

Fichero de Opciones y Variables de Servidor



Ángel David González Quintana

Índice:

1. Ficheros de opciones.....	3
2. Variables del servidor.....	7
2.1. Variables de estado.....	12
2.2. Variables dinámicas.....	17



1. Ficheros de opciones

1. Encuentra el fichero my.ini (en el caso que tu servidor MySQL estuviera sobre Windows) o my.cnf de tu instalación de MySQL (podría no estar en una ubicación no estándar).

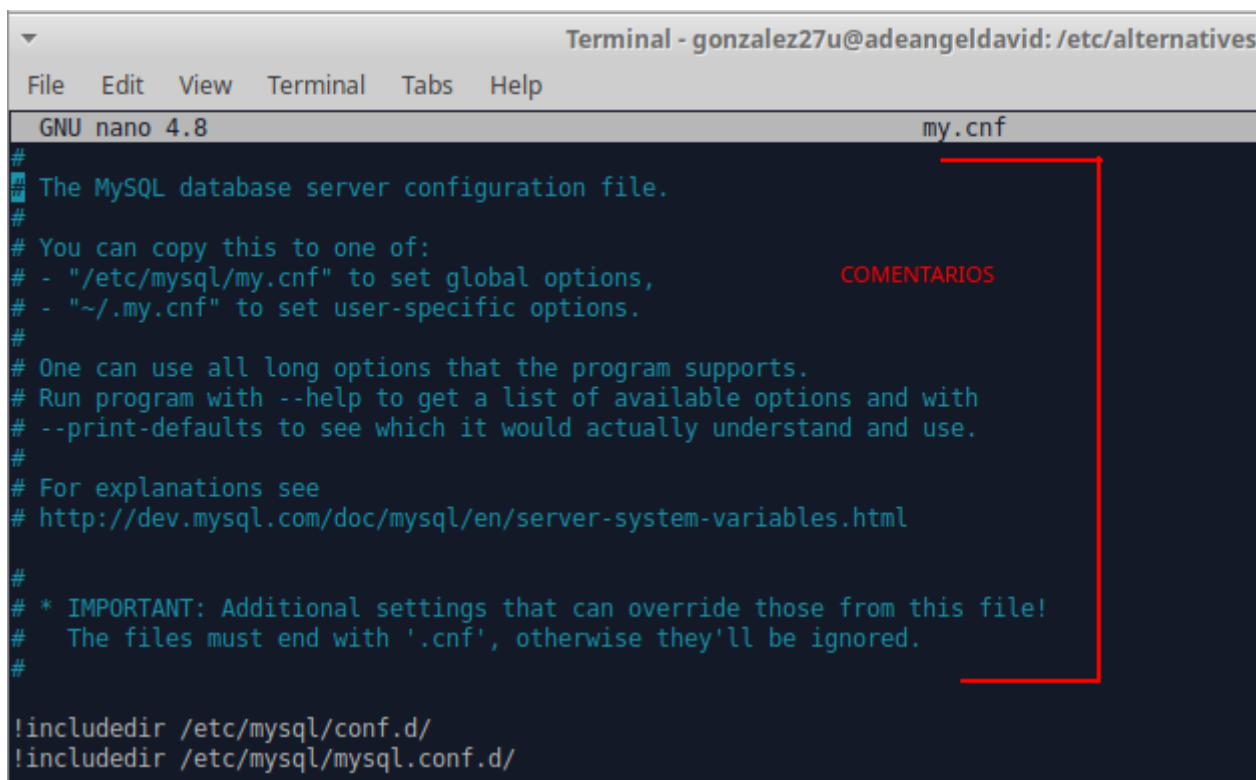
Para encontrar el fichero my.cnf es muy sencillo, simplemente accedemos al directorio /etc/mysql/ y vemos el fichero:

```
gonzalez27u@adeangel david:~$ cd /etc/mysql/
gonzalez27u@adeangel david:/etc/mysql$ ls -la
total 40
drwxr-xr-x  4 root root  4096 Nov 11 12:51 .
drwxr-xr-x 155 root root 12288 Nov 18 08:15 ..
drwxr-xr-x  2 root root  4096 Oct 14 11:27 conf.d
-rwxr-xr-x  1 root root   120 Aug 19 13:32 debian-start
-rw-----  1 root root   317 Nov 11 12:51 debian.cnf
lrwxrwxrwx  1 root root    24 Oct 14 11:27 my.cnf -> /etc/alternatives/my.cnf
-rw-r--r--  1 root root   839 Aug  3 2016 my.cnf.fallback
-rw-r--r--  1 root root   713 Nov 11 12:51 mysql.cnf
drwxr-xr-x  2 root root  4096 Nov 11 12:51 mysql.conf.d
gonzalez27u@adeangel david:/etc/mysql$ cd /etc/alternatives/
gonzalez27u@adeangel david:/etc/alternatives$ ls -la | grep my.cnf
lrwxrwxrwx  1 root root    20 Oct 14 11:27 my.cnf -> /etc/mysql/mysql.cnf
gonzalez27u@adeangel david:/etc/alternatives$
```

Como podemos ver, este mismo fichero está enlazado y está situado en /etc/alternatives/.

2. ¿Cómo se escribe un comentario en este fichero? ¿Y un grupo de opciones? ¿Todas las opciones tienen un valor?

Un comentario se puede escribir como en la siguiente imagen:



```
Terminal - gonzalez27u@adeangeldavid: /etc/alternatives
File Edit View Terminal Tabs Help
GNU nano 4.8 my.cnf
#
# The MySQL database server configuration file.
#
# You can copy this to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html
#
# * IMPORTANT: Additional settings that can override those from this file!
# The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mysql.conf.d/
```

Como podemos ver, todas las líneas que van acompañadas de una almohadilla (#) son todos comentarios.

Para escribir un grupo de opciones escribiremos el siguiente ejemplo:

```
[client]
port=3306
password="telesforo";
```

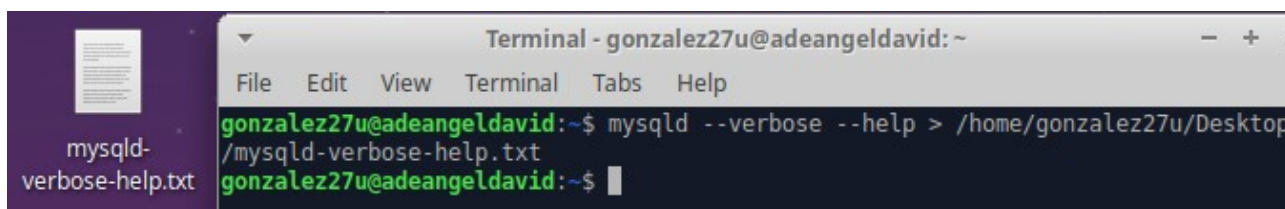
Pero no todas las opciones tienen un valor, es decir, en el ejemplo anterior, tenemos dos opciones con un valor cada uno (opcion=valor), pero hay opciones sin valor, como el siguiente ejemplo que se va a mostrar a continuación:

```
[mysqldump]
quick
```

Como se puede ver, no tiene valor.

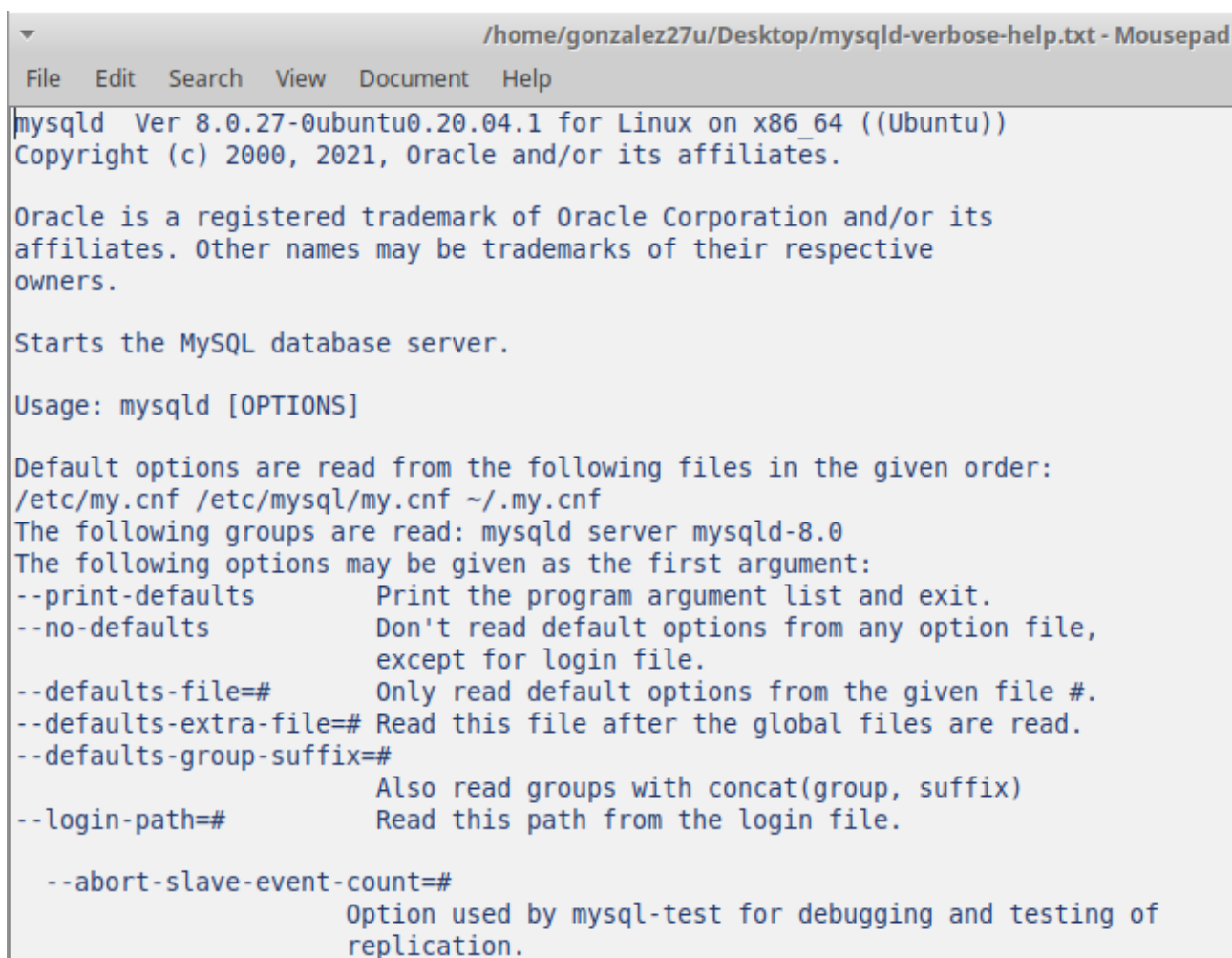
4. Ejecuta "mysql --verbose --help" desde una consola para ver una lista de las variables del servidor. Para ver mejor el texto redirecciona la salida a fichero.

Para ejecutar este comando y redireccionarlo debemos darle una salida, esto lo conseguimos con el comando `mysql --verbose --help > /home/usuario/ruta-de-destino/fichero.txt` :



```
Terminal - gonzalez27u@adeangeldavid: ~
File Edit View Terminal Tabs Help
gonzalez27u@adeangeldavid:~$ mysql --verbose --help > /home/gonzalez27u/Desktop/mysql-verbose-help.txt
gonzalez27u@adeangeldavid:~$
```

Al ejecutarlo crea el fichero en la ruta de destino (en mi caso, ../Desktop/mysql-verbose-help.txt). Simplemente debemos abrir el fichero y vemos todo el contenido del comando:



```
/home/gonzalez27u/Desktop/mysql-verbose-help.txt - Mousepad
File Edit Search View Document Help
mysql Ver 8.0.27-0ubuntu0.20.04.1 for Linux on x86_64 ((Ubuntu))
Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Starts the MySQL database server.

Usage: mysql [OPTIONS]

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 server mysql-8.0
The following options may be given as the first argument:
--print-defaults      Print the program argument list and exit.
--no-defaults         Don't read default options from any option file,
                      except for login file.
--defaults-file=#     Only read default options from the given file #.
--defaults-extra-file=# Read this file after the global files are read.
--defaults-group-suffix=# Also read groups with concat(group, suffix)
--login-path=#        Read this path from the login file.

--abort-slave-event-count=#
                      Option used by mysql-test for debugging and testing of
                      replication.
```

5. Explica qué significan y que se consigue con cada una de las variables del siguiente fichero de configuración:

[client]
port=3306
password="telesforo";

Cliente
Puerto al que se conecta
Contraseña

Esta variable hace que el cliente se conecte por el puerto 3306 y tendrá que introducir la contraseña “telesforo” para poder acceder.

[mysqld]
port=3306
key_buffer_size=16M
max_allowed_packet=8M

Demonio MySQL
Puerto al que se conecta
Tamaño de memoria del búfer
Tamaño máximo de paquetes permitidos

El demonio MySQL se abre por el puerto 3306 con tamaño de memoria búfer de 16 MB y permite sólo paquetes hasta un máximo de 8 MB de tamaño.

[mysqldump]
quick

Esta opción le dice a mysqldump que escriba la salida de volcado a medida que lee cada fila del servidor, lo que será útil para tablas grandes. De forma predeterminada, mysqldump lee todas las filas de una tabla en la memoria antes de escribir la salida; para tablas grandes, esto requiere una gran cantidad de memoria, lo que posiblemente provoque un error en el volcado.

2. Variables del servidor

1. Define qué son las variables del servidor.

Las variables del servidor son parámetros que sirven para configurar el funcionamiento del servidor.

El servidor mantiene muchas variables y cada una tiene un valor predeterminado que se puede configurar con opciones en una línea de comandos o en un archivo de opciones.

2. Usa el comando "SHOW VARIABLES" para conocer el valor de todas las variables y enviar el resultado a un fichero (podemos hacerlo desde consola linux y desde el mysql client usando la orden tee.

Para dar salida a consultas MySQL vamos a usar la orden **tee**.

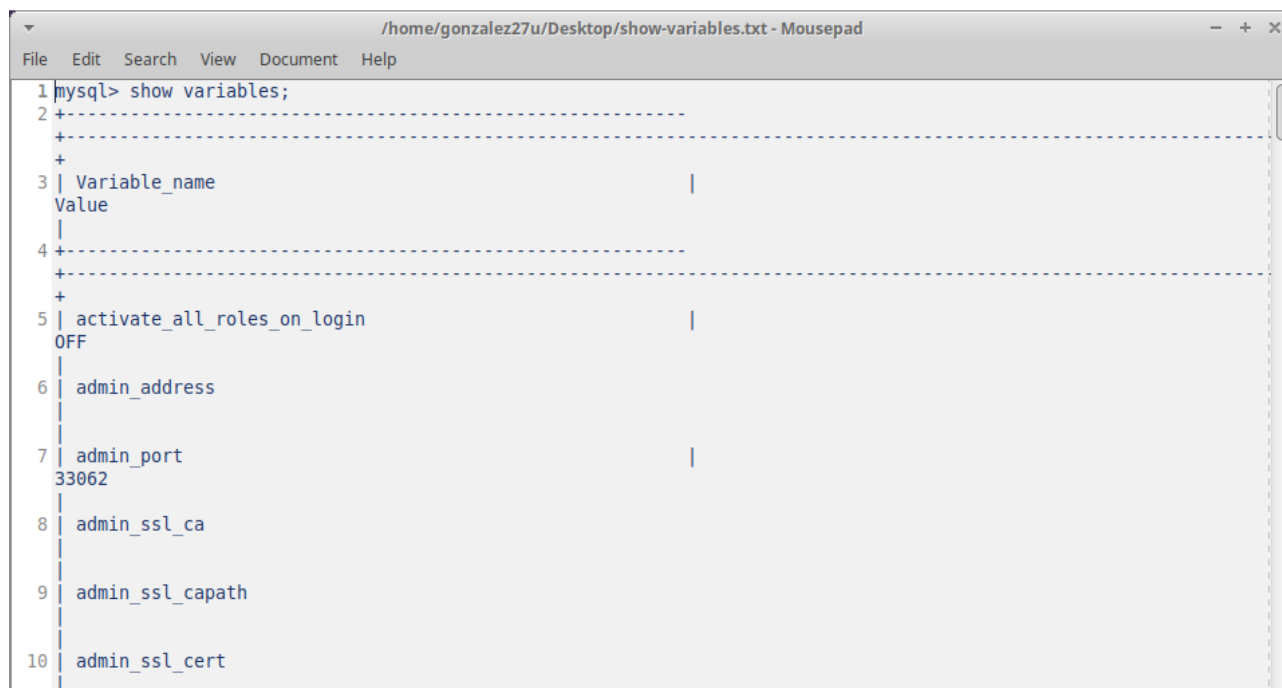
Primero vamos a crear el fichero en la ruta donde vamos a abrir MySQL (en mi caso, ../Desktop/show-variables.txt):

```
gonzalez27u@adeangeldavid:~$ cd Desktop/  
gonzalez27u@adeangeldavid:~/Desktop$ touch show-variables.txt  
gonzalez27u@adeangeldavid:~/Desktop$
```

Luego accedemos a MySQL y ejecutamos la orden `tee fichero.txt`, luego ejecutamos la orden `show variables;` :

```
gonzalez27u@adeangeldavid: ~/Desktop  
mysql> tee show-variables.txt  
Logging to file 'show-variables.txt'  
mysql> show variables;
```

Una vez ejecutado, cerramos MySQL, accedemos al fichero y comprobamos que se han guardado todas las órdenes y resultados que hayamos hecho:

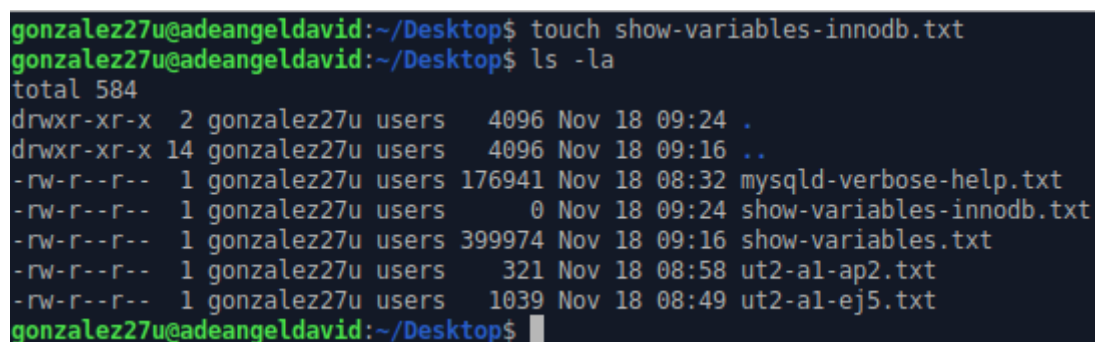


The screenshot shows a text editor window titled "/home/gonzalez27u/Desktop/show-variables.txt - Mousepad". The content of the file is the output of the MySQL command 'show variables;'. The output is a table with two columns: 'Variable_name' and 'Value'. The variables shown are 'activate_all_roles_on_login' (OFF), 'admin_address', 'admin_port' (33062), 'admin_ssl_ca', 'admin_ssl_capath', and 'admin_ssl_cert'.

```
1 mysql> show variables;
2 +-----+
3 | Variable_name |
4 | Value         |
5 +-----+
6 | activate_all_roles_on_login |
7 | OFF           |
8 | admin_address |
9 | admin_port    |
10 | 33062         |
11 | admin_ssl_ca  |
12 | admin_ssl_capath |
13 | admin_ssl_cert |
```

3. Repite lo anterior para mostrar solo las variables relacionadas con el motor "InnoDB".

Hacemos lo mismo. Creamos un fichero nuevo en el mismo directorio anterior (en mi caso, ../Desktop/show-variables-innodb.txt):



The screenshot shows a terminal window with the following commands and output:

```
gonzalez27u@adeangeldavid:~/Desktop$ touch show-variables-innodb.txt
gonzalez27u@adeangeldavid:~/Desktop$ ls -la
total 584
drwxr-xr-x  2 gonzalez27u users  4096 Nov 18 09:24 .
drwxr-xr-x 14 gonzalez27u users  4096 Nov 18 09:16 ..
-rw-r--r--  1 gonzalez27u users 176941 Nov 18 08:32 mysqld-verbose-help.txt
-rw-r--r--  1 gonzalez27u users    0 Nov 18 09:24 show-variables-innodb.txt
-rw-r--r--  1 gonzalez27u users 399974 Nov 18 09:16 show-variables.txt
-rw-r--r--  1 gonzalez27u users   321 Nov 18 08:58 ut2-a1-ap2.txt
-rw-r--r--  1 gonzalez27u users  1039 Nov 18 08:49 ut2-a1-ej5.txt
gonzalez27u@adeangeldavid:~/Desktop$
```


Accedemos a MySQL, ejecutamos la orden `tee fichero.txt`, luego ejecutamos la orden `show variables like 'innodb%';`:

```
gonzalez27u@adeangeldavid: ~/Desktop
mysql> tee show-variables-innodb.txt
Logging to file 'show-variables-innodb.txt'
mysql> show variables like 'inno%';
```

Una vez ejecutado, cerramos MySQL, accedemos al fichero y comprobamos que se han guardado todas las órdenes y resultados que hayamos hecho:

```
/home/gonzalez27u/Desktop/show-variables-innodb.txt - Mousepad
File Edit Search View Document Help

1 mysql> show variables like 'inno%';
2 +-----+-----+
3 | Variable_name | Value |
4 +-----+-----+
5 | innodb_adaptive_flushing | ON |
6 | innodb_adaptive_flushing_lwm | 10 |
7 | innodb_adaptive_hash_index | ON |
8 | innodb_adaptive_hash_index_parts | 8 |
9 | innodb_adaptive_max_sleep_delay | 150000 |
10 | innodb_api_bk_commit_interval | 5 |
11 | innodb_api_disable_rowlock | OFF |
12 | innodb_api_enable_binlog | OFF |
13 | innodb_api_enable_mdls | OFF |
14 | innodb_api_trx_level | 0 |
15 | innodb_autoextend_increment | 64 |
16 | innodb_autoinc_lock_mode | 2 |
17 | innodb_buffer_pool_chunk_size | 134217728 |
18 | innodb_buffer_pool_dump_at_shutdown | ON |
19 | innodb_buffer_pool_dump_now | OFF |
20 | innodb_buffer_pool_dump_pct | 25 |
21 | innodb_buffer_pool_filename | ib_buffer_pool |
22 | innodb_buffer_pool_in_core_file | ON |
23 | innodb_buffer_pool_instances | 1 |
24 | innodb_buffer_pool_load_abort | OFF |
25 | innodb_buffer_pool_load_at_startup | ON |
26 | innodb_buffer_pool_load_now | OFF |
27 | innodb_buffer_pool_size | 134217728 |
28 | innodb_change_buffer_max_size | 25 |
29 | innodb_change_buffering | all |
30 | innodb_checksum_algorithm | crc32 |
31 | innodb_cmp_per_index_enabled | OFF |
```

4. Para gestionar variables tenemos, como hemos visto, el comando SHOW "*comando*":

- cómo mostrar todos los motores de almacenamiento

Para mostrar todos los motores de almacenamiento debemos ejecutar la orden `show engines;` :

```
Terminal - gonzalez27u@adeangeldavid: ~/Desktop
mysql> show engines;
+-----+-----+-----+-----+-----+
| Engine      | Support | Comment                                     | Transactions | XA  | Savepoints |
+-----+-----+-----+-----+-----+
| ARCHIVE     | YES     | Archive storage engine                     | NO           | NO  | NO         |
| BLACKHOLE   | YES     | /dev/null storage engine (anything you write to it disappears) | NO           | NO  | NO         |
| MRG_MYISAM  | YES     | Collection of identical MyISAM tables      | NO           | NO  | NO         |
| FEDERATED   | NO      | Federated MySQL storage engine           | NULL        | NULL | NULL       |
| MyISAM      | YES     | MyISAM storage engine                     | NO           | NO  | NO         |
| PERFORMANCE_SCHEMA | YES     | Performance Schema                       | NO           | NO  | NO         |
| InnoDB      | DEFAULT | Supports transactions, row-level locking, and foreign keys | YES          | YES | YES        |
| MEMORY      | YES     | Hash based, stored in memory, useful for temporary tables | NO           | NO  | NO         |
| CSV         | YES     | CSV storage engine                       | NO           | NO  | NO         |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql>
```

- cómo mostrar el estado actual del servidor

Para mostrar el estado actual del servidor usaremos la orden `show status;` :

```
gonzalez27u@adeangeldavid: ~/Desktop
mysql> show status;
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Aborted_clients | 0 |
| Aborted_connects | 1 |
| Acl_cache_items_count | 0 |
| Binlog_cache_disk_use | 0 |
| Binlog_cache_use | 0 |
| Binlog_stmt_cache_disk_use | 0 |
| Binlog_stmt_cache_use | 0 |
| Bytes_received | 647 |
| Bytes_sent | 34104 |
+-----+-----+
```

- **cómo averiguar todos los clientes que están conectados al servidor**

Para averiguar los clientes conectados usaremos la orden `show processlist;` :

```
gonzalez27u@adeangeldavid: ~/Desktop
mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | User          | Host      | db    | Command | Time | State          | Info          |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 5  | event_scheduler | localhost | NULL  | Daemon   | 5218 | Waiting on empty queue | NULL          |
| 14 | root           | localhost | NULL  | Query    | 0    | init           | show processlist |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

- **cómo conocer todas las tablas que están abiertas**

Para conocer todas las tablas abiertas usaremos la orden `show open tables;` :

```
gonzalez27u@adeangeldavid: ~
mysql> show open tables;
+-----+-----+-----+-----+
| Database | Table                               | In_use | Name_locked |
+-----+-----+-----+-----+
| mysql    | events                             | 0      | 0            |
| mysql    | servers                            | 0      | 0            |
| mysql    | time_zone_transition_type          | 0      | 0            |
| mysql    | slave_master_info                  | 0      | 0            |
| mysql    | time_zone                          | 0      | 0            |
| mysql    | time_zone_leap_second              | 0      | 0            |
| mysql    | password_history                   | 0      | 0            |
| mysql    | global_grants                     | 0      | 0            |
| mysql    | default_roles                     | 0      | 0            |
| mysql    | role_edges                         | 0      | 0            |
| mysql    | proxies_priv                      | 0      | 0            |
| mysql    | procs_priv                        | 0      | 0            |
| mysql    | column_type_elements              | 0      | 0            |
| mysql    | user                              | 0      | 0            |
| mysql    | columns                           | 0      | 0            |
| mysql    | time_zone_name                    | 0      | 0            |
| mysql    | foreign_key_column_usage          | 0      | 0            |
| mysql    | db                                | 0      | 0            |
| mysql    | foreign_keys                      | 0      | 0            |
| mysql    | func                              | 0      | 0            |
| mysql    | indexes                           | 0      | 0            |
| mysql    | slave_worker_info                 | 0      | 0            |
| mysql    | index_partitions                  | 0      | 0            |
| mysql    | index column usage                 | 0      | 0            |
+-----+-----+-----+-----+
```

2.1. Variables de estado

1. Explica qué función tienen las variables de estado.

Su función es proporcionar información sobre el funcionamiento del servidor MySQL.

2. Usa el comando "SHOW STATUS" para conocer el valor de todas las variables.

```
gonzalez27u@adeangeldavid: ~/Desktop
mysql> show status;
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Aborted_clients | 0 |
| Aborted_connects | 1 |
| Acl_cache_items_count | 0 |
| Binlog_cache_disk_use | 0 |
| Binlog_cache_use | 0 |
| Binlog_stmt_cache_disk_use | 0 |
| Binlog_stmt_cache_use | 0 |
| Bytes_received | 855 |
| Bytes_sent | 49832 |
| Caching_sha2_password_rsa_public_key | -----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA2mu3sa3D0cWg8n7lfeMj
wGeoTACHVqE8l3AbMsMR3VXP7bdxBgPf6XvK5kynwUgYiccranHrXgt6qi5hh00a
e5yklMgGS2hUiU0Xc8mAYR+y86hyVpXn4fK77mLB0X4t3zert367ujovalCNweK2
PMzwYeahpPyExklSWz7daB5ZAZETv0Y9PS2b6GDU+9ShTLawNH3cJx0Gx5nZy11N
F0exzRMf78KR9Tosn8/0o3fx9o6Kq7/0pYy24BU8U0bg07kRuk9iVBnfFjios00J
dYeTudQabo5KqoUCdv32No7ED/qc4kRVngIckd25rTmdR4s7tBQJiellleQ1UQw+H
IwIDAQAB
-----END PUBLIC KEY-----
| Com_admin_commands | 0 |
```

3. Haz que uno o más de tus compañeros se conecte a tu servidor (puede que por cuestión de permisos no os podáis conectar).

Dejamos que nuestro compañero (Ayoze Hernández) se conecte a nuestro servidor:

```
ayoze@adeayoze:/etc/mysql$ mysql -u root -p -h 172.19.27.69
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

Dejamos que acceda a nuestra base de datos 'information_schema':

```
mysql> use information_schema;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_information_schema |
+-----+
| ADMINISTRABLE_ROLE_AUTHORIZATIONS |
| APPLICABLE ROLES |
| CHARACTER SETS |
| CHECK CONSTRAINTS |
| COLLATIONS |
| COLLATION_CHARACTER_SET_APPLICABILITY |
| COLUMNS |
| COLUMNS_EXTENSIONS |
| COLUMN_PRIVILEGES |
| COLUMN_STATISTICS |
| ENABLED_ROLES |
| ENGINES |
| EVENTS |
```

Y que ejecute algunas consultas libremente:

```
mysql> describe USER_PRIVILEGES;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| GRANTEE        | varchar(292)  | NO   |     |         |       |
| TABLE_CATALOG | varchar(512)  | NO   |     |         |       |
| PRIVILEGE_TYPE | varchar(64)   | NO   |     |         |       |
| IS_GRANTABLE   | varchar(3)    | NO   |     |         |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select GRANTEE from USER_PRIVILEGES;
+-----+
| GRANTEE |
+-----+
| 'root'@'%' |
+-----+
1 row in set (0.00 sec)

mysql> █
```

4. Comprueba quién está conectado usando el comando correspondiente (Pista: es un comando visto SHOW XYZ).

Comprobamos que nuestro compañero está dentro de nuestro servidor con la orden `show processlist;` :

```
Terminal - gonzalez27u@adeangeldavid: ~
File Edit View Terminal Tabs Help

mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | User          | Host                | db                | Command | Time | State                | Info                |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 5  | event_scheduler | localhost           | NULL              | Daemon  | 674  | Waiting on empty queue | NULL                |
| 16 | root          | 172.19.28.35:57322 | information_schema | Sleep   | 2    |                       | NULL                |
| 17 | root          | localhost           | NULL              | Query   | 0    | init                  | show processlist   |
+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Podemos ver que nuestro compañero accedió con el usuario **root** y está dentro de la base de datos 'information_schema', lo que vamos a memorizar la **ID**.

5. Intenta desconectarlo con el comando "kill"

Lo desconectamos de nuestro servidor usando la orden `kill ID` ; :

```
mysql> kill 16;
Query OK, 0 rows affected (0.00 sec)
```

Finalmente volvemos a revisar si está conectado:

```
mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | User          | Host      | db  | Command | Time | State          | Info          |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 5  | event_scheduler | localhost | NULL | Daemon   | 699  | Waiting on empty queue | NULL          |
| 17 | root          | localhost | NULL | Query    | 0    | init           | show processlist |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

Ahora desde que intente ejecutar cualquier consulta no se ejecutará ya que lo hemos desconectado del servidor:

```
ayoze@adeayoze: /etc/mysql
mysql> describe USER_PRIVILEGES;
+-----+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+-----+
| GRANTEE        | varchar(292)  | NO   |     |         |       |
| TABLE_CATALOG | varchar(512)  | NO   |     |         |       |
| PRIVILEGE_TYPE | varchar(64)   | NO   |     |         |       |
| IS_GRANTABLE   | varchar(3)    | NO   |     |         |       |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select GRANTEE from USER_PRIVILEGES;
+-----+
| GRANTEE |
+-----+
| 'root'@'%' |
+-----+
1 row in set (0.00 sec)

mysql> select GRANTEE from USER_PRIVILEGES;
ERROR 2013 (HY000): Lost connection to MySQL server during query
No connection. Trying to reconnect...
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'172.19.28.35' (using password: YES)
ERROR:
Can't connect to the server
```

6. ¿Cuántas consultas se están ejecutando hasta el momento en tu servidor MYSQL? ¿Y si se trata de consultas lentas?

Con la orden `show processlist;` podemos ver qué consultas se están ejecutando ahora mismo:

```
Terminal - gonzalez27u@adeangeldavid: ~
File Edit View Terminal Tabs Help
mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | User          | Host      | db  | Command | Time | State          | Info          |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 5  | event_scheduler | localhost | NULL | Daemon  | 2033 | Waiting on empty queue | NULL          |
| 23 | root           | localhost | NULL | Query   | 0    | init           | show processlist |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

Como podemos ver, sólo se está ejecutando nuestra consulta.

7. Un estado informa el sobre el máximo de conexiones concurrentes que se ha dado en la sesión de trabajo. ¿Cuál es?

El estado es `'max_used_connections'` y podemos verlo con la orden `show status like 'max_used_connections';` :

```
root@adeangeldavid: /home/gonzalez27u x gonz
mysql> show status like 'max_used_connections';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Max_used_connections | 4     |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```


2.2. Variables dinámicas

1. Detalla los posibles atributos que tendría una variable de servidor como "port".

Command-Line Format	--port=port_num	
System Variable	Name	port
	Variable Scope	Global
	Dynamic Variable	No
Permitted Values	Type	Entero
	Default	3306
	Min Value	0
	Max Value	65535

2. ¿Cómo podemos saber si una variable es dinámica o no?

Una variable es dinámica cuando es imposible de modificar su valor pero cambia según los eventos que ocurran en el servidor.

3. ¿Qué hace la variable "uptime"?

Muestra el tiempo que lleva activo el servidor desde su último reinicio o puesta a 0 de las variables.

- **Indica su valor en tu servidor**

Para verlo debemos ejecutar la orden `show status like 'uptime';` :

```
root@adeangeldavid: /home/gonzalez27u x
mysql> show status like 'uptime';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Uptime       | 4939  |
+-----+-----+
1 row in set (0.00 sec)
```

- ¿Es posible modificar su valor con comando SET?

No, ya que es una variable dinámica y se modifican en tiempo de ejecución.

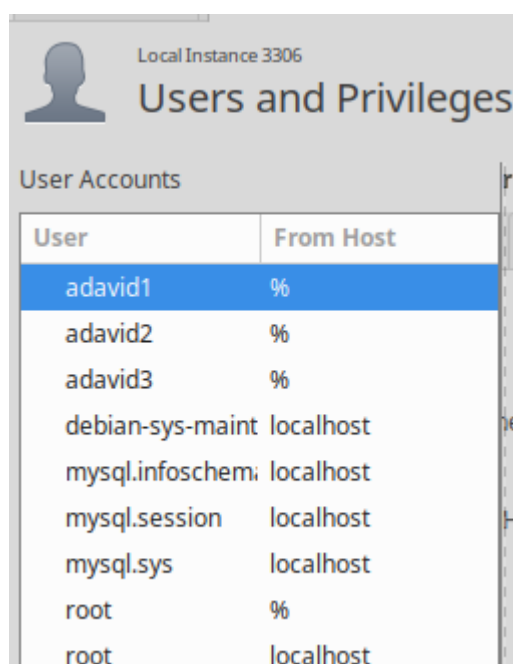
4.- Localiza la variable que establece el límite de conexiones concurrentes. ¿Cuál es?

El estado es 'max_used_connections' y podemos verlo con la orden `show status like 'max_used_connections';` :

```
root@adeangeldavid: /home/gonzalez27u x gonz.  
mysql> show status like 'max_used_connections';  
+-----+-----+  
| Variable_name | Value |  
+-----+-----+  
| Max_used_connections | 4 |  
+-----+-----+  
1 row in set (0.00 sec)  
  
mysql> 
```

- Conéctate con tres instancias de la herramienta cliente mysql utilizando un usuario creado previamente (podéis crear el usuario desde WorkBench, es más sencillo).

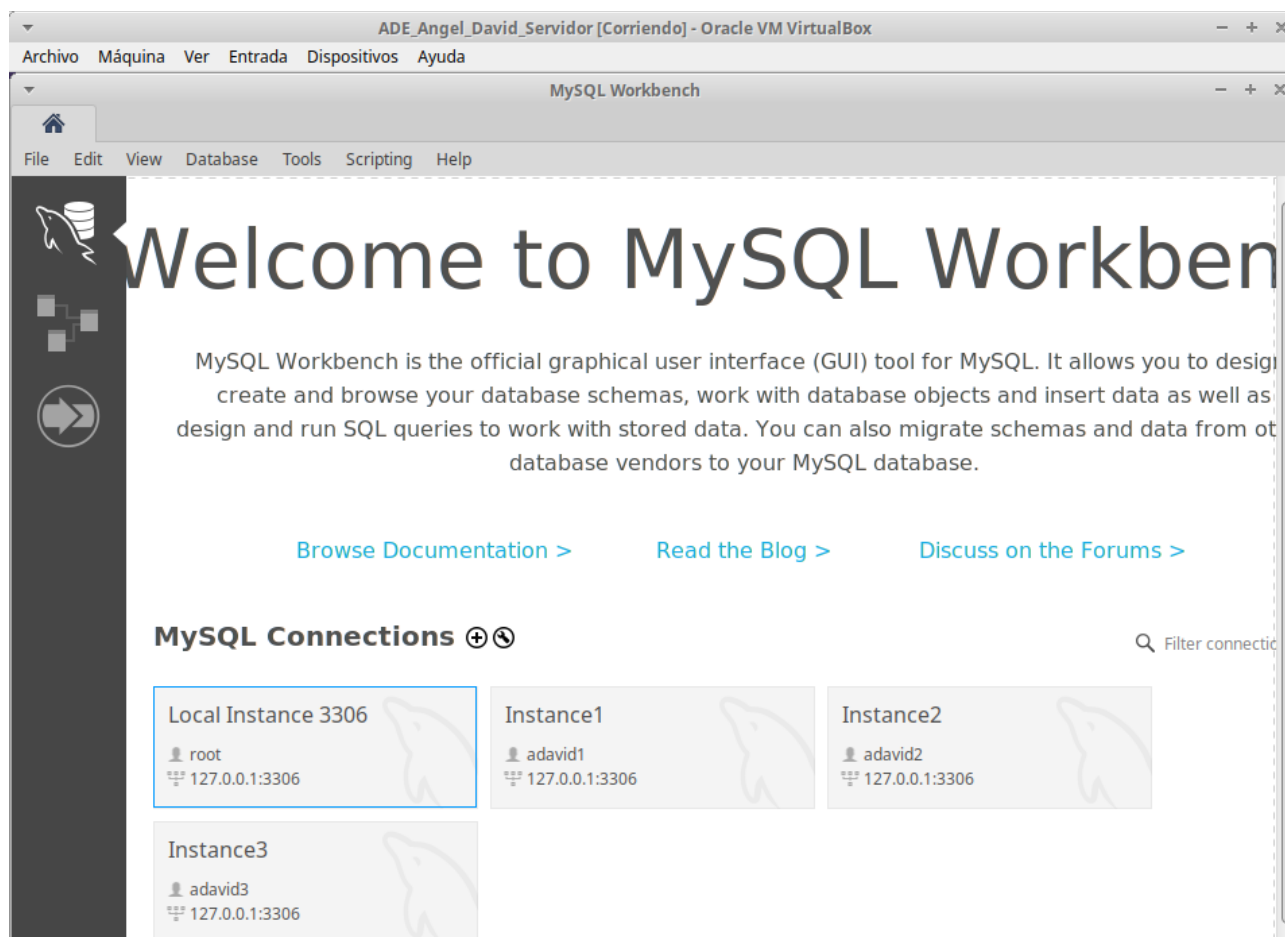
Primero debemos crear el usuario, que en mi caso he creado 3:



The screenshot shows the 'Users and Privileges' window in MySQL Workbench for 'Local Instance 3306'. It displays a table of 'User Accounts' with columns 'User' and 'From Host'.

User	From Host
adavid1	%
adavid2	%
adavid3	%
debian-sys-maint	localhost
mysql.infoschem	localhost
mysql.session	localhost
mysql.sys	localhost
root	%
root	localhost

Luego creamos tres instancias:



Y nos conectamos con las tres. Luego abrimos MySQL y ejecutamos la orden `show processlist;` :

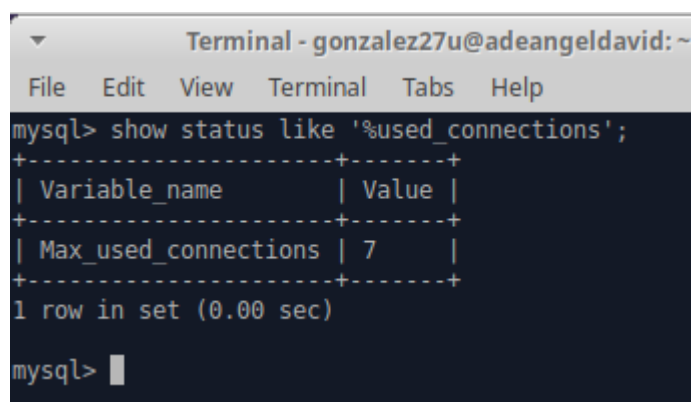
```
mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id | User      | Host      | db  | Command | Time | State          | Info          |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 5  | event_scheduler | localhost | NULL | Daemon  | 418  | Waiting on empty queue | NULL          |
| 20 | adavid1    | localhost:47564 | NULL | Sleep   | 49   |                 | NULL          |
| 21 | adavid1    | localhost:47566 | NULL | Sleep   | 49   |                 | NULL          |
| 23 | adavid2    | localhost:47570 | NULL | Sleep   | 44   |                 | NULL          |
| 24 | adavid2    | localhost:47572 | NULL | Sleep   | 44   |                 | NULL          |
| 26 | adavid3    | localhost:47576 | NULL | Sleep   | 38   |                 | NULL          |
| 27 | adavid3    | localhost:47578 | NULL | Sleep   | 38   |                 | NULL          |
| 28 | root       | localhost  | NULL | Query   | 0    | init           | show processlist |
+-----+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

Podemos ver que las tres instancias están conectadas.

- **Comprueba el valor de la variable de estado que indica cuantas conexiones simultáneas fueron establecidas al servidor Mysql (para ello busca la cadena 'connection' en las variables de estado).**

La variable es 'max_used_connections' y podemos verlo con la orden `show status like 'max_used_connections';`:

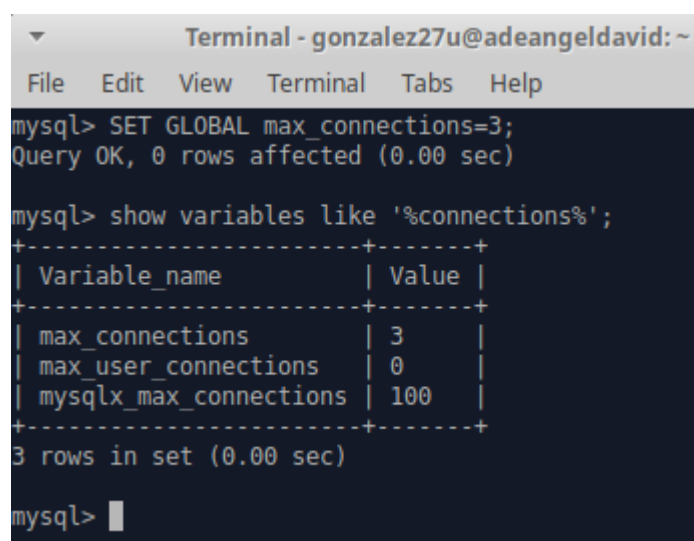


```
Terminal - gonzalez27u@adeangeldavid: ~
File Edit View Terminal Tabs Help
mysql> show status like '%used_connections';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Max_used_connections | 7 |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

- **Modifica la variable del sistema que limita el número de conexiones simultáneas a 3 (busca que variable es de la misma forma que en el paso anterior).**

Modificamos la variable anterior con la orden `set global max_connections=3;`:



```
Terminal - gonzalez27u@adeangeldavid: ~
File Edit View Terminal Tabs Help
mysql> SET GLOBAL max_connections=3;
Query OK, 0 rows affected (0.00 sec)

mysql> show variables like '%connections%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| max_connections | 3 |
| max_user_connections | 0 |
| mysqlx_max_connections | 100 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

Intenta conectarte con una nueva instancia de Mysql utilizando el usuario creado previamente.

- **Deja la variable del sistema a su valor por defecto. (se iguala con la palabra DEFAULT)**

Accedemos a MySQL con el usuario creado anteriormente (adavid1) e intentamos dejar la variable anterior por defecto con la orden `set global max_connections=default;` :

```
gonzalez27u@adeangeldavid:~$ mysql -u adavid1 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.27 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> SET GLOBAL max_connections=DEFAULT;
ERROR 1227 (42000): Access denied; you need (at least one of) the SUPER or SYSTEM_VARIABLES_ADMIN privilege(s) for this operation
mysql> █
```

Pero podemos ver que no podemos porque no tenemos permiso SUPER (root en nuestro caso).