**Fichero de Opciones y Variables de Servidor**



Daniel Escaño Hernández 2º ASIR

# **1. Índice:**

[**2. Ficheros de opciones**](#_juqhwshk34qr) **3**

[2.1 Encuentra el fichero my.ini (en el caso de Windows) o my.cnf de tu instalación de MySQL:](#_5d38uxjyswtl) 3

[2.2 ¿Cómo se escribe un comentario en este fichero?¿Y un grupo de opciones?¿Todas las opciones tienen un valor?](#_yul3wecinz8q) 4

[2.3 Ejecuta "mysqld --verbose --help" desde una consola para ver una lista de las variables del servidor. Para ver mejor el texto mejor redirecciona la salida a fichero:](#_n9aoelx1u45i) 4

[2.4 Explica qué significan y que se consigue con cada una de las variables del siguiente fichero de configuración](#_nqdlfgchwol8) 5

[**3. Variables del servidor**](#_3yuad4t75yoe) **6**

[3.1 Define qué son las variables del servidor.](#_b714u3g09f01) 6

[3.2 Usa el comando "SHOW VARIABLES" para conocer el valor de todas las variables y enviar el resultado a un fichero:](#_jgv1uzsx0hf2) 7

[3.3 Repite lo anterior para mostrar solo las variables relacionadas con el motor "InnoDB".](#_tcabfbfdcwb2) 8

[3.4 Para gestionar variables tenemos, como hemos visto, el comando SHOW "comando":](#_22ap7ylllt2v) 10

[**4. Variables de estado**](#_t90ouojoal2y) **12**

[4.1 Explica qué función tienen las variables de estado.](#_c587vcl8tkih) 12

[4.2 Usa el comando "SHOW STATUS" para conocer el valor de todas las variables](#_8fiq9qscd7ks) 12

[4.3 Haz que uno o más de tus compañeros se conecten a tu servidor:](#_shf5oehlop2a) 13

[4.4 Comprueba quién está conectado usando el comando correspondiente](#_kzvlmb6kucmn) 13

[4.5 Intenta desconectarlo con el comando "kill"](#_ma3vw2y2i3xs) 14

[4.6 ¿Cuántas consultas se están ejecutando hasta el momento en tu servidor MYSQL? ¿Y si se trata de consultas lentas?](#_3gviekxi7amx) 14

[4.7 Un estado informa sobre el máximo de conexiones concurrentes que se ha dado en la sesión de trabajo.](#_99h1djx4u5ty) 15

[**5. Variables dinámicas**](#_6617vhums4nh) **15**

[5.1 Detalla los posibles atributos que tendría una variable de servidor como "port".](#_j5m7d51h94w8) 15

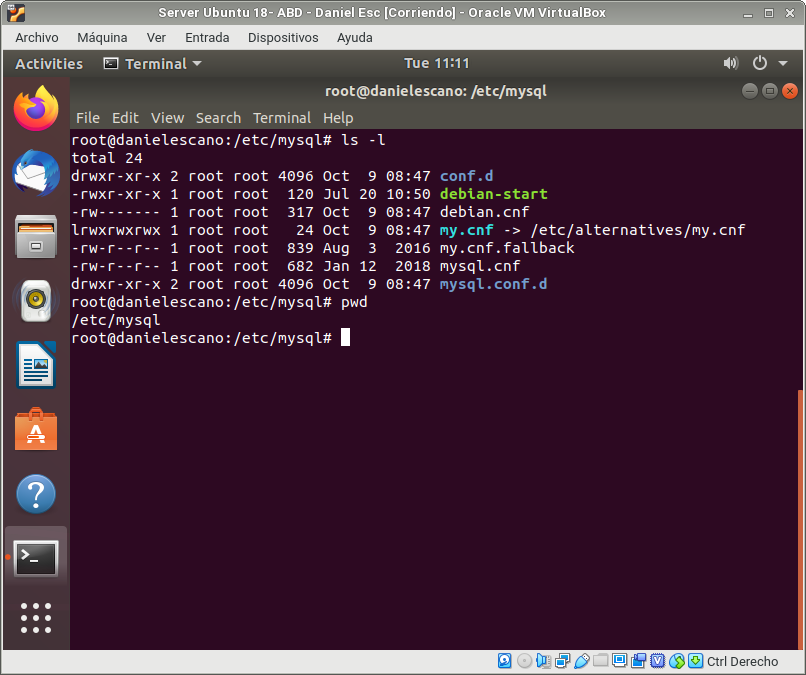
[5.2 ¿Cómo podemos saber si una variable es dinámica o no?](#_j23iapm68gv0) 16

[5.3 ¿Qué hace la variable "uptime"?](#_xfm9okg2iv3c) 16

[5.4 Localiza la variable que establece el límite de conexiones concurrentes.](#_raofn7u4q3it) 17

# **2. Ficheros de opciones**

### **2.1 Encuentra el fichero my.ini (en el caso de Windows) o my.cnf de tu instalación de MySQL:**



En nuestro caso podemos encontrar el fichero de configuración en la ruta: /etc/mysql

Este archivo se utiliza para indicar todo tipo de parámetros, configuraciones y valores a nuestro MYSQL. Se puede decir que contiene las opciones de arranque.

Estas son otras rutas típicas donde se puede llegar a encontrar el archivo de configuración en caso de que no lo encontremos.

Ruta Objeto

/etc/my.cnf Opciones Globales

/etc/mysql/my.cnf Opciones Globales

*SYSCONFDIR*/my.cnf Opciones Globales

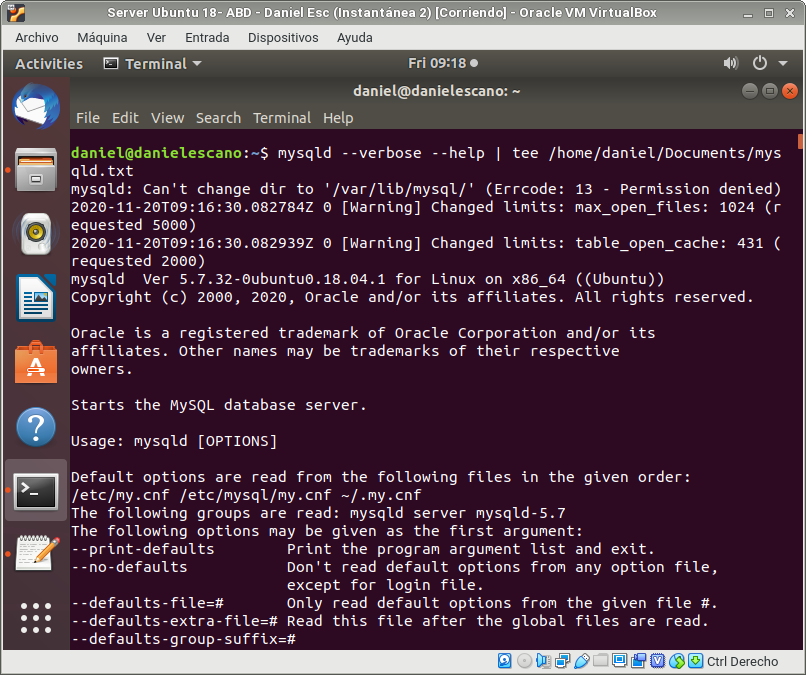
### **2.2 ¿Cómo se escribe un comentario en este fichero?¿Y un grupo de opciones?¿Todas las opciones tienen un valor?**

Para escribir un comentario en nuestro fichero de configuración necesitaremos de una ‘#’ al principio de la línea para que sea considerado un comentario dicha línea.

Para escribir un grupo de opciones son necesarios 2 elementos, el primero es indicar el nombre del programa o grupo al cual le quieres añadir las opciones entre *[corchetes]* y el segundo elemento son las opciones que podemos introducir para que estas, carguen al arrancar.

Contamos con 2 formas de declarar opciones, unas necesitan valores para funcionar al ser archivos con varias opciones y las otras, no es necesario indicarle un valor, su principal rasgo detector es si no contiene un ‘=’ para ser declarada.

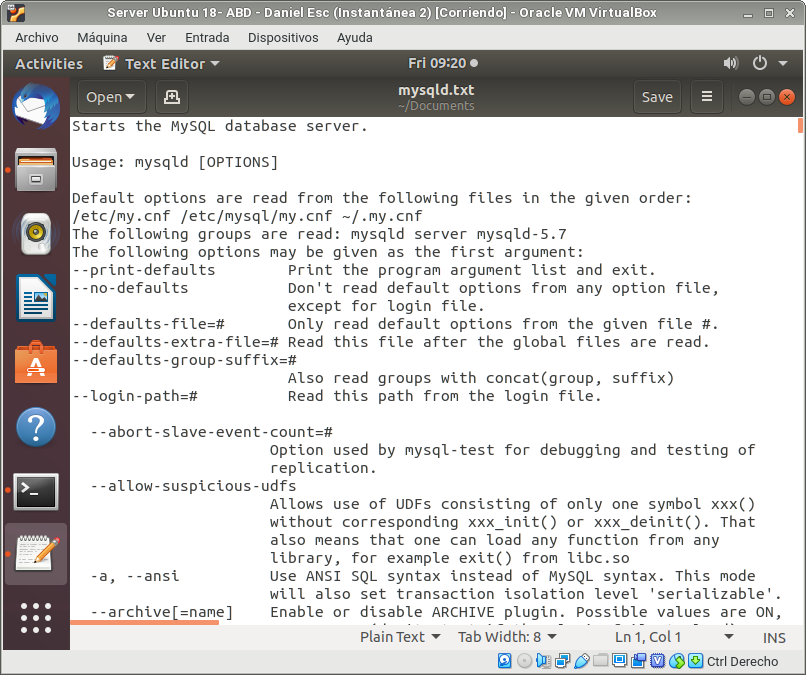
### **2.3 Ejecuta "mysqld --verbose --help" desde una consola para ver una lista de las variables del servidor. Para ver mejor el texto mejor redirecciona la salida a fichero:**



Para esto utilizamos el comando *tee* y nos quedaría una línea de comando así:

| mysqld --verbose --help | tee /home/daniel/Documents/mysqld.txt |
| --- |

Y nos quedaría este archivo:



### **2.4 Explica qué significan y que se consigue con cada una de las variables del siguiente fichero de configuración**

| [client]  port=3306  password="telesforo";  [mysqld]  port=3306  key\_buffer\_size=16M  max\_allowed\_packet=8M  [mysqldump]  quick |
| --- |

En el primer bloque de la configuración del grupo ‘client’, nos está indicando el puerto y la contraseña de los clientes, con estos parámetros nos está indicando que dicho puerto y contraseña se utiliza como predeterminada en todos los usuarios MYSQL.

El bloque del grupo ‘mysqld’, nos indica un puerto, la capacidad o tamaño máximo de buffer que acepta para llaves, 16 Megabytes en este parámetro y además nos indica la capacidad máxima permitida de paquete, estableciendolo en 8 Megabytes.

La última parte, que se refiere a ‘Mysqldump’, añade la opción quick. Con Mysqldump se puede recuperar y volcar el contenido de las tablas fila por fila, o puede recuperar todo el contenido de una tabla y almacenarlo en la memoria antes de volcarlo. El almacenamiento en búfer en la memoria puede ser un problema si estás volcando tablas grandes.

Para volcar tablas fila por fila, se utiliza la opción --quick.

# **3. Variables del servidor**

### **3.1 Define qué son las variables del servidor.**

El servidor MySQL mantiene muchas variables del sistema que configuran su funcionamiento.

Cada variable del sistema tiene un valor predeterminado. Las variables del sistema se pueden configurar al inicio del servidor usando opciones en la línea de comando o en un archivo de opciones.

La mayoría de ellos se pueden cambiar dinámicamente en tiempo de ejecución utilizando la instrucción SET, que le permite modificar el funcionamiento del servidor sin tener que detenerlo y reiniciarlo.

También puede utilizar valores de variables del sistema en expresiones.

Hay varias formas de ver los nombres y valores de las variables del sistema, entre ellas:

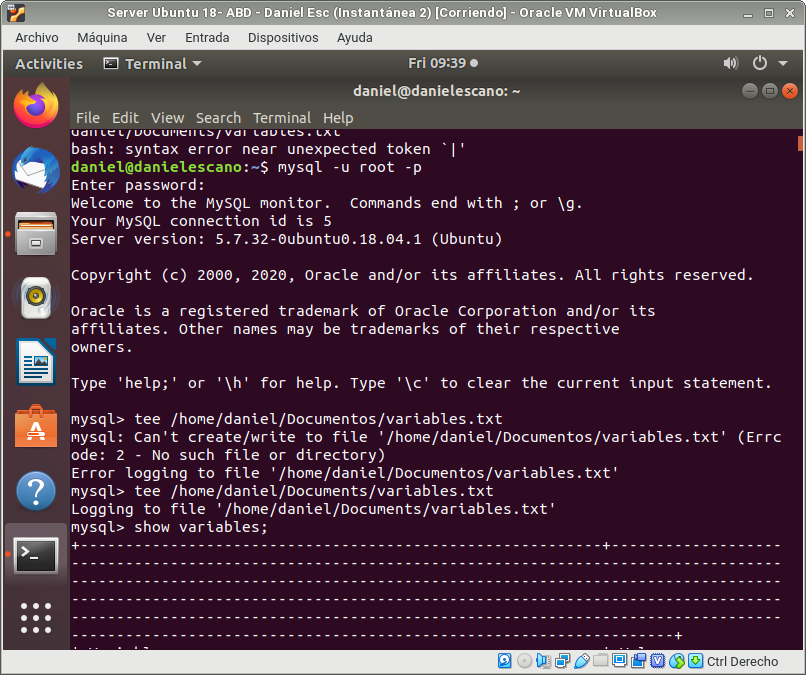
| mysqld --verbose --help |
| --- |
| mysqld --no-defaults --verbose --help |

### **3.2 Usa el comando "SHOW VARIABLES" para conocer el valor de todas las variables y enviar el resultado a un fichero:**

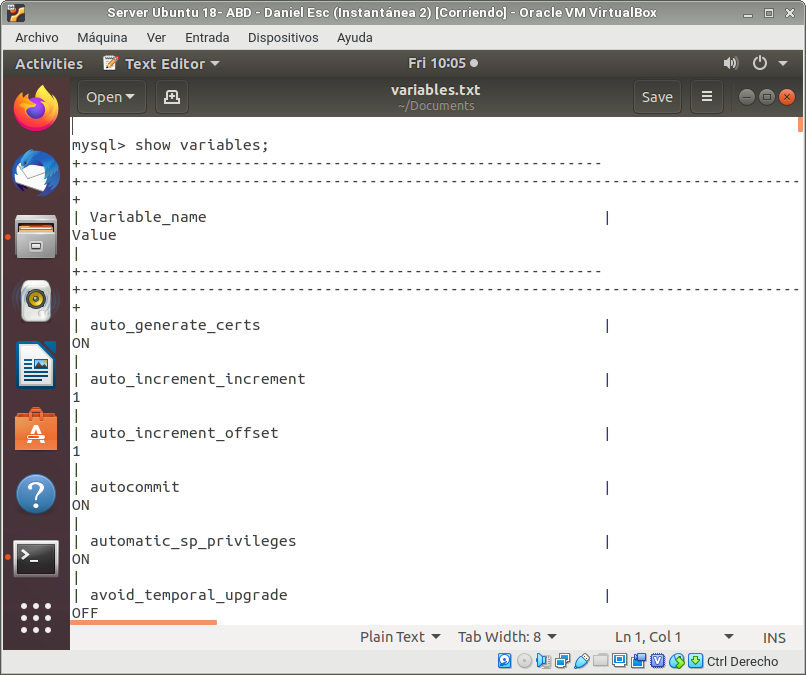
El comando ‘show variables’ se utiliza para ver los valores actuales de las variables utilizados por el servidor en ejecución.

Necesitamos ejecutar la consulta y desviar su salida a un fichero, para ello utilizamos el comando *tee,* de forma que resulta de la siguiente secuencia de comandos:

| mysql -u root -p |
| --- |
| tee /home/daniel/Documents/variables.txt |
| show variables; |
| notee; |



Nos quedaría un archivo con formato .txt de la siguiente forma:

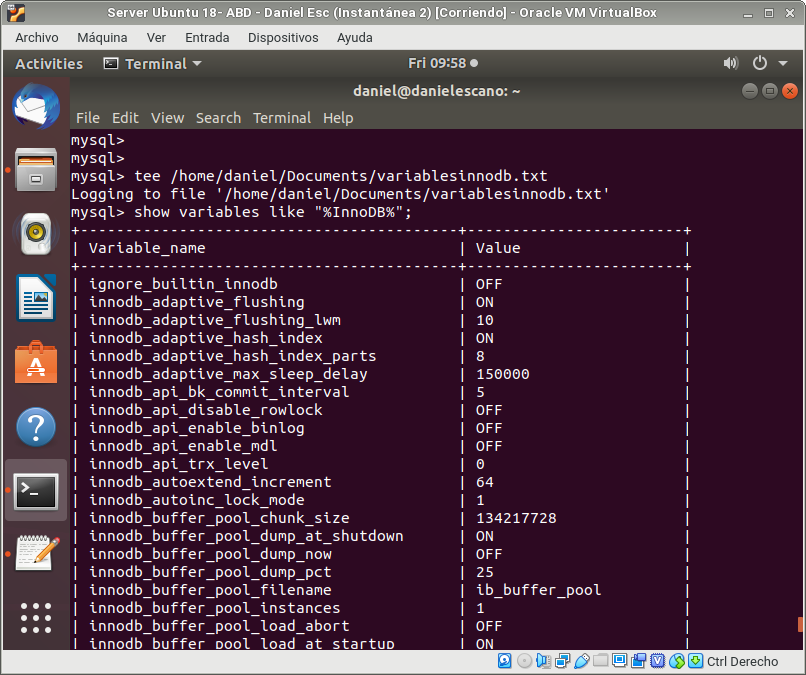


### **3.3 Repite lo anterior para mostrar solo las variables relacionadas con el motor "InnoDB".**

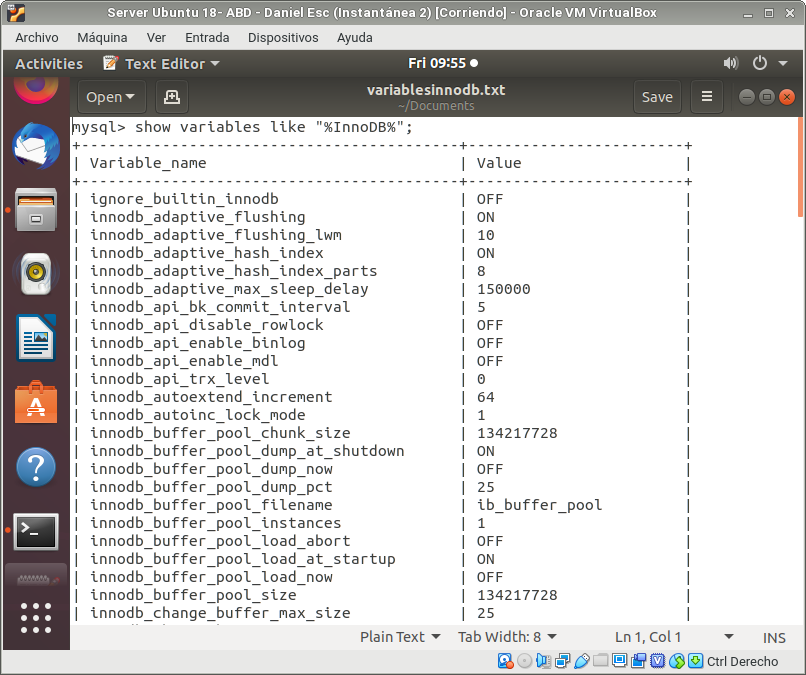
Para visualizar las variables relacionadas con el motor InnoDB necesitaremos del comando ‘show variables’ que se utiliza para visualizar los valores actuales de las variables utilizados por el servidor en ejecución, como acabamos de ver.

Se nos propone que ejecutemos esta consulta y creemos un fichero donde se almacenará la salida de esta consulta, para esta tarea utilizamos nuevamente el comando *tee* y resultaría en los siguientes comandos:

| mysql -u root -p |
| --- |
| tee /home/daniel/Documents/variablesinnodb.txt |
| show variables like “%InnoDB%”; |
| notee; |



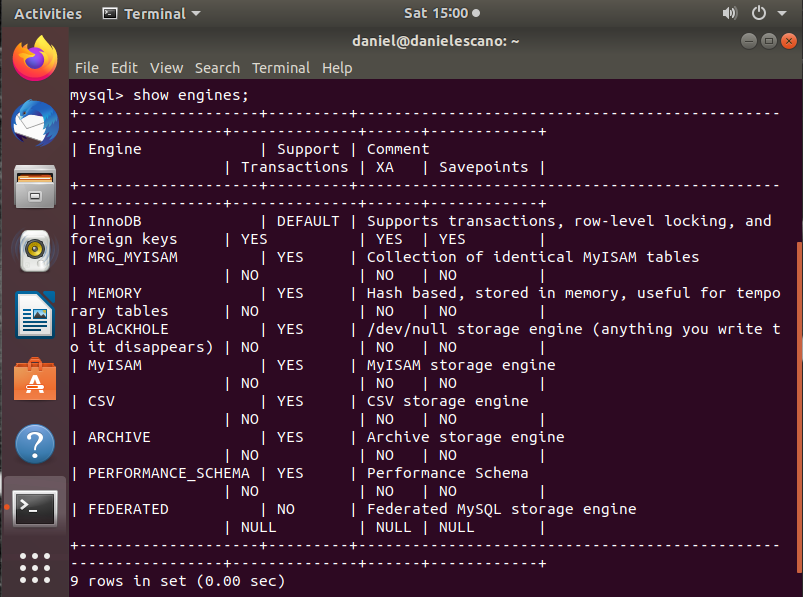
Y nos resultaría un archivo en formato .txt de la siguiente forma:



### **3.4 Para gestionar variables tenemos, como hemos visto, el comando SHOW "comando":**

Mostrar todos los motores de almacenamiento:

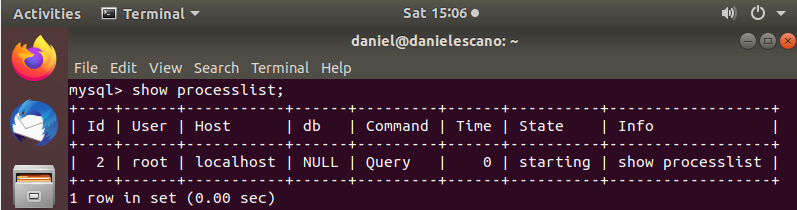
| show engines; |
| --- |



Esto es particularmente útil para verificar si un motor de almacenamiento es compatible o para ver cuál es el motor predeterminado.

Mostrar el estado actual del servidor y Averiguar todos los clientes que están conectados al servidor:

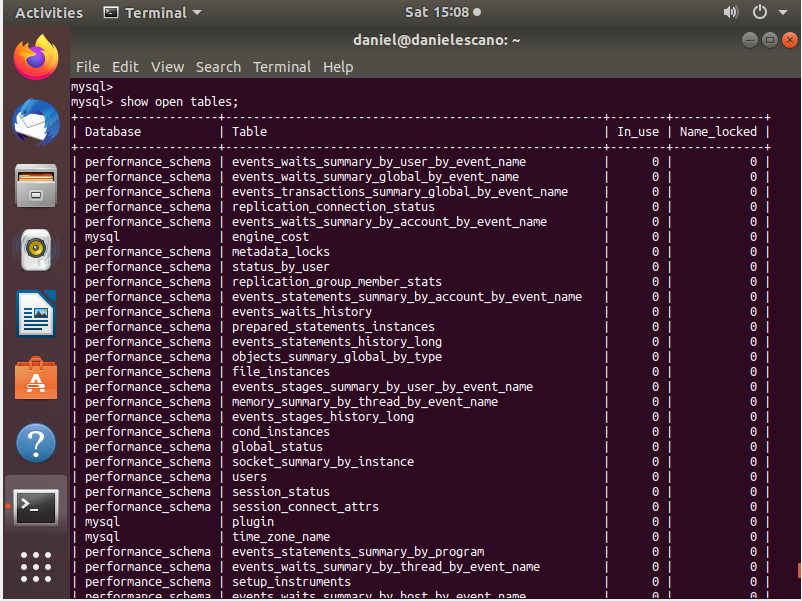
| show processlist; |
| --- |



Aparecerá tanto la lista de procesos como los usuarios que los están ejecutando.

Conocer todas las tablas que están abiertas:

| show open tables; |
| --- |



Esta consulta enumera las tablas no temporales que están abiertas actualmente en la caché de tablas.

# **4. Variables de estado**

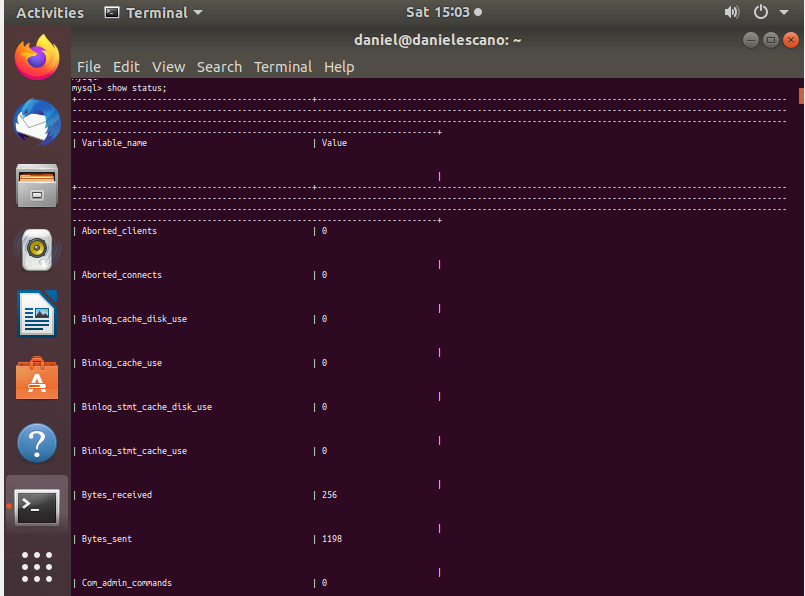
### **4.1 Explica qué función tienen las variables de estado.**

MySQL almacena en sus tablas de sistema, información acerca de las operaciones realizadas en el servidor, estas son las variables de estado.

Esta información puede ser de utilidad para monitorizar nuestra base de datos o para identificar posibles fallos o errores.

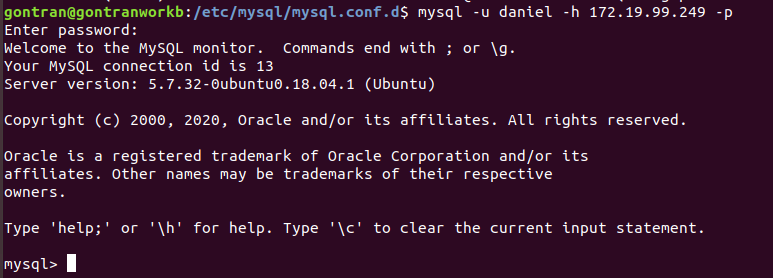
Con ayuda de las variables de estado, nos anticipamos a posibles fallos críticos de nuestro servidor con una correcta configuración y gestión.

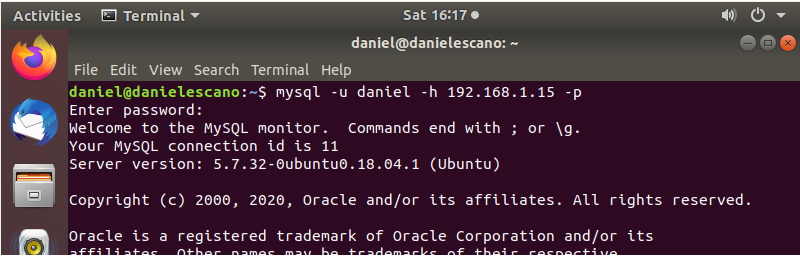
### **4.2 Usa el comando "SHOW STATUS" para conocer el valor de todas las variables**



Esta declaración no requiere ningún privilegio. Solo requiere la capacidad de conectarse al servidor. Nos muestra el nombre y el valor de las variables.

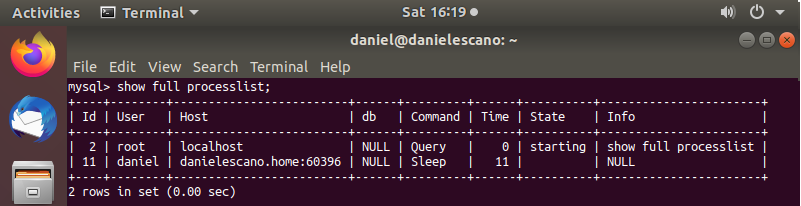
### **4.3 Haz que uno o más de tus compañeros se conecten a tu servidor:**





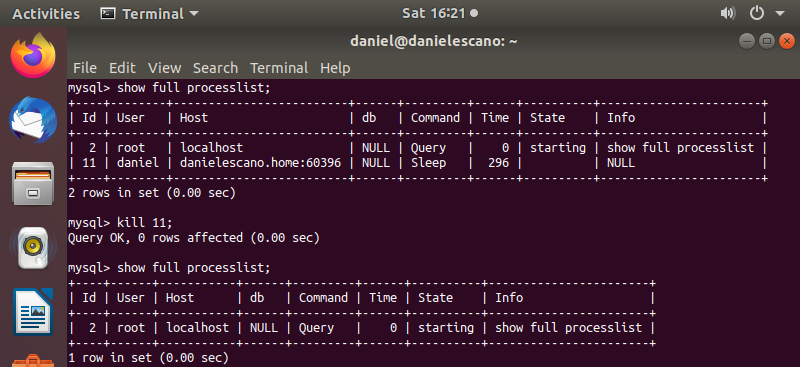
Hice una segunda conexión yo mismo debido a que no contaba con compañeros para terminar las cuestiones.

### **4.4 Comprueba quién está conectado usando el comando correspondiente**



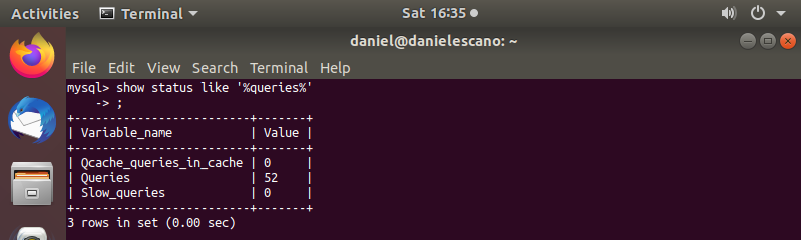
Utilizamos el comando *show processlist* para obtener la identificación del proceso sobretodo, además de parámetros como nombre de usuario, host, base de datos a la que está accediendo, tiempo de conexión, estado y la consulta.

### **4.5 Intenta desconectarlo con el comando "kill"**



Sabiendo la identificación de la sesión, podemos ejecutar el comando kill añadiendo la ID de la sesión a finalizar, en este caso: 11.

### **4.6 ¿Cuántas consultas se están ejecutando hasta el momento en tu servidor MYSQL? ¿Y si se trata de consultas lentas?**

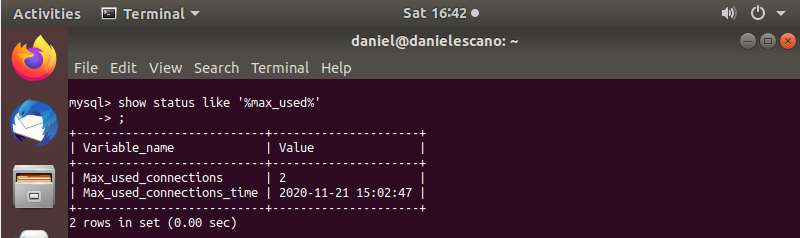


Utilizamos la consulta

| show status like ‘%queries%’; |
| --- |

Nos mostrará el número total de consultas realizadas hasta el momento, que es 52 y el número total que, según la configuración establecida, considera consultas lentas, en este caso 0.

### **4.7 Un estado informa sobre el máximo de conexiones concurrentes que se ha dado en la sesión de trabajo.**



Utilizando la consulta

| show status like ‘%max\_used%’; |
| --- |

Nos saca por pantalla lo pedido, el máximo de conexiones simultáneas, además de la fecha y hora.

# **5. Variables dinámicas**

### **5.1 Detalla los posibles atributos que tendría una variable de servidor como "port".**

| **Command-Line Format** | --port=port\_number | | |
| --- | --- | --- | --- |
| **System Variable** | **Name** | port | |
| **Variable Scope** | Global | |
| **Dynamic Variable** | No | |
| **Permitted Values** | **Type** | Integer / Entero | |
| **Default** | 3306 | |
| **Min Value** | 0 | |
| **Max Value** | 65535 | |

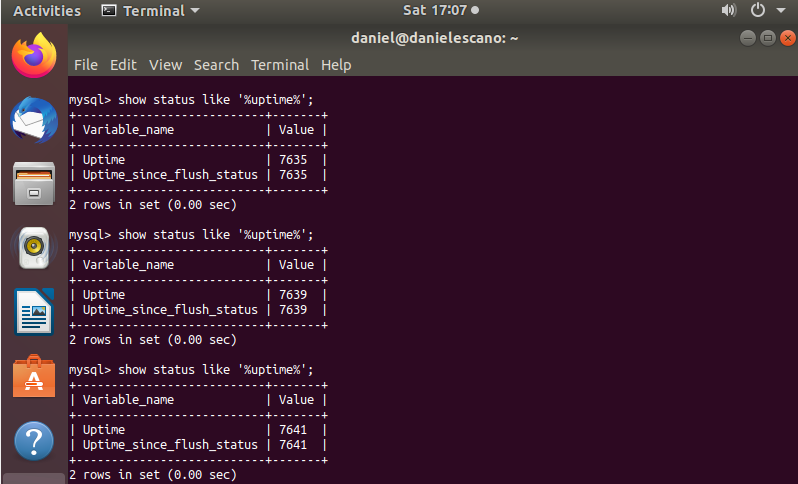
### **5.2 ¿Cómo podemos saber si una variable es dinámica o no?**

Podemos saberlo de forma sencilla, si la consulta SET funciona con la variable, eso nos indica que es dinámica, ya que la sentencia SET únicamente funciona con variables que pueden ser modificadas en tiempo de ejecución como lo son las dinámicas.

O consultando directamente la información oficial en específico de la variable que queramos saber.

### **5.3 ¿Qué hace la variable "uptime"?**

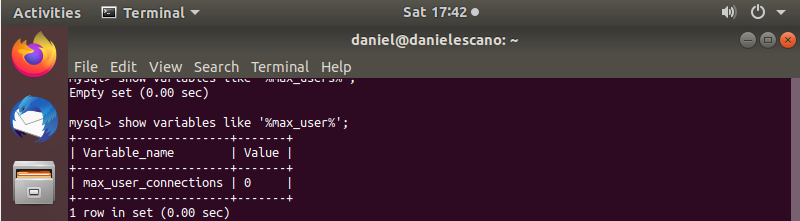
* + Indica su valor en tu servidor
  + ¿Es posible modificar su valor con comando SET?



La variable uptime nos indica el tiempo que lleva activo el servidor desde su último reinicio, ya sea general o de configuración.

No es posible su modificación con la consulta SET al tratarse de una variable de estado.

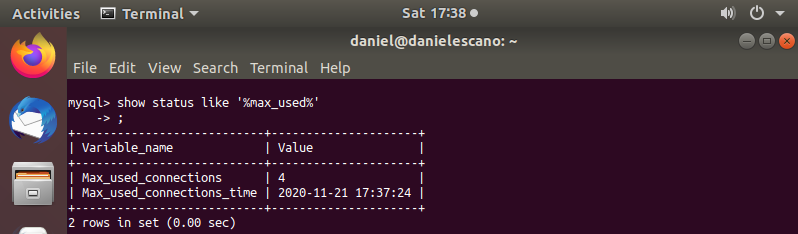
### **5.4 Localiza la variable que establece el límite de conexiones concurrentes.**



Para ello utilizamos la siguiente consulta:

| show variables like ‘%max\_user%’; |
| --- |

* Conéctate con tres instancias de la herramienta cliente **mysql** utilizando un usuario creado previamente.
* Comprueba el valor de la variable de estado que indica cuántas conexiones simultáneas fueron establecidas al servidor Mysql.



Esto lo podemos conseguir con la próxima consulta:

| show status like ‘%max\_used%’; |
| --- |

Aquí podemos observar como hay 4 conexiones activas en nuestro servidor.

* Modifica la variable del sistema que limita el número de conexiones simultáneas a 3



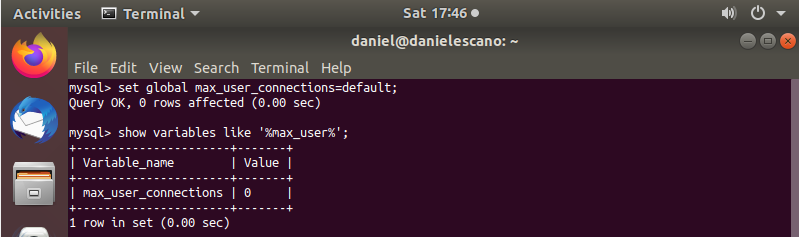
Estas sentencias han de ser ejecutadas por el usuario root o un usuario que contenga los permisos iguales a los de root para evitar cualquier fallo por denegación de permisos.

Introducimos la siguiente consulta:

| set global max\_user\_connectios=3; |
| --- |

Y comprobamos que el cambio ha surgido efecto.

* Deja la variable del sistema a su valor por defecto.



Con la sentencia escrita a continuación:

| set global max\_user\_connections=default; |
| --- |

Conseguimos devolver la variable a su valor por defecto o predeterminado.