Re-Executing SQL Statements

As you execute SQL statements in the SQL Commander, DbVisualizer saves them as History entries, along with information about the Connection, Catalog, Schema and the execution result. This makes it easy to locate statements and scripts you have executed in the past.

- Using Previous and Next in the SQL Commander
- Using the SQL History Window
    - Reusing a History Entry
    - Saving a History Entry as a Bookmark or Other File
- Using Quick Load

Using Previous and Next in the SQL Commander

If you just want to go back and forth between statements you have executed recently, you can use the **Get Previous from History** and **Get Next from History** toolbar buttons in the SQL Commander.

Using the SQL History Window

To look through all saved statements, you can display the History entries by clicking the **SQL History** toolbar button in the main window or select the corresponding operation from the **Tools** menu.

The entries are ordered with the most recently executed entries at the top by default, but you can reorder them by clicking on the column headers. The complete content of the selected entry is shown below the list, unless you disable it by clicking the Show Details toolbar button.

The columns show when the entry was executed, a part of the script/statement, the size of the complete statement/script, and then the database type and connection it was executed for, and how long it took to execute. The **Success** and **Errors** columns show how many of the statements in a script were executed successfully or that caused an error. Finally, the **Rows** column show the number of rows retrieved or affected by the script.

You can use the field at the top right corner in the dialog to search for entries matching a critera. Clicking on the magnifying glass reveals a configuration menu where you can, among other things, specify which columns to search in and if you want to search through the complete script rather than just the part of the script shown in the Script/SQL Statement column.

In the Tool Properties dialog, in the **SQL History** category under the General tab, you can specify that sequential executions of the same SQL statement /script should be collected into a single history entry. When this feature is enabled, the **Count** field number is increased for each execution. In the same Tool Properties category you can also specify rules for when not to create a history entry.

Reusing a History Entry

When you have found the entry you're looking for, you can open it in an SQL Commander by double-clicking it or clicking the corresponding toolbar button.

You can also add the content of an entry to the current content of an SQL Editor. Select the entry in the list, drag it with the mouse key depressed to the position in the editor where you want to add it, and drop it there by releasing the mouse button.
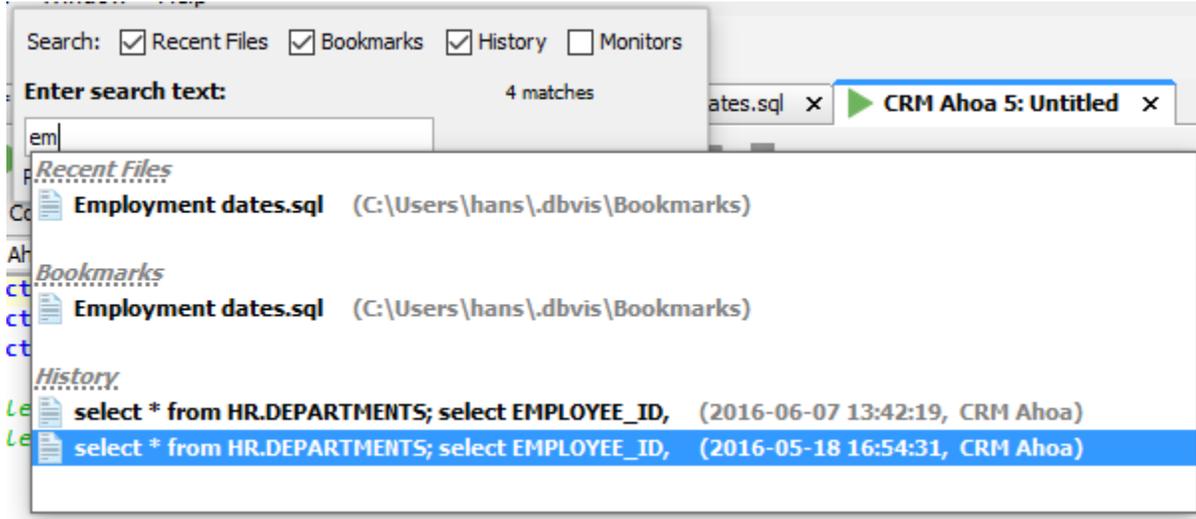
Saving a History Entry as a Bookmark or Other File

If you realize that you need easy access to a History entry in the future, you can save it is as a Bookmark or other file. Just select the entry and use the **Sav e As** operation, or just drag it to the **Bookmarks** tree and drop it.

You can also locate the file holding the history entry in the file system using the the **Open Enclosing Directory** right-mouse menu entry or toolbar button. This opens a file chooser for the directory holding the file.

Using Quick Load

An alternative to locating Bookmarks in the Scripts tab and History entries in the History window is to use the **Quick Load** feature, by default bound to the **Ctrl+Alt+O** key combination. It is also available via a main toolbar button as well as in the **File->Quick File Open** menu.

The Quick Load feature locates files with partly matching names from the categories you have selected, as you type. You can use an asterisk ("*") as a wildcard in the search string. Use the **Max** field to limit the number of matching files to display in the list.

When you see the file you're looking for, just select it and click **Enter** to load it into an SQL Commander editor. If the file is already loaded in an editor, that tab is made visible instead.