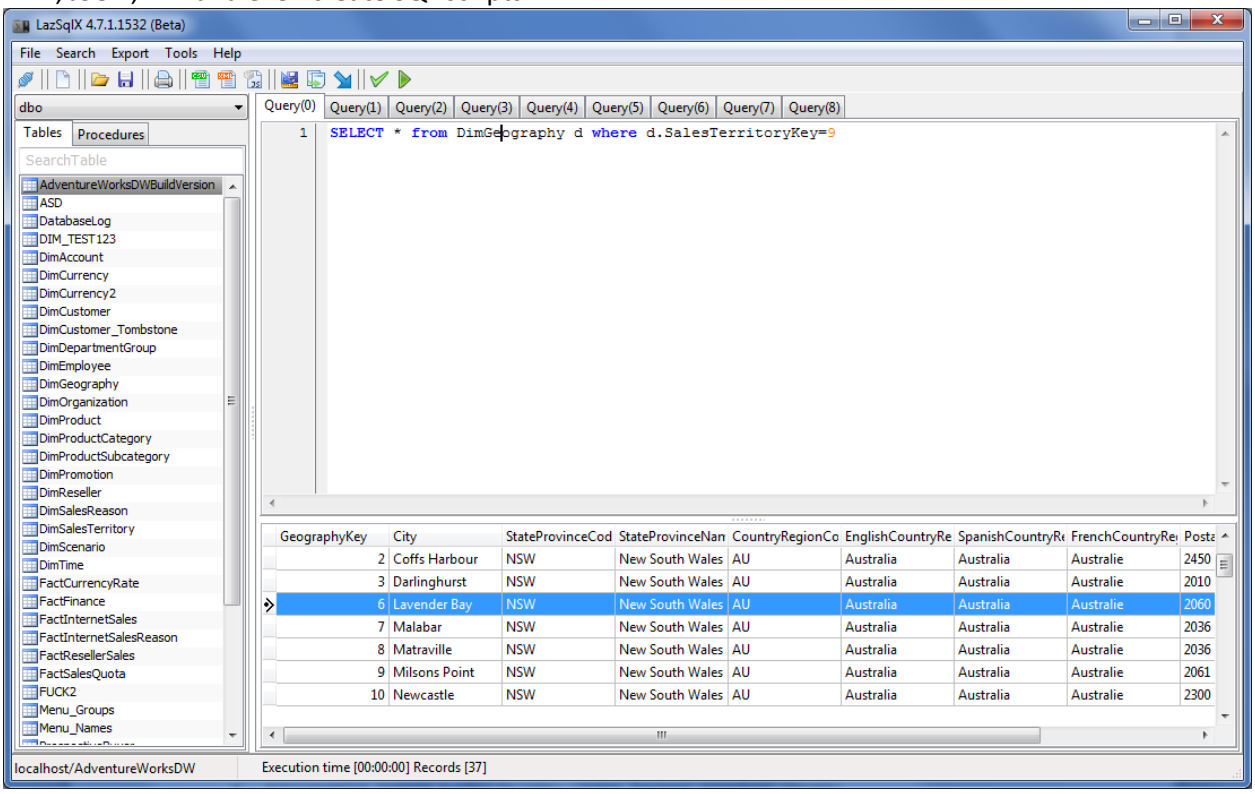
LazSQLX

Introduction

LazSqlX comes in handy to any developer who works with MsSql, Oracle, MySql, SQLite, Firebird and PostgreSQL. It provides them with an intuitive database management tool that can be used to browse the database structure, generate and execute queries and stored procedures, edit table structure and create new tables. It also allows you to edit data in a userfriendly form (Open As Form) with appropriate UI controls according to column datatypes, including a quick filter and printing ability of current data displayed on the grid. LazSqlX allows you to export data to various formats, such as CSV, XML, JSON, RTF and even create SQL scripts.



Features

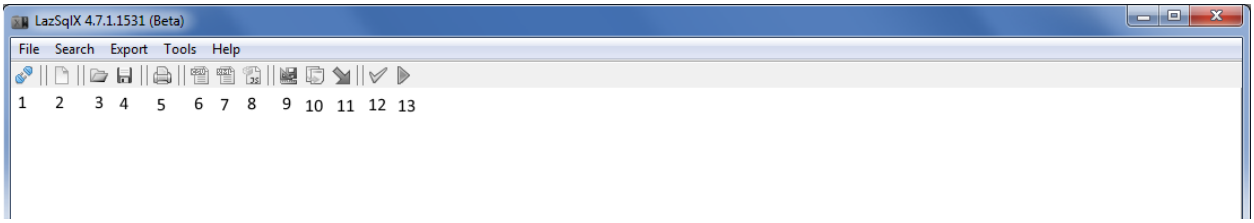
- Connect to MsSQL, Oracle, MySql, Sqlite and Firebird
- Run queries against supported databases
- Export data returned from query or opened from table to xml,csv,json,sql script
- Modify data from GUI (Open Tabledata)
- Query generations for select, selectItem,insert,update and delete

- Query Designer (select query generation)
- Run procedures via GUI Helper for supported databases
- BLOB field view/edit and filetype detection by content of the binary file (dblClick on blob field)
- Autocompletion available in editor for tables, columns, functions, procedures, variables
- Multiple editor tabs
- Portable
- For windows users, no db client installed need, db libs are shipped with the setup package
- Import Data from delimited text file to a table
- Copy result set rows as delimited text, sql insert script
- Open table as data entry form with
- Create/Edit tables
- Find/Replace text in query editor

And many more…

Main Toolbar

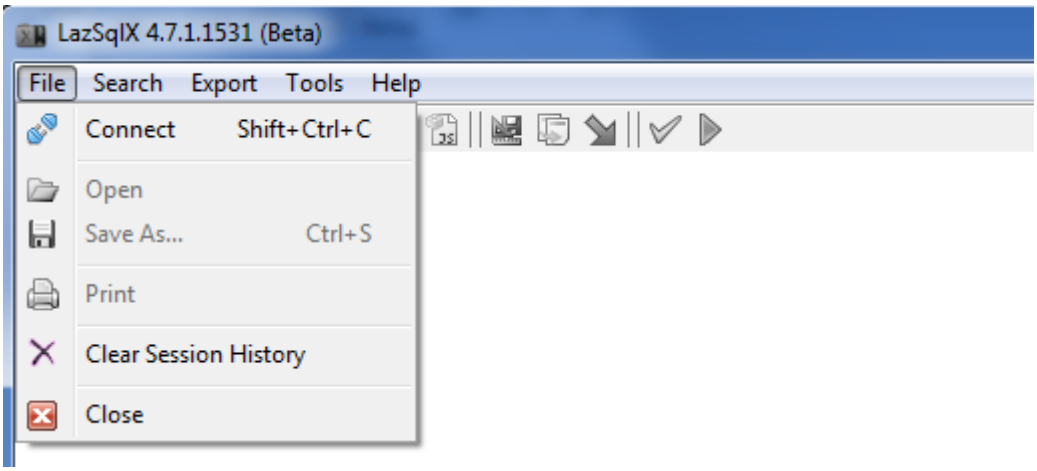
Toolbar buttons



1. Connect/Disconnect to/from a database
2. New Query Editor
3. Load a text file into the active query editor
4. Save active query editor to file
5. Print result of active query editor
6. Export result of the active query editor to XML
7. Export result of the active query editor CSV
8. Export result of the active query editor JSON
9. Query Designer tool
10. Structure Database Cloner tool
11. Data Importer tool
12. Syntax Check of the active query editor
13. Run Query from the active query editor

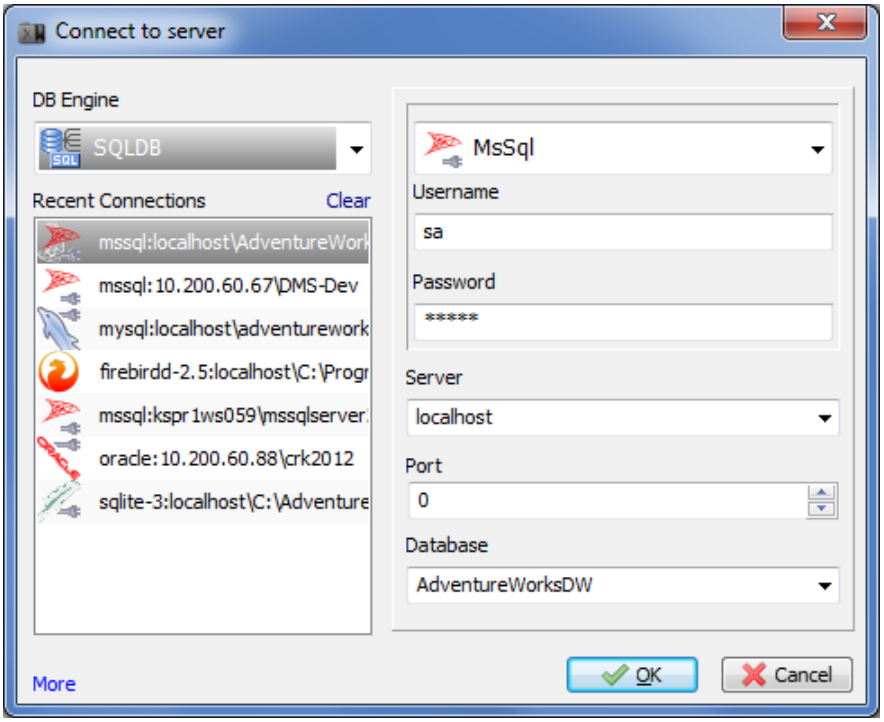
Main Menu

File



Connect .- Opens database connection dialog where you can select the database type you want to connect to (supported databases MsSQL, Oracle, MySql, Firebird, Sqlite).

DbEngine is used to determine which db connector engine should be used to connect and run queries. There are two available SQLDB and ZEOS. When selecting Database type (Right side), the most suitable DbEngine for selected database type is autoselected, but you can change it after that (Except for MySql which only runs with Zeos db components)



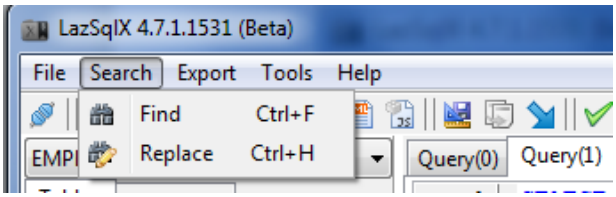
Open .- Loads a text file into active tab

Save As .- Saves active tab's text to a file

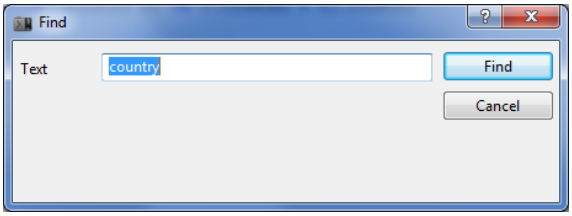
Print .- Prints the data grid of active tab's query result. You can resize and/or reorder columns in grid as you want them to appear when printing.

Clear Session History .- Clears saved content of tabs so when you open LazSqlX and connect to the same databse next time, it will be with one tab/empty query editor (otherwise it saves all tab's tabs/queries and reloads them the next time you open LazSqlx and connect to that same database)

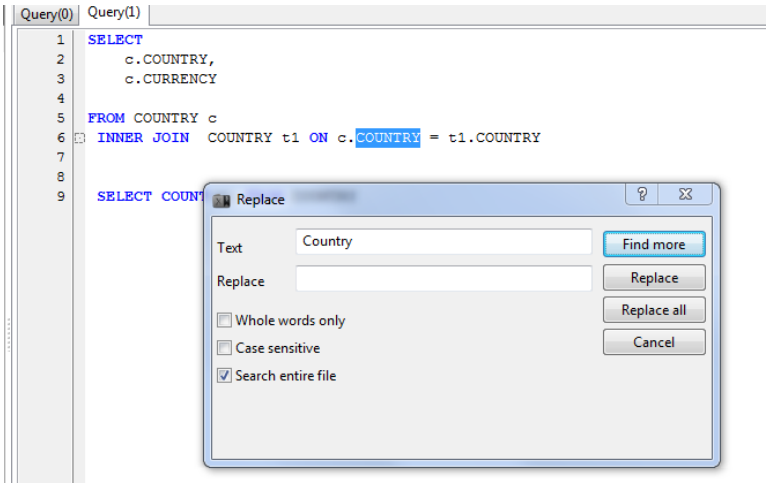
Search



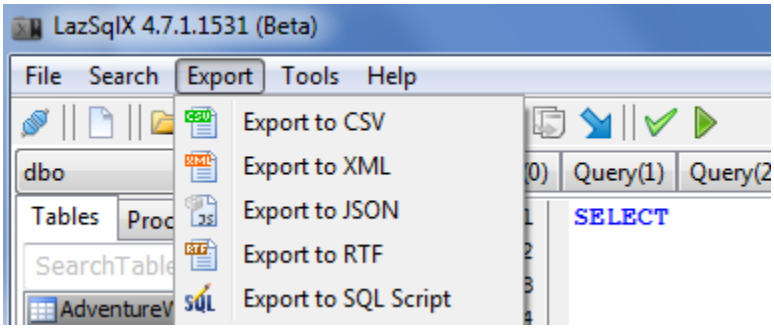
Find (keyboard shortcut Ctrl+F) .- Opens Find Dialog which find given text in active tab's query editor



Replace (keyboard shortcut CTRL+H) .- Opens Find/Replace Dialog which is finds and/or replace given texts in the ative tab's query editor



Export



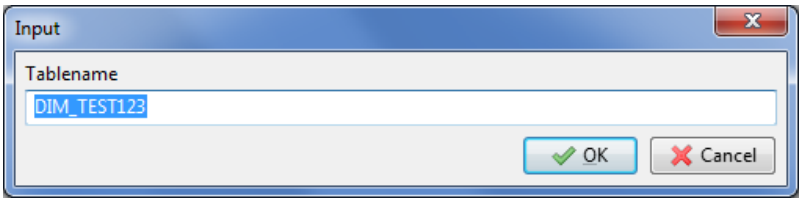
Export to CSV .- exports active tab's result to a text file with semicolon seperated format. Executes the save dialog to specify location where to save the file.

Export to XML.- exports active tab's result to a XML format then executes save dialog to specify location where to save the file

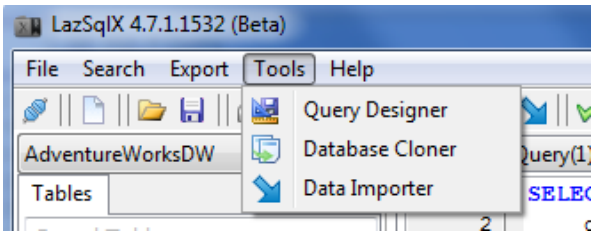
Export to JSON .- exports active tab's result to a JSON format then executes save dialog to specify location where to save the file

Export to RTF .- exports active tab's result to a RTF format then executes save dialog to specify location where to save the file

Export to SQL Script .- exports active tab's result to a SQL Insert Script file, prior to that, it asks for tablename so it knows what to write as a table name (example as in screenshot will generate "insert into [*DIM_TEST123*]({all column name}) values ({all column values}" for each row in query result)



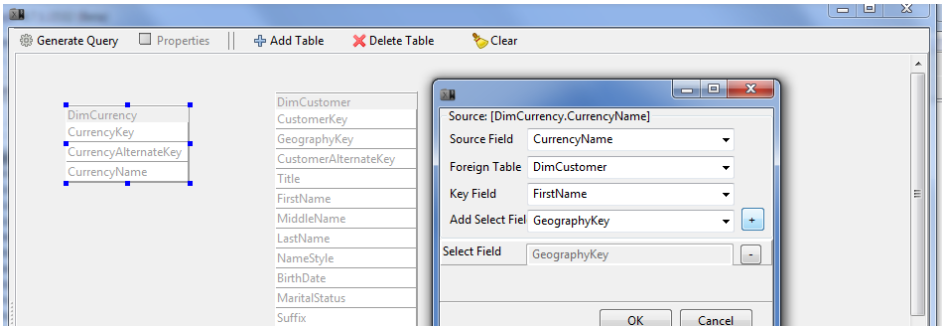
Tools



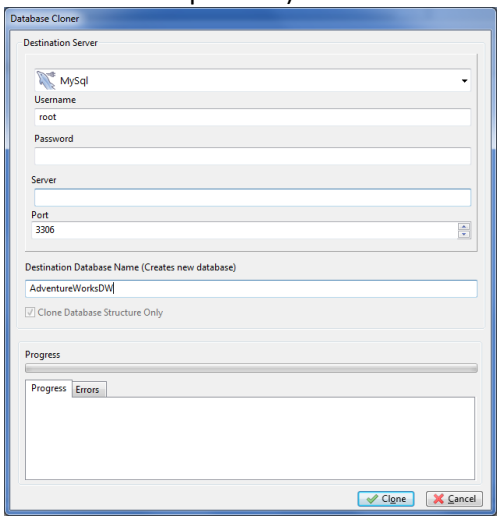
Query Designer .- opens a very simple query designer , which should enable you to make queries by drag and drop from one table column to another table column, making links (virtual relations, not modifying the actual relations in db).

When you drag from one column of a table, to another column of the other table, a dialog appears where you can set parameters to make a virtual relation, then add select fields from referenced tables.

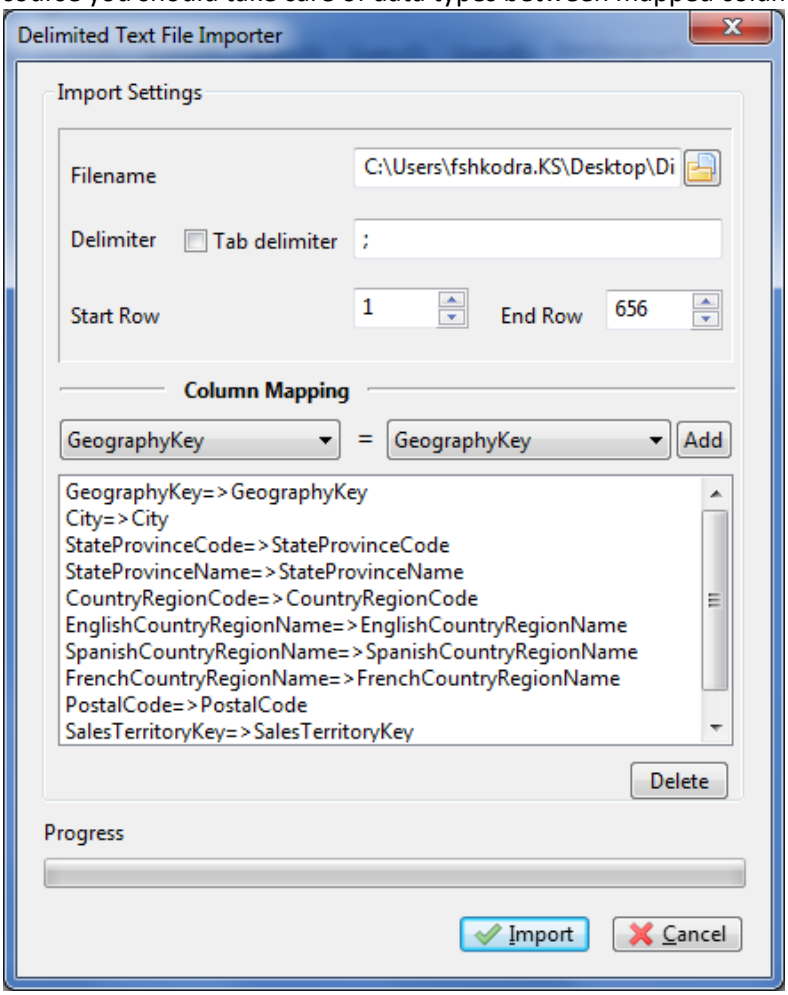
When everything set, click on the table and then press "*Generate Query*" after this click ok on the main QueryDesigner form and the generated query will be copied to active tab's query editor



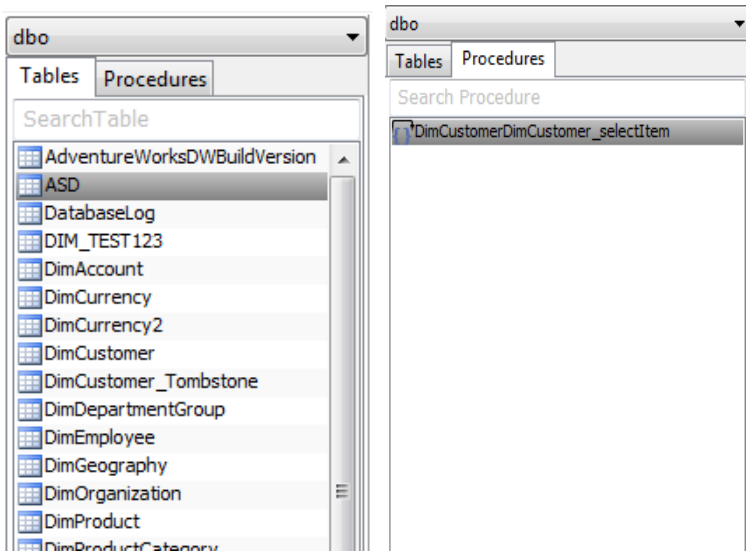
Databse Cloner .- opens an experimental database structure copier, which should make possible for you to export the structure of the current connected database to another database type (tables with column names and suitable datatypes for target database type and also table relations when possible).



Data Importer .- Opens the data importer tool to import CSV data files into the given table (great contributions to this made by Reinier Olislagers known as BigChimp in lazarus community, a great contributor to lazarus community). Automatically maps source columns with destination columns but you can also make your own mapping and then import data to destination table. Of course you should take care of data types between mapped columns



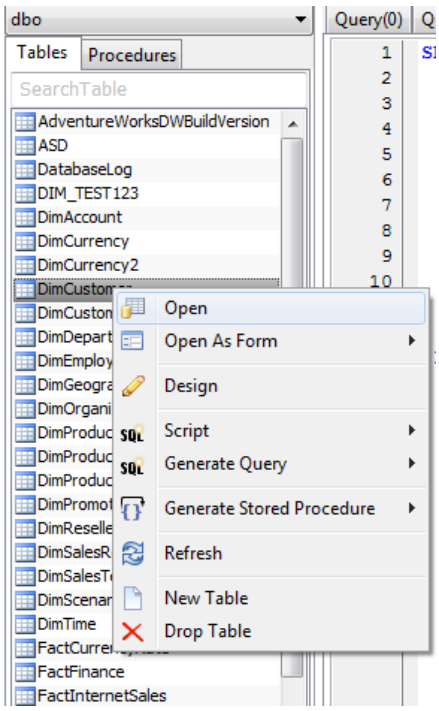
Left Pane – Tables and Stored Procedures



Tables

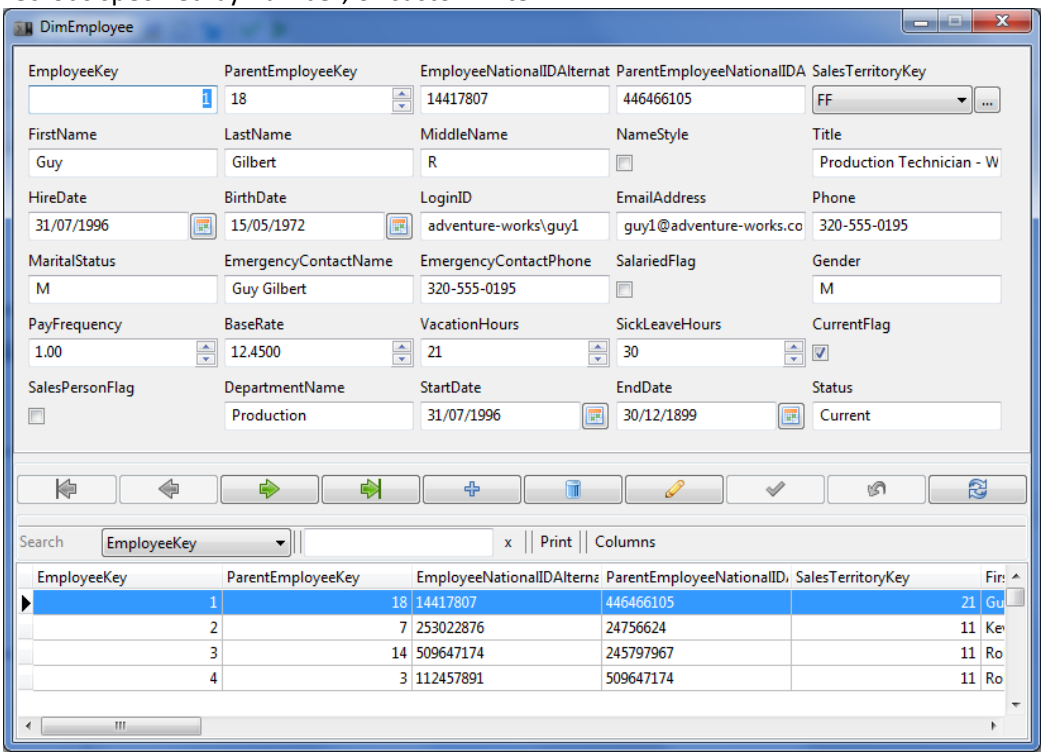
The combo box is filled with schema names (if supported by current database type, else it's just the db filename).
Tables's tab is filled with table names of selected schema (or just all tables if database type supports no schema)

Tables' context menu (PopUp)



Open.- Opens selected table as data table in a new tab, with possibility to edit data

Open As Form .- Opens selected table as data entry form, with possibility to edit data. It generates a gui form with different ui controls according to column types. Prior to opening the Form, you have to choose wether you want all data rertrieved from a database, limited count of recrods specified by number, or custom filter.

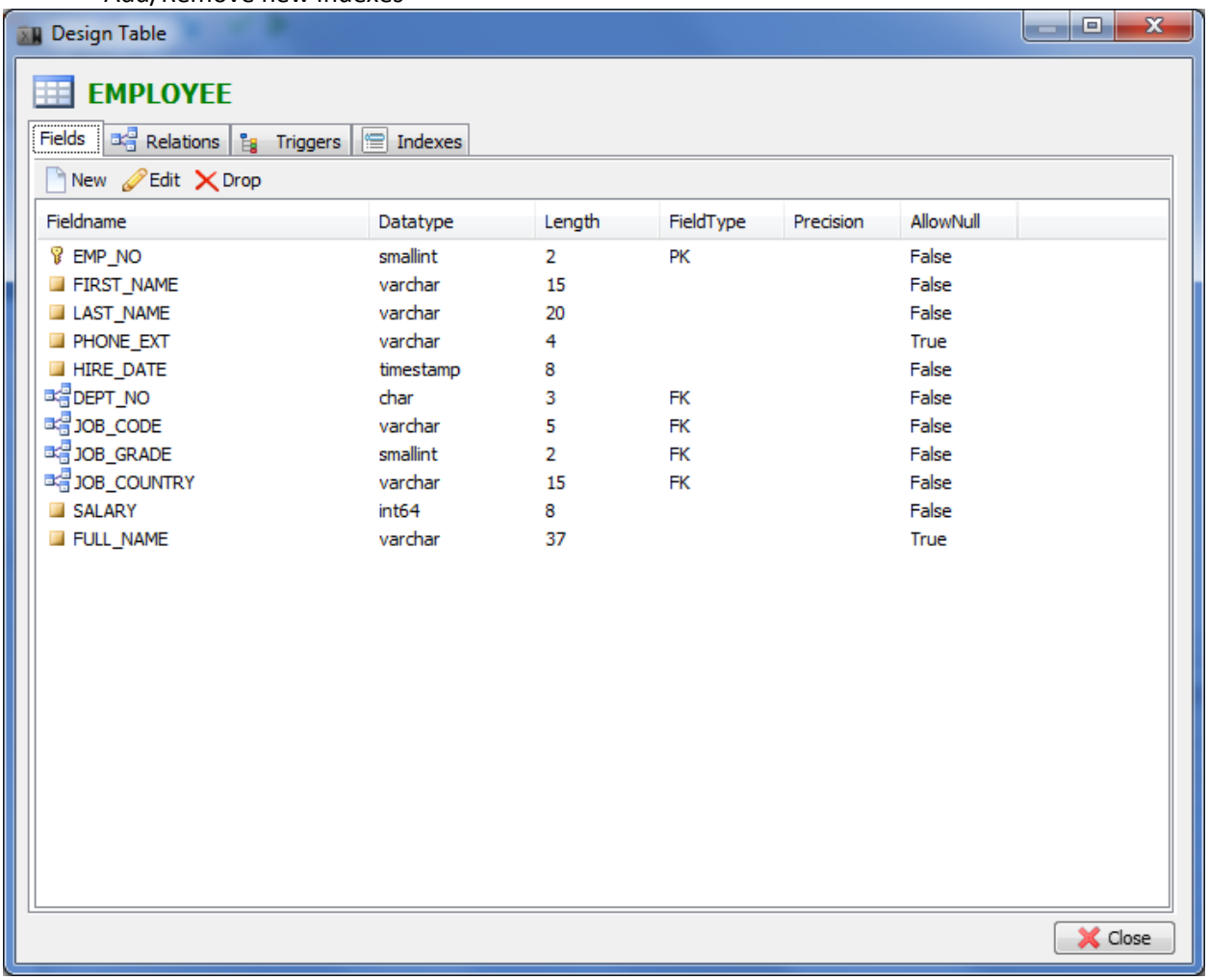


So the form in the screenshot is generated from DimEmployee table (from MsSql sample AdventureWorks).

- Numeric fields: Numeric updown control
- Reference fields: ComboBox displaying the first text field of the foreign table
- Date fields: datecontrol with a calendar
- Boolean fields (bit [1]): CheckBoxes.
- BLOB fields(binary): a control with the ability to add new from file or save current binary as file

Design.- Opens the table editor which has the following features

-Add new/edit/drop columns
-Add/Remove new relations
-View tiggers
-Add/Remove new indexes



Script.- Generates a simple “Create table” script for selected table

Generate Query .- Generates select and select item (with joins to reference tables), insert,update,delete queries for selected table.

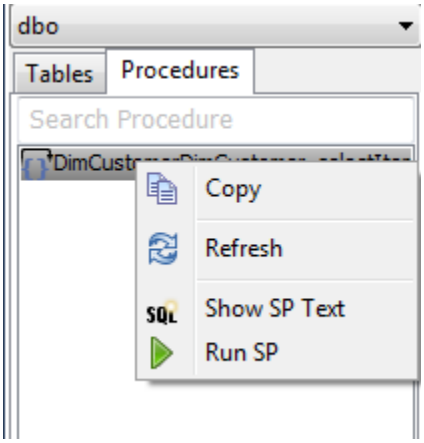
Generate Stored Procedure .- Generates “Create Procedure” select and select item (with joins to reference tables), insert,update,delete queries for selected table.

Refresh .- Refresh (refills) tables from server

New Table .- Creates new table, opens a dialog that asks you for tablename then takes you to table designer

Drop Table .- Drops selected table from the database

Procedures' context menu (popup)

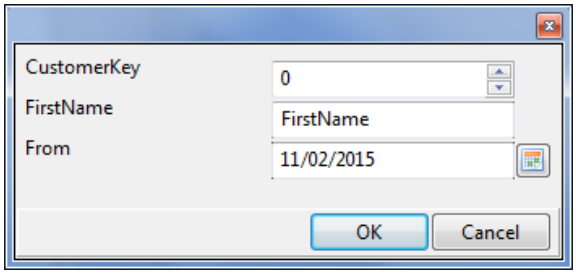


Copy .- Copies selected procedure to clipboard as a script for execution with parameters set (example: exec DimCustomerDimCustomer_selectItem @CustomerKey = 0, @FirstName = 'John', @From = '')

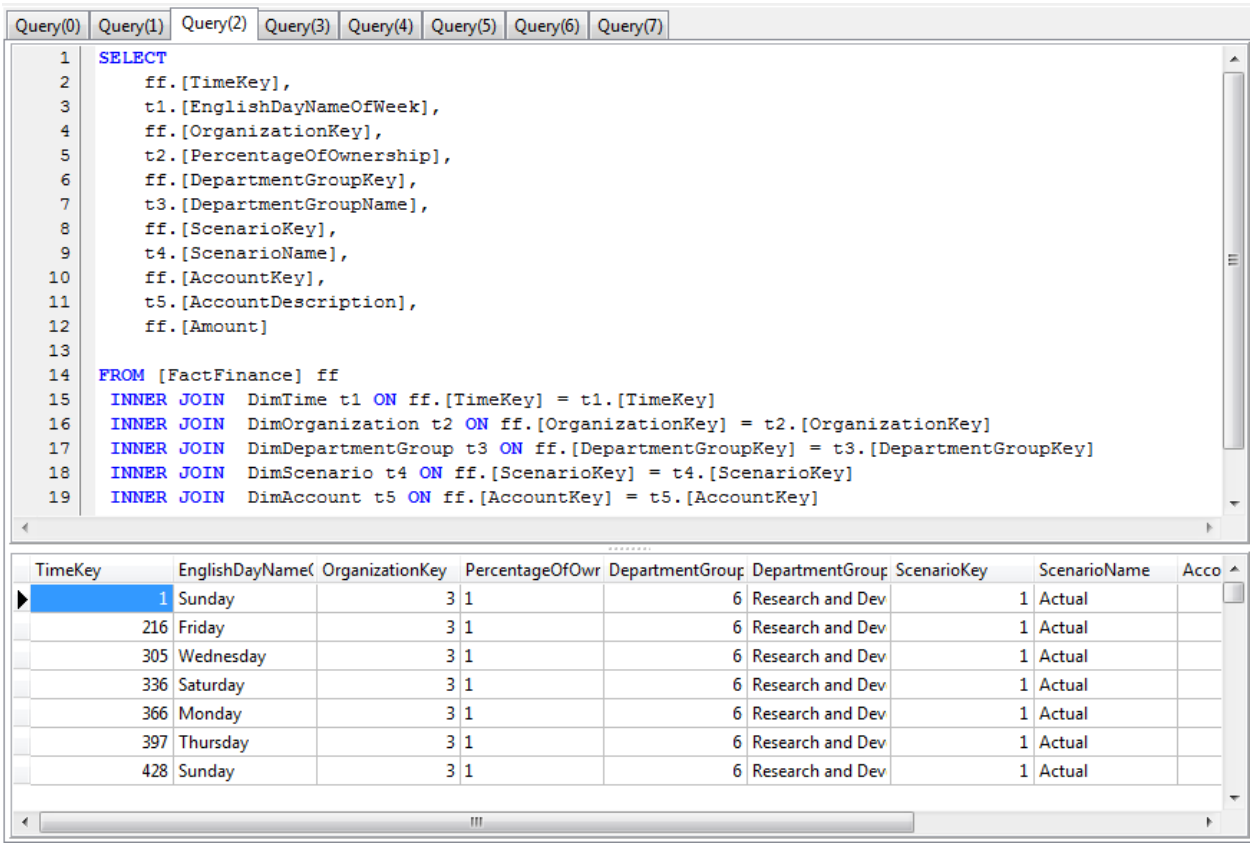
Refresh.- Refreshes (refills) stored procedures from database server

Show SP Text .- Shows stored procedure's text content

Run SP.- Opens a dialog with possibility to give parameters (generated according to stored procedure's parameters), before executing the stored procedure according to the given parameters

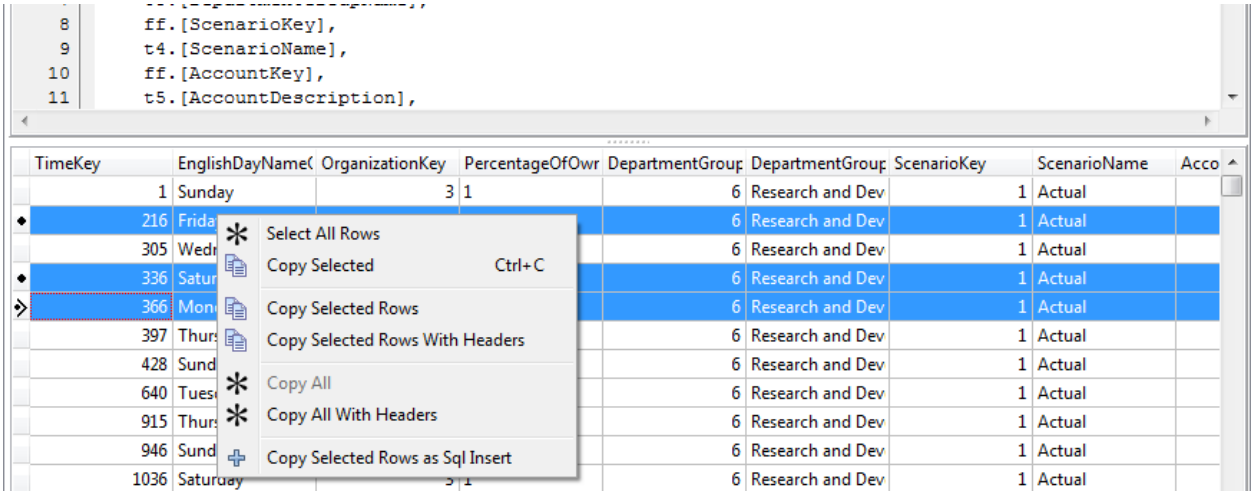


Query Editor



Query editor supports multiple tabs and has some basic autocomplete feature invoked with CTRL+Space. You can write you sql query and execute it (shortcut F9) to see the results. The results are displayd in a grid on the bottom.

Query Editor Grid



The Query Editor Grid supports custom multi-select which can be achieved through holding CTRL and selecting wanted rows with the mouse. It has the following features that can be accessed via its context menu (popup via right-click):

Select All Rows .- Select all rows in the grid

Copy Selected.- Copies selected cell's value to clipboard

Copy Selected Rows.- Copies selected rows' values to clipboard as tab delimited text

Copy Selected Rows With Headers .- Copies selected rows' values (including their headers-column names) to clipboard as tab delimited text

Copy All .- Copies all rows' values to clipboard as tab delimited text

Copy All With Headers .- Copies all rows' values (including their headers-column names) to clipboard as tab delimited text

Copy Selected Rows as Sql Inser .- Copies selected rows to clipboard as Sql Insert script