SQLite Database Converter

User's guide

Copyright © 2002-2024 SQL Maestro Group

Table of Contents

Foreword 0

I Welcome to SQLite Database Converter! 1

1 System Requirements 2
2 Installation 3
3 How can I purchase SQLite Database Converter? 4
4 License Agreement 5
5 About SQL Maestro Group 7
6 What's new 10

II Getting Started 11

1 Connecting to the source data 12
MySQL 12
PostgreSQL 15
MS SQL Server 18
Firebird 18
Oracle 21
SQLite 21
2 Connecting to the target database 24
3 Projects 26
4 Command line options 27

III Source objects 28

1 Data type mappings 29
2 Naming conventions 30

IV Target objects 31

V Data import 32

VI Converting options 34

VII Options 36

1 Application 37
Common 37
Confirmations 38
2 Editors & Viewers 40
General 40
Display 41
SQL highlight 42
XML highlight 43
PHP highlight 44

© 2024 SQL Maestro Group

Code Insight 45
Code Folding 46
3 Appearance 48
Bars and menus 48
Trees and lists 49
Edit controls 50
Check boxes 51
Buttons 52
Page controls 53
Group boxes 54
Splitters 55
Index 57

© 2024 SQL Maestro Group

1 Welcome to SQLite Database Converter!

SQLite Database Converter is a powerful tool for migrating schema and data from any ADO-compatible database to SQLite . It allows you to transfer tables, indexes, foreign key constraints. With this tool you can convert source objects directly to the target database or generate SQL scripts with objects definitions and INSERT statements to be used later.

Key features:

- Copying tables, indexes, and foreign keys
- Native support for most popular database servers
- Support for all other data sources accessible via ODBC / OLE DB
- Bulk data loading
- Flexible and customizable data type mappings
- Smart database integrity checker
- Command-line interface to automate the conversion

© 2024 SQL Maestro Group

1.1 System Requirements

Client environment

- Pentium PC or higher;
- Windows NT4/2000/XP/Vista/Windows 7/Windows 8/Windows 10/Windows 11;
- 512 MB RAM (1 GB recommended);
- 25 MB of free hard disk space;
- SVGA-compatible video adapter.

Server environment

- SQLite 2.8/3.x.

© 2024 SQL Maestro Group

1.2 Installation

To install **SQLite Database Converter** on your PC:

- download the SQLite Database Converter distribution package from the download page at our site;
- run setup.exe from the local folder and follow the instructions of the installation wizard;
- find the SQLite Database Converter shortcut in the corresponding program group of the Windows Start menu after the installation is completed.

© 2024 SQL Maestro Group

1.3 How can I purchase SQLite Database Converter?

Thank you for your interest in purchasing **SQLite Database Converter**!

You can select licensing options and register SQLite Database Converter at its on-line order page. It is possible to purchase on-line, by fax, mail, toll-free phone call, or place a purchase order. We send the software activation key by email within 24 hours after completion of the order process. If you have not received the activation key within this period, please contact our sales department.

All our products and bundles are shipped with 12 months of free upgrades (minor and major ones) or with 36 months of free upgrades for a quite small additional fee. After this period you may renew your license for the next 12(36) months with a 50% discount.

SQLite Database Converter has a free 30-day trial. Upon purchasing the product you confirm that you have tested it and you are completely satisfied with its current version.

To obtain technical support, please visit the appropriate section on our website or contact us by email to support@sqlmaestro.com.

© 2024 SQL Maestro Group

1.4 License Agreement

Notice to users: carefully read the following legal agreement. The use of the software provided with this agreement (the "SOFTWARE") constitutes your acceptance of these terms. If you do not agree to the terms of this agreement, do not install and/or use this software. The use of this software is conditioned upon the user's compliance with the terms of this agreement.

- **License grant**. SQL Maestro Group grants you a license to use one copy of the version of this SOFTWARE on any single hardware product for as many licenses as you purchase. "You" means a company, an entity or an individual. "Use" means storing, loading, installing, executing or displaying the SOFTWARE. You may not modify the SOFTWARE or disable any licensing or control features of the SOFTWARE except as an intended part of the SOFTWARE's programming features. This license is not transferable to any other company, entity or individual. You may not publish any registration information (serial numbers, registration keys, etc.) or pass it to any other company, entity or individual.

- **Ownership**. The SOFTWARE is owned and copyrighted by SQL Maestro Group. Your license confers no title or ownership of the SOFTWARE and should not be construed as a sale of any rights for the SOFTWARE.

- **Copyright**. The SOFTWARE is protected by the United States copyright law and international treaty provisions. You acknowledge that no title to the intellectual property in the SOFTWARE is transferred to you. You further acknowledge that title and full ownership rights to the SOFTWARE will remain the exclusive property of SQL Maestro Group and you will not acquire any rights to the SOFTWARE except as expressly set forth in this license. You agree that any copies of the SOFTWARE will contain the same proprietary notices which appear on and in the SOFTWARE.

- **License and distribution**. An unregistered copy of the SOFTWARE ("UNREGISTERED SOFTWARE") may be used for evaluation purposes. The UNREGISTERED SOFTWARE may be freely copied and distributed to other users for their evaluation. If you offer this UNREGISTERED SOFTWARE installation package for download, then you agree to:

- replace existing version of the UNREGISTERED SOFTWARE installation package with the new package immediately after a new version of the SOFTWARE is released by SQL Maestro Group, or
- delete an obsolete version of the UNREGISTERED SOFTWARE installation package immediately upon written email notice by SQL Maestro Group.

A registered copy of the SOFTWARE ("REGISTERED SOFTWARE") allows you to use the SOFTWARE only on a single computer and only by a single user at a time. If you wish to use the SOFTWARE for more than one user, you will need a separate license for each individual user. You are allowed to make one copy of the REGISTERED SOFTWARE for back-up purposes.

- **Reverse engineering**. You affirm that you will not attempt to reverse compile, modify, translate, or disassemble the SOFTWARE in whole or in part.

- **Unauthorized use**. You may not use, copy, rent, lease, sell, modify, decompile, disassemble, otherwise reverse engineer, or transfer the SOFTWARE except as provided in this agreement. Any such unauthorized use shall result in immediate and

© 2024 SQL Maestro Group

automatic termination of this license.

- **No other warranties**. SQL Maestro Group does not warrant that the SOFTWARE is error-free. SQL Maestro Group disclaims all other warranties with respect to the SOFTWARE, either express or implied, including but not limited to implied warranties of merchantability, fitness for a particular purpose and noninfringement of third party rights. Some jurisdictions do not allow the exclusion of implied warranties or limitations on how long an implied warranty may last, or the exclusion or limitation of incidental or consequential damages, so the above given limitations or exclusions may not apply to you. This warranty gives you specific legal rights and you may also have other rights which vary from jurisdiction to jurisdiction.

- **Limited warranty**. This SOFTWARE is provided on an "AS IS" basis. SQL Maestro Group disclaims all warranties relating to this SOFTWARE, whether expressed or implied, including but not limited to any implied warranties of merchantability or fitness for a particular purpose. Neither SQL Maestro Group nor anyone else who has been involved in the creation, production, or delivery of this SOFTWARE shall be liable for any indirect, consequential, or incidental damages arising out of the use or inability to use such SOFTWARE, even if SQL Maestro Group has been advised of the possibility of such damages or claims. The person using the SOFTWARE bears all risk as to the quality and performance of the SOFTWARE.

Some jurisdictions do not allow limitation or exclusion of incidental or consequential damages, so the above given limitations or exclusion may not apply to you to the extent that liability is by law incapable of exclusion or restriction.

- **Severability**. In the event of invalidity of any provision of this license, the parties agree that such invalidity shall not affect the validity of the remaining portions of this license.

- **No liability for consequential damages**. In no event shall SQL Maestro Group or its suppliers be liable to you for any consequential, special, incidental or indirect damages of any kind arising out of the delivery, performance or use of the SOFTWARE, even if SQL Maestro Group has been advised of the possibility of such damages. In no event will SQL Maestro Group's liability for any claim, whether in contract, tort or any other theory of liability, exceed the license fee paid by you, if any.

- **Entire agreement**. This is the entire agreement between you and SQL Maestro Group which supersedes any prior agreement or understanding, whether written or oral, relating to the subject matter of this license.

- **Reserved rights**. All rights not expressly granted here are reserved to SQL Maestro Group.

© 2024 SQL Maestro Group

1.5 About SQL Maestro Group

SQL Maestro Group is a privately-held company producing high-quality software for database administrators and developers. The united team of eminently qualified developers is pleased to create new software products for commercial, academic and government customers worldwide. We do our best to design and develop products that remove complexity, improve productivity, compress time frames, and increase database performance and availability. We are glad to realize that our products take usual chores upon themselves, so that our customers could have more time left for their creative work.

The company was founded in 2002 as an essential partner for every business that is trying to harness the explosive growth in corporate data. SQL Maestro Group employs an international team concentrating their efforts on cutting-edge DBA tools development.

The slogan of our company is **The Shortest Path to SQL**. It is aimed to denote that we set to create easy-to-use products meant for those who appreciate comfort, friendly program interface and support when working with SQL servers.

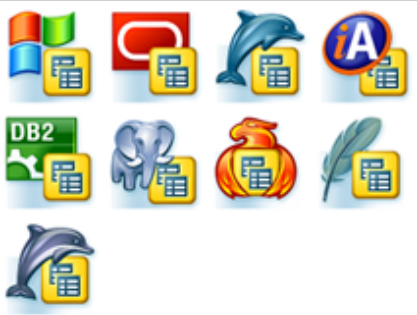
- We are pleased to facilitate your job.
- We aim at being of considerable assistance to our clients.
- We feel contented doing our beloved work.

At present, our company offers a series of Windows GUI admin tools for SQL management, control and development of the following servers: **MySQL, Microsoft SQL Server, PostgreSQL, Oracle, SQL Anywhere, DB2, SQLite, Firebird,** and **MaxDB**. We also produce universal tools to be used for administering any database engine accessible via ODBC driver or OLE DB provider. Such products may be the clear-cut decision for those who constantly work with several database servers.

SQL Maestro is the premier Windows GUI admin tool for database development, management, and control.

It provides you with the ability to perform all the necessary database operations such as creating, editing, copying, extracting and dropping database objects; moreover, you can build queries visually, execute queries and SQL scripts, view and edit data including BLOBs, represent data as diagrams, export and import data to/from most popular file formats, manage users and their privileges (if possible), and use a lot of other tools designed for making your work with your server comfortable and efficient.

© 2024 SQL Maestro Group

SQL PHP Generator is a powerful tool for creating database-driven web applications visually. It allows you to generate high-quality PHP scripts for working with tables, views and queries through the web. You needn't have any programming background to use it.	
SQL Data Wizard is a high-capacity Windows GUI utility for managing your data. It provides you with a number of easy-to-use wizards for performing the required data manipulation easily and quickly. The tool allows you to export data from SQLite tables and queries to most popular formats, import data into the tables, generate SQL dump of selected tables, and export/ import BLOB fields from/to files.	
SQL Code Factory is a premier GUI tool aimed at the SQL queries and scripts development. It allows you to manage SQL queries and scripts using such useful features as code folding, code completion and syntax highlighting, build query visually, execute several queries at a time, execute scripts from files, view and edit result data with filtering, sorting and grouping abilities, export data to as many as 14 file formats including Excel, RTF and HTML, import data from Excel, CSV, XML and text files, view and edit BLOBs in various way, build diagrams based on Oracle data, and much more.	
Database Converter is a user friendly tool to migrate any local or remote ADO-compatible database to SQLite . Such tools transfer database schema and data and are equipped with native support for the most popular database servers.	
Data Sync is a powerful and easy-to-use tool for database contents comparison and synchronization. Such tools can be useful for database administrators, developers and testers that need a quick, easy and reliable way to compare and synchronize their data.	

© 2024 SQL Maestro Group

The software products are constantly optimized for the latest server versions support.

You can use the following contact information if necessary:

Our web-site www.sqlmaestro.com

Postal address: **SQL Maestro Group**
140 Broadway, Suite 706
New York City, New York 10005
United States

Thank you for your interest to our company!

© 2024 SQL Maestro Group

1.6 What's new

Please find out the latest SQLite Database Converter news at http://www.sqlmaestro.com/products/sqlite/converter/news/

© 2024 SQL Maestro Group

2 Getting Started

To convert a database with its structure and data,

- Set connection options of source[12] and target[24] databases;
- Select objects[28] to convert;
- Adjust options[31] of the objects to create;
- Select tables[32] for data import;
- Specify other transfer options[34].

SQLite Database Converter allows you to save and restore all the options set during a session. You need not to specify all options each time you work with the application anew; instead you can load all settings from a project and change them if necessary. When working with a project, all the session parameters are loaded from a project file and may be edited if necessary. To run a wizard with a project, follow More... > Load Project on the first wizard step and enter the name of the project file, recently used projects are also available from this popup menu. Find out more about working with projects[26].

© 2024 SQL Maestro Group

2.1 Connecting to the source data

SQLite Database Converter supports native connection to MySQL 12, PostgreSQL 15, Firebird 18, MS SQL Server 18, SQLite 21, and Oracle 21. To transfer schema and data from data sources accessible via ODBC driver or OLE DB provider such as MS Access, Text files, MS Excel, etc. specify the corresponding connection string.

2.1.1 MySQL

Specify the following credentials to connect to MySQL.

- **Direct connection**
  It is the most natural and the most preferable connection mode. Use it each time it is possible. Most of hosting companies allow direct connections to databases. However in most cases you have to go to your control panel and add your home/office computer IP address or domain name to the Access List - list of IP addresses allowed accessing from outside. More information.

- **SSH tunnel connection**
  If your SQLite server does not allow direct connections from your remote workstations, you can establish connection to an allowed intermediate SSH server and forward all MySQL commands through the Secure SHell (SSH) tunnel.

- **HTTP tunnel connection**
  HTTP tunneling is a technique used in conditions of restricted network connectivity including firewalled networks, networks behind proxy servers, and NATs. It is the slowest way and is recommended to use if the others are impossible.

Irrespectively of a connection mode you should specify common credentials as follows:

Host
The host name of the MySQL server.

Port number
The TCP/IP port to use (default MySQL port is 3306).

User name
The username used to connect to MySQL.

Password
The password for the user account on server.

Allow old style password
Turn this option ON only if your MySQL server still uses 16 bytes long password hashes. More information.

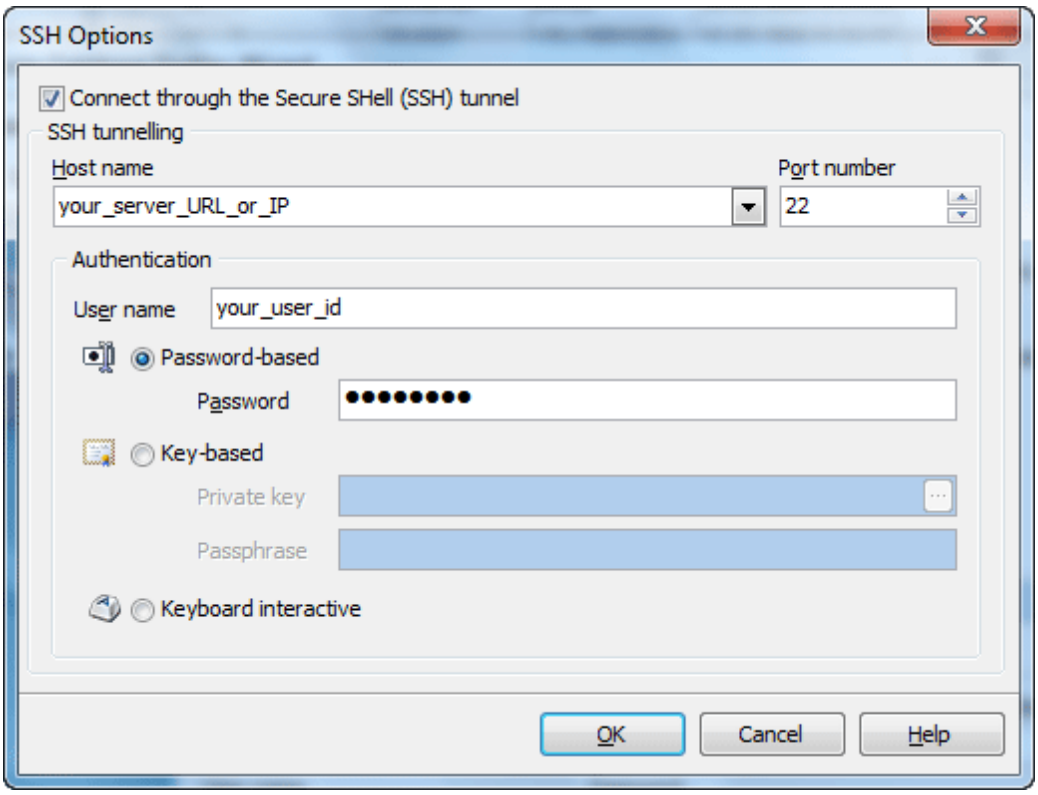
More about SSH tunnel connection

To establish connection to intermediate SSH server and forward all SQLite commands through the secure tunnel, you need to:

1. Check I can connect to the server directly or via SSH tunneling.

© 2024 SQL Maestro Group

2. Follow the Configure SSH options link to open the SSH Options window.



3. Check Connect through the Secure Shell (SSH) tunnel and complete the following fields:

Host name
Specify the host name or IP of your site. Note, that SQLite host name always should be set relatively to the SSH server. For example, if both of SQLite and SSH servers are located on the same computer, you should specify localhost as Host name instead of server's external host name or IP address.

Port number
Enter the port number for the SSH server.

4. Enter valid User name for the remote server and select the Authentication method and set corresponding credentials.

Password-based
Set the password corresponding to the specified user.

Key-based
Specify the path to the Private key file with the corresponding Passphrase to log in to the remote server. SQLite Database Converter accepts keys in **ssh.com** or **OpenSSH** formats. To convert a private key from PuTTY's format to one of the formats supported by our software, use the PuTTYgen utility that

© 2024 SQL Maestro Group

can be freely downloaded from the PuTTY website.

Keyboard interactive
Keyboard authentication is the advanced form of password authentication, aimed specifically at the human operator as a client. During keyboard authentication zero or more prompts (questions) is presented to the user. The user should give the answer to each prompt (question). The number and contents of the questions are virtually not limited, so certain types of automated logins are also possible.

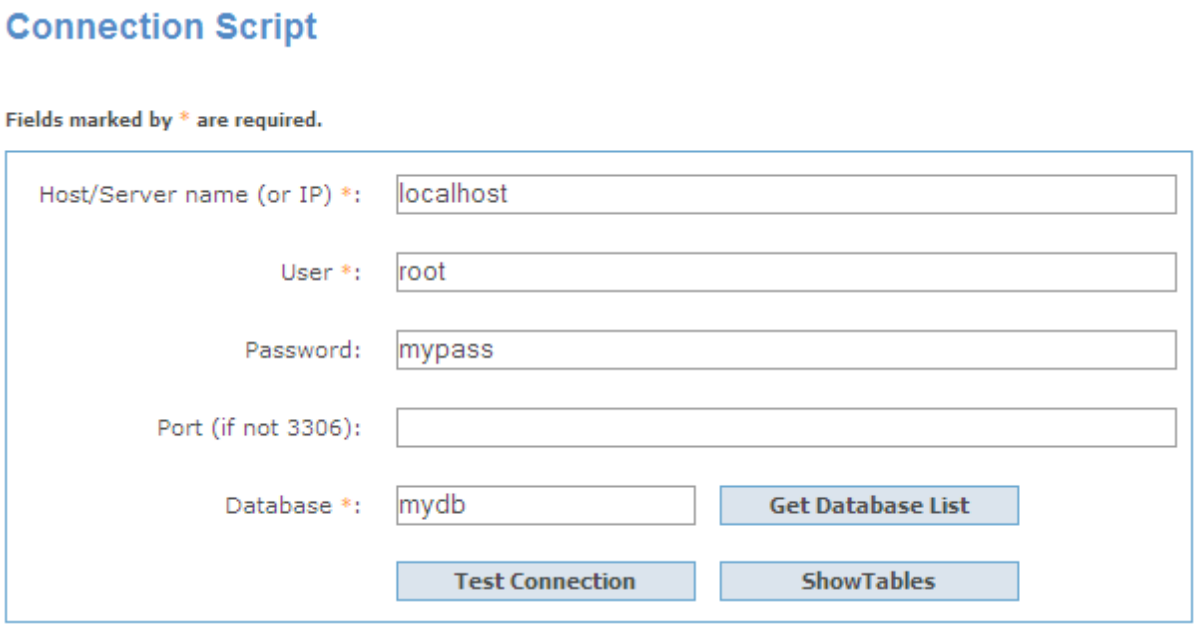
More about connection via HTTP tunnel

To connect to a remote server using an HTTP tunnel, you need to:

1. Upload the connection PHP script to your website. The installation folder, usually *C:\Program Files\SQL Maestro Group\SQLite Database Converter,* contains two scripts: *mysqli_tunnel.php* and *mysql_tunnel.php*. We would recommend you to **use the *mysqli_tunnel.php* script always if possible** as it operates through the MySQLi PHP extension (available since PHP 5) while *mysql_tunnel.php* uses the original MySQL PHP API that is deprecated as of PHP 5.5.

2. Select the I have to use HTTP tunneling radio button.

3. Enter the connection PHP script URL, e.g. *www.yoursite.com/files/sqlite_tunnel.php*. You can test the connection before the profile is created. Just use Test script using default browser to open connection script in your browser, enter all the required connection parameters and click the Test connection button.



4. In case using of a proxy server use Configure tunnelling options to open the HTTP tunnelling options window and specify your proxy server connection parameters and HTTP authentication.

© 2024 SQL Maestro Group

Note: You are actually connecting to your database through the PHP script on the server, so in most cases the host/server name is "localhost" unless the target database server is not installed on the same computer as the Web server.

2.1.2 PostgreSQL

Specify the following credentials to connect to PostgreSQL.

- **Direct connection**
  It is the most natural and the most preferable connection mode. Use it each time it is possible.

- **SSH tunnel connection**
  If your PostgreSQL server does not allow direct connections from your remote workstations, you can establish connection to an allowed intermediate SSH server and forward all PostgreSQL commands through the Secure SHell (SSH) tunnel.

- **HTTP tunnel connection**
  HTTP tunneling is a technique used in conditions of restricted network connectivity including firewalled networks, networks behind proxy servers, and NATs. It is the slowest way and is recommended to use if the others are impossible.

Irrespectively of a connection mode you should specify common credentials as follows:

Host
The host name of the PostgreSQL server.

Port number
The TCP/IP port to use. Default PostgreSQL port is 5432.

User name
The username used to connect to PostgreSQL.

Password
The password for the user account on server.

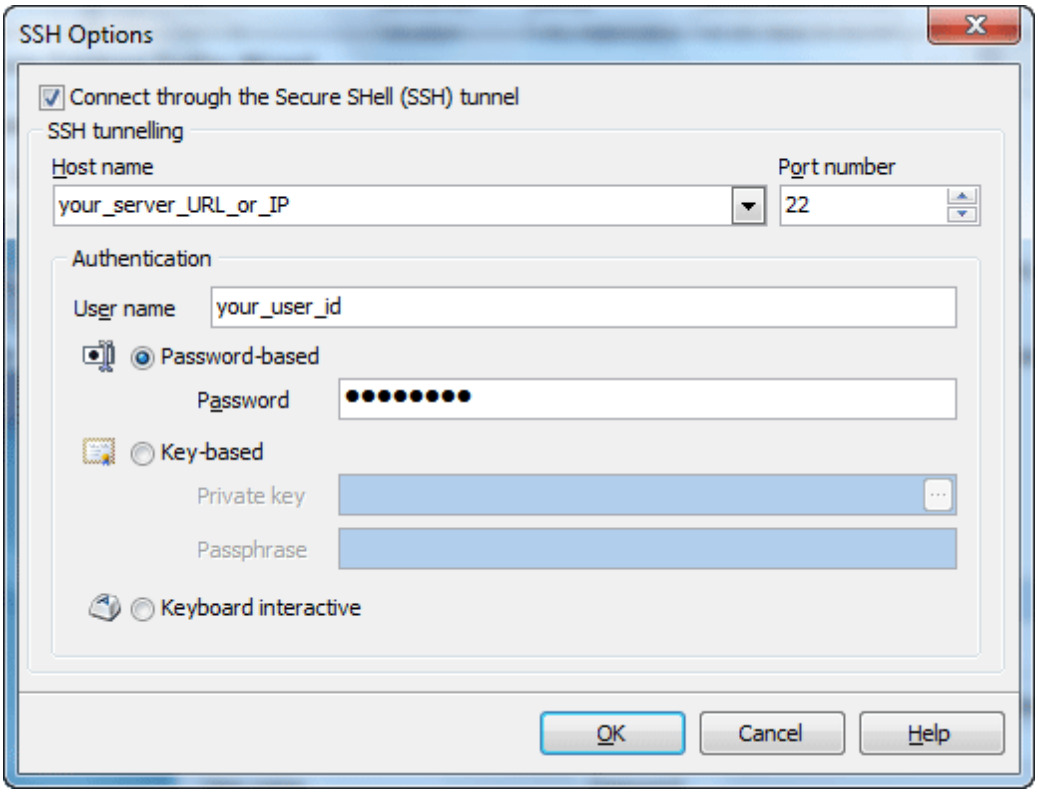
More about SSH tunnel connection

To establish connection to intermediate SSH server and forward all SQLite commands through the secure tunnel, you need to:

1. Check I can connect to the server directly or via SSH tunneling.

2. Follow the Configure SSH options link to open the SSH Options window.

© 2024 SQL Maestro Group



3. Check Connect through the Secure Shell (SSH) tunnel and complete the following fields:

Host name
Specify the host name or IP of your site. Note, that SQLite host name always should be set relatively to the SSH server. For example, if both of SQLite and SSH servers are located on the same computer, you should specify localhost as Host name instead of server's external host name or IP address.

Port number
Enter the port number for the SSH server.

4. Enter valid User name for the remote server and select the Authentication method and set corresponding credentials.

Password-based
Set the password corresponding to the specified user.

Key-based
Specify the path to the Private key file with the corresponding Passphrase to log in to the remote server. SQLite Database Converter accepts keys in **ssh.com** or **OpenSSH** formats. To convert a private key from PuTTY's format to one of the formats supported by our software, use the PuTTYgen utility that can be freely downloaded from the PuTTY website.

Keyboard interactive

© 2024 SQL Maestro Group

Keyboard authentication is the advanced form of password authentication, aimed specifically at the human operator as a client. During keyboard authentication zero or more prompts (questions) is presented to the user. The user should give the answer to each prompt (question). The number and contents of the questions are virtually not limited, so certain types of automated logins are also possible.

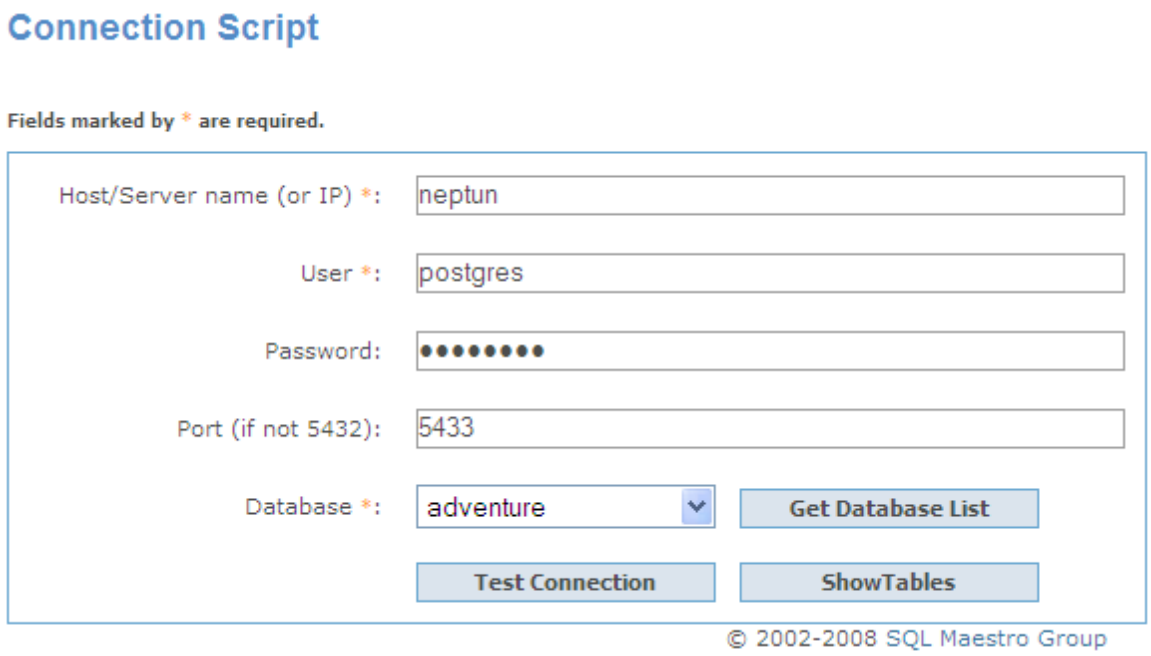
More about connection via HTTP tunnel

To connect to a remote server using an HTTP tunnel, you need to:

1. Upload the connection PHP script to your website. The script is named *sqlite_tunnel.php* and can be found under the installation folder, usually *C:\Program Files\SQL Maestro Group\SQLite Database Converter*.

2. Select the I have to use HTTP tunneling radio button.

3. Enter the connection PHP script URL, e.g. *www.yoursite.com/files/sqlite_tunnel.php*. You can test the connection before the profile is created. Just use Test script using default browser to open connection script in your browser, enter all the required connection parameters and click the Test connection button.



4. In case using of a proxy server use Configure tunnelling options to open the HTTP tunnelling options window and specify your proxy server connection parameters and HTTP authentication.

Note: You are actually connecting to your database through the PHP script on the server, so in most cases the host/server name is "localhost" unless the target database server is not installed on the same computer as the Web server.

2.1.3 MS SQL Server

Specify the following credentials to connect to Microsoft SQL Server.

Server
The full name of SQL Server you want to connect to. Can be specified as *computer_name* (for default SQL Server instances) or as *computer_name\server_name* (for named instances). To connect to an SQL Server running on a non-default port, specify the value of this field as *computer_name,port_number* (or *computer_name \server_name, port_number*). Press the arrow button to scan for SQL Servers accessible in your network.

Provider
The application allows you to connect to SQL Server using any of SQL Server clients installed on your computer. To choose a client you want to use, select the appropriate item in the Provider combobox. The table below shows the correspondence between the value selected in this combobox and SQL Server client to be used.

Value	SQL Server client	Introduced with
SQLOLEDB	Microsoft OLE DB Provider for SQL Server (default value)	Comes with Windows
SQLNCLI	SQL Server Native Client	SQL Server 2005
SQLNCLI10	SQL Server 2008 [R2] Native Client	SQL Server 2008 [R2]
SQLNCLI11	SQL Server 2012 Native Client (recommended)	SQL Server 2012

We would recommend you to install and use **SQL Server 2012 Native Client** as it (and only it) supports SQL Server Express LocalDB. Also it provides the best support for data types implemented in the recent versions of SQL Server.

Windows Authentication (more preferable)
Microsoft Windows Authentication mode allows a user to connect through a Windows user account.

SQL Server Authentication
When a user connects with a specified Login name and Password from a non-trusted connection, SQL Server performs the authentication itself by checking to see if a SQL Server login account has been set up and if the specified password matches the one previously recorded.

2.1.4 Firebird

Specify the following credentials to connect to Firebird.

- **Direct connection**

Direct connection is the most preferable connection mode.

© 2024 SQL Maestro Group

- **SSH tunnel connection**

If your Firebird server does not allow direct connections from remote workstations, you can establish connection to intermediate SSH server and forward all Firebird commands through the Secure SHell (SSH) tunnel.

Irrespectively of a connection mode you should specify common credentials as follows:

Protocol
Select "Standard server" to connect to a local or remote standalone Firebird server or "Embedded server 2.5-" / "Embedded server 3.0+" to use the embedded Firebird servers that come with the software (2.5 and 3.0 accordingly).

Host
The name of machine the Firebird server and database file resides on. It is usually possible to specify TCP/IP address instead of name (like 192.168.12.34), however this functionality is not built into Firebird, instead, it is provided by underlying network layer (e.g. WinSock2), and for some implementations/configurations it can be much slower than using name. In case the server is not localhost, specify the TCP/IP port as Port number.

Login information
Since Firebird 2.1, Windows authentication has been used for configuring the server authentication mode along with the traditional login info, requiring users to log in using a user name and password defined in the security database. To use Windows authentication, check the corresponding box. Under the right conditions, this may be the most secure way to authenticate on Windows. Otherwise, set the following server security options:

User name	The username used to connect to Firebird. User names are case insensitive on the server.
Password	The password for the user account on server. The server checks the user name and password against the security database. Case sensitivity is retained for the comparison. Only first 8 characters of Password are verified.
Role	Specify the role that the user adopts on connection to the database or leave it blank. Regardless of role memberships granted, the user has the privileges of a role at connect time only if a Role clause is specified in the connection. The user must have previously been granted membership in the role to gain the privileges of that role.

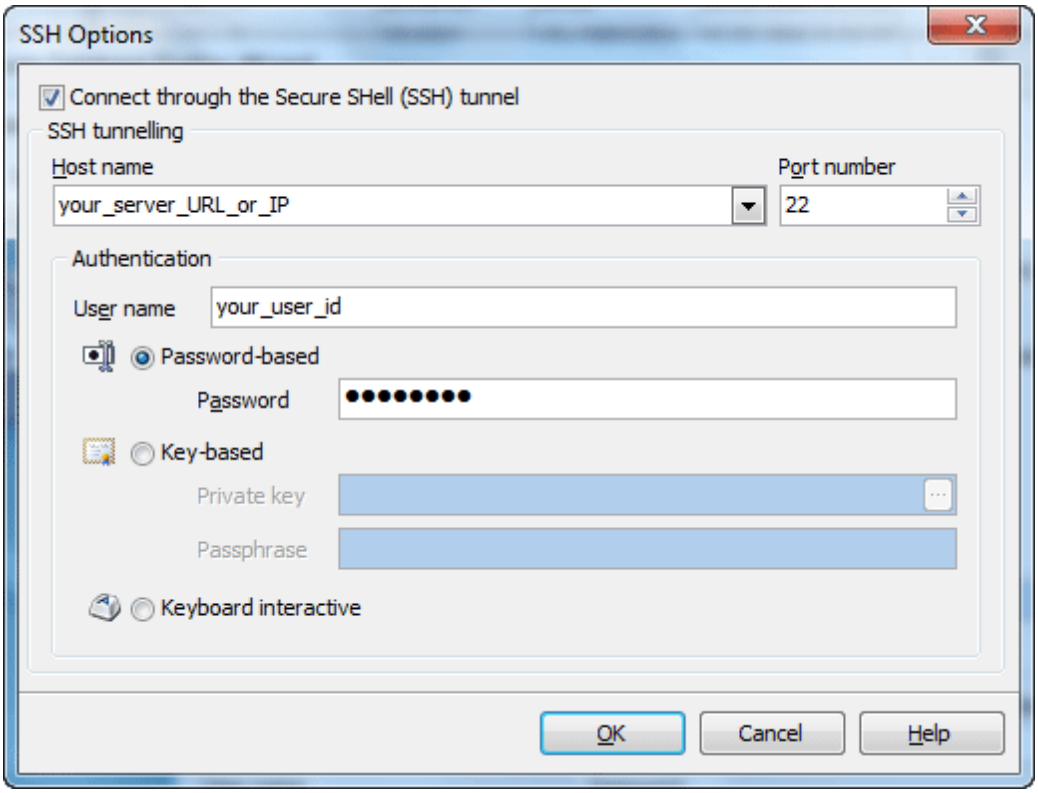
Database name
This is either a full name of the database file (as seen from the server), or an alias defined in the server's aliases.conf file.

More about SSH tunnel connection

To establish connection to intermediate SSH server and forward all SQLite commands through the secure tunnel, you need to:

1. Check I can connect to the server directly or via SSH tunneling.
2. Follow the Configure SSH options link to open the SSH Options window.

© 2024 SQL Maestro Group



3. Check Connect through the Secure Shell (SSH) tunnel and complete the following fields:

Host name
Specify the host name or IP of your site. Note, that SQLite host name always should be set relatively to the SSH server. For example, if both of SQLite and SSH servers are located on the same computer, you should specify localhost as Host name instead of server's external host name or IP address.

Port number
Enter the port number for the SSH server.

4. Enter valid User name for the remote server and select the Authentication method and set corresponding credentials.

Password-based
Set the password corresponding to the specified user.

Key-based
Specify the path to the Private key file with the corresponding Passphrase to log in to the remote server. SQLite Database Converter accepts keys in **ssh.com** or **OpenSSH** formats. To convert a private key from PuTTY's format to one of the formats supported by our software, use the PuTTYgen utility that can be freely downloaded from the PuTTY website.

© 2024 SQL Maestro Group

Keyboard interactive
Keyboard authentication is the advanced form of password authentication, aimed specifically at the human operator as a client. During keyboard authentication zero or more prompts (questions) is presented to the user. The user should give the answer to each prompt (question). The number and contents of the questions are virtually not limited, so certain types of automated logins are also possible.

2.1.5 Oracle

To connect to an Oracle database with SQLite Database Converter, specify the following connection options:

User name
Use the field to specify the username to be used to connect to Oracle.

Password
Enter the password for the user account on server.

Database name
An entry from TNSNames.ora.

Connect mode
Allows you to connect with required administrative privileges (SYSDBA or SYSOPER). More information.

Check Use Operating system authentication to allow Oracle to pass control of user authentication to the operating system. The technology works as follows:

- First, create an OS user (if it doesn't exist).
- Check a value of the Oracle *OS_AUTHENT_PREFIX* initialization parameter. Current value of this parameter can be retrieved using the followng query:

  ```
  SELECT VALUE FROM  V$PARAMETER
  WHERE NAME  = 'os_authent_prefix'
  ```

  The default value is OPS$. The initialization parameter can be modified with the ALTER SYSTEM command.

- Create a database user. The user must use the external identification and its name must be the prefix value concatenated to the OS username (on Windows platforms you would expect an Oracle username of "OPS$DOMAIN\MY_USER" for the user "my_user").

2.1.6 SQLite

As SQLite is implemented as an embedded database engine contained in a single DLL, SQLite databases usually are stored locally or in the shared folders. To connect to such database, you should provide only a full database file name (e.g. C:\Data\SQLite \MyDatabase.db3) and a password (only for encrypted databases).

To read and write encrypted databases, SQLite Database Converter uses the free wxSQLite3 library that is included into the installation package. This means it can

© 2024 SQL Maestro Group

operate only with encrypted databases created by itself or by any other tool that uses the same library. Unfortunately, our software cannot connect to databases encrypted by any other library because different SQLite security extensions use different algorithms, which are not compatible with each other.

SQLite engine does not support network connections, however SQLite Database Converter allows you to manage remote SQLite databases using the HTTP tunneling technique. For this purpose, you need to have a webserver running on a computer that stores the database file. Of course this webserver should be accessible from your workstation and you should be able to upload files there.

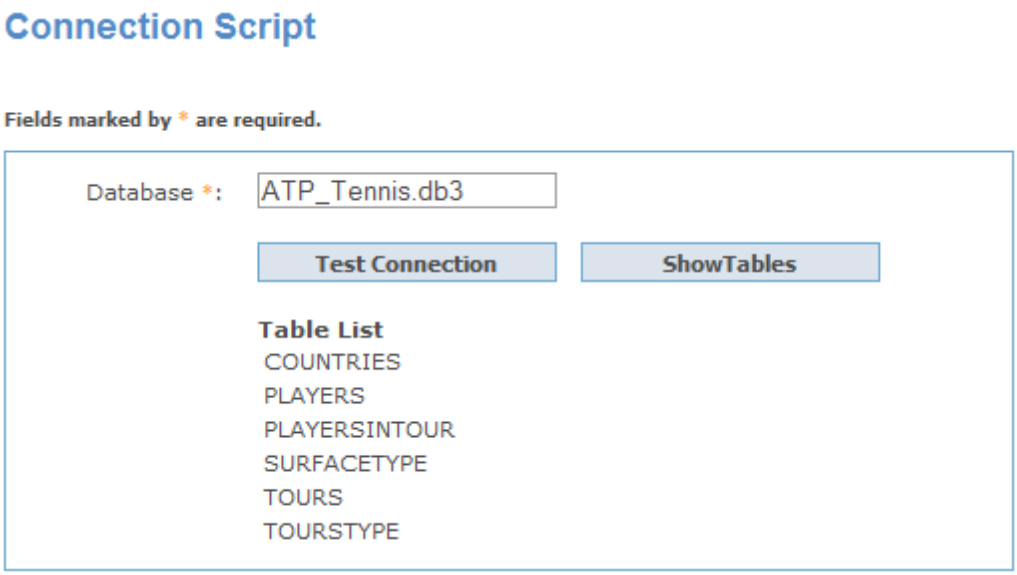
More about connection via HTTP tunnel

To connect to a remote SQLite database using an HTTP tunnel, you need to:

1. Upload the connection PHP script to your website. The scripts are named *sqlite_tunnel.php* and *sqlite3_tunnel.php* for SQLite databases versions 2 and 3 accordingly and can be found under the installation folder, usually *C:\Program Files\SQL Maestro Group\SQLite Database Converter*.

2. Turn ON the I have to use HTTP tunneling checkbox.

3. Enter the connection PHP script URL, e.g. *www.yoursite.com/files/sqlite_tunnel.php*. You can test the connection before the profile is created. Just use Test script using default browser to open connection script in your browser, enter all the required connection parameters and use the Test connection button.



4. In case using of a proxy server use Configure tunnelling options to open the HTTP tunnelling options window and specify your proxy server connection parameters and HTTP authentication.

Note 1. Do not forget to enable read/write permissions for a database file and read/write/execute permissions for the directory where the database file is

© 2024 SQL Maestro Group

stored.

Note 2 (only for SQLite 3 databases). The webserver PDO_SQLite library must be compatible (not earlier in the most cases) with the library the database was created with. If they are not compatible, you will get an error message "Could not retrieve table list from _database_name_ ... " on getting a table list at the connection script. If you've got the message, check the PDO_SQLite library version using, for example, the *phpinfo()* function, download a compatible library from the SQLite official website, get an SQL dump of the database and create a new one from the dump file with this library.

© 2024 SQL Maestro Group

2.2 Connecting to the target database

As SQLite is implemented as an embedded database engine contained in a single DLL, SQLite databases usually are stored locally or in the shared folders. To connect to such database, you should provide only a full database file name (e.g. C:\Data\SQLite \MyDatabase.db3) and a password (only for encrypted databases).

To read and write encrypted databases, SQLite Database Converter uses the free wxSQLite3 library that is included into the installation package. This means it can operate only with encrypted databases created by itself or by any other tool that uses the same library. Unfortunately, our software cannot connect to databases encrypted by any other library because different SQLite security extensions use different algorithms, which are not compatible with each other.

SQLite engine does not support network connections, however SQLite Database Converter allows you to manage remote SQLite databases using the HTTP tunneling technique. For this purpose, you need to have a webserver running on a computer that stores the database file. Of course this webserver should be accessible from your workstation and you should be able to upload files there.

More about connection via HTTP tunnel

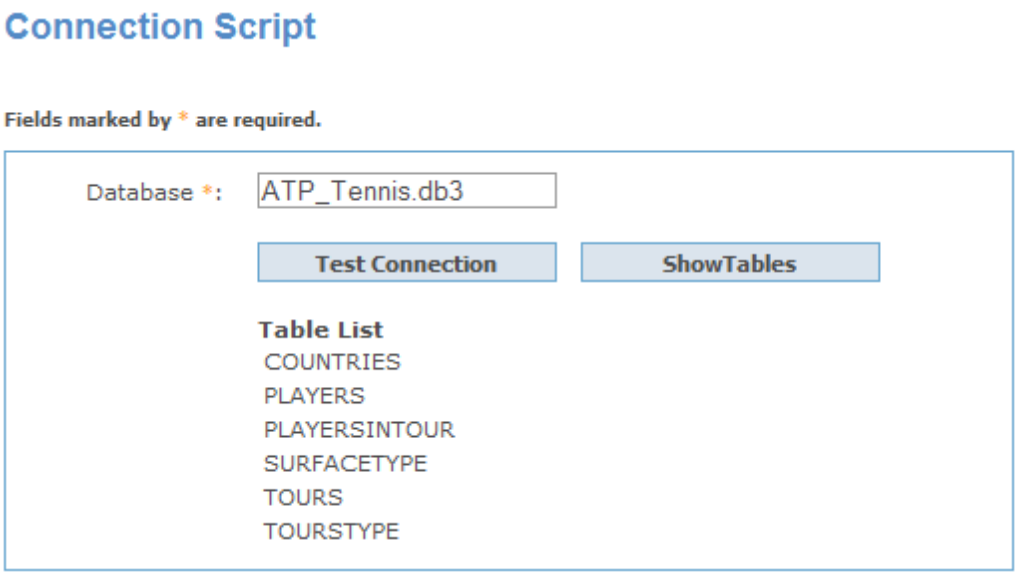
To connect to a remote SQLite database using an HTTP tunnel, you need to:

1. Upload the connection PHP script to your website. The scripts are named *sqlite_tunnel.php* and *sqlite3_tunnel.php* for SQLite databases versions 2 and 3 accordingly and can be found under the installation folder, usually *C:\Program Files\SQL Maestro Group\SQLite Database Converter*.

2. Turn ON the I have to use HTTP tunneling checkbox.

3. Enter the connection PHP script URL, e.g. *www.yoursite.com/files/sqlite_tunnel.php*. You can test the connection before the profile is created. Just use Test script using default browser to open connection script in your browser, enter all the required connection parameters and use the Test connection button.

© 2024 SQL Maestro Group



4. In case using of a proxy server use Configure tunnelling options to open the HTTP tunnelling options window and specify your proxy server connection parameters and HTTP authentication.

Note 1. Do not forget to enable read/write permissions for a database file and read/write/execute permissions for the directory where the database file is stored.

Note 2 (only for SQLite 3 databases). The webserver PDO_SQLite library must be compatible (not earlier in the most cases) with the library the database was created with. If they are not compatible, you will get an error message "Could not retrieve table list from _database_name_ ... " on getting a table list at the connection script. If you've got the message, check the PDO_SQLite library version using, for example, the *phpinfo()* function, download a compatible library from the SQLite official website, get an SQL dump of the database and create a new one from the dump file with this library.

© 2024 SQL Maestro Group

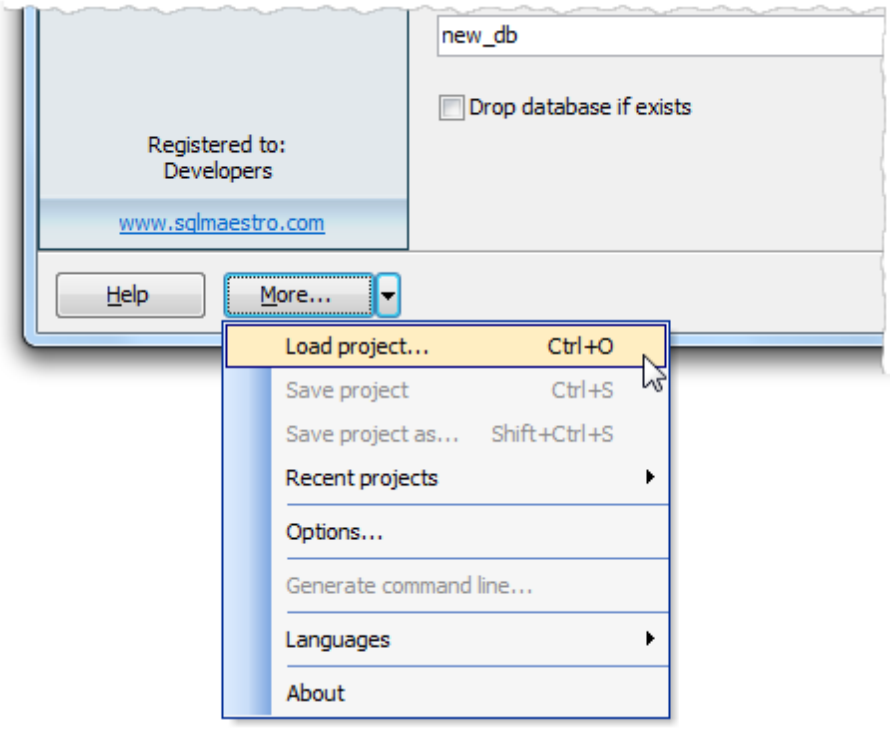
2.3 Projects

SQLite Database Converter allows you to save and restore all the options set during a session. You need not to specify all options each time you work with the application anew; instead you can load all settings from a project and change them if necessary.

Projects are very useful when working with SQLite Database Converter. If you will close the application without saving a project, all carefully adjusted settings will be lost. To set the same options next time, you'll need to repeat the process step by step again while with a project all the session parameters can be restored in a few mouse clicks.

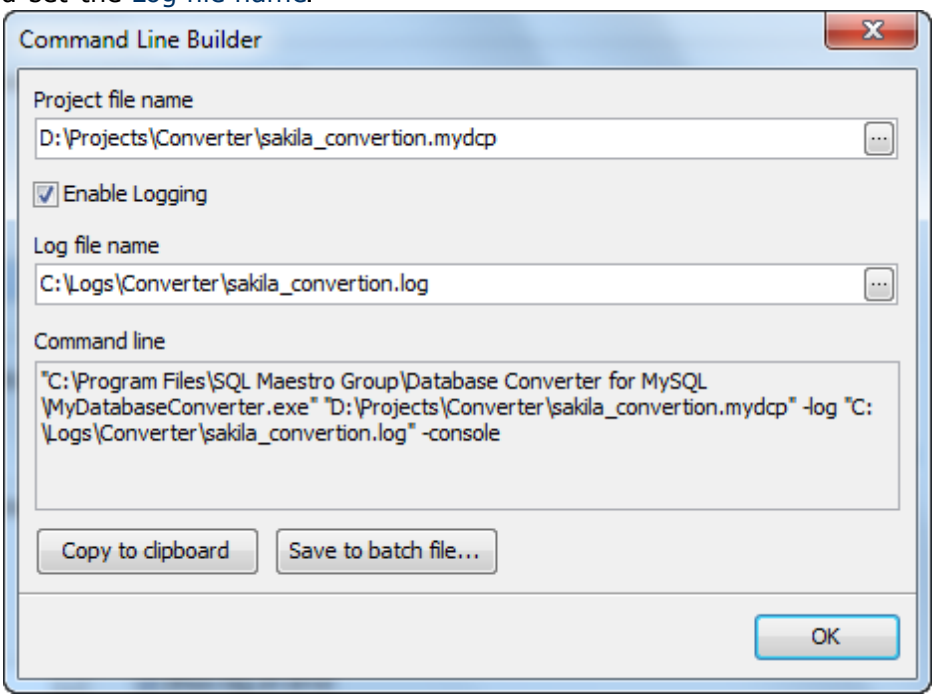
To create a project, click More > Save Project. All the settings you have made will be saved to a file.

To restore previously saved settings from a project, click More > Load Project at the first wizard step. Recently used projects are available from the More > Recent Projects popup menu.



2.4 Command line options

SQLite Database Converter supports a number of command line options that make it possible to fully automate database migration. To generate the command line automatically, load the project to be used or specify the comparison options and click More > Generate command line. To log the conversion process, check the corresponding option and set the Log file name.



The SQLite Database Converter command line syntax is as follows:

MyDatabaseConverter[.exe] <project_file_name> [-l|log <log_file_name>] -console

MyDatabaseConverter[.exe]	The SQLite Database Converter program file.
<project_file_name>	The project 26 with all the task's settings.
-c|console	Runs the wizard in console mode. This option is required.
-l|log	Enables logging, requires the log file name.

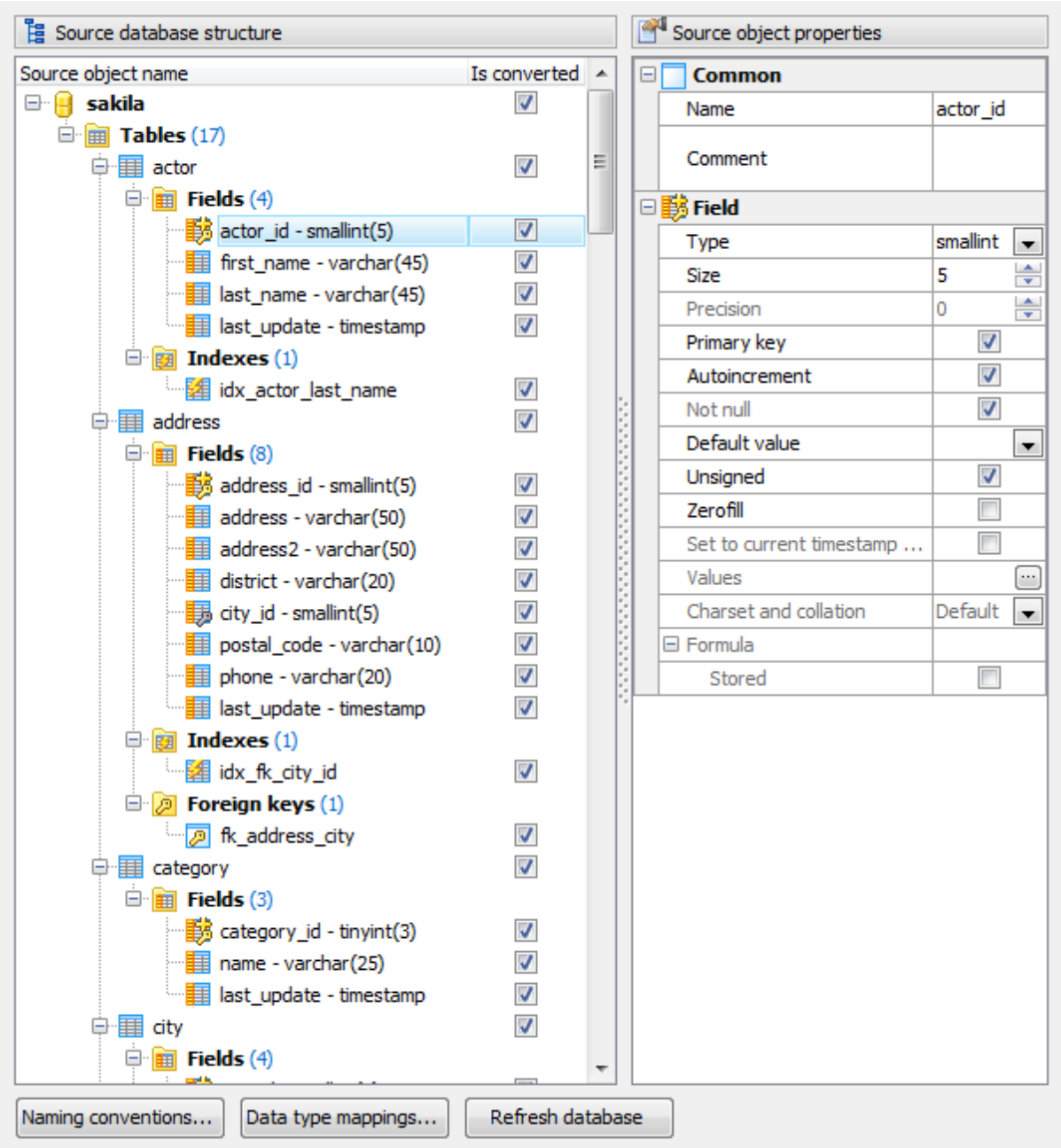
Examples
The example below assumes that you are entering the command lines in the SQLite Database Converter program directory. Don't forget to enclose all paths and filenames containing spaces in quotes.
"C:\Program Files\SQL Maestro Group\Database Converter for MySQL \MyDatabaseConverter.exe" "D:\Projects\sakila_conversion.mydcp" -console

© 2024 SQL Maestro Group

3 Source objects

The Source database structure tab represents the schema of the source database and allows you to specify objects to be transferred to the target one. To include an object to the conversion process, turn on the corresponding "Is converted" option. To get the actual database objects tree, use the Refresh database button. The Source object properties tab allows you to browse the selected object's options.

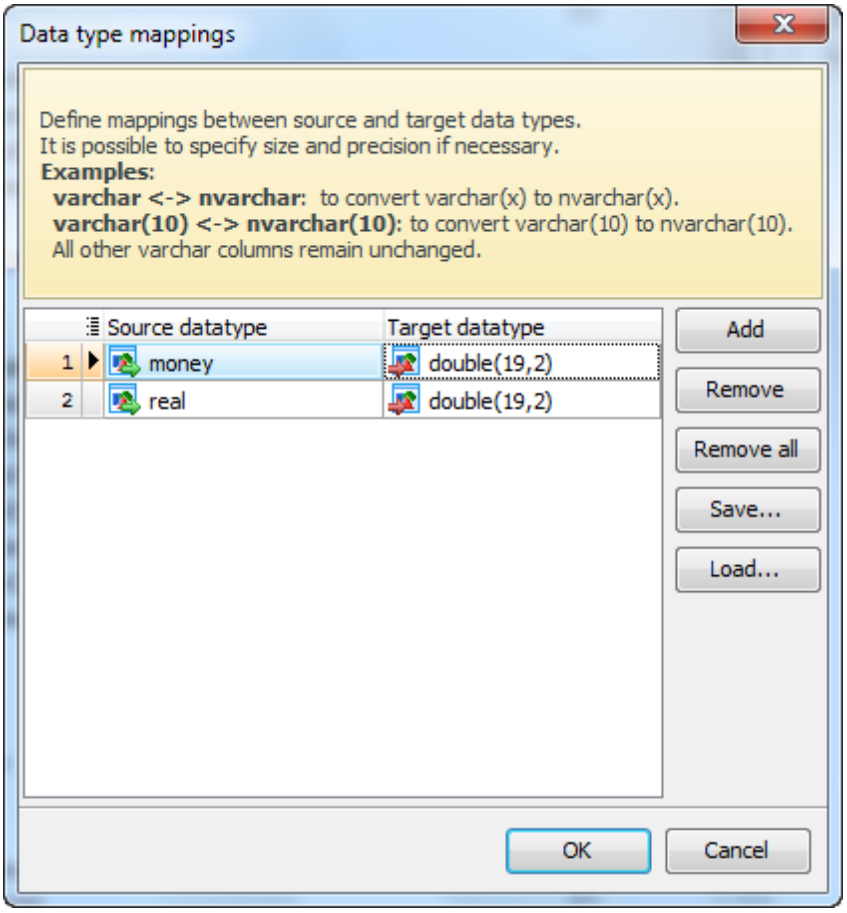
To set rules of the data types conversion 29 and naming conventions 30 to be applied during the schema transferring, use the corresponding windows.



3.1 Data type mappings

The necessary condition of successful conversion is the accurate mappings between source and target data types. To set rules of the data type conversion, use the Datatype mapping... button at the Selecting source objects step and define the rules to be used for the transferring data.

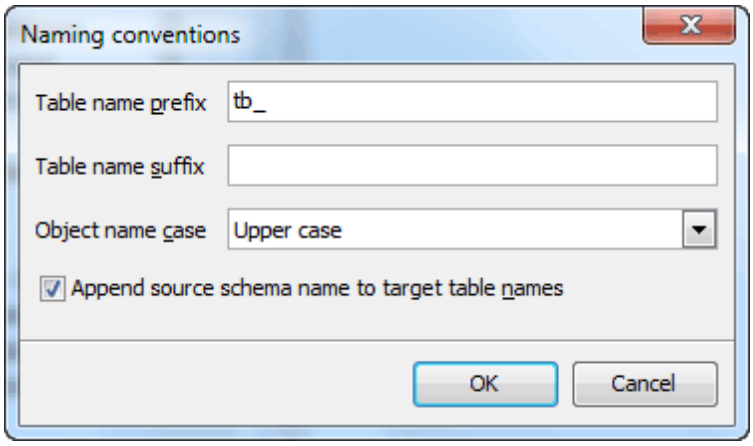
To define a mapping, use the Add button and select Source and Target datatypes from the corresponding drop-down lists. To set size and precision, type it directly to the according cell. You can also save specified mapping to a file to be used for other conversions.



© 2024 SQL Maestro Group

3.2 Naming conventions

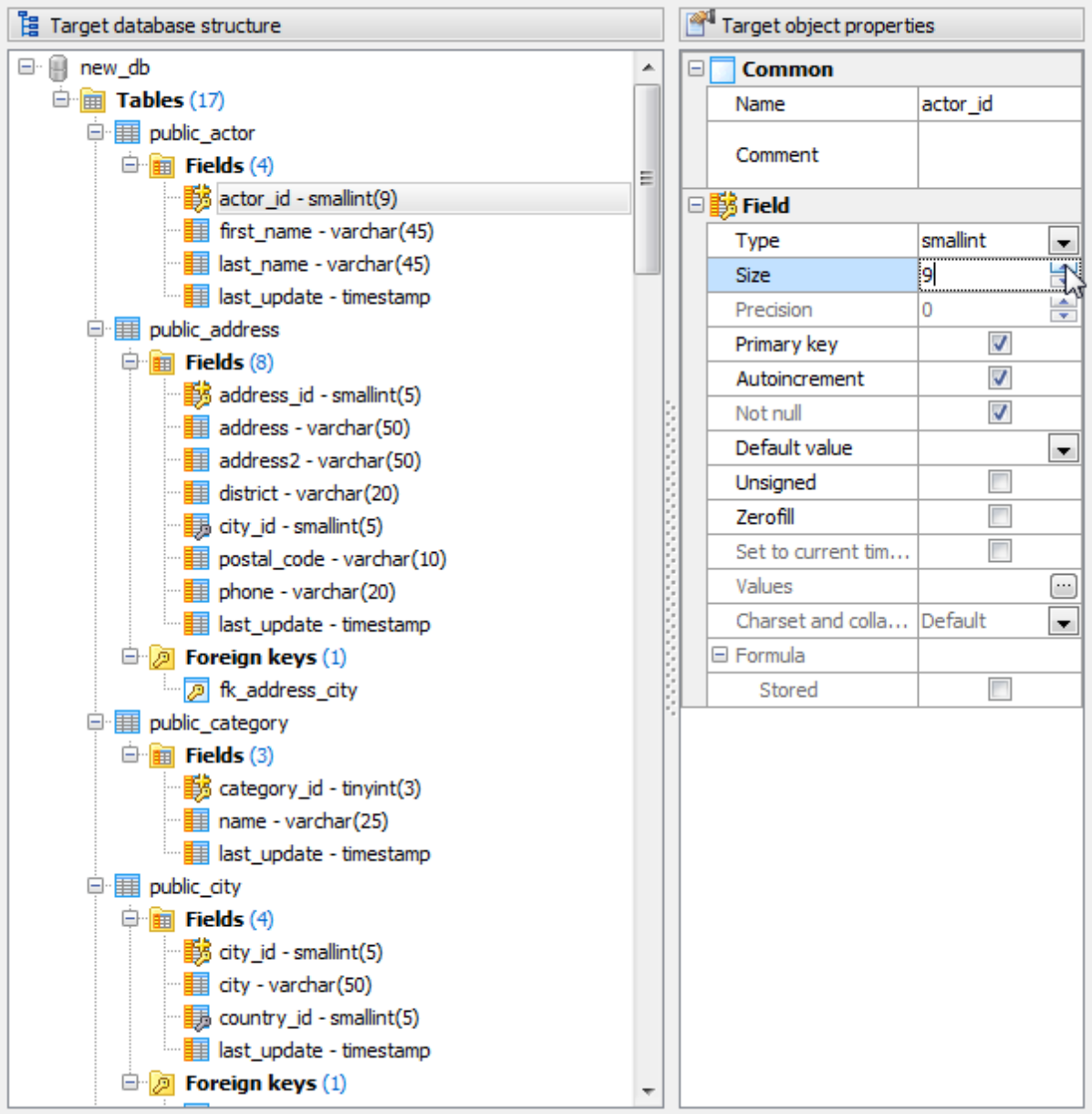
The Naming conventions window allows you to define rules to be applied for the names of the created objects. Here you can specify prefix and suffix for the target tables as well as the case of names of objects to be created. The Append source schema name to target table names checkbox allows you to choose whether source schema name will be added to target table names.



© 2024 SQL Maestro Group

4 Target objects

On this step you can customize the objects to be created in the target database. The Target object properties area on the right of the window allows you to change characteristics of the selected object such as name, comment, data type, etc.



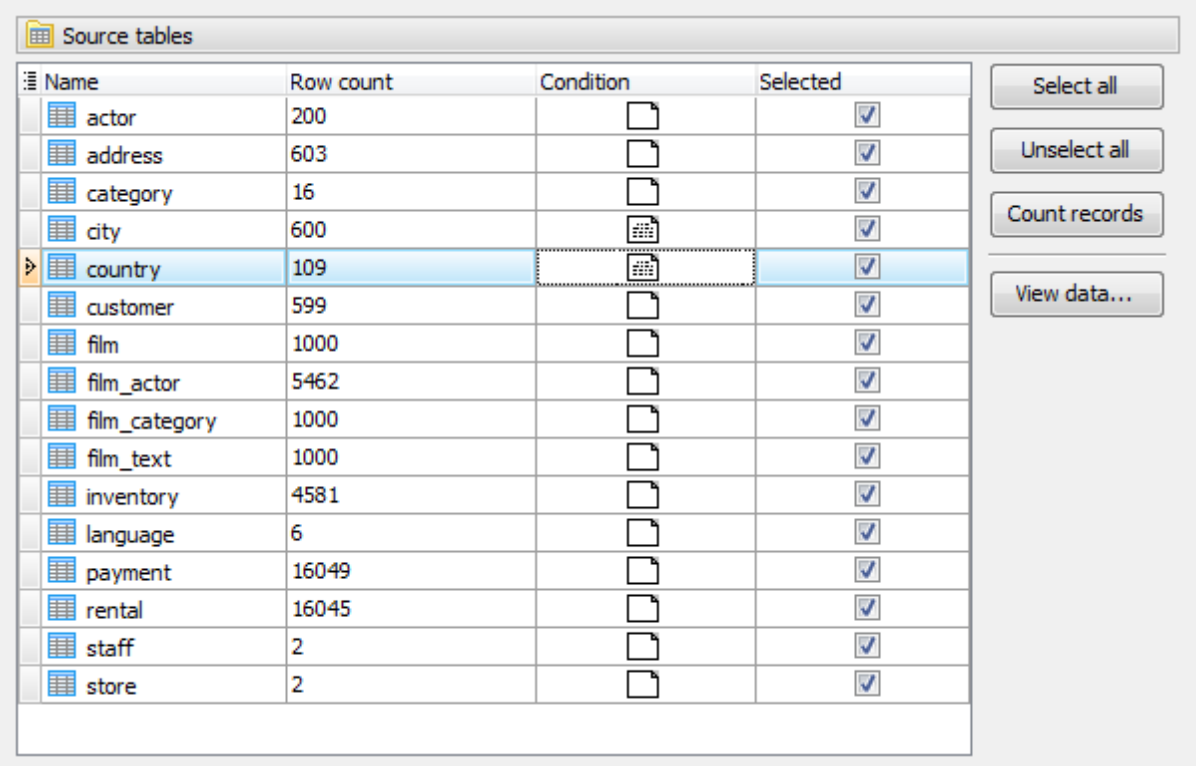
To validate the integrity of the target database after all modifications, invoke the Database integrity errors dialog with the corresponding button. SQLite Database Converter recognizes if an identifier name is longer than allowed by SQLite or if a field in foreign key and referenced field have no similar data types. To define the behavior of the software in these cases, use the corresponding application options 37.

© 2024 SQL Maestro Group

5 Data import

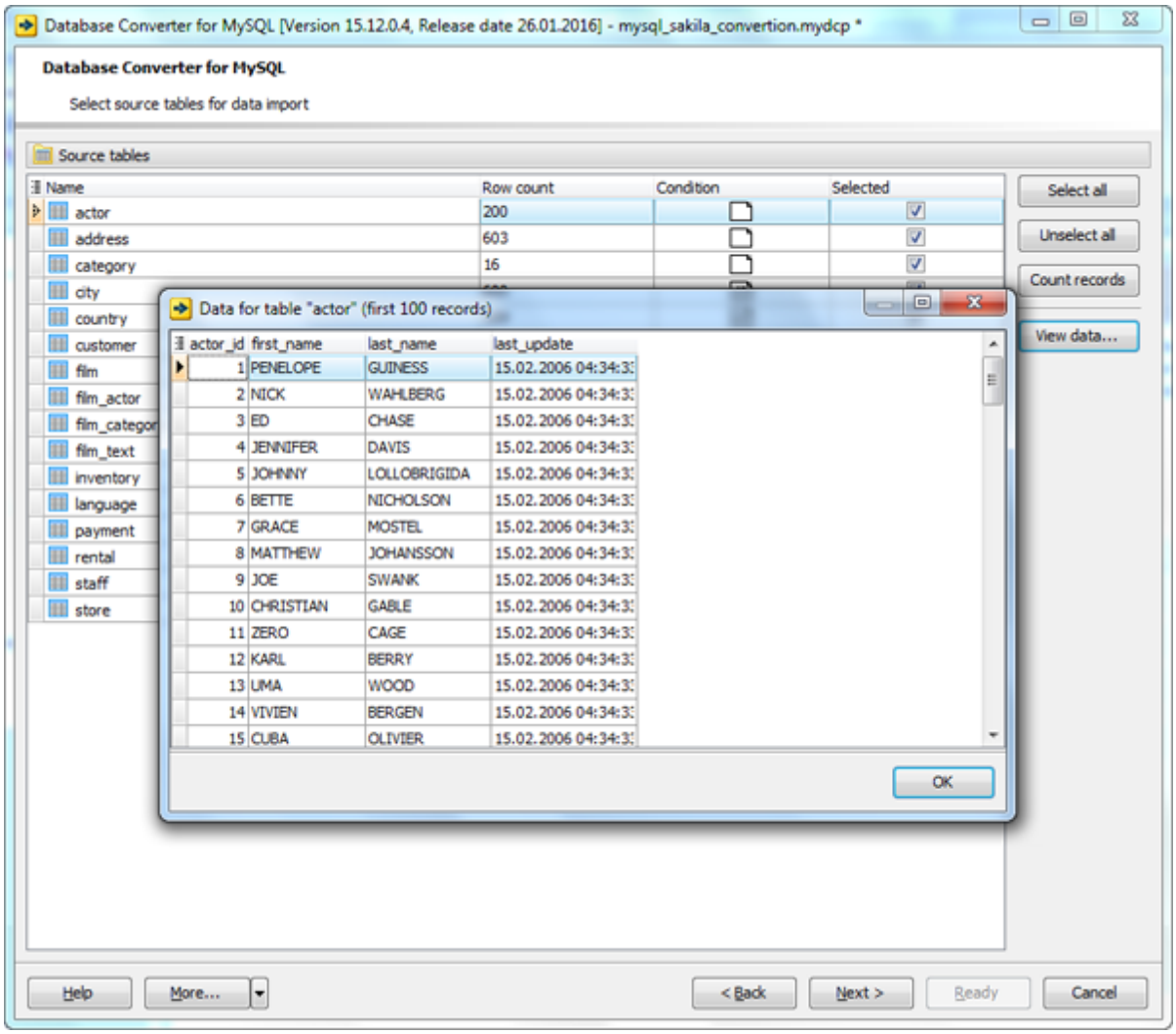
Use this step to setup data conversion. The Source tables
To import data from a table, turn on the corresponding Selected option.

By default the number of records of the source table are not displayed. To find out them, use the Count records button. This process may take some time especially if you have a slow database connection. You can restrict the number of transferred rows with a filter condition.



The View data button allows you to browse the source table content.

© 2024 SQL Maestro Group



© 2024 SQL Maestro Group

6 Converting options

Use this step to define whether the source objects will be converted directly to the target database or/and SQL scripts with objects definitions and INSERT statements will be generated.

To generate objects directly to the target database, turn on the Generate database objects checkbox (do not activate this option if you want only to generate an SQL script). To import data to selected tables, turn the Insert records into database option "ON".

Import Data Wizard supports the LOAD DATA INFILE command to insert data to the table. This feature can speed up the import process up to 10 times so it is recommended to use it always if possible. Uncheck this option to use INSERT statements instead.

To generate SQL scripts to a file to be executed later using any appropriate software product, use the Script generation
checkbox group. To save SQL definitions of the target objects to .sql file, turn on the Save metadata script to file option and specify the file path. To equip this file with insert statements, turn on the corresponding option.

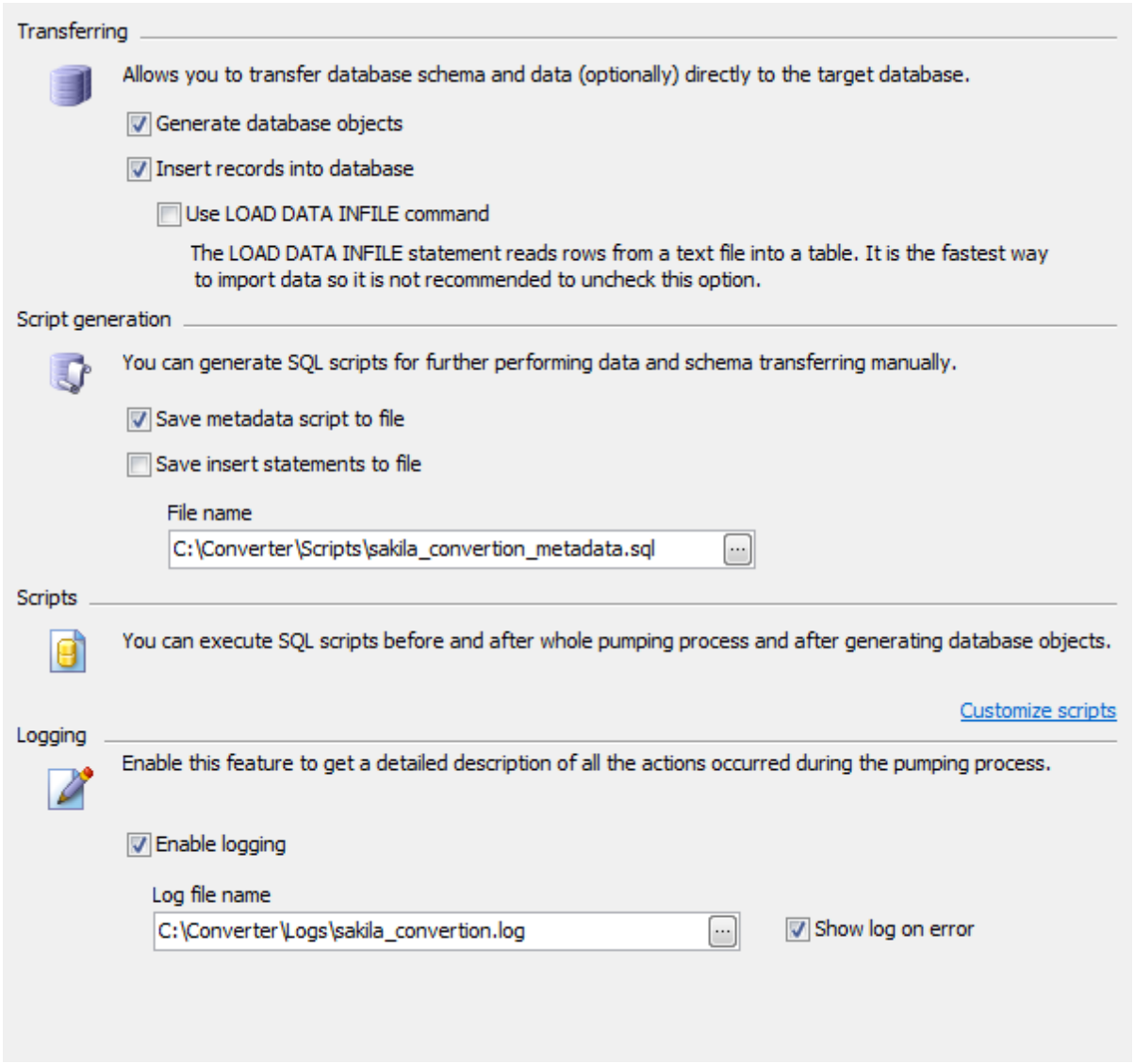
Scripts
Follow the Customize scripts link to specify SQL scripts to be executed during the transfer. The scripts are executed as follows:

- *After connect*: after connecting to the target database but before creating schema objects;
- *Before data importing*: after creating schema objects but before transferring tables' data;
- *After converting*: after completing the deal.

Logging
This option group allows you to enable/disable logging of the converting process.

Projects
On this step you can save all parameters that have been set during the wizard session to a project file 26 for a further using. For this purpose, click the More... button and select the corresponding link.

© 2024 SQL Maestro Group



© 2024 SQL Maestro Group

7 Options

SQLite Database Converter allows you to customize the way it works within the Options dialog.

The window allows you to customize the options grouped by the following sections:

- **Application** 37
  General SQLite Database Converter options: environment style, confirmations, window restrictions
- **Editors & Viewers** 40
  Customizing of all the SQL editors.
- **Appearance** 48
  Customizing program interface - bars, trees, menus, etc.

It is a good idea to check through these settings before you start working with SQLite Database Converter. You may be surprised at all the things you can adjust and configure!

© 2024 SQL Maestro Group

7.1 Application

The Application section allows you to customize common rules of SQLite Database Converter behavior. The section consists of several tab; follow the links to find out more about each of them.

- Common 37
- Confirmations 38

7.1.1 Common

Use this tab to define rules to be used for converting process.

Converting error behavior

On creating

Select the action to be executed when an error occurs on creating a schema object: continue or abort objects creation.

On importing

Select the action to be executed when an error occurs during the data importing: abort importing, skip current table, or continue importing.

Action on database integrity error

Identifier name is too long

Use this option to define the application behavior in case the target schema contains an identifier which name is longer than allowed by SQLite .

- Select Ignore to use this name in the object definition. In this case this identifier will be included into the SQL definition with the specified name.
- Use Set name manually to enter a new name allowed by server in the Database integrity errors 31 dialog window or in the Target object properties 31 tab.
- It the Define name automatically option is selected, a name like "Table01", "Index03", and so on will be assigned for this object by the server.

Field in foreign key and referenced field have no similar data types

This option defines whether the application will change the data type of the foreign field to the data type of the referenced field automatically or will ignore this integrity error.

Options

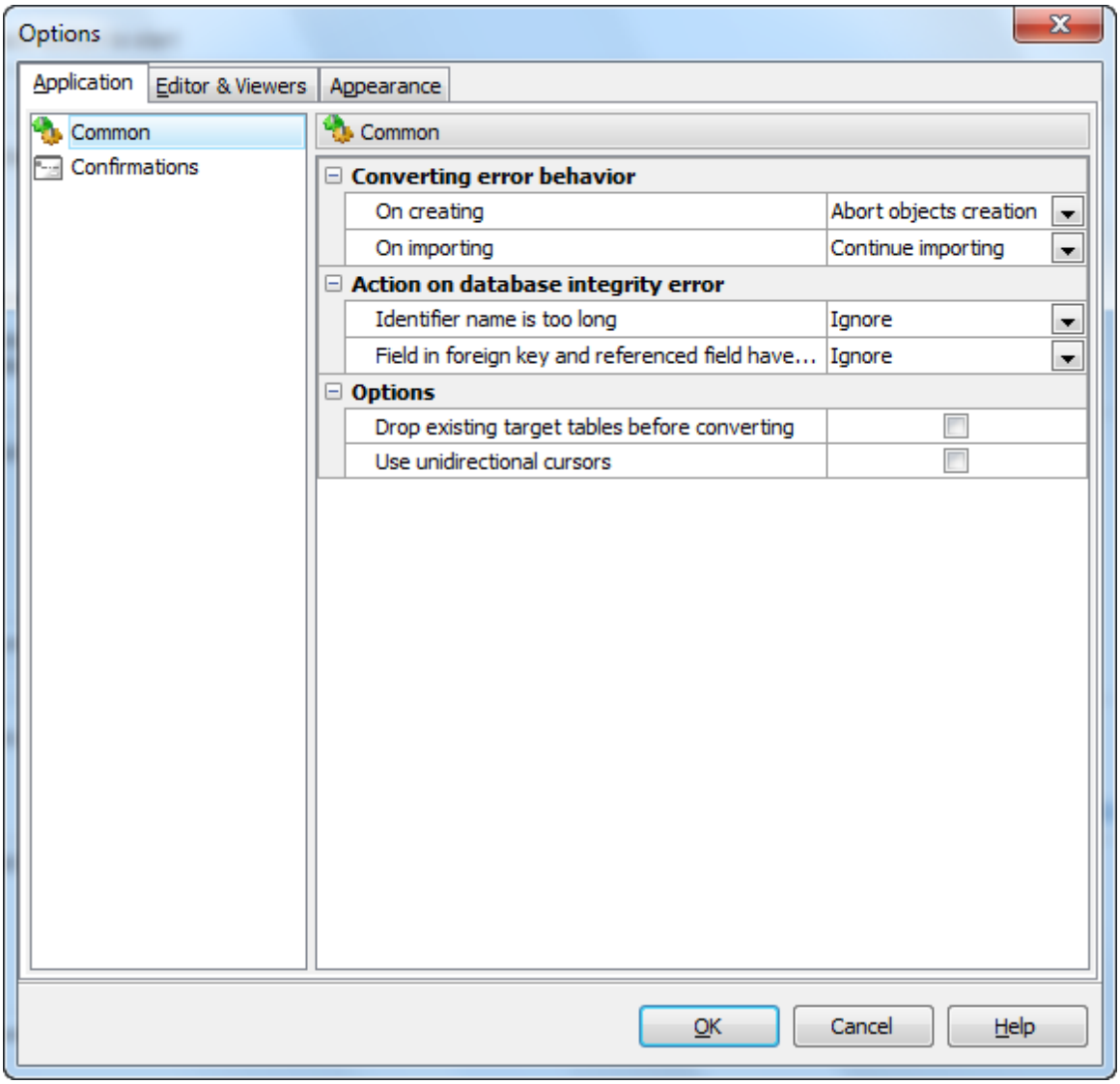
☑ Drop existing target tables before converting

Check this option to recreate existing target tables.

☑ Use unidirectional cursors

To improve data import performance, set this option to true, restricting a cursor to forward movement through a result set.

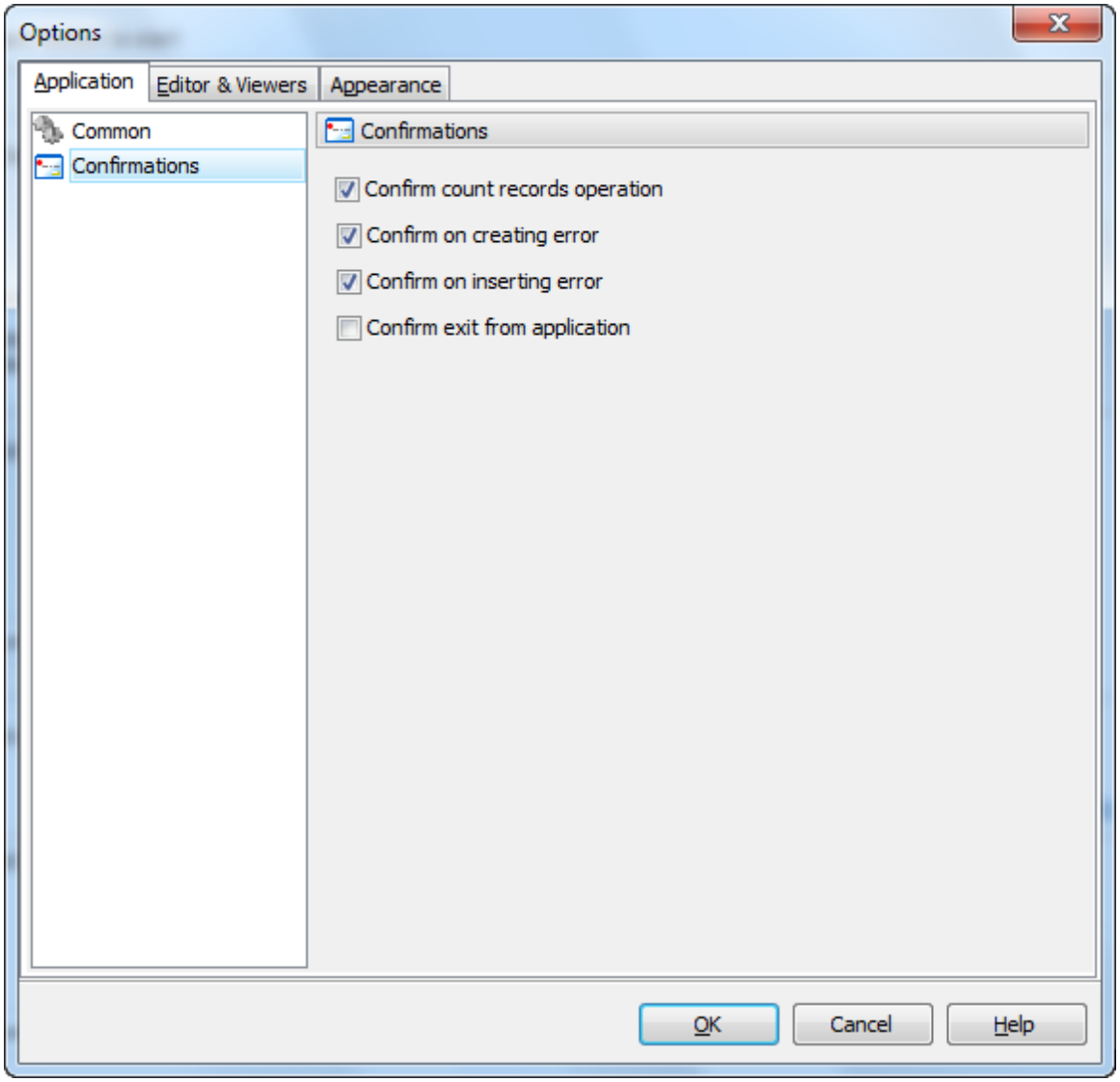
© 2024 SQL Maestro Group



7.1.2 Confirmations

Use this tab to manage application confirmations.

© 2024 SQL Maestro Group



Confirm count records operation

If this option is checked, the program asks you to confirm for count records operation.

Confirm on creating error

If this option is checked, the program requires confirmation each time an error occurs during schema creating.

Confirm on inserting error

If this option is checked, the program requires confirmation each time an error occurs during data inserting.

Confirm exit from application

If this option is checked, the program requires confirmation when you want to exit <% PRODUCT_NAME%.

© 2024 SQL Maestro Group

7.2 Editors & Viewers

The Editors & Viewers section allows you to set the parameters of viewing and editing the SQL statements within SQLite Database Converter.

- General 40
- Display 41
- SQL highlight 42
- PHP highlight 44
- XML highlight 43
- Code Insight 45
- Code Folding 46

7.2.1 General

If the Auto indent option is checked, each new indention is the same as the previous when editing SQL text.

☑ Insert mode
If this option is checked, insert symbols mode is default on.

☑ Use syntax highlight
Enables syntax highlight in the object editor window.

☑ Always show links
If this option is checked, hyperlinks are displayed in the editor window. To open a link click it with the **Ctrl** button pressed.

☑ Show line numbers
If this option is checked, line numbers are displayed in the editor window.

☑ Show special chars
If this option is checked, special chars (like line breaks) are displayed in the editor window.

☑ Use smart tabs
With this option on the number of tab stops is calculated automatically, depending on the previous line tab.

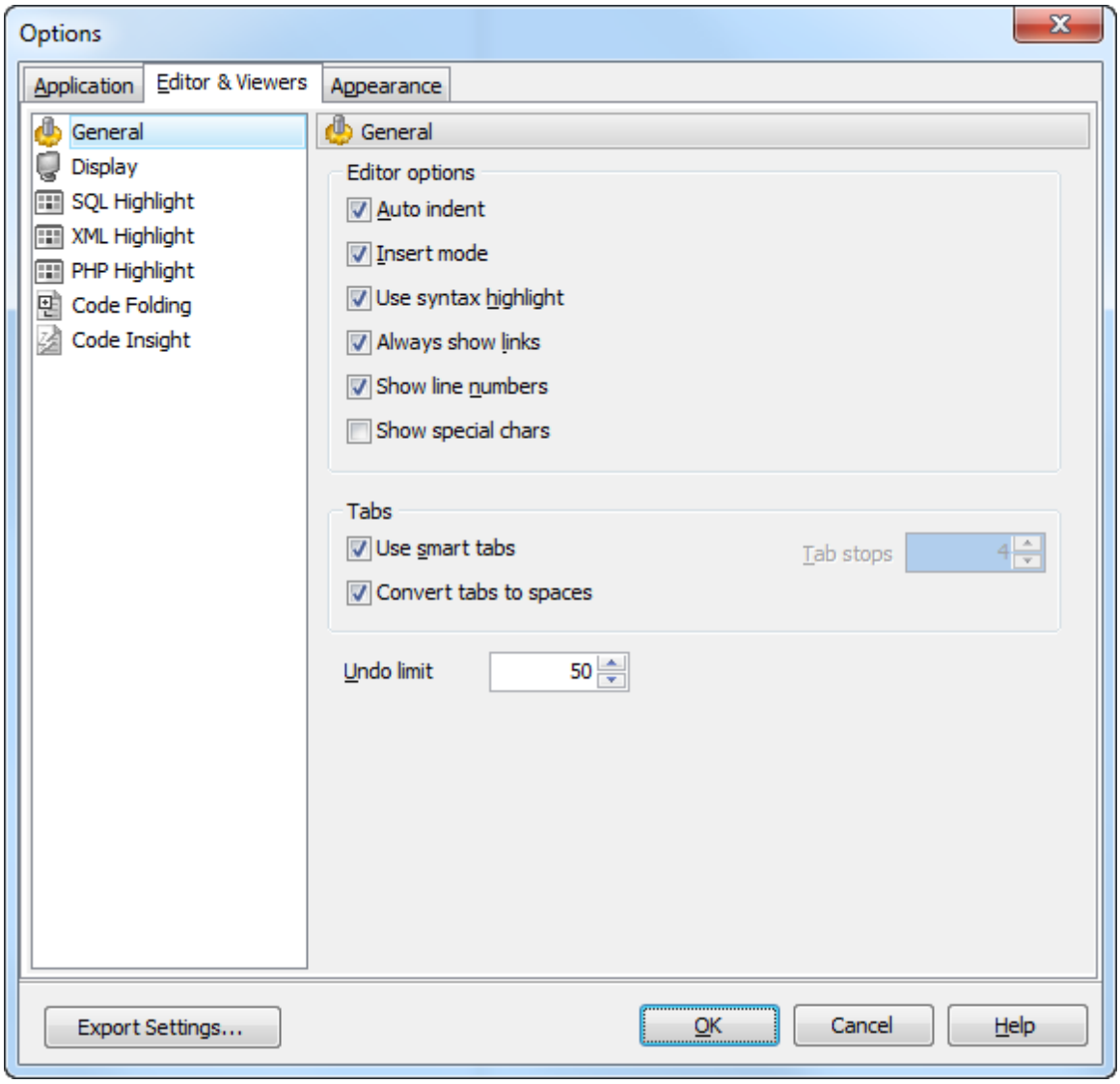
☑ Convert tabs to spaces
If this option is checked, each time you press the Tab key, the appropriate number of spaces will be added to the edited text.

Tab Stops
Defines the tab length, used when editing text.

Undo Limit

© 2024 SQL Maestro Group

Defines the maximum number of changes possible to be undone.

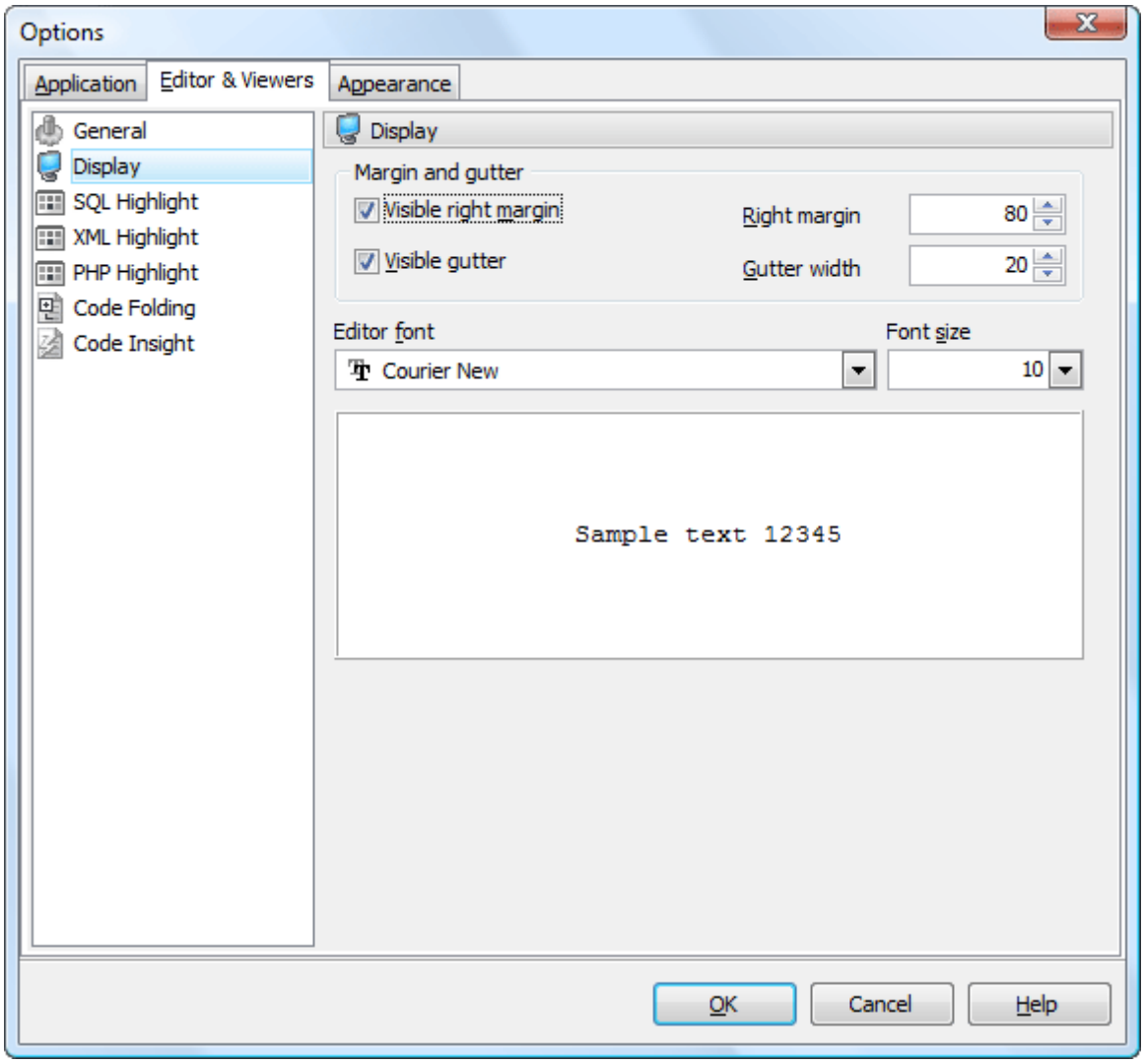


7.2.2 Display

You can disable/enable the right text margin and the gutter of the editor area, set the position of the right text margin as Right margin, and the Gutter width.

Use the Editor font and Font size to define the font used in all program editors and viewers. The panel below displays the sample of the selected font.

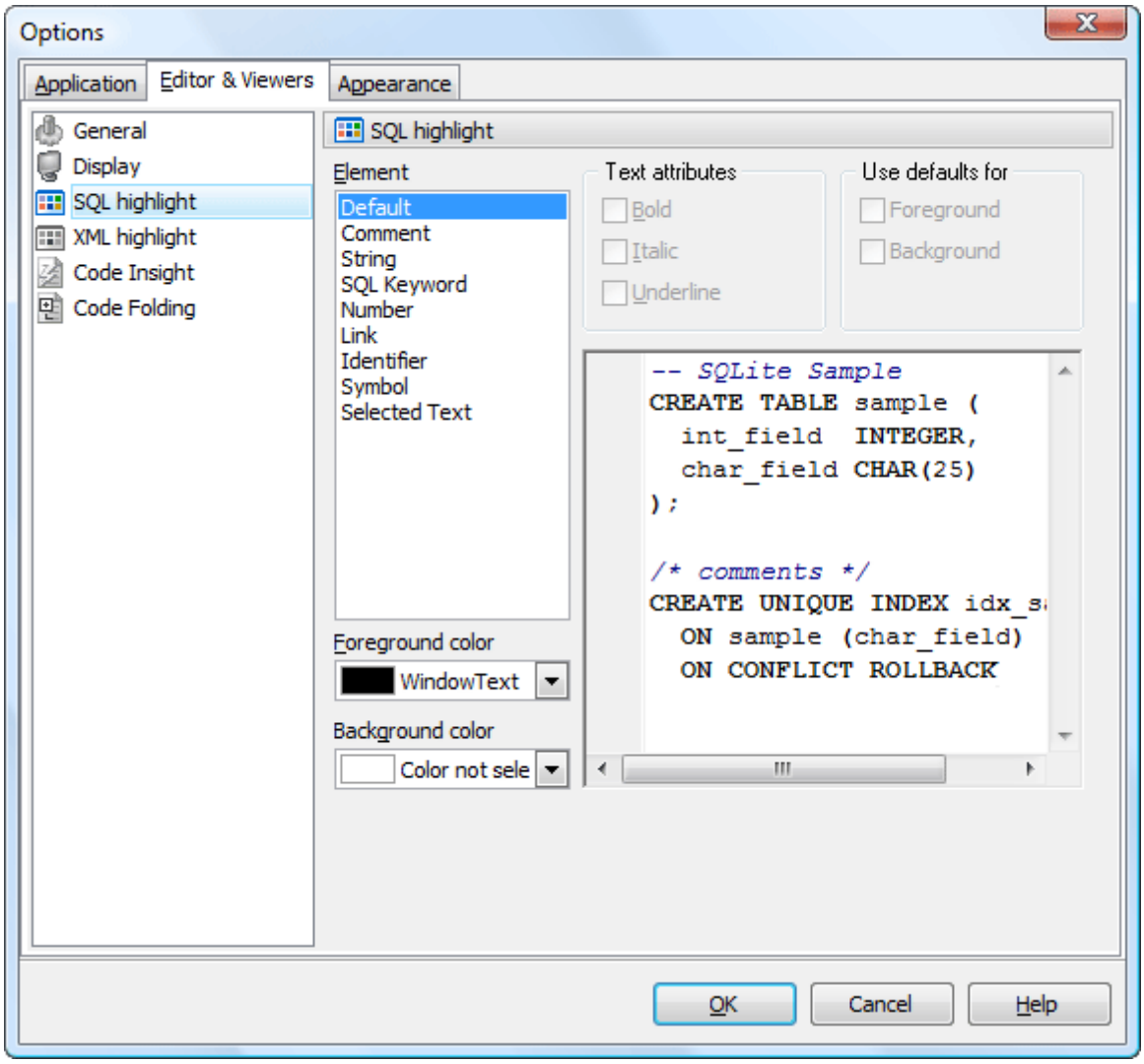
© 2024 SQL Maestro Group



7.2.3 SQL highlight

Use the SQL highlight item to customize syntax highlight in all SQL editors and viewers. Select the text element from the list, e.g. *comment* or *SQL keyword* and adjust its foreground color, background color and text attributes according to your preferences.

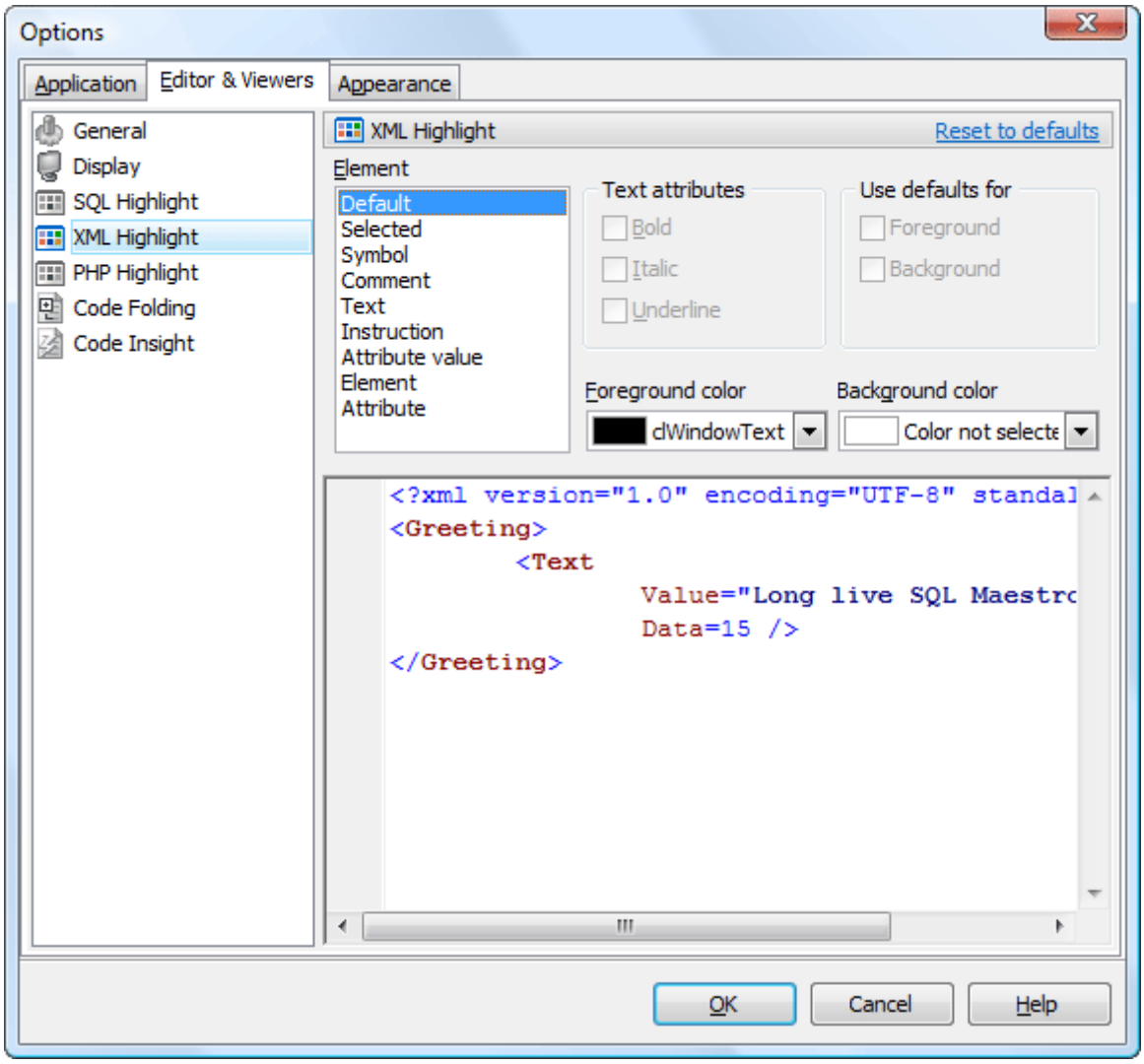
© 2024 SQL Maestro Group



7.2.4 XML highlight

Use the XML highlight item to customize XML syntax highlight for the text representation of BLOBs. Select the text element from the list, e.g. attribute or attribute value and adjust its foreground color, background color and text attributes according to your wishes.

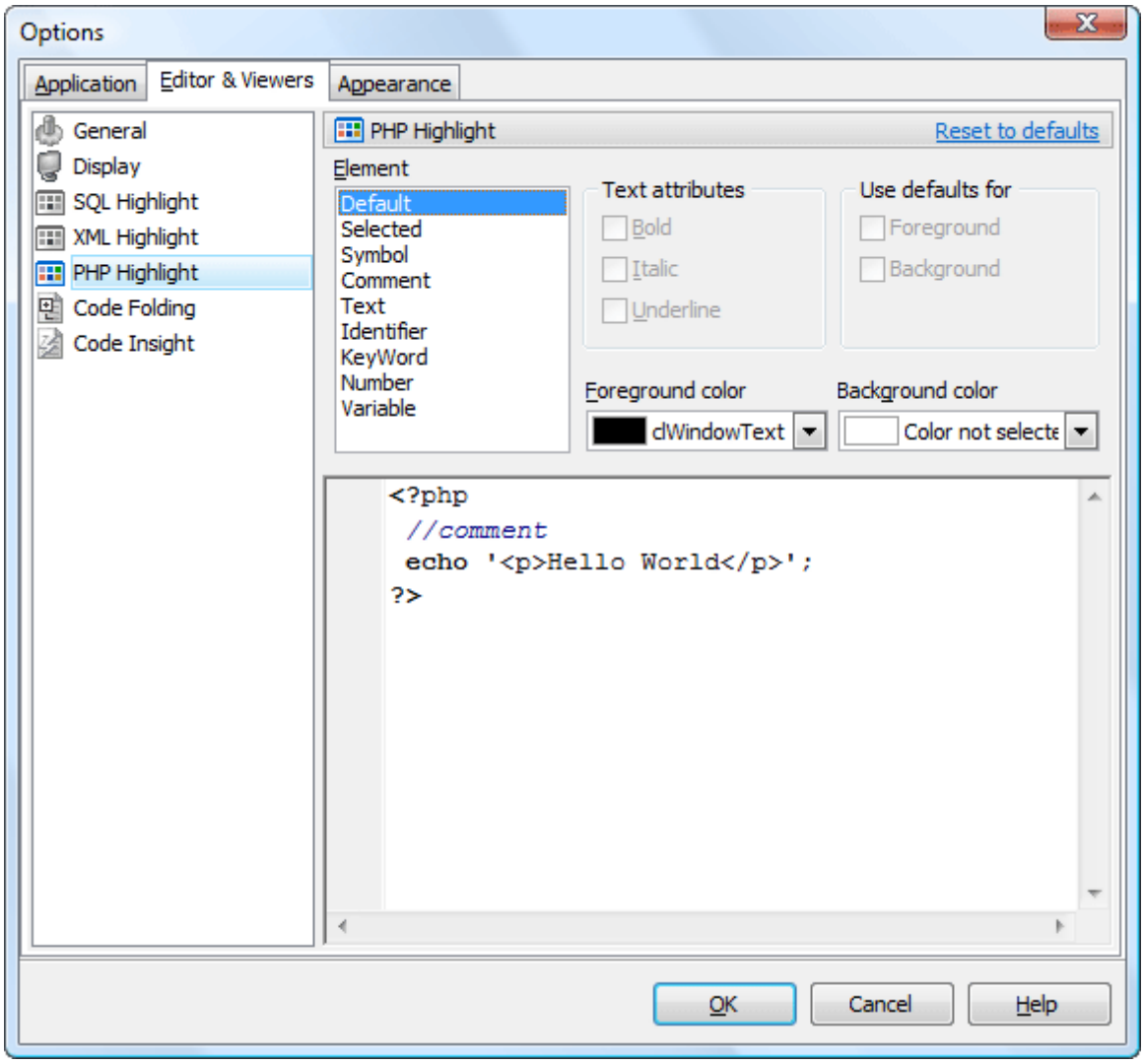
© 2024 SQL Maestro Group



7.2.5 PHP highlight

Select the text element from the list (e.g. Keyword, Comment, Identifier), and adjust its foreground color, background color and text attributes according to your wishes.

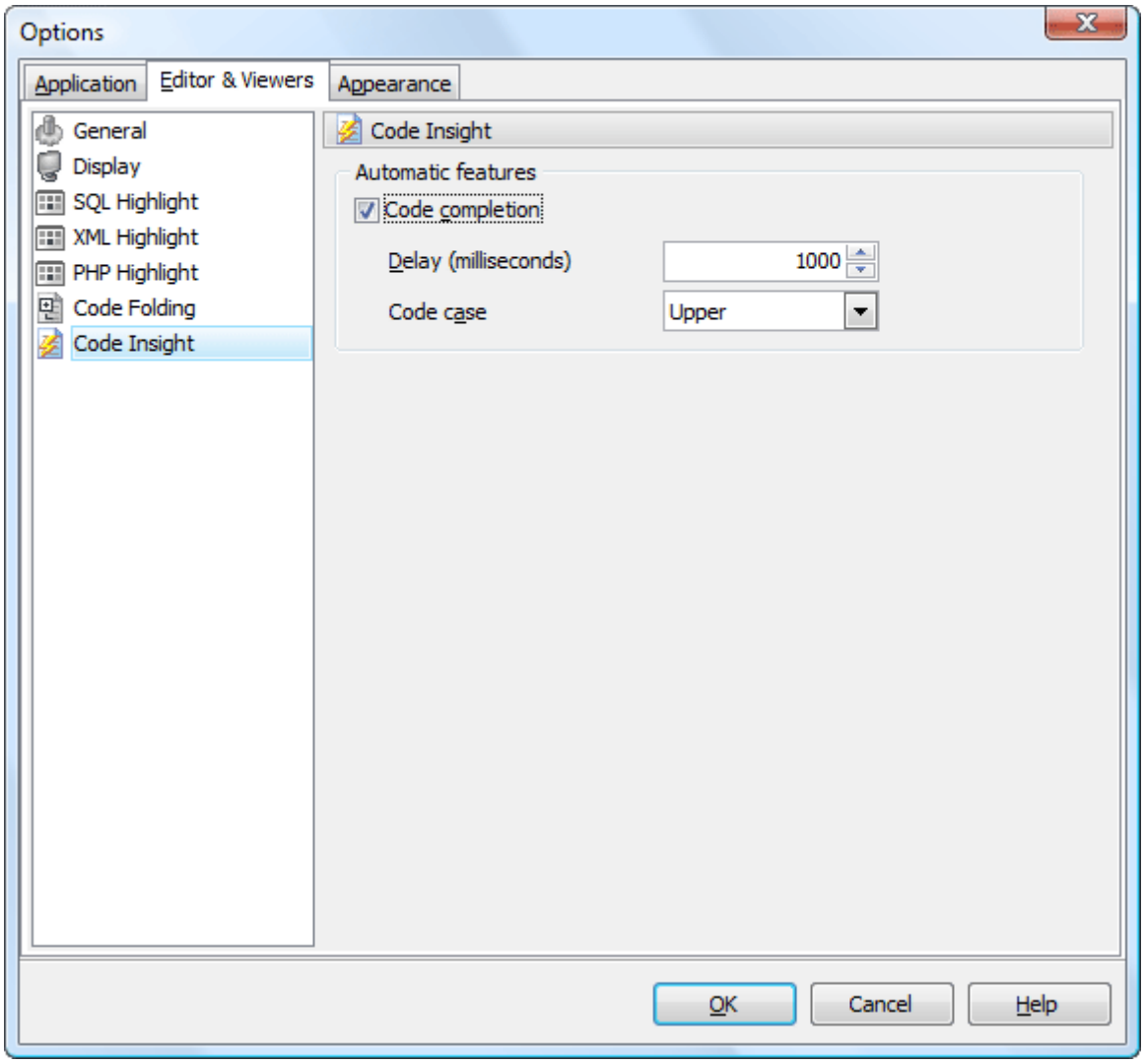
© 2024 SQL Maestro Group



7.2.6 Code Insight

You can disable/enable the code completion with the corresponding option and also set the time is appears as Delay, and case of words inserted automatically.

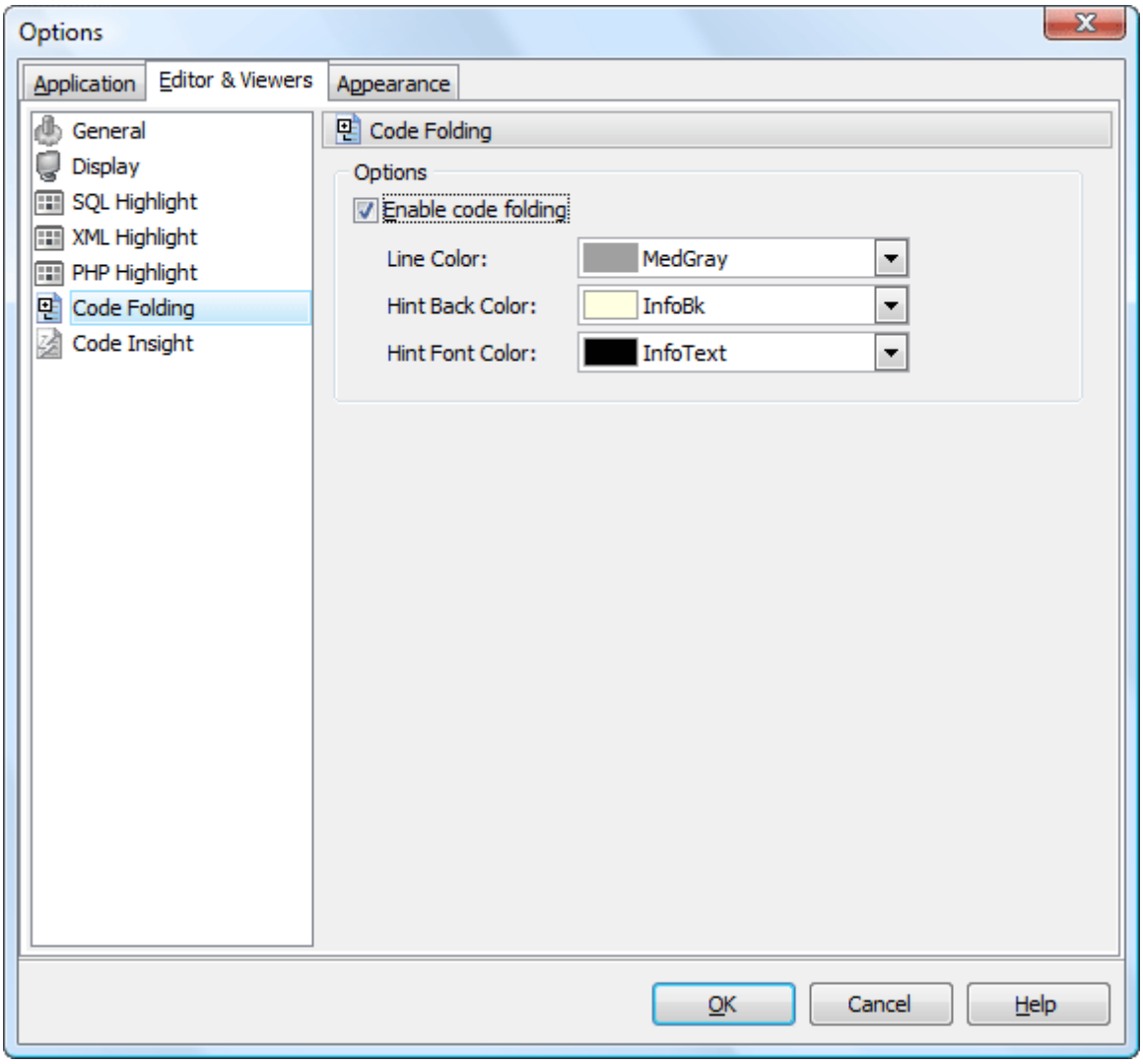
© 2024 SQL Maestro Group



7.2.7 Code Folding

The Code Folding item group makes it possible both to view the whole text and to divide it into logical parts (regions). Each part can be collapsed and extended. In extended mode the whole text is displayed (set by default), in collapsed mode the text is hidden behind one text line denoting the first line of the collapsed region.

© 2024 SQL Maestro Group



You can enable/disable code folding in SQL editors and viewers and customize the colors of its items.

© 2024 SQL Maestro Group

7.3 Appearance

The Appearance section allows you to customize the application interface style to your preferences.

Use the Scheme name box to select the interface scheme you prefer: *Office XP style*, *Windows XP native style*, etc. You can create your own interface schemes by customizing any visual options (Bars and menus, Trees and lists, Edit controls, Check boxes, Buttons, etc.) and clicking the Save As button. All the customized options are displayed on the sample panel.

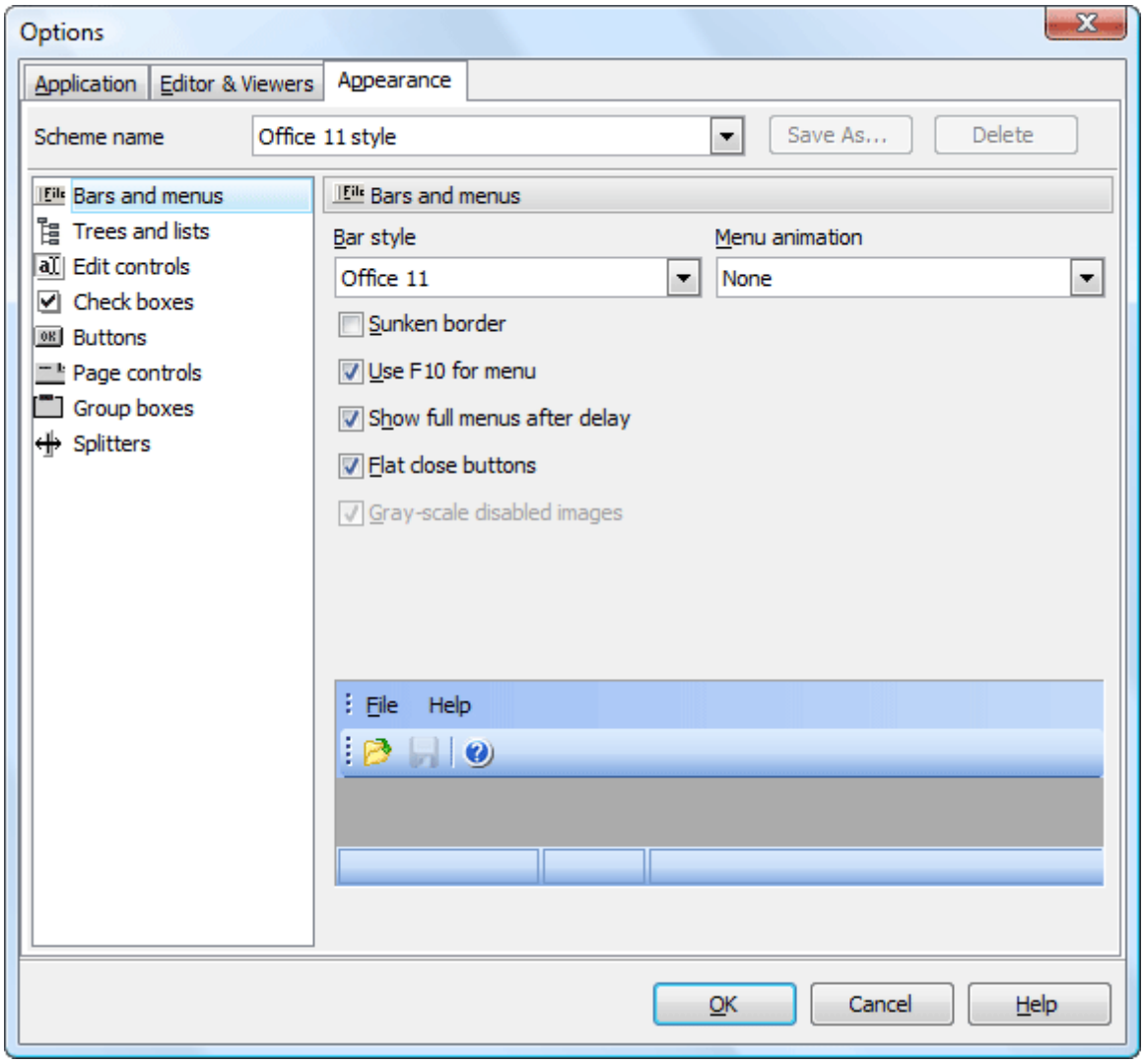
- Bars and menus 48
- Trees and lists 49
- Edit controls 50
- Check boxes 51
- Buttons 52
- Page controls 53
- Group boxes 54
- Splitters 55

7.3.1 Bars and menus

Use the Bars and menus item to customize SQLite Database Converter toolbars style and menus animation.

The item allows you to select Bar style and menu animation from the corresponding drop-down lists and to enable or disable such options as sunken border, F10 key for opening menu, viewing full menus after delay, flat close buttons, gray-scale images.

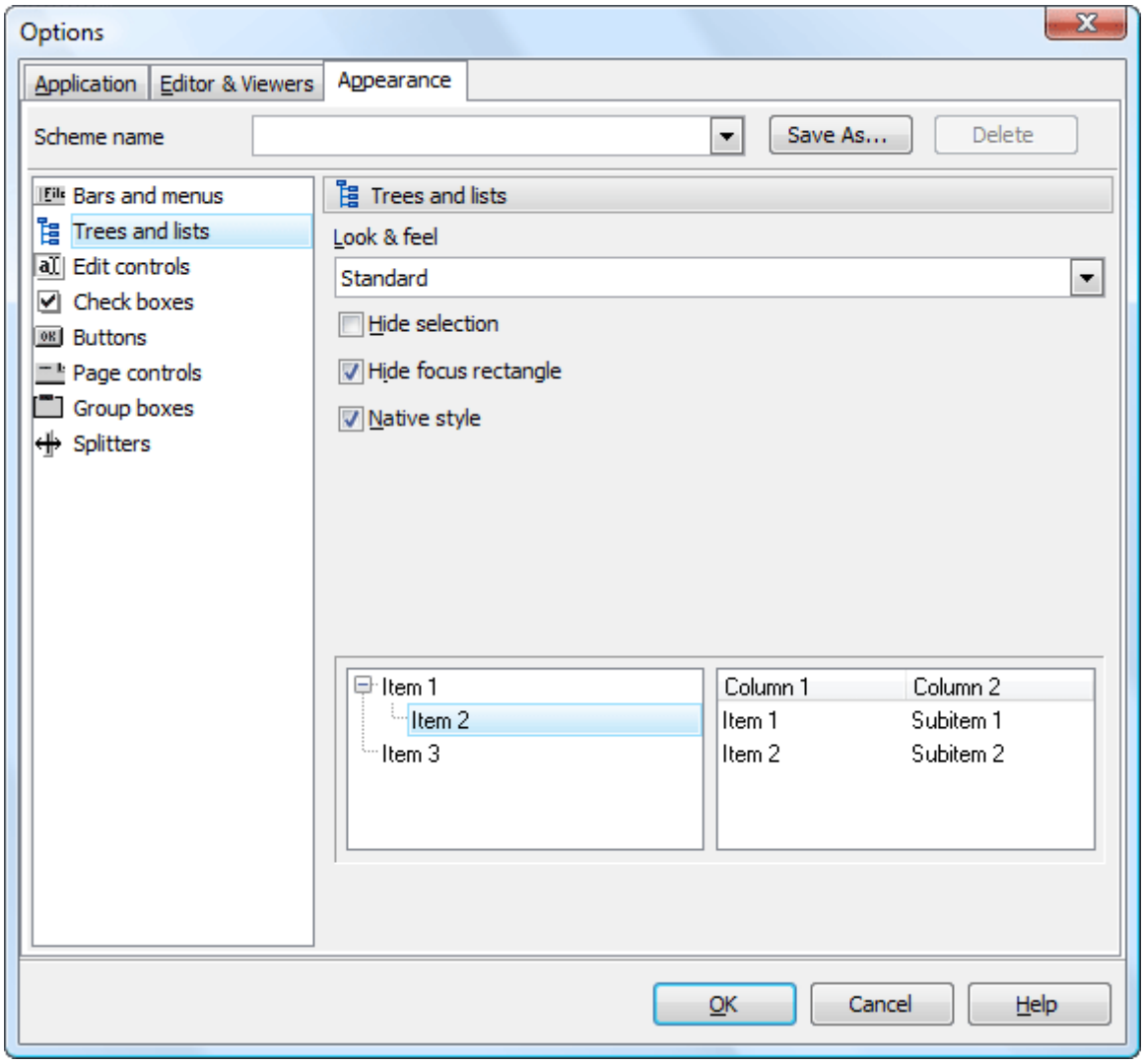
© 2024 SQL Maestro Group



7.3.2 Trees and lists

Use the Trees and lists item to select various tree view options. Use the item to select *standard*, *flat* or *ultraflat* styles, check or uncheck the *hide selection*, *hide focus rectangle* and *native style* options.

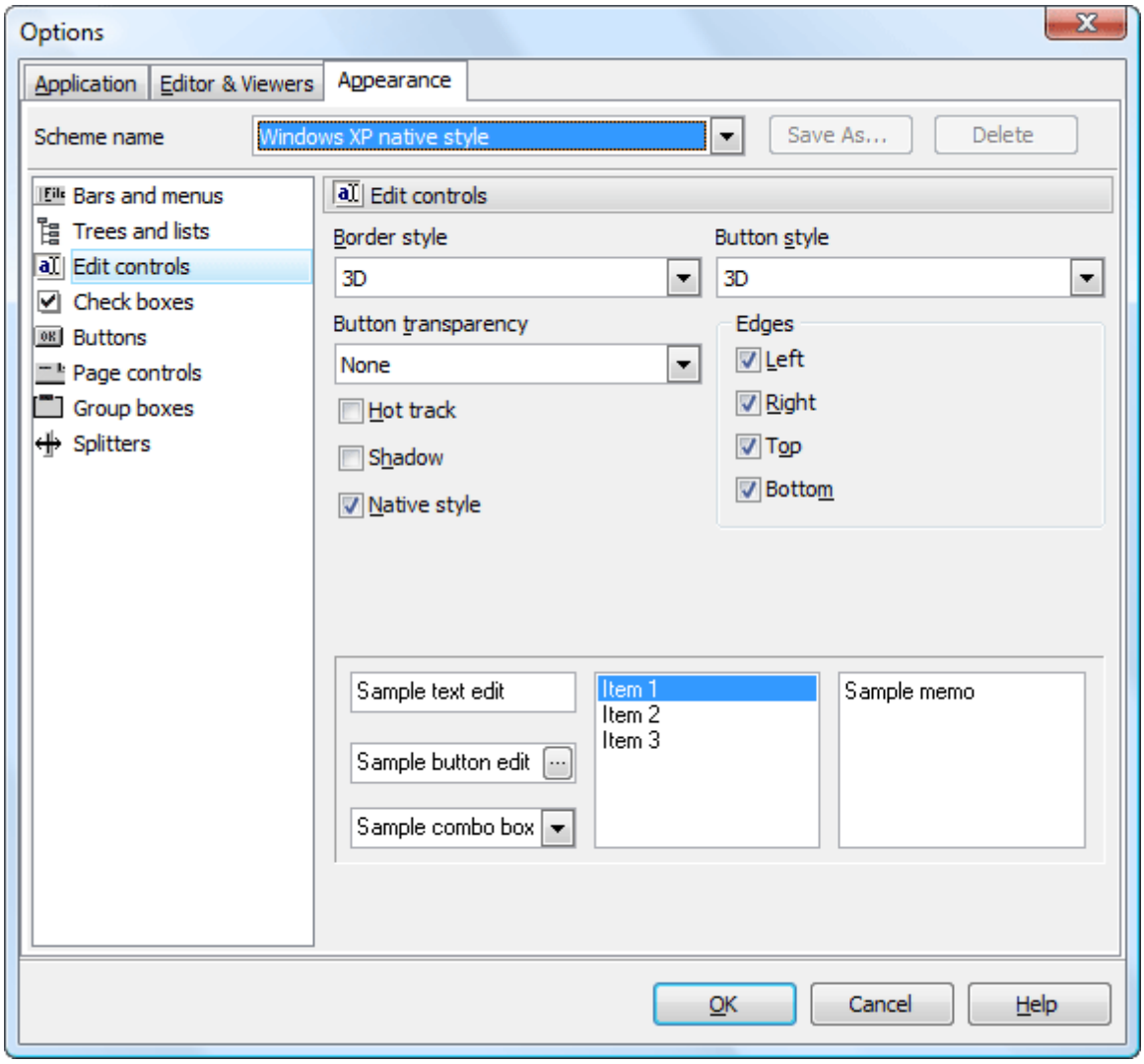
© 2024 SQL Maestro Group



7.3.3 Edit controls

Use the Edit controls item to customize the appearance of different SQLite Database Converter edit controls. The tab allows you to select the edit controls border style, button style and transparency, enable/disable hot tracks, shadows, native style and customize edges. It is also possible to define samples for the text edit, button edit and combo box controls.

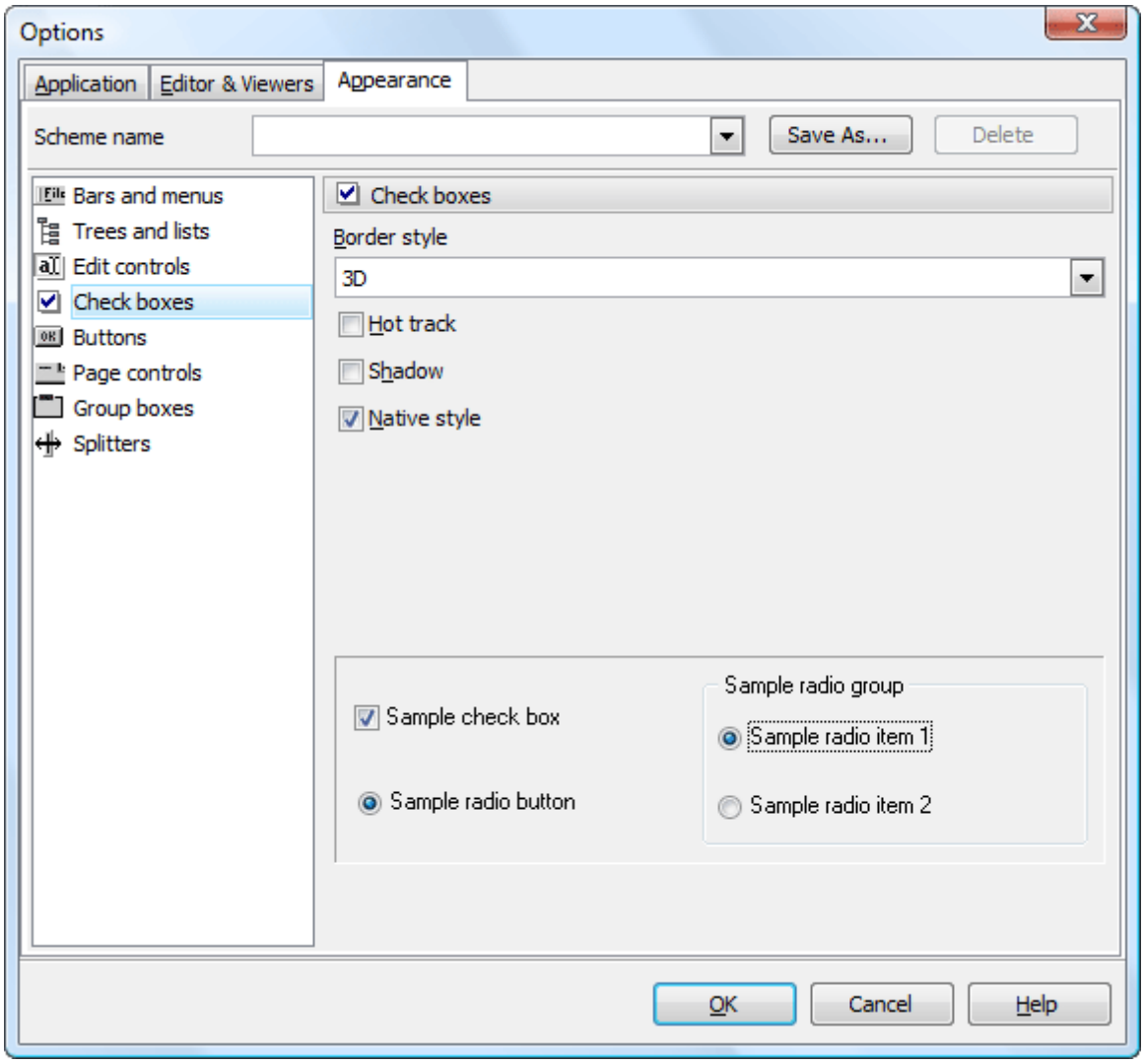
© 2024 SQL Maestro Group



7.3.4 Check boxes

The Check boxes item allows you to customize the appearance of check boxes and radio buttons. The tab allows you to customize the appearance of check boxes: set border style, enable/disable hot tracks, shadows, native style. It is also possible to define samples for check boxes and radio buttons.

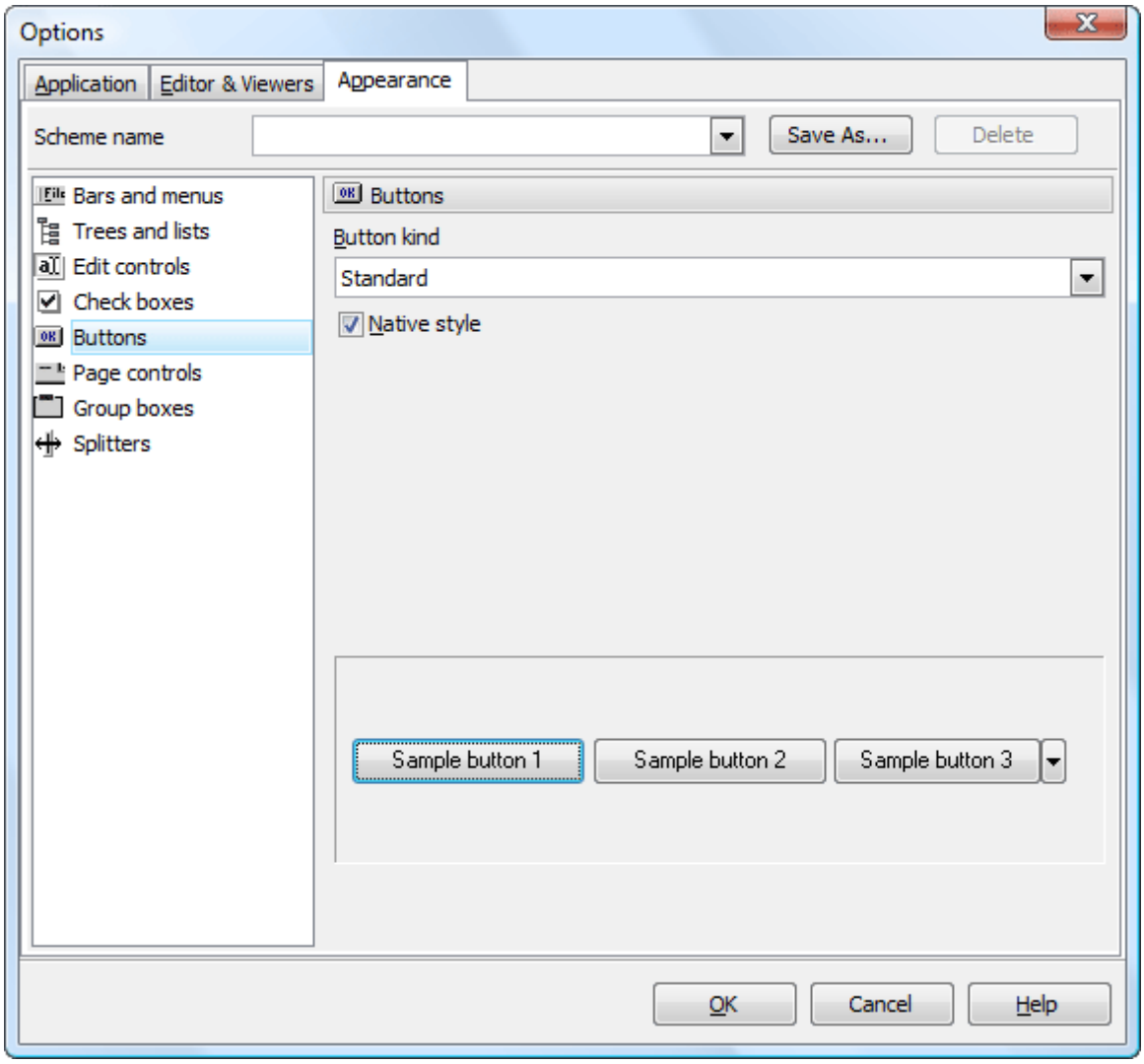
© 2024 SQL Maestro Group



7.3.5 Buttons

Use the Buttons item to customize SQLite Database Converter buttons. The tab allows you to adjust the appearance of buttons and define sample buttons as well.

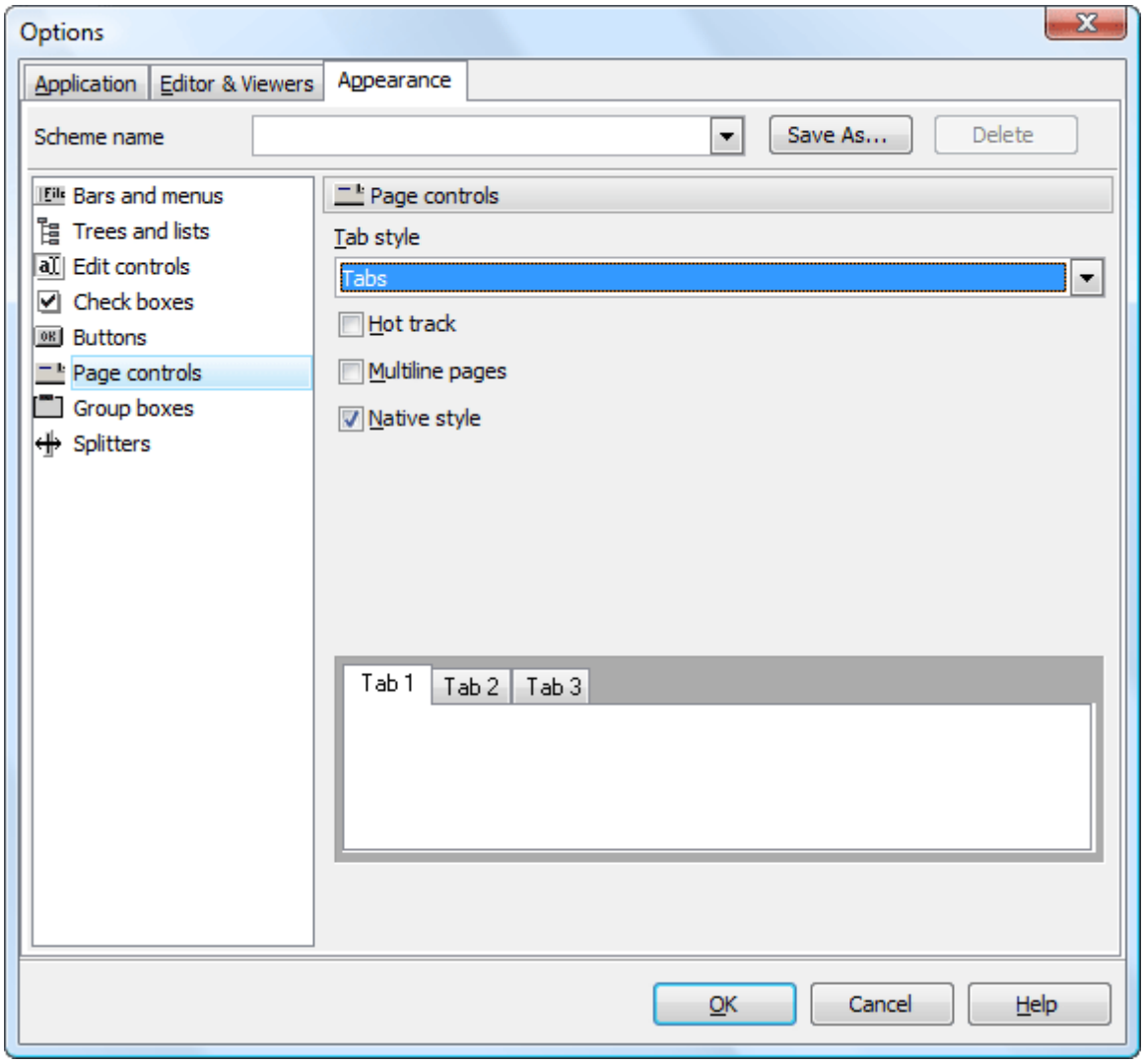
© 2024 SQL Maestro Group



7.3.6 Page controls

The Page controls item allows you to customize the style of all SQLite Database Converter page controls. The tab allows you to select tab styles, enable/disable hot track, multi-line pages and native style.

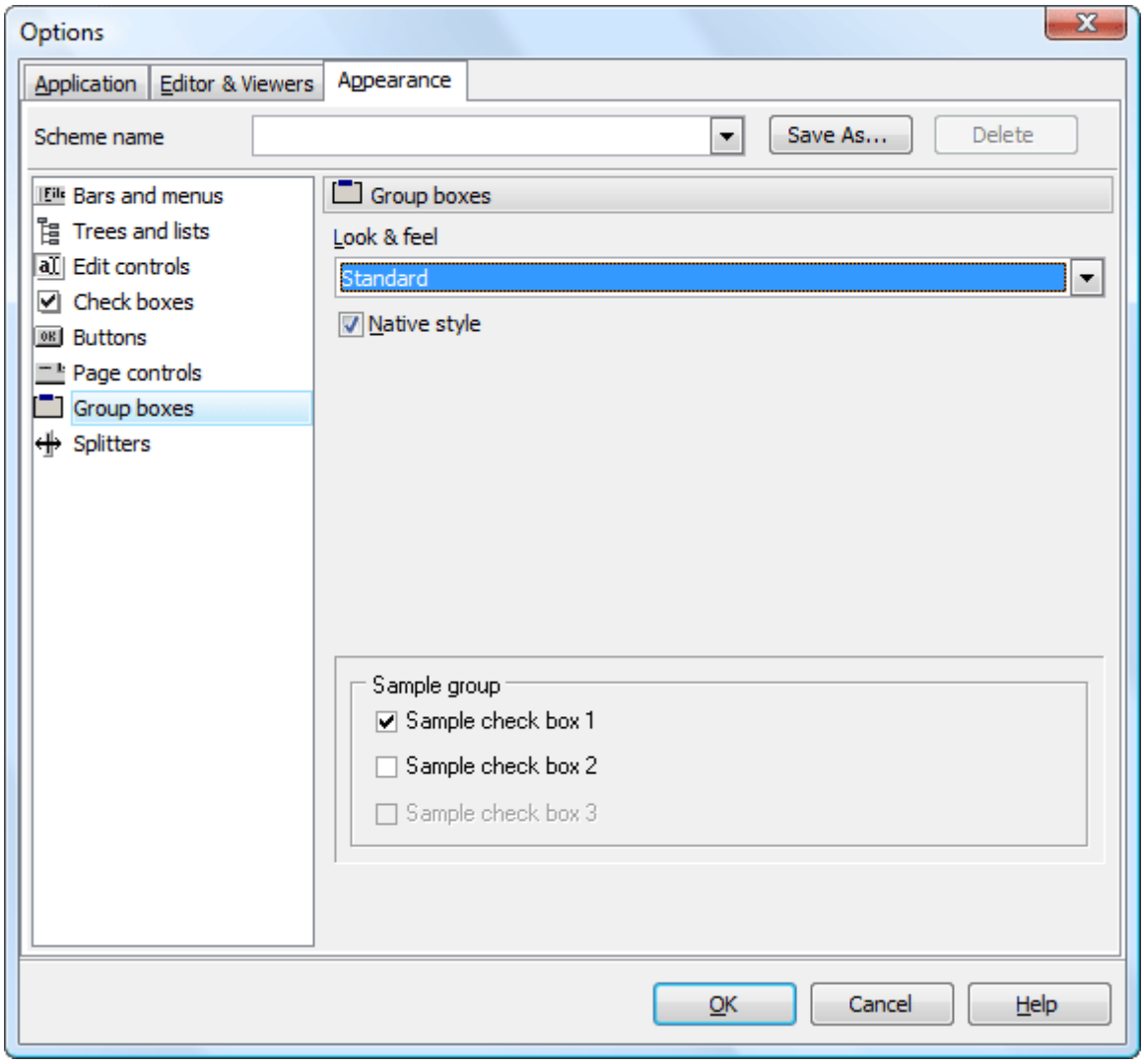
© 2024 SQL Maestro Group



7.3.7 Group boxes

Use the Group boxes item to customize all SQLite Database Converter group boxes according to your preferences. Use tab to apply styles for group boxes, enable/disable native style and define samples.

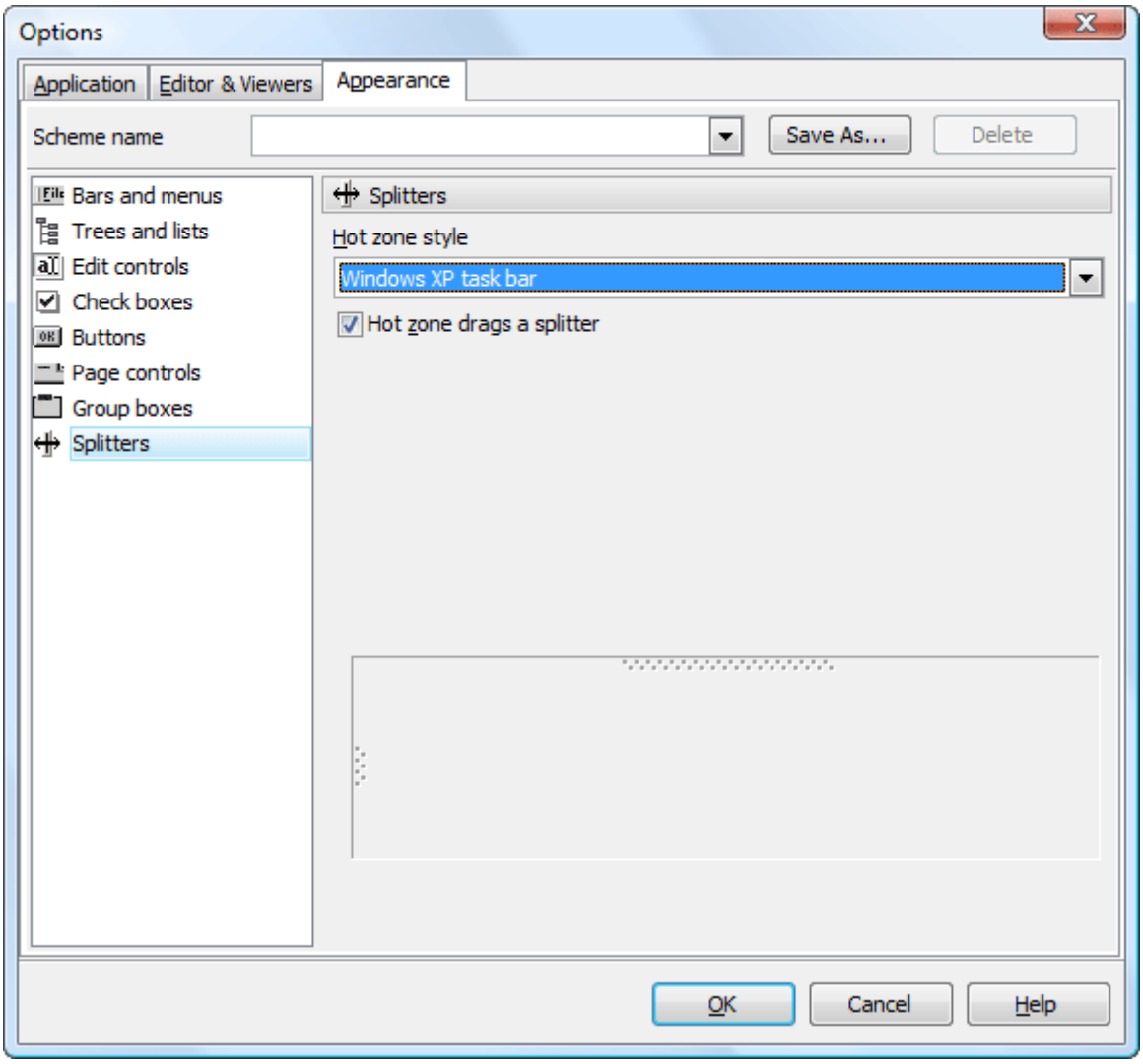
© 2024 SQL Maestro Group



7.3.8 Splitters

Use the Splitters item to customize all SQLite Database Converter splitters according to your preferences. Use the tab to select hot zone style (*Windows XP task bar*, *Media Player 8*, *Media Player 9*, *Simple* or *none*) and specify the Hot zone drags a splitter option.

© 2024 SQL Maestro Group



© 2024 SQL Maestro Group

Index

- A -

Appearance Options
- Bar and menus 48
- Buttons 52
- Check boxes 51
- Edit controls 50
- Group boxes 54
- Page controls 53
- Splitters 55
- Trees and lists 49

- C -

Converting options 34

- D -

Data type mapping 29

- E -

Editor & Viewer Options
- Code Folding 46
- Code Insight 45
- Display 41
- General 40
- PHP highlight 44
- SQL highlight 42
- XML highlight 43

EULA 5

- I -

Installation instructions 3

- L -

License Agreement 5

- N -

Naming conventions 30

- O -

Options 36
- Appearance 48
- Application 37
- Application confirmations 38
- Common 37
- Editor & Viewers 40

- P -

Purchase SQLite Database Converter 4

- R -

Registration 4

- S -

Setting databases 12
Source objects 28
Source tables for data import 32
SQLite Database Converter
- Command line options 27
- Connection options 24
- Getting Started 11
- Installation 3
- License agreement 5
- Overview 1
- Projects 26
- Registration 4
- System requirements 2

System requirements 2

- T -

Target objects 31

© 2024 SQL Maestro Group