

\c Clear the current input. Use this if you change your mind about executing the statement that you are entering.

List of all MySQL commands:

Note that all text commands must be first on line and end with ';'.

? (\?) Synonym for 'help'.

clear (\c) Clear command.

connect (\r) Reconnect to the server. Optional arguments are db and host.

delimiter (\d) Set statement delimiter.

edit (\e) Edit command with \$EDITOR.

ego (\G) Send command to mysql server, display result vertically.

exit (\q) Exit mysql. Same as quit.

go (\g) Send command to mysql server.

help (\h) Display this help.

nopager (\n) Disable pager, print to stdout.

notee (\t) Don't write into outfile.

pager (\P) Set PAGER [to_pager]. Print the query results via PAGER.

print (\p) Print current command.

prompt (\R) Change your mysql prompt.

quit (\q) Quit mysql.

rehash (\#) Rebuild completion hash.

source (\.) Execute an SQL script file. Takes a file name as an argument.

status (\s) Get status information from the server.

system (\!) Execute a system shell command.

tee (\T) Set outfile [to_outfile]. Append everything into given outfile.

use (\u) Use another database. Takes database name as argument.

charset (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.

warnings (\W) Show warnings after every statement.

nowarning (\w) Don't show warnings after every statement.

If you get stuck in an INSERT command with a MySQL command line that looks something like:

```
'>/c  
'>exit;  
'>help;  
'>/c  
'>nothing i type here does anything!!!  
'>
```

MySQL is trying to tell you that you have a “single quote” that needs to be “closed” in order to complete the SQL statement.

Generally typing in a “single quote” and semi-colon (eg: ‘;’) will complete the SQL statement. **Warning:** Chances are that the SQL statement you complete will be invalid and do nothing, **BUT**, if the statement is valid, unintended changes to your data can occur – resulting in bad happenings.

The other solution is to use task manager to kill the MySQL process; if you do this DO NOT kill the MySQLd process, which is the MySQL server process.