog* → *Export to SQL Dialog*.

Selecting Data Records Using Search Conditions

Use

To only display certain data records of a table, you can enter search conditions in the Form dialog [Seite 26].

Procedure

1. To display the window area for entering the search conditions, select the *Filter* flag.
2. Formulate your search conditions for the appropriate table columns. Select the required predicate, and enter comparison values or relational expressions.
3. Enter how the formulated search condition is to be linked with the next search condition.
4. Choose *Refresh*. SQL Studio downloads the data records from the database.

   For information on the syntax of predicates, see the *SQL Reference Manual*.

Changing Data Records

Prerequisites

You need to choose change mode in order to execute the Form dialog [Seite 26].

- You need to have the required user authorizations.

Procedure

1. On the screen display, scroll to the data record that you want to change.
2. Enter your changes.

The names of key columns are displayed in bold.

3. Click *Update* to confirm. SQL Studio transfers the changes to the database.

Inserting Data Records

Use

You can use the Form dialog [Seite 26] to insert new data records in tables on a screen display.

Prerequisites

You need to choose change mode in order to execute the Form dialog [Seite 26].

- You need to have the required user authorizations.

Procedure

1. In the screen display, choose *Form Dialog* → *New*. An empty input form opens.
2. Enter your data. Note the following points when doing this:
    - The names of key columns are displayed in bold.
    - For a NULL value, enter the NULL value displayed according to your user settings [Seite 9] (**?**, for example).
    - If you leave a field blank when you insert a data record, the system interprets this as follows:
        - If a DEFAULT value is defined for the relevant column, this value is copied to the database table.
        - If there is **no** DEFAULT value defined for the relevant column, the NULL value is copied to the database table.
    - You can import and export contents from LOB columns.
3. Choose *Insert*. SQL Studio stores the new data record in the table.
   An additional input form opens. You can enter further data on this form directly and insert this as data records.
4. If you do not want to insert any more data records, deselect *Form Dialog* → *New.*
5. To view the new data records in a list, choose *Refresh* and then *Report*.

   When inserting data records, you can use an existing data record from the same table as a template. Scroll to this data record, enter your changes, and choose *Insert*. The data record is inserted in the table as a new data record.

Deleting a Data Record

Prerequisites

You need to choose change mode in order to execute the Form dialog [Seite 26].

- You need to have the required user authorizations.

Procedure

1. On the screen display, scroll to the data record that you want to delete.
2. Choose *Delete*.

⚠

SQL Studio deletes the data record from the database immediately.

Importing and Exporting LOB Columns

Use

You can use the Form dialog [Seite 26] to import the content of LOB columns from a file or the clipboard, and export them to a file or the clipboard.

Procedure

1. To import the content of a LOB column from a file, click the input field of this column in the screen display, and choose *Form Dialog →Import LOB → Import from File.*
2. To import the content of a LOB column from the clipboard, click the input field of this column in the screen display, and choose *Form Dialog → Import LOB → Import Text from Clipboard.*
3. To export the content of a LOB column to a file, click the input field of this column in the screen display, and choose *Form Dialog → Export LOB → Export to File.*
4. To export the content of a LOB column to the clipboard, click the input field of this column in the screen display, and choose *Form Dialog → Export LOB → Export to Clipboard as Text.*

Visual Query Dialog

If you want the system to provide you with visual support when you formulate database queries, you can use the Visual Query dialog.

The following options are available in the Visual Query dialog:

- Creating and executing a Visual Query [Seite 30]
- Linking table columns with JOINS [Seite 31]
- Entering search conditions [Seite 31]
- Setting parameters for search conditions [Seite 32]
- Defining a result view [Seite 32]

If you want to create statements directly in SQL or want to use other database options in addition to database queries, use the SQL dialog [Seite 24].

Creating and Executing a Visual Query

Use

In the Visual Query dialog [Seite 29], you can create and save SQL statements in a number of different isolation levels.

Procedure

1. Choose *View* → *Visual Query.* The system opens a window for the Visual Query dialog. The window consists of the following areas:
   - The top half of the window is the area for selecting tables.
   - The bottom half of the window is the area for defining the result.
2. The default value that you set in the user settings is entered for the isolation level. If you want to use a different isolation level, select the relevant isolation level in the lower part of the dialog box.
3. Choose *Visual Query* → *Get Table*. Select the table that you want to use in your database query, and choose *OK* to confirm. In this way, you can copy up to 16 tables to the area for selecting tables. For each of the tables copied, the system opens a window containing the names of the table columns.
   - To rearrange the tables, choose *Visual Query* → *Arrange Tables* or *Visual Query* → *Cascade Tables*.
   - To remove a table from the table selection, close the relevant window.
   - To remove all tables from the table selection, choose *Visual Query* → *Clear Window*.
4. Select the names of the table columns that your database query relates to, and use the drag and drop function to copy them to the area for defining the result.

   If you enter *, you copy all table columns of a table into the area for defining the result. To remove a table column from the result definition, select the column and choose *Visual Query* → *Delete Column*.
5. You can link table columns with JOINS [Seite 31], enter search conditions [Seite 31] for the data records, and define the result view [Seite 32].
6. To use or edit your Visual Query again later, save it as an SQL Studio object [Seite 11] by choosing Visual Query → Save As.
7. To execute the database query that you created with Visual Query, choose *Visual Query* → *Execute*.
8. To export the SQL statements of your Visual Query to the SQL dialog, choose *Visual Query* → *Export to SQL Dialog.*.

Result when Executing a Visual Query

The status line of the statement window indicates whether your SQL statements were executed successfully. If an error occurred, an error message is displayed in the status window [Seite 34] of the SQL Studio. Depending on your settings, additional information such as the time it took to execute your SQL statements is also displayed there.

If your database query returns a result table, this is displayed in a separate window.

Linking Table Columns with Joins

Use

You can use the Visual Query dialog [Seite 29] to link table columns with Joins in database queries.

Prerequisites

When you created the Visual Query [Seite 30], you copied the tables containing the columns you want to link together to the area for selecting tables.

Procedure

1. In the table selection, select the table column that you want to link, and choose the drag and drop function to link it to the relevant column in the other table. The system displays a connecting line between the two table columns.
2. To specify the Join in more detail, select it, and choose *Show Join Definition* in the context menu. The system opens a window for the Join definition. Select the Join type and relational operator, and choose *OK*.
3. To delete a Join, select it and choose *Delete Join* in the context menu.

> If you copy the same table to the table selection twice, SQL Studio assigns an alias name. In this way, you can create SELF JOINS.

Entering Search Conditions

Use

In the Visual Query dialog [Seite 29], you can enter search conditions when you formulate your database query.

Prerequisites

When you created the Visual Query [Seite 30], you copied the table columns that your database query relates to into the area for result definition.

Procedure

In the relevant table column, enter your search condition in the *Predicate* field.
You can set parameters for the search condition [Seite 32].

> If you specify a search condition for a column with the data type CHAR, VARCHAR, DATE or TIME, you must put the relational expression between single quotation marks.

You can formulate several search conditions with AND links or OR links. If you want to use one column several times to formulate search conditions, but only want to display it once, you can hide it in the result view [Seite 32].

- To formulate several search conditions with AND links, enter these conditions in the same row.
- To formulate several search conditions with OR links, enter each of these conditions in a new row.

For information on the syntax of search conditions and predicates, see the *SQL Reference Manual*.

Setting Parameters for Search Conditions

Use

In the Visual Query dialog [Seite 29], you can set parameters for search conditions for database queries. When you execute the database query a dialog window then opens in which you must enter a value for the parameter.

Procedure

1. When you create a search condition [Seite 31], insert square brackets in all the places you want a variable entry to be possible.
2. If you want to formulate a prompt for the dialog window, enter this prompt within the square brackets.

   You can set parameters for the search condition for the `Order_No` column of the table `Purchase orders` as follows:
   `=[order number:]`

Defining the Result View

Use

In the Visual Query dialog [Seite 29], you can specify how SQL Studio displays the result of your database query.

Prerequisites

When you created the Visual Query [Seite 30], you copied the table columns that your database query relates to into the area for defining the result.

Procedure

1. To assign a synonym to a column name, enter the name of the synonym in the *Synonym* line.
2. To hide a column in the result view, choose *No* in the *Visible* line of the relevant column.

   This is particularly useful if you only need the column in order to formulate search conditions [Seite 31].
3. To group the results according to the column values, choose *Yes* in the *Group* line of the relevant column.

   You can use this grouping feature to define average values, maximum values, minimum values, totals, and the number of found data records in the result view.

4. To sort the data records in ascending or descending order, in the *Order* field, select *Ascending* or *Descending* in the column that you want to use for sorting.

Result Tables

SQL Studio displays the results of each database query in a separate area of the window. You can use the navigation functions in the context menu of a result to scroll through the result table.

The following options are also available:

- Adjusting the result view [Seite 33]
- Saving a result [Seite 34]

If you execute several SQL statements at once in the SQL dialog [Seite 24], several individual result tables are generated. You can use the selection list in the status line of the result window to switch between these result tables.

Adjusting the Result View

Use

You can display the result table [Seite 33] in segments or consecutively. You can rename or hide columns, or fix (freeze) them in the result table to make scrolling easier, and you can change the font of the result table.

Procedure

1. To always display the result table in segments, choose *Clipped Result View* in the user settings [Seite 9].
   In the SQL dialog [Seite 24] and the Visual Query dialog [Seite 29], you can change this setting for the current dialog window. To do this, choose
   SQL Dialog → Clipped Result View or Visual Query → Clipped Result View.
2. To rename, hide or fix columns, choose *Customize* in the context menu for the result. Make the following settings:
   - Change the column names, if required.
   - Change the *Visible* setting for the individual columns, if required.
   - In *Fixed Columns*, enter the number of columns that you want to fix, starting from the left*.*

   Confirm your settings with *OK.*
3. To change the font of the result table, choose *Font* in the context menu for the result. Select the required font, and confirm with *OK*.

Saving a Result

Use

In the SQL mode [Extern] INTERNAL, you can save the complete result [Seite 33] of a database query as a text file, or export it to Microsoft Excel. You can save individual cells as a text file, or display them in a separate zoom window.

Procedure

To save the complete result table as a text file, choose *Export Result to → File* in the context menu for the result.
Depending on your user settings [Seite 9], the content of the result table is attached to the existing file content, or a new file content is generated.
When SQL Studio saves the result, it separates the columns with tabs, and the rows with line breaks. Column contents with the data type CHARACTER or LOB are cut off after 16 characters. Characters that cannot be displayed are replaced by blanks.

- To export the complete result table to Microsoft Excel, choose *Export Result to → Excel* in the context menu for the result.
  Microsoft Excel opens automatically, and transfers a maximum of 1024 bytes per cell. Data with type CHARACTER are cut off after 1021 bytes.
- To save a cell as a text file, choose *Export Cell to → File.*
  Depending on your user settings, the content of the result table is attached to the existing file content, or a new file content is generated.
- To display a cell in a separate zoom window, choose *Export Cell to → Zoom Window.* In the user settings you can decide how many bytes are displayed in the zoom window.

> Exporting and/or displaying the contents of cells is particularly helpful for column contents that have the data type LOB.

Status Window

In the status window on the lower part of the screen of the SQL Studio, the system displays information such as error messages or the time it takes to execute particular SQL statements.

You can search for words or terms, copy the text in the status window, or export it as a text file.

The following options are also available:

- If, in addition to error messages, you want to see information about your actions in the SQL Studio, choose *Status Window → Trace All*.
- If you want to select the entire content of the status window, choose
  Status Window → Select All.
- If you want to delete the content of the status window, choose *Status Window → Clear.*
- If you want to show or hide the status window, choose *View → Status Window*.

Ending the Database Session

To end the current database session, select *Session → Disconnect*.

The database system automatically ends the database session if:

- You close SQL Studio.
- You log on to the database instance as a different database user.
- The timeout value is exceeded.