

# Using the GUI: Does SQLyog GUI understand the meaning of special MySQL KEYWORDS.

Yes it does!

Starting from SQLyog version 5.1 specific MySQL KEYWORDS are given a special treatment by the GUI. The KEYWORDS supported like this currently are:

- NULL
- CURRENT\_USER
- CURRENT\_TIMESTAMP
- UTC\_TIMESTAMP
- localtime

Also (from version 6.1) in the special context of the 'default' column of the CREATE/ALTER TABLE dialogue

- "

(two singlequotes) is treated like a KEYWORD to make it possible to define the EMPTY STRING as default for char, varchar, varbinary, set and enum columns (with other column types SQLyog will display a warning).

Finally SQLyog implements the word

- default

to be treated like a KEYWORD

Also read [here](http://faq.webyog.com/content/8/99/en/does-sqlyog-gui-understand-the-meaning-of-special-mysql-keywords.html) about FUNCTIONS support in SQLyog GUI.

To insert **CURRENT\_USER** (like 'memyself@localhost') you simply only need to enter "CURRENT\_USER" (without quotes) from SQLyog DATA and RESULT panes. When the row is saved SQLyog sends the string 'CURRENT\_USER' (also without quotes) and the server does the rest - inserts the user like 'memyself@localhost'. Similar for the four other KEYWORDS. With NULL there is that special thing that SQLyog always displayed NULL as (NULL) in the GRIDs. It still does. To enter NULL in a cell you can write "NULL" or "(NULL)" (without quotes) as you like. This is to facilitate 'old' as well as 'new' users.

# Using the GUI: Does SQLyog GUI understand the meaning of special MySQL KEYWORDS.

In case you for some reason should want to enter the **literal string** 'NULL' or 'CURRENT\_TIMESTAMP' etc. SQLyog 5.1 introduces the backquote character as a special 'code' that means: **this is a literal**. You can surround the KEYWORD with backquotes like "CURRENT\_TIMESTAMP" (the KEYWORD surrounded with backquotes and no other quotes) and SQLyog sends "CURRENT\_USER" (with singlequotes but without doublequotes) and "CURRENT\_USER" is inserted as a literal. To insert the backquote itself you must thus write "" (three backquotes). Actually you can enclose any literal string like 'mystring' with backquotes too, but there is no need for that - is only needed for those strings that are supported special MySQL KEYWORDS **with two exceptions/additions**: 1) SQLyog 5.1 implements the word 'Default' as a special supported KEYWORD **of it's own**. Writing 'Default' (without quotes and no matter the letter-case) will set the cell value to the column default (if there is no default for the column an empty string will be used). So to enter the literal 'default' you must also here use backquotes like **`default`**! As with "NULL"/"(NULL)" "(DEFAULT)" is synonym with "DEFAULT".

This also solves this issue often requested by users lately: To create a TIMESTAMP column as the first timestamp column in the table With MySQL >= 4.1 defined as

## Default CURRENT\_TIMESTAMP not NULL on update CURRENT\_TIMESTAMP

...the only thing you will need to do is to check the 'Not Null' checkbox. SQLyog and the MySQL server will do the rest. This is fully in accordance with the 'standard' behavior of TIMESTAMPS as described by MySQL AB [here](#). When UPDATING such 'on update ...' TIMESTAMP SQLyog will not either send the old value to MySQL as before. And the server will then update with a new CURRENT\_TIMESTAMP (as it was meant to be!). More information on defining TIMESTAMP properties with SQLyog [here](#).

All this of course applies **only** to the GUI functionalities of SQLyog - not the SQL pane.

We can expect more of that type KEYWORDS with MySQL in the future. SQLyog now has the features needed to implement them as soon as they become available.

Starting from SQLyog 5.1 you can also set 'Default', 'NULL' and 'Empty String' by right clicking a cell in the grid. As SQLyog uses the " (empty) sql\_mode for the connection with MySQL 5.x, the server will actually set the value of a **numerical field** to 0 (zero) when you are setting it 'empty string'. MySQL versions before 5.x do the same unless the GLOBAL sql\_mode has been changed from the default setting in the configuration.

Starting from SQLyog version 5.2 it is possible to turn of this feature from 'preferences' if you do not want it.

Unique solution ID: #1082

Author: Peter Laursen

Last update: 2009-02-16 05:37