



# [A]DB\_EXPLORER v.1.1

## User Guide

TiReks33



[Non-Commercial License Requires Attribution] This logo's original image was designed by Wannapik [URL: <https://www.wannapik.com/vectors/22949>].

## Description

ADB-Explorer – simple and handy C++ GUI for MariaDB/MySQL that automates standard DB-administrators routines, such as:

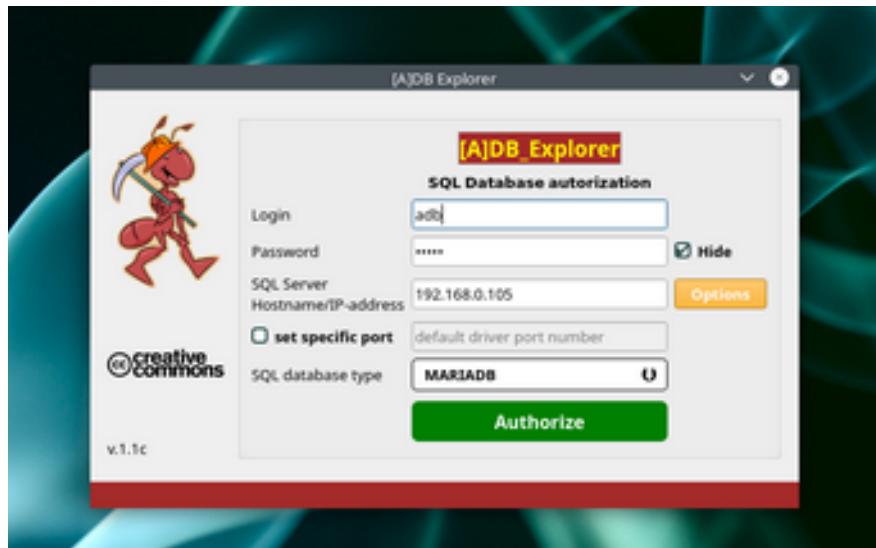
- creating/deleting user/roles, changing passwords;
- granting/revoking privileges to user/roles with different levels;
- creating, deleting databases, tables;
- sending SQL-queries, exporting results of queries;
- creating backups.

That sort of tasks can be easily solved with various instruments, that provides ADB-Explorer.

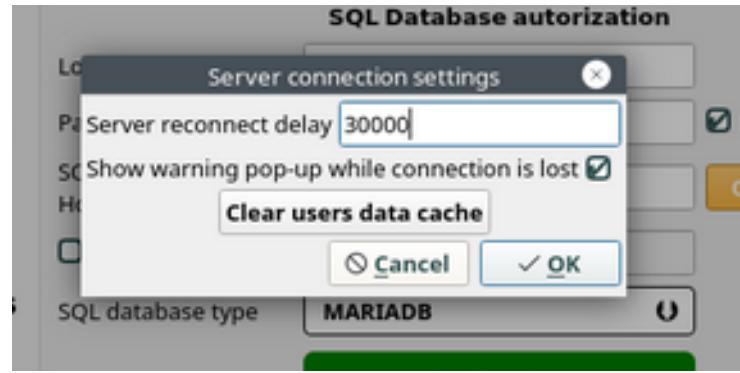
## Brief summary

### 1. Startup window

On startup (login) window you can enter your credentials, ip address/hostname of SQL server, set specific port (if necessary), driver, set specific options of connection.



Login window



### Additional connection settings

All program cache places in “**\$HOME/.adb-explorer**” directory (*hidden*).

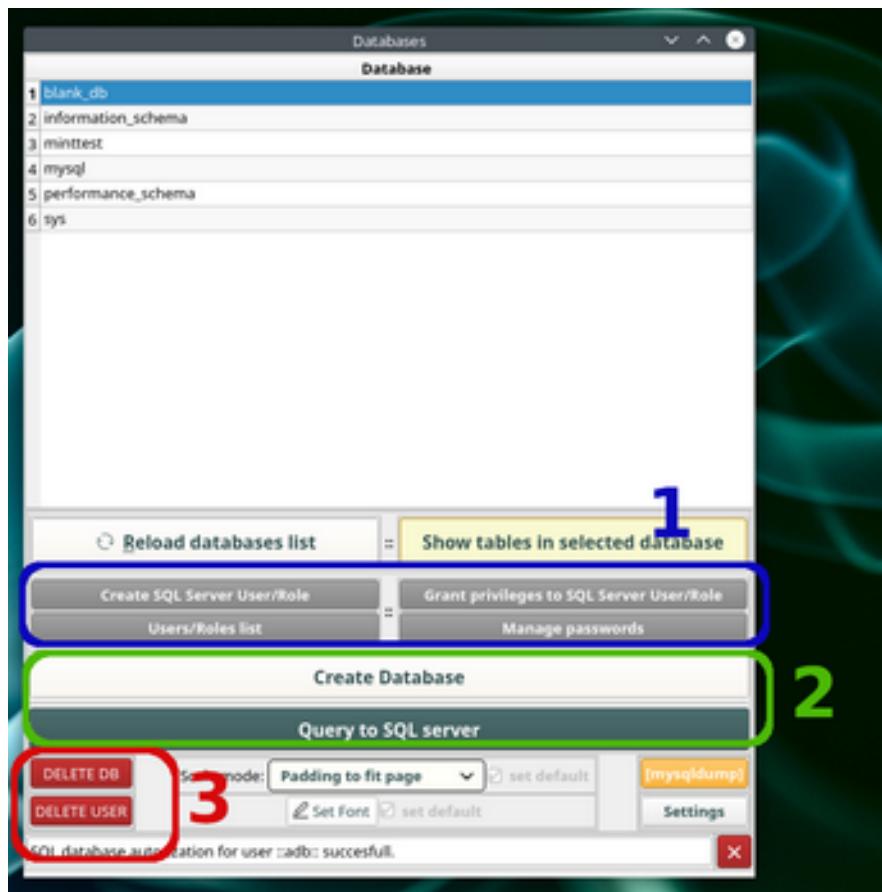
## 2. Databases

After SQL-user successfully login, «Databases» window are opened.

Main list showing current server’s databases that you can pick interactively.

Scaling, font size and other view parameters of list can be configured at bottom panels (Additional info about scale modes can be found in Appendix A).

A lot of base functionality with server’s databases and user management places here. Main functional buttons of this window can be separated to several functionality groups:



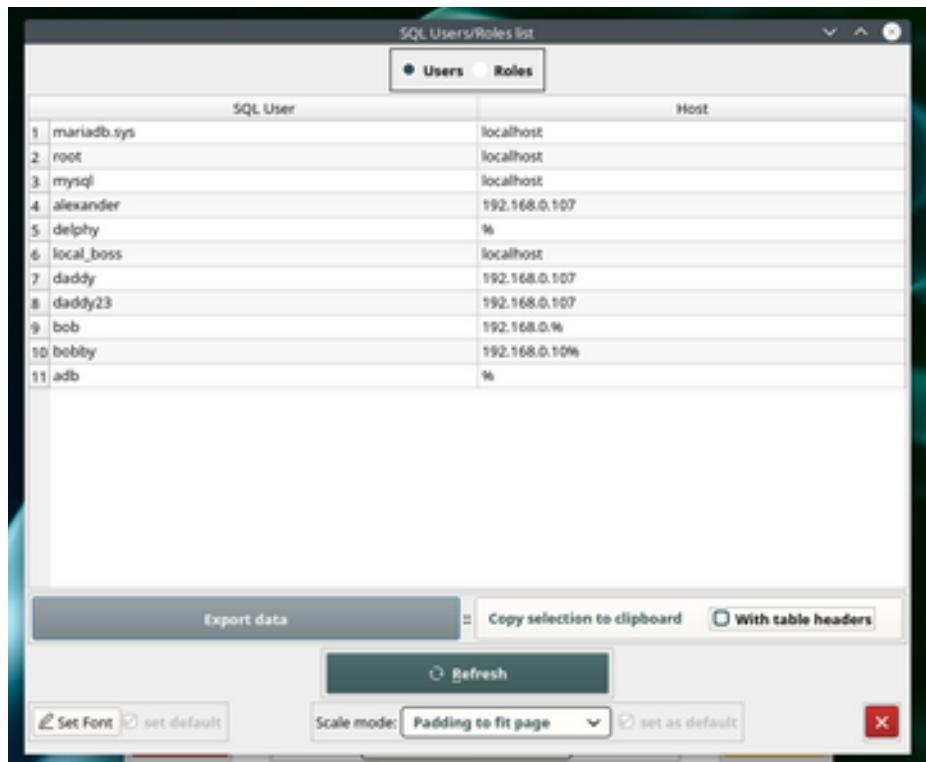
Functionality groups of databases window

- **Group 1 (blue)** – user/roles management group. This group is responsible for SQL user and roles administration: user/role creation, privileges and password management.  

**Note:** the logged user must have corresponding privileges to be able to use all the functionality of this group.
- **Group 2 (green)** – server queries group. This group response to custom user's queries to the SQL server and database creation.
- **Group 3 (red)** – “remove” group. This group responsible to removing databases, users and roles from the server.

## 2.1 User/roles management

### 2.1.1 User/roles list



The screenshot shows a software window titled "SQL Users/Roles list". At the top, there are two tabs: "Users" (which is selected) and "Roles". Below the tabs is a table with two columns: "SQL User" and "Host". The table contains 11 rows of data:

SQL User	Host
1 mariadb.sys	localhost
2 root	localhost
3 mysql	localhost
4 alexander	192.168.0.107
5 delphy	%
6 local_boss	localhost
7 daddy	192.168.0.107
8 daddy23	192.168.0.107
9 bob	192.168.0.%
10 bobby	192.168.0.10%
11 adb	%

At the bottom of the window, there are several buttons: "Export data", "Copy selection to clipboard" (with options "With table headers" and "Without table headers"), "Refresh", "Set Font", "Scale mode: Padding to fit page", "Set as default", and a close button.

### User/roles management

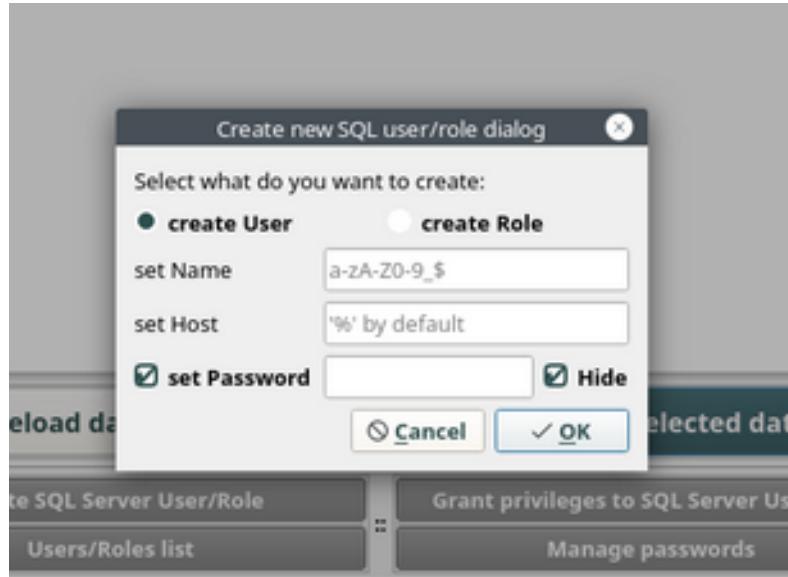
This window show current servers Users and Roles with hostnames/ip addresses, that corresponds to them.

Users/Roles groups can be switched at top panel.

You can select and copy enters to clipboard interactively, or by «Copy selection to clipboard» button (with or without table headers).

List can be exported to .csv format (Excel, Calc and other compatible software format) by «Export data» button.

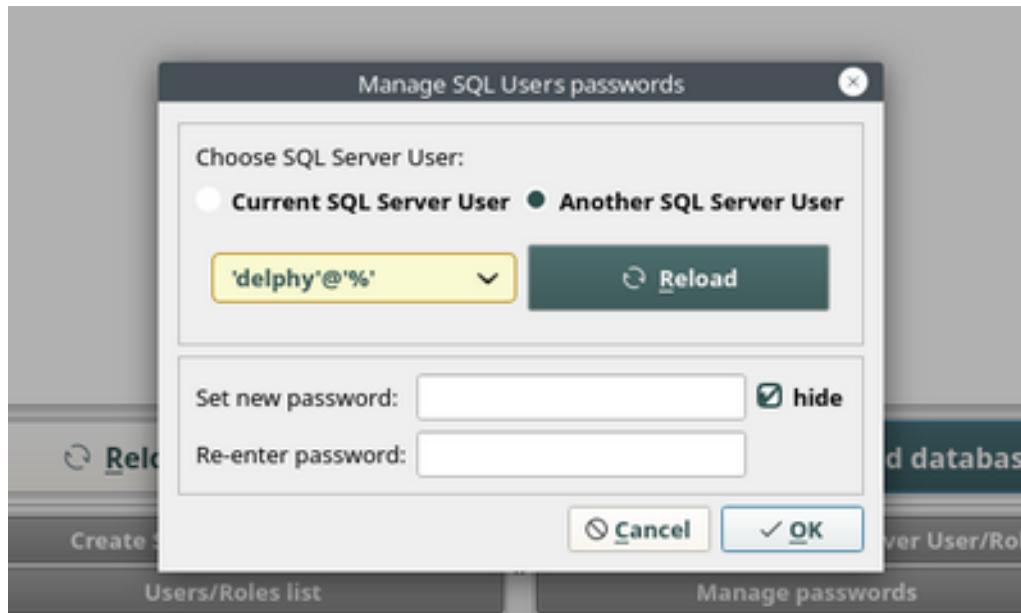
## 2.1.2 Create SQL User/Roles



**SQL User/Roles creation**

In this form you can create user or user's role entry and assign password to it. Toggle radio buttons “create User” and “create Role” to switch between specific forms.

## 2.1.3 Passwords management



**Passwords manager**

In passwords manager, you can set password for logged user or another user.  
If you want to set password for role, you must choose “Another SQL Server User”.

**Note:** you must have specific SQL privileges to switch password for another users/roles.

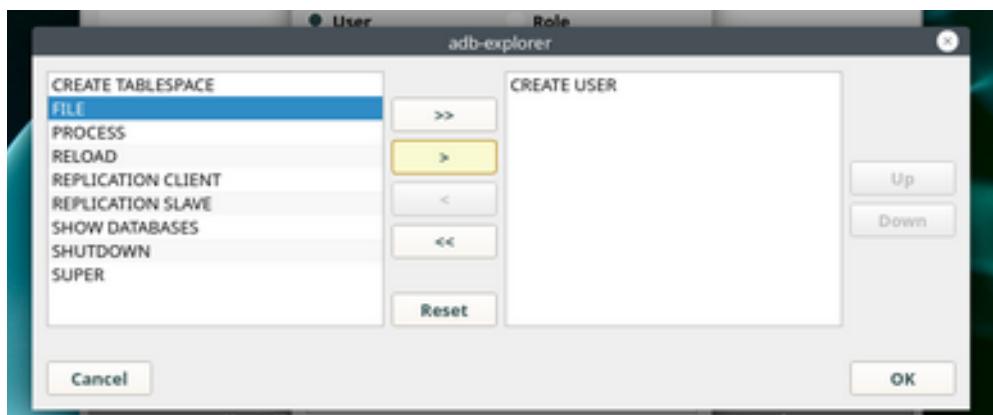
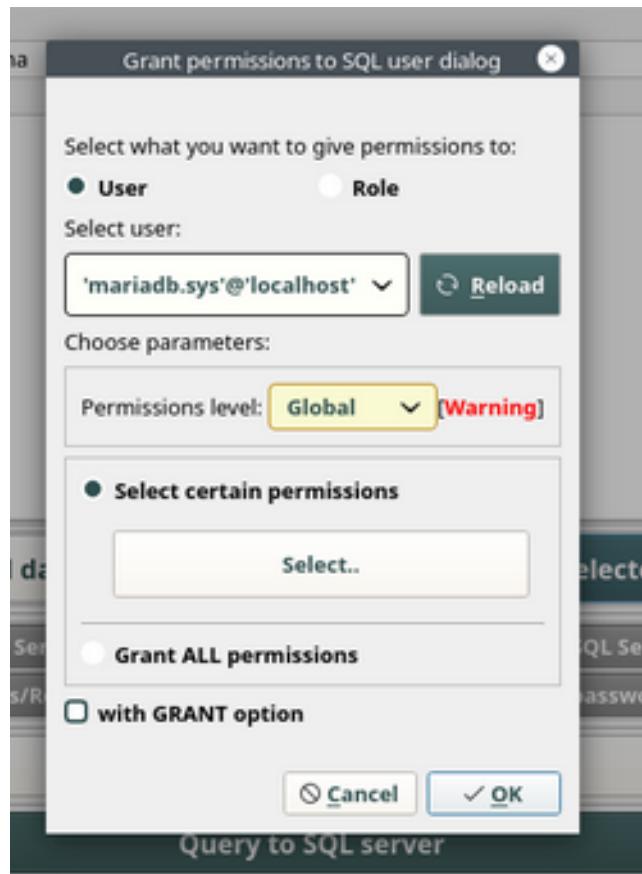
## 2.1.4 Grant SQL privileges

In this form you can choose, what type (level) of privileges you want to grant for selected user/role. There is 3 types:

- a) Global permissions;
- b) Database access permissions;
- c) Table access permissions.

In each of this you may choose certain privileges, or set “Grant ALL” button to provide User/Role full access on this level.

### a) Global permissions



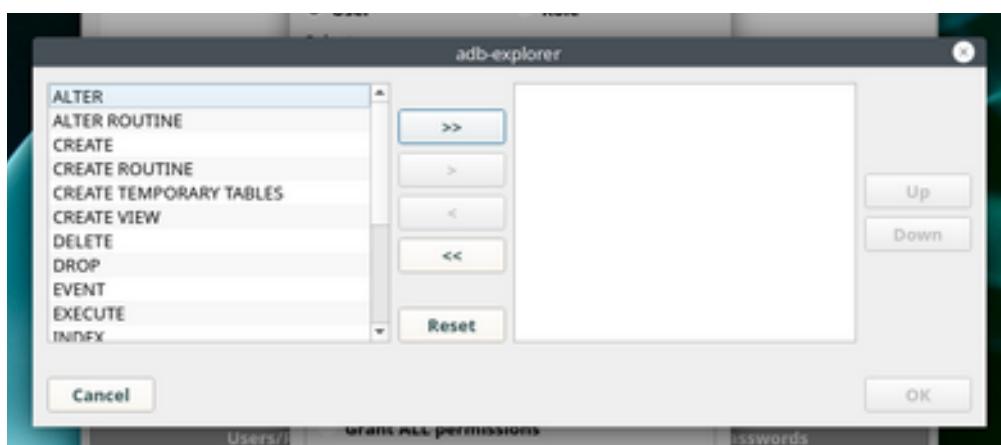
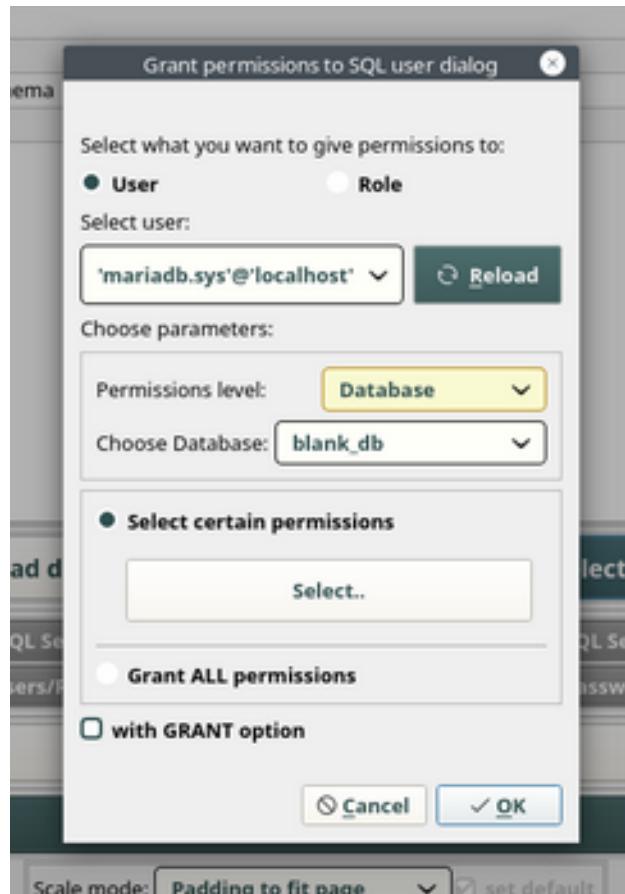
Granting permissions to SQL user/role (Global permissions)

This type refers to SQL server (non-specific to databases and tables) kind of permissions, such as “CREATE USER” privilege, or “SHOW DATABASES” to view the database list.

**WARNING!!! Granting “ALL permissions” on this group give your SQL User/Role full control on server (root access)!!!**

### b) Database access permissions

This type of permissions refers to specific database, such as selecting, tables creation, deleting (“DROP”) etc.

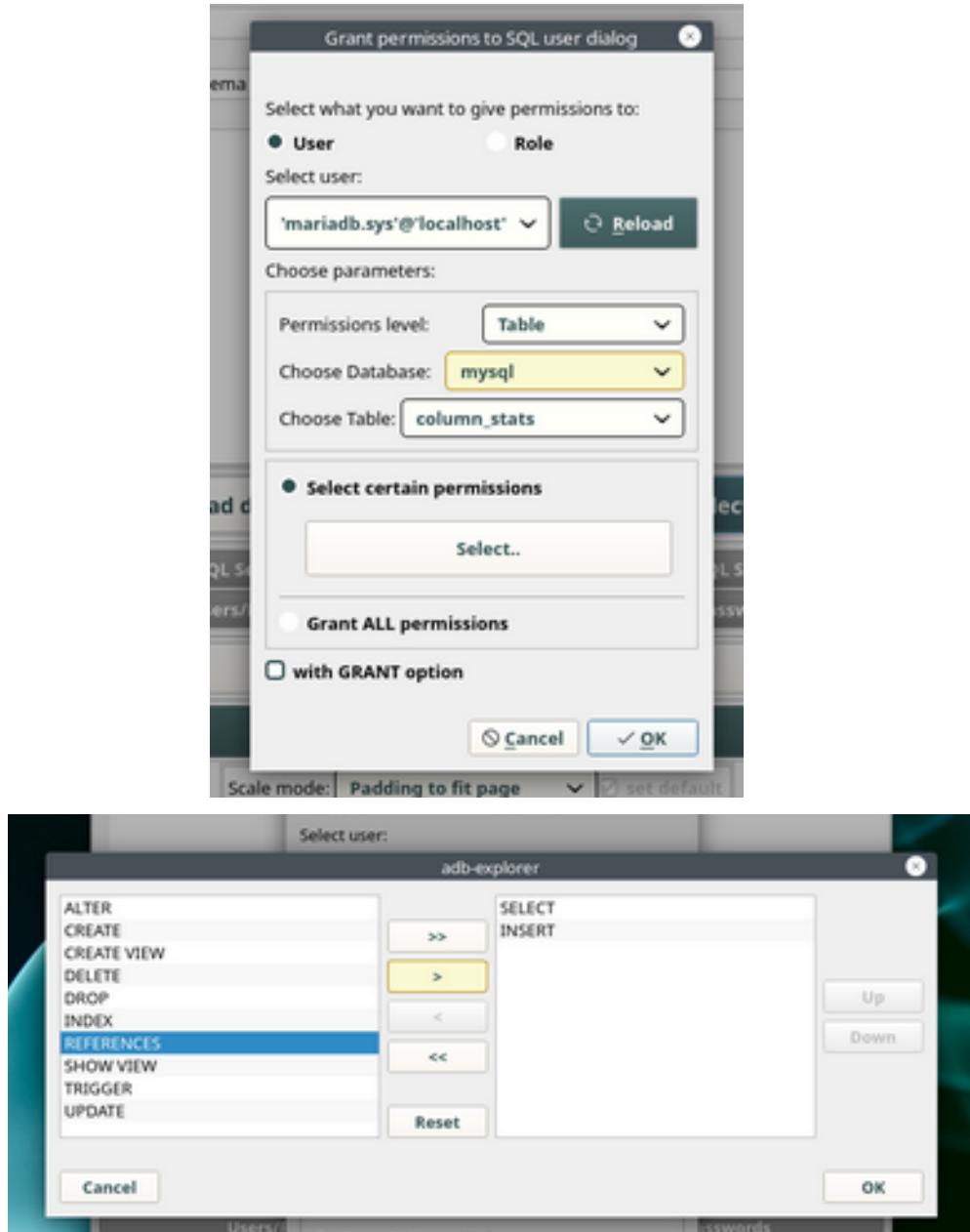


Granting permissions to SQL user/role (Database access permissions)

“Grant All” in this case provides full access on selected DB.

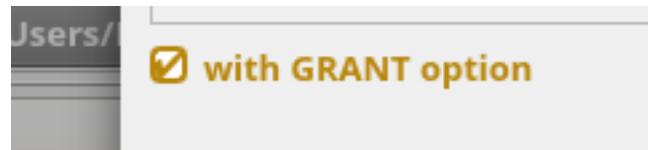
### c) Table access permissions

This “lowest” level provides privileges for operations on specific table in database, such as “CREATE TABLE”, “INSERT INTO TABLE”.



### Granting permissions to SQL user/role (Table access permissions)

“Grant ALL” in this level of permissions provides full access on specific table in database.

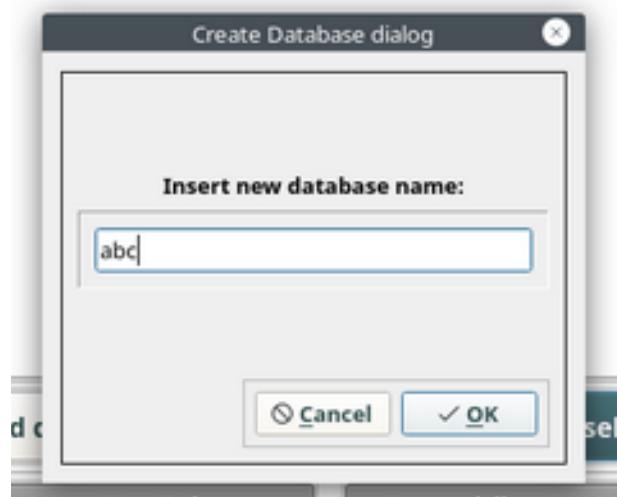


**GRANT option**

Functional check button “with GRANT option” means, that privileges, provided in this form for your user/role, also may have provided by this user/group for other user/s or role/s. **Be careful with this!!**

## 2.2 Server queries

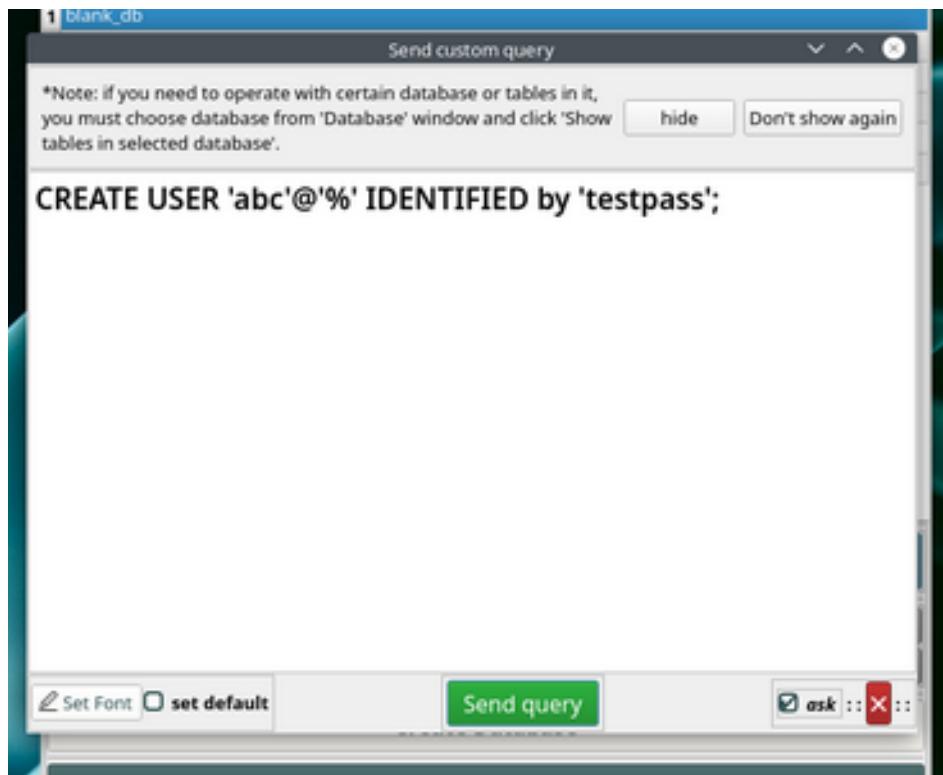
### 2.2.1. Databases creation



**“Create database” form**

Very simple form to create new database on server. If database with entered name already exists, corresponding warning message will show.

### 2.2.2. SQL-Server queries



Server user queries form

Custom user's queries to SQL server form. You can customize font in left-bottom panel.

**Note:** this form designed for non-database specific server queries. If you want provide database-specific query, you must accompany table names with database prefix (like showed on pictures below), or choose database first, and then perform operations in next “Tables” form.

```
5 F  
5 S select user,host from mysql.user;  
  
insert into minttest.table4(column1) values (11);
```

Database-specific queries with prefix

“*USE <DATABASE>*” queries will not work with that form.

## 2.3 “Removing” group

### 2.3.1 Remove database

Remove database by its name.

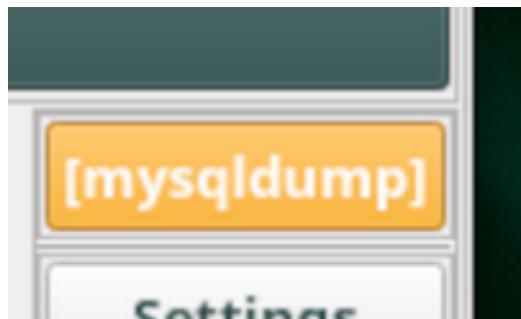
### 2.3.2 Remove user



Remove SQL User/Role dialog

There is a switcher in top panel for User/Role deletion.

### 2.4. Dumping

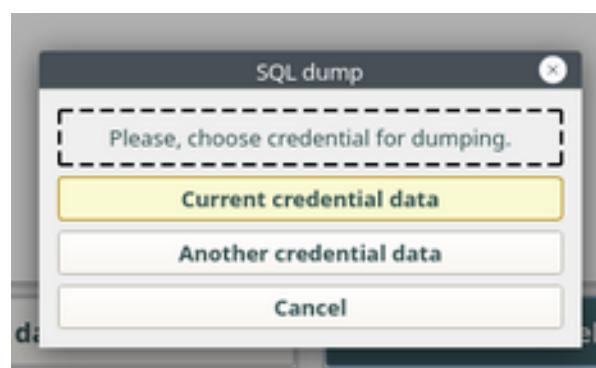


Dump functional button

Various dumping interactive customization settings have place here.

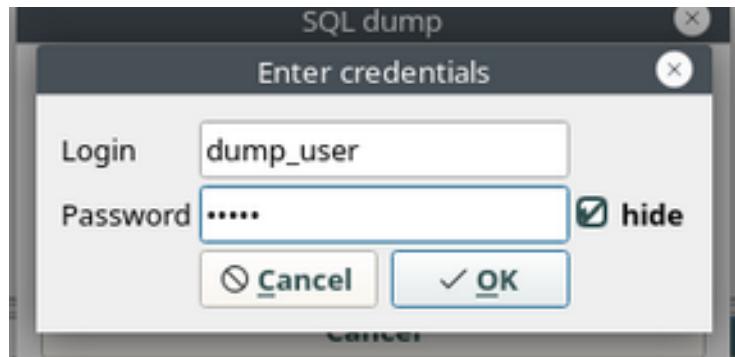
#### 2.4.1 User choose

In this window you must choose, what user will be proceed to dump your server information:



Dumping credentials

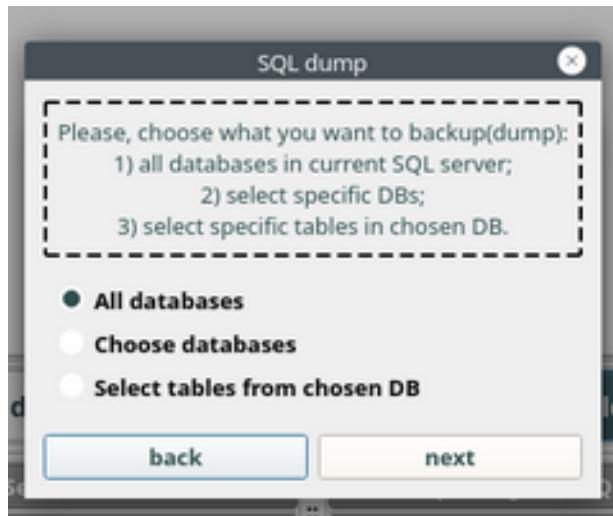
If you choose “Another credential data” via this dialog, you will be prompted to specify existing SQL-server user entry:



### Another credentials specification

Of course, for successful dumping, entered user must already have corresponding privileges for such operations.

#### 2.4.2 Dumping level



### Data level specifying

Next you have to choose, what data do you want to backup – full server dump, specific database/s, or only selecting tables in specific database. Related dialogs will be provided after “next” button clicking.

#### 2.4.3 Dump name

On last step you have to select name for your dump. Type postfix “\*.sql” will be added automatically if you not write it by yourself.

Default folder for dumps is “\$HOME/.adb-explorer/SQL\_dumps”.

**Note:** dump folder is hidden, so you must choose “show hidden” setting in your file manager. If this dump name already exist, program will offer you to overwrite existing dump-file.



### Dump name setup

After choosing name, wait for dumping process until finished.

## 2.5 Moving to “Tables” window

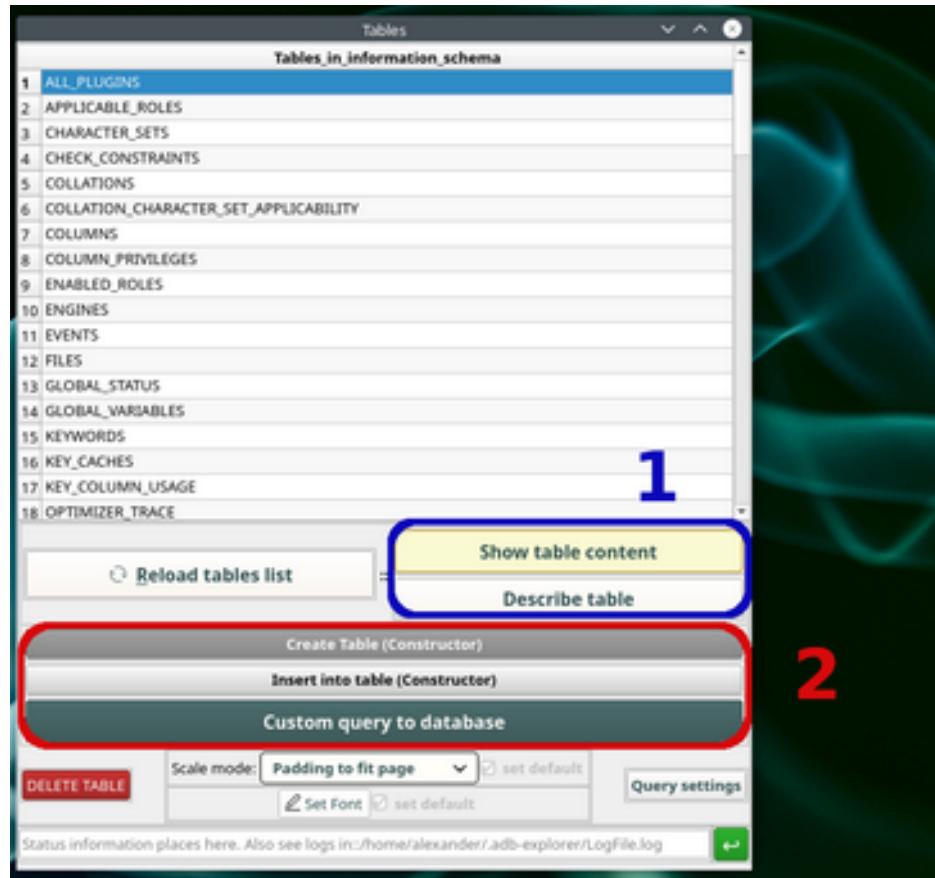
To move for the next window, specify database name interactively in upper list, and click on “Show tables in specific database”.

## 3. Tables

This window designed for tables-specific operations with DB.

2 functional groups have place here (showed on picture below):

- **Group 1(blue) – content display for specific tables**
- **Group 2(red) – custom SQL-queries management**



**Tables window**

### 3.1 Content display

#### 3.1.1 Show table content

This functional button provides displaying tables corteges (entries).

You can export all table content data in .csv file, or interactively select data partially and copy it (with table headers or without), for potential paste to .txt file or table document (like .xls).

Window customization, such as fonts and scaling mode, can be provided in bottom panels (additional info about scale modes can be found in Appendix A).

LUGIN_NAM	UGIN_VERSI	LUGIN_STATU	PLUGIN_TYPE	IN_TYPE	VER	UGIN_LIBRA	I_LIBRARY_VI	UGIN_AUTHN	UGIN_DESCRIP	UGIN_LICEN	DAD_OPTIO	GIN_MATUR	UGIN_AUTH_VI
1 binlog	1.0	ACTIVE	STORAGE ENGINE	100616.0				MySQL AB	This is a pseudo storage engine to represent the binlog in a transaction	GPL	FORCE	Stable	1.0
2 mysql_nati...	1.0	ACTIVE	AUTHENTI...	2.2				R.J.Silk, Sergel Golubchik	Native MySQL authentic...	GPL	FORCE	Stable	1.0
3 mysql_old...	1.0	ACTIVE	AUTHENTI...	2.2				R.J.Silk, Sergel Golubchik	Old MySQL-4.0 authentic...	GPL	FORCE	Stable	1.0
4 CSV	1.0	ACTIVE	STORAGE ENGINE	100616.0				Brian Aker, MySQL AB	Stores tables as CSV files	GPL	FORCE	Stable	1.0
5 MEMORY	1.0	ACTIVE	STORAGE ENGINE	100616.0				MySQL AB	Hash based, stored in memory, useful for temporary tables	GPL	FORCE	Stable	1.0
6 Aria	1.5	ACTIVE	STORAGE ENGINE	100616.0				MariaDB Corporation Ab	Crash-safe tables with MyISAM heritage. Used for internal temporary tables and privilege tables	GPL	FORCE	Stable	1.5
									Non-transactio... emline with				

Export data      Copy selection to clipboard       With table headers

Refresh      Scale mode: Padding to fit page       set as default

## Tables corteges displaying

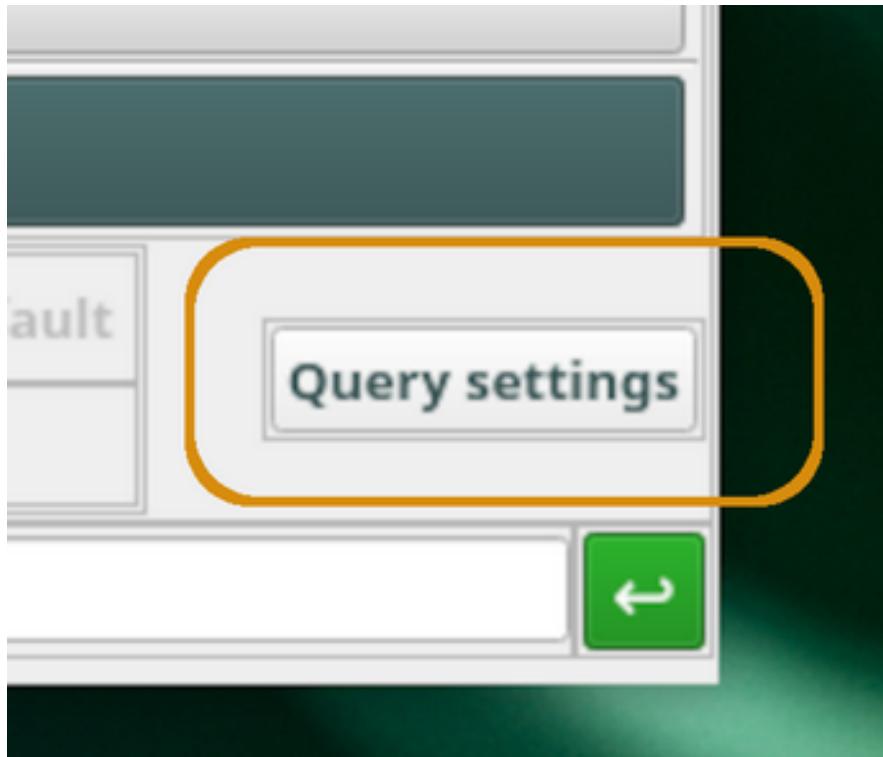
### 3.1.2 Table description

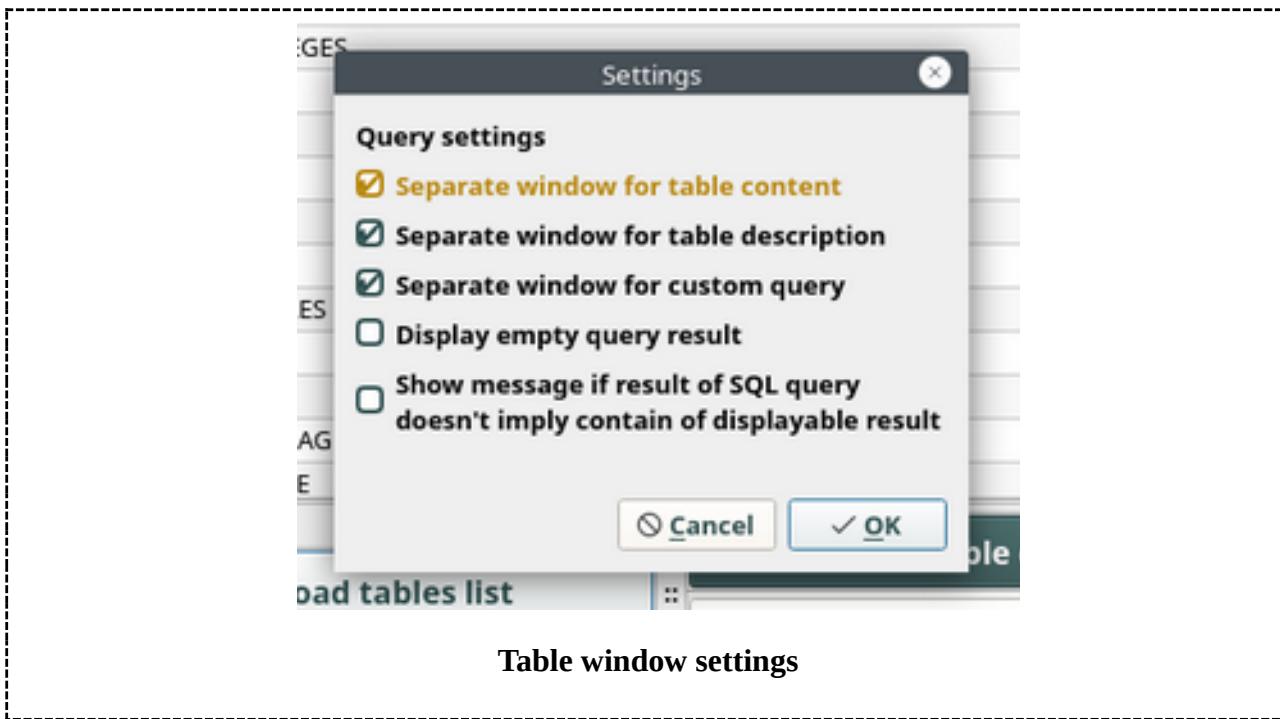
Similar to previous functional button, this button displaying provides detailed table description, such as data types and corteges constraints.

Field	Type	Null	Key	Default	Extra
1 PLUGIN_NAME	varchar(64)	NO			
2 PLUGIN_VERSION	varchar(20)	NO			
3 PLUGIN_STATUS	varchar(16)	NO			
4 PLUGIN_TYPE	varchar(80)	NO			
5 PLUGIN_TYPE_VE...	varchar(20)	NO			
6 PLUGIN_LIBRARY	varchar(64)	YES			
7 PLUGIN_LIBRARY_...	varchar(20)	YES			
8 PLUGIN_AUTHOR	varchar(64)	YES			
9 PLUGIN_DESCRIP...	longtext	YES			
10 PLUGIN_LICENSE	varchar(80)	NO			
11 LOAD_OPTION	varchar(64)	NO			
12 PLUGIN_MATURITY	varchar(12)	NO			
13 PLUGIN_AUTH_VE...	varchar(80)	YES			

### Table description displaying

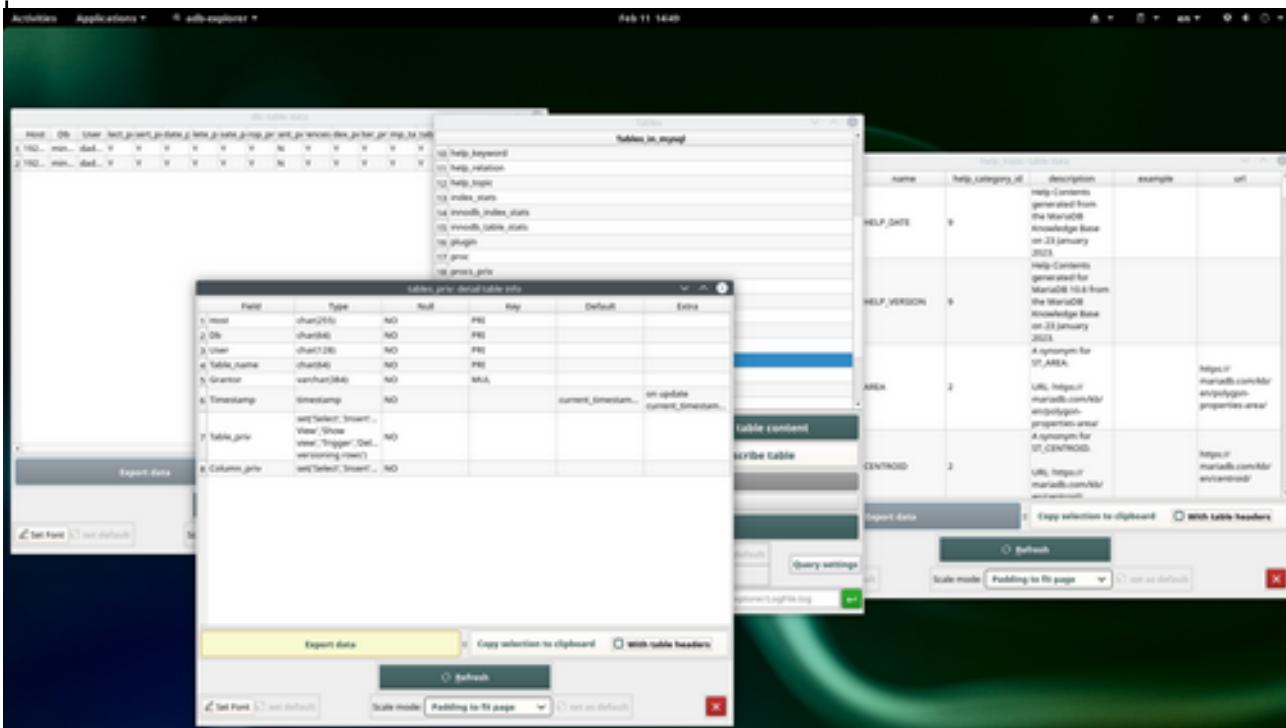
**Note:** you can switch the table content/description displaying mode, that the data will be displayed in same “Tables” window, or in new windows (by default), by uncheck the “Separate window for table content” and “Separate window for table description” settings:





### Table window settings

**Note №2:** for ease of interaction with SQL databases, many of functional forms (such as table content display or custom user query results) supports **multiply windows at the same time**; you only need to click on functional button again to call another window instance:



### Multiply query tables in the same time functionality example

Clicking on main window/sub-window will shift focus to that window.

Some program's functionality doesn't support this.

## 3.2 Custom SQL-queries management

This group includes user-friendly constructors for table queries.

### 3.2.1 “Create table” constructor

#### 3.2.1.1 Table name choose

Provide name for new table and click “next”:

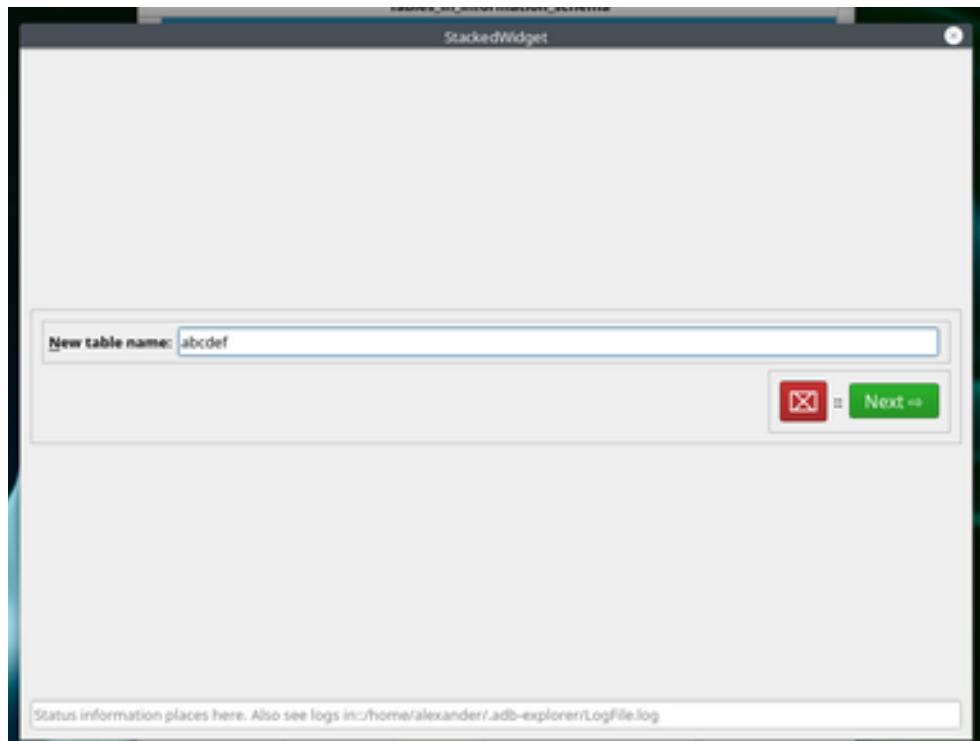
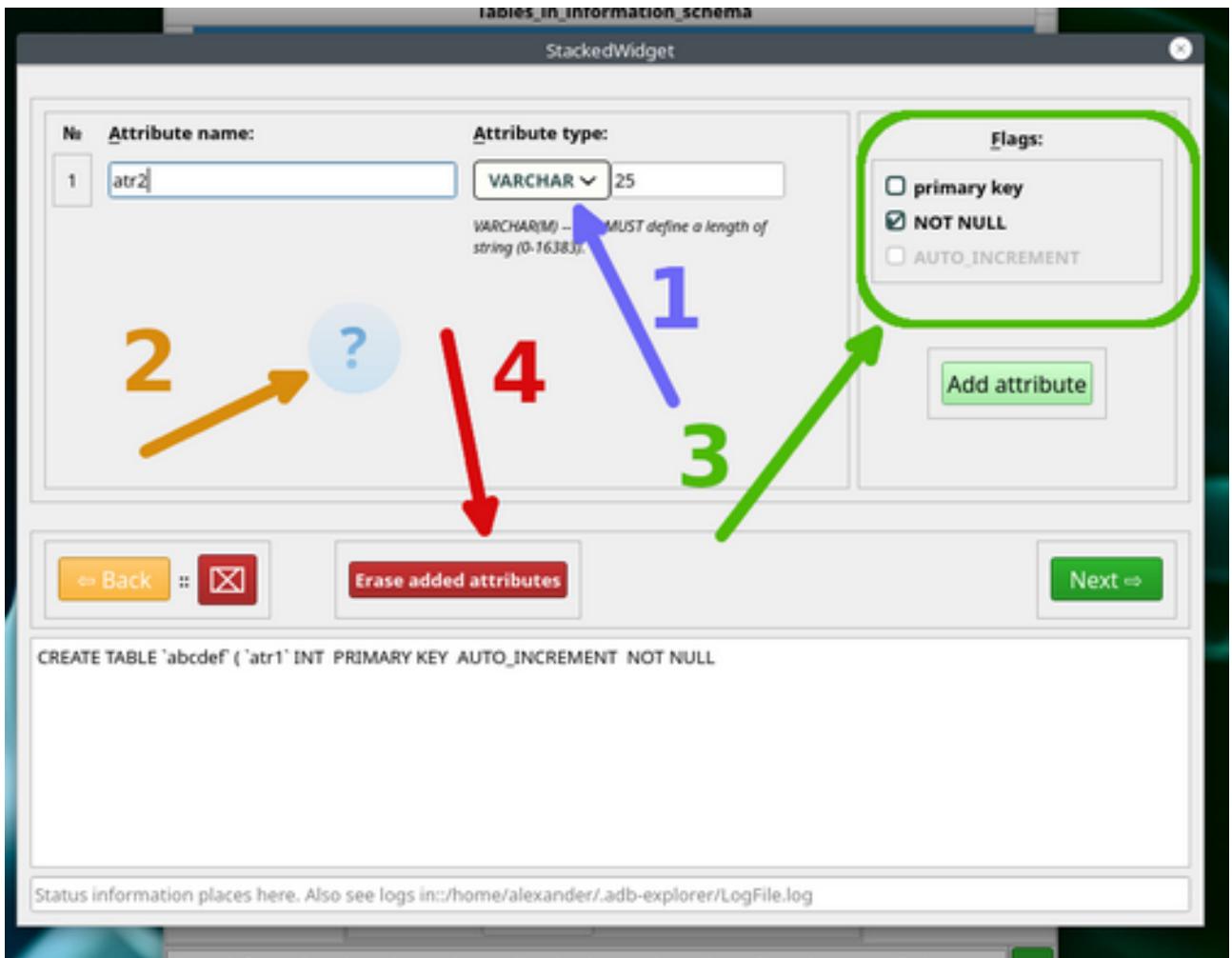


Table constructor window

#### 3.2.1.2 Add attributes (columns) for new table



### Adding attributes to new table

Here things have to be described more detailed.

First, set name for new attribute. Secondly, choose type for new attribute (“1” in picture above). For detailed info about main data types in SQL DB’s, click on “?” button (“2”). Next, set needed flags for your attribute (“3”) and click on “add attribute”.

After you add new attribute, in a bottom panel you will have see new entry/s in future emerging query that will send to SQL-Server after you finished setup your table in constructor.

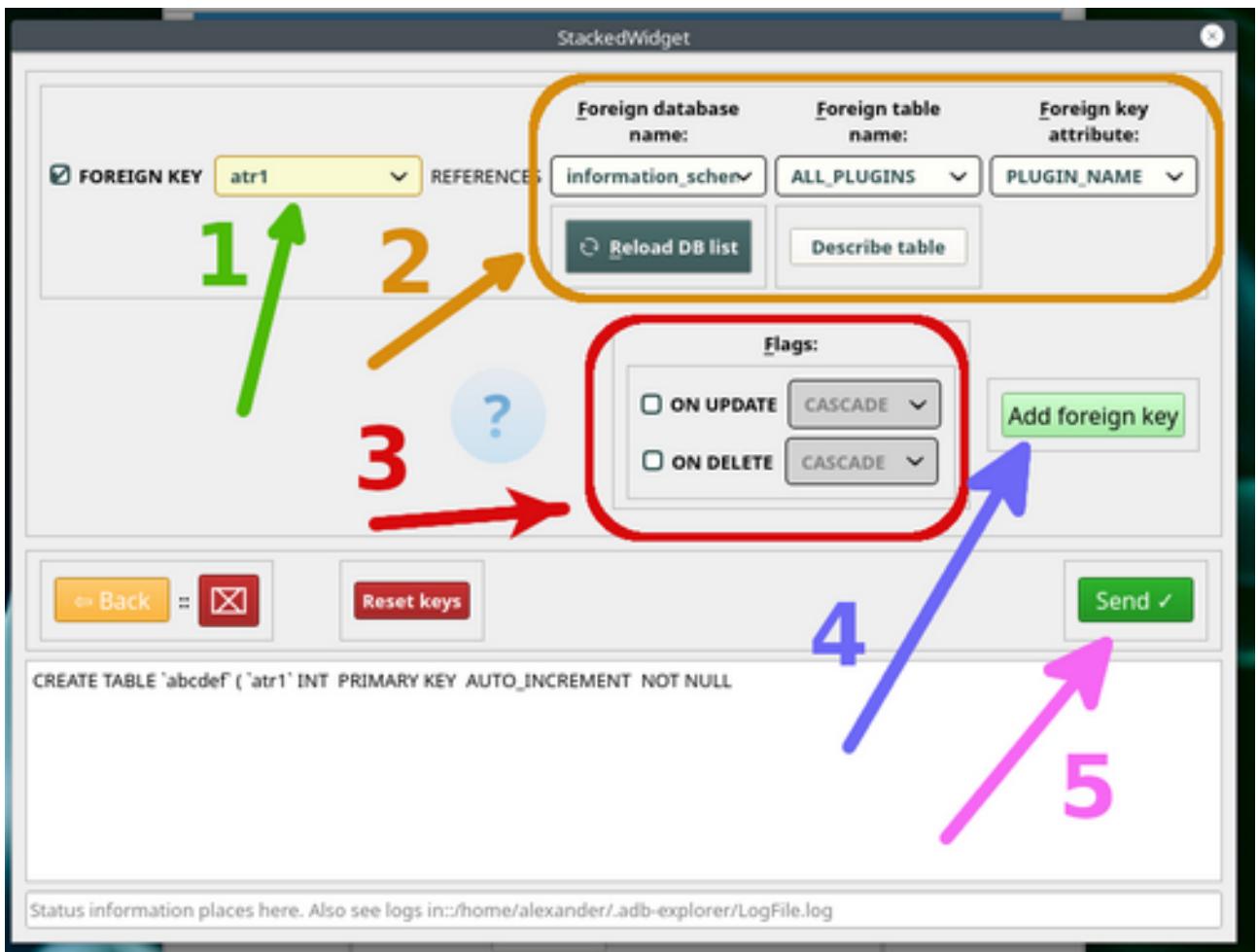
If something goes wrong and you will want to erase newly added entries, click “Erase added attributes” (“4”), and last changes will be reverted.

After attributes added, click “Next” to proceed to the next form.

#### 3.2.1.3 Foreign keys set up

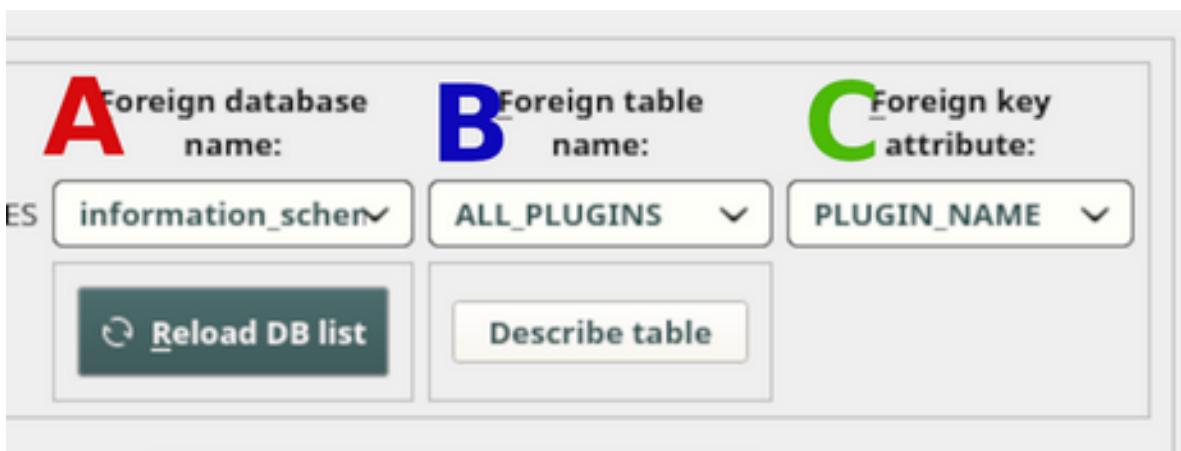
In third window you can optionally specify foreign keys for your new table; choose options for actions when foreign keys are updating or removing; and finally send your query to the SQL-Server.

Let’s review this things more detailed.



### Foreign keys setup in table constructor

First, we have to check, what attribute we will use as foreign key for table (“1” in picture above). Secondly(“2”), we have to choose, what foreign [database(A).table(B).attribute(C)] chain this key will be referred to:



If you do not remember, what datatype of foreign table attribute used, or you need other detailed table info, click on “Describe table” button.

Next, you may (optionally) specify actions for your key in foreign attribute's update and deletion("3").

Add your foreign key reference by clicking "Add foreign key" button("4").

Once you complete table setup, click "Send" for sending generated query to SQL-Server("5").

### 3.2.2 Insert into table ("Create Tuple") constructor

This constructor is specially designed to simplify a routine of "INSERT INTO" queries for multiply inserts with user-friendly interactive form:

The screenshot shows a window titled "Form" with a dark header bar. The main area is divided into five horizontal stages, each numbered 1 through 5.

- Stage 1:** Shows the SQL command "INSERT INTO" followed by a dropdown menu set to "table6". A note "<-- Choose table to insert" is next to it, and a green "Update tables list" button is to the right.
- Stage 2:** Shows the opening parenthesis of a tuple "( Add attributes )". A note "<-- Choose attributes to add from table" is to its right.
- Stage 3:** Shows the first attribute "digit\_atr, str\_atr" selected in a grey box. To its right is a note "<-- Chosen attributes" and a "Describe table" button.
- Stage 4:** Shows the value "7, "seven"" entered into the tuple. To its right is a note ")" and an "Add tuple" button.
- Stage 5:** Shows the total values added: "values added: 2" and a red "Reset" button. At the bottom right are three buttons: a red "X" (Close), a grey double-dot separator "...", and a green "OK" button. Below the buttons is a status message: "Status information places here. Also see logs in:/home/alexander/.adb-explorer/LogFile.log".

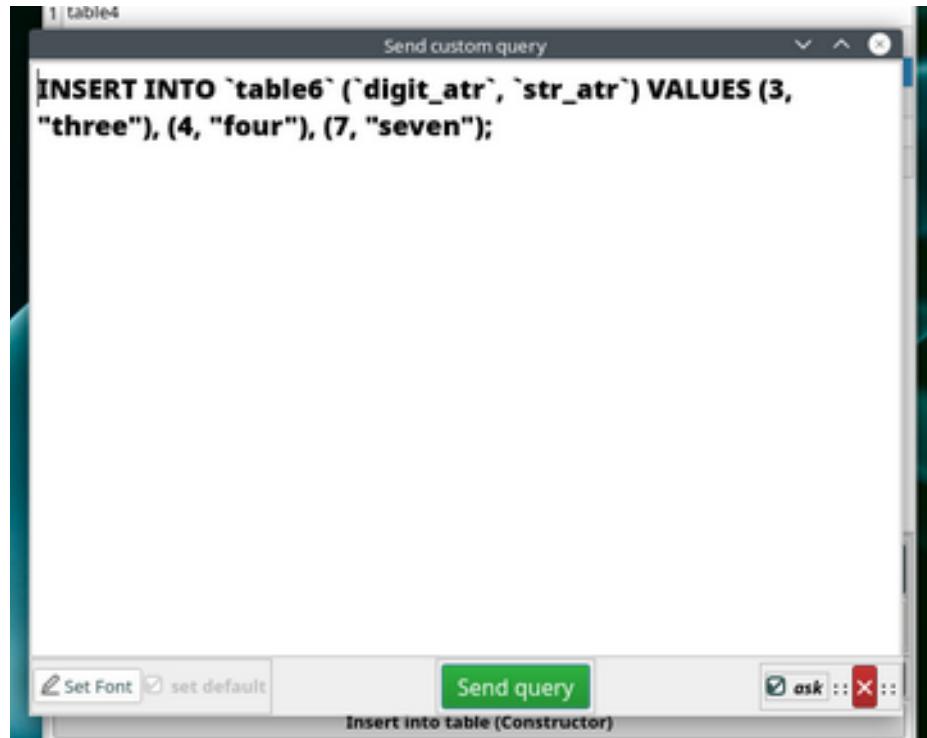
#### "Insert into table" constructor

This constructor defines 3 interactive stages:

1. Define table to insert;
2. Choose attributes from this table;

### 3. Insert new cortege 1-by-1.

After finalization of all stages and click on “OK” button, constructor will close, and you will see generated query in sending form for you to check (on screenshot below), making necessary corrections (if needed), and sending to SQL-Server.

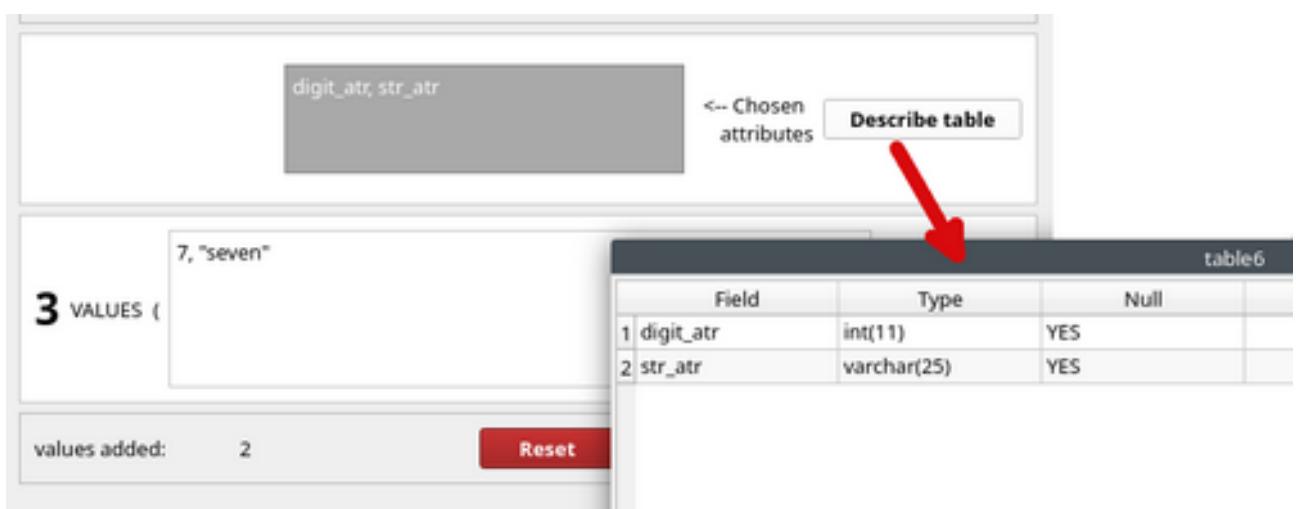


```
1 | table4
Send custom query
INSERT INTO `table6`(`digit_atr`, `str_atr`) VALUES (3, "three"), (4, "four"), (7, "seven");
```

The screenshot shows a MySQL Workbench window titled 'table4'. In the main pane, there is a text area containing an SQL insert statement: `INSERT INTO `table6`(`digit_atr`, `str_atr`) VALUES (3, "three"), (4, "four"), (7, "seven");`. Below the text area are several buttons: 'Set Font', 'set default', 'Send query' (which is highlighted in green), 'ask :: X ::', and 'Insert into table (Constructor)'. The status bar at the bottom shows the text 'Query to check & send'.

**Query to check & send**

In a middle panel you will see display of chosen table attributes to insert and “Describe table” button. This button helpful for checking data types, NULL flags of attributes etc.:



The screenshot shows the 'Insert into table (Constructor)' window for 'table6'. At the top, there is a list of 'Chosen attributes': 'digit\_atr, str\_atr'. To the right of this list is a 'Describe table' button, which is highlighted with a red arrow. Below this, there is a table showing the values being inserted: '7, "seven"' and '3 VALUES ('. The table has columns 'Field', 'Type', and 'Null'. The first row shows 'digit\_atr' as 'int(11)' and 'YES'. The second row shows 'str\_atr' as 'varchar(25)' and 'YES'. At the bottom left, it says 'values added: 2', and at the bottom right is a 'Reset' button.

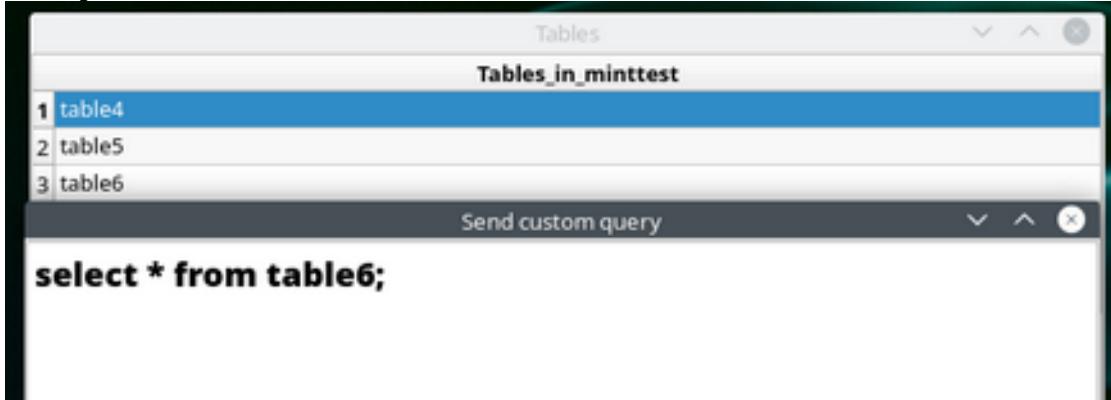
**“Describe” button in “Insert into table” constructor**

If you need to start from beginning, or choose another table, click on “Reset” button (or close and open constructor again).

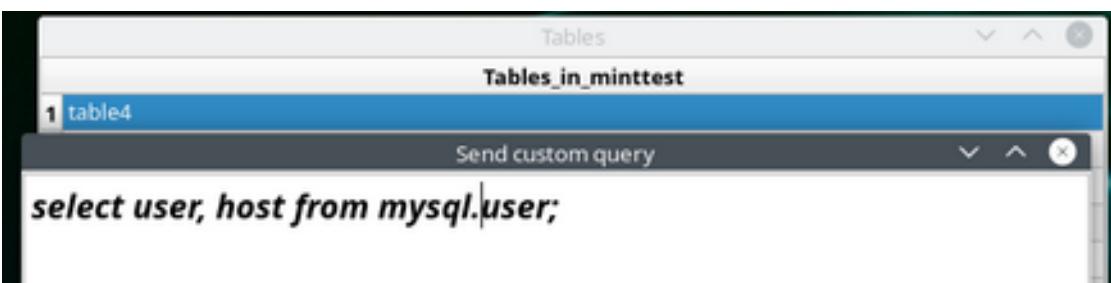
### 3.2.3. Custom query form

This form is similar to custom query form from 2.2.2 clause; difference is that this form is supposed to be used with chosen DB; if you will refer to another database entity/es, you also need to accompany your (another database) queries with prefix.

Example:



**Query to current database's table (without prefix)**



**Query, referred to another database's table (with prefix)**

**Note:** “USE <DATABASE>” query will work in this form.

Don't forget to click “Reload” button on “Tables” window after such query to refresh tables list in display form for new DB. If you already have opened query results form, this “hot switching” not erase your existing form data; but you will need to “return” to switched database (by “USE” query) if you need to refresh data in this opened forms.

## Appendix A

### Query tables scale mode

Scale mode: **Padding to fit page**  **set as default**

#### Scale mode embedded panel

There are 3 pre-defined scale modes in query result display forms:

##### **1) Padding to fit page (set by default)**

This mode make table's rows(R) and columns(C) fixed size to fit current window size, ignoring R and C content size. If your current window size not big enough, the contents may become unreadable.

For example, window have big size:

ALL_PLUGINS: table data								
PLUGIN_NAME	PLUGIN_VERSION	PLUGIN_STATUS	PLUGIN_TYPE	PLUGIN_TYPE_VERSION	PLUGIN_LIBRARY	PLUGIN_LIBRARY_VERSION	PLUGIN_AUTHOR	PLUGIN_DESCR
1 binlog	1.0	ACTIVE	STORAGE ENGINE	100616.0			MySQL AB	This is a pseudo storage engine representing the binlog in a transaction
2 mysql_native_password	1.0	ACTIVE	AUTHENTICATION	2.2			R.J.Silk, Sergei Golubchik	Native MySQL authentication
3 mysql_old_password	1.0	ACTIVE	AUTHENTICATION	2.2			R.J.Silk, Sergei Golubchik	Old MySQL 4.0 authentication
4 CSV	1.0	ACTIVE	STORAGE ENGINE	100616.0			Brian Aker, MySQL AB	Stores tables as files
5 MEMORY	1.0	ACTIVE	STORAGE ENGINE	100616.0			MySQL AB	Hash based, so memory, useful for temporary tables
6 Aria	1.5	ACTIVE	STORAGE ENGINE	100616.0			MariaDB Corporation AB	Crash-safe table MyISAM inheritance. Used for internal temporary table privilege tables
7 MyISAM	1.0	ACTIVE	STORAGE ENGINE	100616.0			MySQL AB	Non-transactional engine with good performance at small data footprint
8 MRG_MyISAM	1.0	ACTIVE	STORAGE ENGINE	100616.0			MySQL AB	Collection of individual MyISAM tables
9 SPATIAL_REF_SYS	1.0	ACTIVE	INFORMATION SCHEMA	100616.0			MariaDB	Lists all geometry columns
10 GEOMETRY_COLUMNS	1.0	ACTIVE	INFORMATION SCHEMA	100616.0			MariaDB	Lists all geometry columns
11 inet6	1.0	ACTIVE	DATA TYPE	100616.0			MariaDB Corporation	Data type INET
12 inet_aton	1.0	ACTIVE	FUNCTION	100616.0			MariaDB Corporation	Function INET_ATON
13 inet_ntoa	1.0	ACTIVE	FUNCTION	100616.0			MariaDB Corporation	Function INET_NTOA
14 inet6_aton	1.0	ACTIVE	FUNCTION	100616.0			MariaDB Corporation	Function INET6_ATON

Everything looks OK! But if we decrease window size:

	GIN_N	IN_VER	IN_ST	GIN_T	TYPE_V	IN_LIB	LIBRARY	IN_AU	_DESC	IN_LIC	D_OPT	N_MAT	AUTH_
13	inet...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... INE...	GPL	FOR...	Sta...	1.0
14	inet...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... INE...	GPL	FOR...	Sta...	1.0
15	inet...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... INE...	GPL	FOR...	Sta...	1.0
16	is_ip...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... IS_L...	GPL	FOR...	Sta...	1.0
17	is_ip...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... IS_L...	GPL	FOR...	Sta...	1.0
18	is_ip...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... IS_L...	GPL	FOR...	Sta...	1.0
19	is_ip...	1.0	ACT...	FUN...	100...			Mar... Cor...	Fun... IS_L...	GPL	FOR...	Sta...	1.0
20	CLIE...	2.0	ACT...	INF... SCH...	100...			Perc... and Serg... Gol...	Client Stati...	GPL	FOR...	Sta...	2.0
21	IND...	2.0	ACT...	INF... SCH...	100...			Perc... and Serg... Gol...	Index Stati...	GPL	FOR...	Sta...	2.0
22	TAB...	2.0	ACT...	INF... SCH...	100...			Perc... and Serg...	Table Stati...	GPL	FOR...	Sta...	2.0

That's it, the corteges display is on unreadable state.

## 2) Interactive resize

This mode gives the user interactive correction of R and C sizes. After switching to this mode, window scaling will be ignored:

	PLUGIN_NAME	PLUGIN_VERSION	PLUGIN_STATUS	PLUG
1	binlog	1.0	ACTIVE	STORAGE
2	mysql_native_password	1.0	ACTIVE	AUTHENT
3	mysql_old_password	1.0	ACTIVE	AUTHENT
4	CSV	1.0	ACTIVE	STORAGE
5	MEMORY	1.0	ACTIVE	STORAGE
6				




### 3) Wrap text to fit content

This mode ignores set R and C size fixed depending on attributes and corteges sizes; window scaling will be ignored, and you may scroll trough the table's content by mouse wheel, using form's scroll bars or keyboard arrows:

PLUGIN_NAME	PLUGIN_VERSION	PLUGIN_STATUS	PLUGIN_TYPE	PLUGIN_TYPE_VERSION	PLUGIN_LI
28 INNODB_LOCKS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
29 INNODB_LOCK_WAITS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
30 INNODB_CMP	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
31 INNODB_CMP_RESET	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
32 INNODB_CMPMEM	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
33 INNODB_CMPMEM_RESET	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
34 INNODB_CMP_PER_INDEX	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
35 INNODB_CMP_PER_INDEX_RESET	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
36 INNODB_BUFFER_PAGE	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
37 INNODB_BUFFER_PAGE_LRU	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
38 INNODB_BUFFER_POOL_STATS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
39 INNODB_METRICS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
40 INNODB_FT_DEFAULT_STOPWORD	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
41 INNODB_FT_DELETED	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
42 INNODB_FT_BEING_DELETED	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
43 INNODB_FT_CONFIG	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
44 INNODB_FT_INDEX_CACHE	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
45 INNODB_FT_INDEX_TABLE	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
46 INNODB_SYS_TABLES	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
47 INNODB_SYS_TABLESTATS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
48 INNODB_SYS_INDEXES	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
49 INNODB_SYS_COLUMNS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
50 INNODB_SYS_FIELDS	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	
51 INNODB_SYS_FOREIGN	10.6	ACTIVE	INFORMATION SCHEMA	100616.0	

## TABLE OF CONTENTS

Description .....	1
Brief summary .....	1
1. Startup window .....	1
2. Databases .....	2
2.1 User/Roles management .....	3
2.1.1 User/roles list .....	3
2.1.2 Create SQL User/Roles .....	4
2.1.3 Passwords management .....	4
2.1.4 Grant SQL privileges .....	5
2.2 Server queries .....	8
2.2.1 Databases creation .....	8
2.2.2 SQL-Server queries .....	8
2.3 “Removing” group .....	9
2.3.1 Remove database .....	9
2.3.2 Remove user .....	10
2.4 Dumping .....	10
2.4.1 User Choose .....	10
2.4.2 Dumping level .....	11
2.4.3 Dump name .....	11
2.5 Moving to “Tables” window .....	12
3. Tables .....	12
3.1 Content display .....	13
3.1.1 Show table content .....	13
3.1.2 Table description .....	14
3.2 Custom SQL-queries management .....	17
3.2.1 “Create table” constructor .....	17
3.2.1.1 Table name choose .....	17
3.2.1.2 Add attributes (columns) for new table .....	17
3.2.1.3 Foreign keys set up .....	18
3.2.2 Insert into table (“Create Tuple”) constructor .....	20
3.2.3. Custom query form .....	22
Appendix A .....	23