Home » Resources » Knowledge Base » MariaDB » MariaDB Documentation » Clients and Utilities » mysql Client » mysql Command-line Client

mysql Command-line Client

About the mysql command-line client

mysql is a simple SQL shell (with GNU readline capabilities). It supports interactive and non-interactive use. When used interactively, query results are presented in an ASCII-ta format. When used non-interactively (for example, as a filter), the result is presented in tab-separated format. The output format can be changed using command options.

If you have problems due to insufficient memory for large result sets, use the `--quick` option. This forces mysql to retrieve results from the server a row at a time rather than retrieving the entire result set and buffering it in memory before displaying it. This is done by returning the result set using the `mysql_use_result()` C API function in the client/server library rather than `mysql_store_result()` .

Using mysql is very easy. Invoke it from the prompt of your command interpreter as follows:

```
mysql db_name
```

Or:

```
mysql --user=user_name --password=your_password db_name
```

Then type an SQL statement, end it with “;”, \g, or \G and press Enter.

Typing Control-C causes mysql to attempt to kill the current statement. If this cannot be done, or Control-C is typed again before the statement is killed, mysql exits.

You can execute SQL statements in a script file (batch file) like this:

```
mysql db_name < script.sql > output.tab
```

mysql options

Options can be specified on the command line or in the `[mysql]` , `[client]` , [client-server] or [client-mariadb] option file groups of a my.cnf file. The full list of supported option can be obtained with:

```
mysql --verbose --help
```

The following options are available:

Option	Description
`-?` , `--help`	Display this help and exit.
`-I` , `--help`	Synonym for `-?`
`--abort-source-on-error`	Abort 'source filename' operations in case of errors.
`--auto-rehash`	Enable automatic rehashing. This option is on by default, which enables database, table, and column name completion. Use `--disable-auto-rehash` , `--no-auto-rehash` or `skip-auto-rehash` to disable rehashing. That causes mysql to start faster, but you must is the rehash command if you want to use name completion. To complete a name, enter the first part and press Tab. If the name is unambiguous, mysql completes it. Otherwise, you can press Tab again to see the possible names that begin with what you have typed so Completion does not occur if there is no default database.
`-A` , `--no-auto-rehash`	No automatic rehashing. One has to use 'rehash' to get table and field completion. This gives a quicker start of mysql and disables rehash on reconnect.
`--auto-vertical-output`	Automatically switch to vertical output mode if the result is wider than the terminal width.
`-B` , `--batch`	Print results using tab as the column separator, with each row on a new line. With this option, mysql does not use the history file. Batch mo results in nontabular output format and escaping of special characters. Escaping may be disabled by using raw mode; see the description the `--raw` option. (Enables `--silent` .)
`--binary-mode`	By default, ASCII '\0' is disallowed and '\r\n' is translated to '\n'. This switch turns off both features, and also turns off parsing of all client commands except \C and DELIMITER, in non-interactive mode (for input piped to mysql or loaded using the 'source' command). This is necessary when processing output from mysqlbinlog that may contain blobs.
`--character-sets-dir=name`	Directory for character set files.
`--column-names`	Write column names in results. (Defaults to on; use `--skip-column-names` to disable.)
`--column-type-info`	Display column type information.
`-c` , `--comments`	Preserve comments. Send comments to the server. The default is `--skip-comments` (discard comments), enable with `--comments` .
`-C` , `--compress`	Compress all information sent between the client and the server if both support compression.
`--connect-timeout=num`	Number of seconds before connection timeout. Defaults to zero.

`-D` , `--database=name`	Database to use.
`-# [options]` , `--debug[=options]`	On debugging builds, write a debugging log. A typical debug_options string is `d:t:o,file_name` . The default is `d:t:o,/tmp/mysql.trace` .
`--debug-check`	Check memory and open file usage at exit.
`-T` , `--debug-info`	Print some debug info at exit.
`--default-auth=plugin`	Default authentication client-side plugin to use.
`--default-character-set=name`	Set the default character set. A common issue that can occur when the operating system uses utf8 or another multibyte character set is th output from the mysql client is formatted incorrectly, due to the fact that the MariaDB client uses the latin1 character set by default. You ca usually fix such issues by using this option to force the client to use the system character set instead. If set to `auto` the character set is tak from the client environment (`LC_CTYPE` on Unix).
`--defaults-extra-file=file`	Read this file after the global files are read. Must be given as the first option.
`--defaults-file=file`	Only read default options from the given *file*. Must be given as the first option.
`--defaults-group-suffix=suffix`	In addition to the given groups, also read groups with this suffix.
`--delimiter=name`	Delimiter to be used. The default is the semicolon character (“;”).
`-e` , `--execute=name`	Execute statement and quit. Disables `--force` and history file. The default output format is like that produced with `--batch` .
`-f` , `--force`	Continue even if we get an SQL error. Sets `--abort-source-on-error` to 0.
`-h` , `--host=name`	Connect to host.
`-H` , `--html`	Produce HTML output.
`-U` , `--i-am-a-dummy`	Synonym for option `--safe-updates` , `-U` .
`-i` , `--ignore-spaces`	Ignore space after function names. Allows one to have spaces (including tab characters and new line characters) between function name a '('. The drawback is that this causes built in functions to become reserved words.
`--init-command=str`	SQL Command to execute when connecting to the MariaDB server. Will automatically be re-executed when reconnecting.
`--line-numbers`	Write line numbers for errors. (Defaults to on; use `--skip-line-numbers` to disable.)
`--local-infile`	Enable or disable LOCAL capability for LOAD DATA INFILE. With no value, the option enables LOCAL. The option may be given as `--local-infile=0` or `--local-infile=1` to explicitly disable or enable LOCAL. Enabling LOCAL has no effect if the server does not als support it.
`--max-allowed-packet=num`	The maximum packet length to send to or receive from server. The default is 16MB, the maximum 1GB.
`--max-join-size=num`	Automatic limit for rows in a join when using `--safe-updates` . Default is 1000000.
`-G` , `--named-commands`	Enable named commands. Named commands mean mysql's internal commands (see below) . When enabled, the named commands can used from any line of the query, otherwise only from the first line, before an enter. Long-format commands are allowed, not just short-form commands. For example, `quit` and `\q` are both recognized. Disable with `--disable-named-commands` . This option is disabled by default.
`--net-buffer-length=num`	The buffer size for TCP/IP and socket communication. Default is 16KB.
`-b` , `--no-beep`	Turn off beep on error.
`--no-defaults`	Don't read default options from any option file. Must be given as the first option.
`-o` , `--one-database`	Ignore statements except those those that occur while the default database is the one named on the command line. This filtering is limited and based only on USE statements. This is useful for skipping updates to other databases in the binary log.
`--pager[=name]`	Pager to use to display results (Unix only). If you don't supply an option, the default pager is taken from your ENV variable PAGER. Valid pagers are *less*, *more*, *cat [> filename]*, etc. See interactive help (\h) also. This option does not work in batch mode. Disable with `--disable-pager` . This option is disabled by default.
`-p` , `--password[=name]`	Password to use when connecting to server. If you use the short option form (-p), you cannot have a space between the option and the password. If you omit the password value following the `--password` or `-p` option on the command line, mysql prompts for one. Specifying password on the command line should be considered insecure. You can use an option file to avoid giving the password on the command li
`--plugin-dir=name`	Directory for client-side plugins.
`-P` , `--port=num`	Port number to use for connection or 0 for default to, in order of preference, my.cnf, $MYSQL_TCP_PORT, /etc/services, built-in default (3306).
`--print-defaults`	Print the program argument list and exit. Must be given as the first option.
`--progress-reports`	Get progress reports for long running commands (such as ALTER TABLE). (Defaults to on; use `--skip-progress-reports` to disable.)
`--prompt=name`	Set the mysql prompt to this value. See prompt command for options.
`--protocol=name`	The protocol to use for connection (tcp, socket, pipe, memory).
`-q` , `--quick`	Don't cache result, print it row by row. This may slow down the server if the output is suspended. Doesn't use history file.

`-r` , `--raw`	For tabular output, the “boxing” around columns enables one column value to be distinguished from another. For nontabular output (such is produced in batch mode or when the `--batch` or `--silent` option is given), special characters are escaped in the output so they can be identified easily. Newline, tab, NUL, and backslash are written as `\n` , `\t` , `\0` , and . The `--raw` option disables this character escaping.
`--reconnect`	Reconnect if the connection is lost. This option is enabled by default. Disable with `--disable-reconnect` or `skip-reconnect` .
`-U` , `--safe-updates`	Allow only those UPDATE and DELETE statements that specify which rows to modify by using key values. If you have set this option in an option file, you can override it by using `--safe-updates` on the command line. See using the --safe-updates option for more.
`--secure-auth`	Refuse client connecting to server if it uses old (pre-MySQL4.1.1) protocol. Defaults to false (unlike MySQL since 5,6, which defaults to tru
`--select-limit=num`	Automatic limit for SELECT when using --safe-updates. Default 1000.
`--server-arg=name`	Send embedded server this as a parameter.
`--shared-memory-base-name=name`	Shared-memory name to use for Windows connections using shared memory to a local server (started with the --shared-memory option). Case-sensitive.
`--show-warnings`	Show warnings after every statement. Applies to interactive and batch mode.
`--sigint-ignore`	Ignore SIGINT signals (usually CTRL-C).
`-s` , `--silent`	Be more silent. This option can be given multiple times to produce less and less output. This option results in nontabular output format and escaping of special characters. Escaping may be disabled by using raw mode; see the description for the `--raw` option.
`-L` , `--skip-auto-rehash`	Don't write line number for errors. See `--auto-rehash` .
`-N` , `--skip-column-names`	Don't write column names in results. See `--column-names` .
`-L` , `--skip-comments`	Don't write line number for errors. See `--comments` .
`-L` , `--skip-line-numbers`	Don't write line number for errors. See `--line-numbers` .
`-L` , `--skip-progress-reports`	Don't write line number for errors. See `--progress-reports` .
`-L` , `--skip-reconnect`	Don't write line number for errors. See `--reconnect` .
`-S` , `--socket=name`	For connections to localhost, the Unix socket file to use, or, on Windows, the name of the named pipe to use.
`--ssl`	Enable secure connections (TLS) for connection (automatically enabled with other flags).
`--ssl-ca=name`	CA file in PEM format (check OpenSSL docs, implies `--ssl`).
`--ssl-capath=name`	CA directory (check OpenSSL docs, implies `--ssl`).
`--ssl-cert=name`	X509 cert in PEM format (implies `--ssl`).
`--ssl-cipher=name`	TLS cipher to use (implies `--ssl`).
`--ssl-key=name`	X509 key in PEM format (implies `--ssl`).
`--ssl-crl=name`	Certificate revocation list (implies `--ssl`).
`--ssl-crlpath=name`	Certificate revocation list path (implies `--ssl`).
`--ssl-verify-server-cert`	Verify server's "Common Name" in its cert against hostname used when connecting. This option is disabled by default.
`-t` , `--table`	Display output in table format. This is the default for interactive use, but can be used to produce table output in batch mode.
`--tee=name`	Append everything into outfile. See interactive help (\h) also. Does not work in batch mode. Disable with `--disable-tee` . This option is disabled by default.
`-n` , `--unbuffered`	Flush buffer after each query.
`-u` , `--user=name`	User for login if not current user.
`-v` , `--verbose`	Write more. (-v -v -v gives the table output format).
`-V` , `--version`	Output version information and exit.
`-E` , `--vertical`	Print the output of a query (rows) vertically. Use the `\G` delimiter to apply to a particular statement if this option is not enabled.
`-w` , `--wait`	If the connection cannot be established, wait and retry instead of aborting.
`-X` , `--xml`	Produce XML output. See the mysqldump --xml option for more.

Default options are read from the following files in the given order: /etc/my.cnf /etc/mysql/my.cnf /.my.cnf The following groups are read: mysql client client-server client-mariadb

The output you get when running the above may differ depending on any customizations you may have in your local my.cnf file. See Configuring MariaDB with my.cnf.

How to specify which protocol to use when connecting to the mysqld server

The following is true for all MySQL and MariaDB command line clients:

You can force which protocol to be used to connect to the `mysqld` server by giving the `protocol` option one of the following values: `tcp` , `socket` , `pipe` or `memory` .

If `protocol` is not specified, then the following happens:

Linux/Unix

- If `hostname` is not specified or `hostname` is `localhost` , then Unix sockets are used.
- In other cases (`hostname` is given and it's not `localhost`) then a tcpip connection through the `port` option is used.

Note that `localhost` is a special value. Using 127.0.0.1 is not the same thing. The latter will connect to the mysqld server through tcpip.

Windows

- If `shared-memory-base-name` is specified and `hostname` is not specified or `hostname` is `localhost` , then the connection will happen through shared memory.
- If `shared-memory-base-name` is not specified and `hostname` is not specified or `hostname` is `localhost` , then the connection will happen through windows named pipes.
- Named pipes will also be used if the `libmysql` / `libmariadb` client library detects that the client doesn't support tcpip.
- In other cases then a tcpip connection through the `port` option is used.

How to test which protocol is used

The `status` command shows you information about which protocol is used:

```
shell> mysql test

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 10
Server version: 10.2.2-MariaDB-valgrind-max-debug Source distribution

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [test]> status;
--------------
mysql  Ver 15.1 Distrib 10.0.25-MariaDB, for Linux (x86_64) using readline 5.2

Connection id:          10
Current database:       test
Current user:           monty@localhost
...
Connection:             Localhost via UNIX socket
...
UNIX socket:            /tmp/mysql-dbug.sock
```

mysql commands

There are also a number of commands that can be run inside the client. Note that all text commands must be first on line and end with ';'

Command	Description
`?` , `\?`	Synonym for `help'.
`clear` , `\c`	Clear the current input statement.
`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. See prompt command for options.
`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.

The mysql_history file

On Unix, the mysql client writes a record of executed statements to a history file. By default, this file is named `.mysql_history` and is created in your home directory. To specify different file, set the value of the MYSQL_HISTFILE environment variable.

The .mysql_history file should be protected with a restrictive access mode because sensitive information might be written to it, such as the text of SQL statements that contain passwords.

If you do not want to maintain a history file, first remove .mysql_history if it exists, and then use either of the following techniques:

- Set the MYSQL_HISTFILE variable to /dev/null. To cause this setting to take effect each time you log in, put the setting in one of your shell's startup files.
- Create .mysql_history as a symbolic link to /dev/null:

```
shell> ln -s /dev/null $HOME/.mysql_history
```

You need do this only once.

prompt command

The prompt command reconfigures the default prompt `\N [\d]>` . The string for defining the prompt can contain the following special sequences.

Option	Description
`\c`	A counter that increments for each statement you issue.
`\D`	The full current date.
`\d`	The default database.
`\h`	The server host.
`\l`	The current delimiter.
`\m`	Minutes of the current time.
`\n`	A newline character.
`\O`	The current month in three-letter format (Jan, Feb, ...).
`\o`	The current month in numeric format.
`\P`	am/pm.
`\p`	The current TCP/IP port or socket file.
`\R`	The current time, in 24-hour military time (0–23).
`\r`	The current time, standard 12-hour time (1–12).
`\S`	Semicolon.
`\s`	Seconds of the current time.
`\t`	A tab character.
`\U`	Your full user_name@host_name account name.
`\u`	Your user name.
`\v`	The server version.
`\w`	The current day of the week in three-letter format (Mon, Tue, ...).
`\Y`	The current year, four digits.
`\y`	The current year, two digits.
`\_`	A space.
`\ `	A space (a space follows the backslash).
`\'`	Single quote.
`\"`	Double quote.
`\ \`	A literal “\” backslash character.

`\x`	x, for any “x” not listed above.

mysql tips

This section describes some techniques that can help you use **`mysql`** more effectively.

Displaying query results vertically

Some query results are much more readable when displayed vertically, instead of in the usual horizontal table format. Queries can be displayed vertically by terminating the que with \G instead of a semicolon. For example, longer text values that include newlines often are much easier to read with vertical output:

```
mysql> SELECT * FROM mails WHERE LENGTH(txt) < 300 LIMIT 300,1\G
*************************** 1. row ***************************
  msg_nro: 3068
    date: 2000-03-01 23:29:50
time_zone: +0200
mail_from: Monty
    reply: monty@no.spam.com
  mail_to: "Thimble Smith" <tim@no.spam.com>
      sbj: UTF-8
      txt: >>>>> "Thimble" == Thimble Smith writes:
Thimble> Hi.  I think this is a good idea.  Is anyone familiar
Thimble> with UTF-8 or Unicode? Otherwise, I´ll put this on my
Thimble> TODO list and see what happens.
Yes, please do that.
Regards,
Monty
    file: inbox-jani-1
    hash: 190402944
1 row in set (0.09 sec)
```

Using the --safe-updates option

For beginners, a useful startup option is `--safe-updates` (or `--i-am-a-dummy` , which has the same effect). It is helpful for cases when you might have issued a `DELETE FROM tbl_name` statement but forgotten the `WHERE` clause. Normally, such a statement deletes all rows from the table. With `--safe-updates` , you can delete rows only b specifying the key values that identify them. This helps prevent accidents.

When you use the `--safe-updates` option, mysql issues the following statement when it connects to the MariaDB server:

```
SET sql_safe_updates=1, sql_select_limit=1000, sql_max_join_size=1000000;
```

The SET statement has the following effects:

- You are not allowed to execute an UPDATE or DELETE statement unless you specify a key constraint in the WHERE clause or provide a LIMIT clause (or both). For example:

```
UPDATE tbl_name SET not_key_column=val WHERE key_column=val;
UPDATE tbl_name SET not_key_column=val LIMIT 1;
```

- The server limits all large `SELECT` results to 1,000 rows unless the statement includes a `LIMIT` clause.
- The server aborts multiple-table `SELECT` statements that probably need to examine more than 1,000,000 row combinations.

To specify limits different from 1,000 and 1,000,000, you can override the defaults by using the `--select_limit` and `--max_join_size` options:

```
mysql --safe-updates --select_limit=500 --max_join_size=10000
```

Disabling mysql auto-reconnect

If the mysql client loses its connection to the server while sending a statement, it immediately and automatically tries to reconnect once to the server and send the statement ag However, even if mysql succeeds in reconnecting, your first connection has ended and all your previous session objects and settings are lost: temporary tables, the autocommi mode, and user-defined and session variables. Also, any current transaction rolls back. This behavior may be dangerous for you, as in the following example where the server shut down and restarted between the first and second statements without you knowing it:

```
mysql> SET @a=1;
Query OK, 0 rows affected (0.05 sec)
mysql> INSERT INTO t VALUES(@a);
ERROR 2006: MySQL server has gone away
No connection. Trying to reconnect...
Connection id:    1
Current database: test
Query OK, 1 row affected (1.30 sec)
mysql> SELECT * FROM t;
+------+
| a    |
+------+
| NULL |
+------+
```

The @a user variable has been lost with the connection, and after the reconnection it is undefined. If it is important to have mysql terminate with an error if the connection has been lost, you can start the mysql client with the `--skip-reconnect` option.

See also

- Readline commands and configuration

↑ mysql Client ↑   Delimiters in the mysql Client →

Comments

No comments