

# MySQL (I)

- MySQL es un Sistema Gestor de Bases de Datos (SGBD) relacionales.
- Su propietario actual es Oracle Corporation.
- Es software bajo licencia GNU GPL.
  - Si se utiliza en productos con licencia privativa debe adquirirse una licencia para su uso.
- Funciona en multitud de sistemas operativos:
  - Linux con kernel 2.0 o superior.
  - Windows 7, 8, Server 2003, Server 2008.
  - FreeBSD 2.X y superior.
  - Mac OS X.

# MySQL (II)

- Soporta dos motores distintos de almacenamiento:
  - MyISAM:
    - No posee control de transacciones.
    - Lectura de datos muy rápida.
    - Ideal para aplicaciones web con uso intensivo de lectura de datos y pocas modificaciones.
  - InnoDB:
    - Posee control de transacciones.
    - Uso general como cualquier otra base de datos.
  - Los dos motores pueden funcionar de forma simultánea.

# MySQL (III)

- Las conexiones a la base de datos se realizan de forma general mediante TCP/IP.
  - En Windows permite conexiones usando named pipes.
  - En Linux permite conexiones usando ficheros sockets.
- El tamaño máximo del espacio de tablas es de 64 Tbytes.
  - La limitación la impone el sistema operativo:
    - Linux 2.4 o superior: 16 Tbytes en ext4.
    - Windows: 16 Tbytes en NTFS.
- Utiliza privilegios y contraseñas seguro.
  - El tráfico de contraseñas se encripta en la conexión.

# Instalación de MySQL (I)

- La instalación de MySQL esta formada por varios paquetes:
  - mysql: Programas cliente y librerías compartidas.
  - mysql-libs: Librerías compartidas.
  - mysql-server: Servidor de MySQL.
- Existen además otros paquetes adicionales:
  - mysql-bench: Scripts para comprobar el rendimiento.
  - mysql-connector-java: JDBC driver para MySQL.
  - mysql-connector-odbc: ODBC driver para MySQL.
  - mysql-devel: Ficheros para desarrollar aplicaciones que usen MySQL.

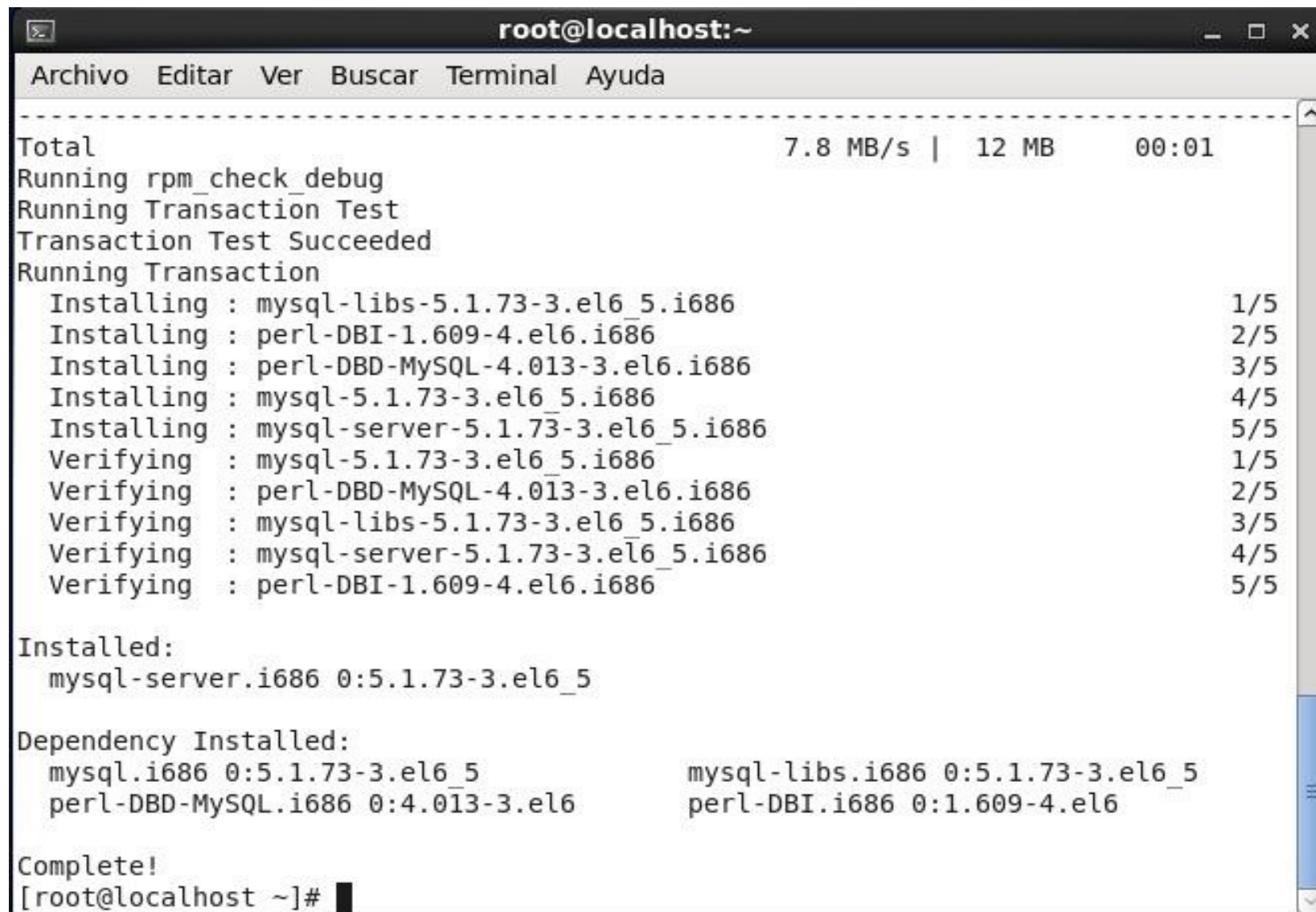
# Instalación de MySQL (II)

- La instalación de MySQL se realiza ejecutando el comando:

```
yum -y install mysql-server
```

- El comando instala el paquete `mysql-server` con todas las dependencias necesarias.

# Instalación de MySQL (III)

A terminal window titled 'root@localhost:~' with a menu bar (Archivo, Editar, Ver, Buscar, Terminal, Ayuda). It shows the progress of installing MySQL packages. A progress bar at the top indicates 7.8 MB/s, 12 MB, and 00:01. The output shows the transaction test succeeded and the installation of mysql-server and its dependencies. The installation is complete.

```
root@localhost:~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
-----
Total                                         7.8 MB/s | 12 MB    00:01
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : mysql-libs-5.1.73-3.el6_5.i686      1/5
  Installing : perl-DBI-1.609-4.el6.i686           2/5
  Installing : perl-DBD-MySQL-4.013-3.el6.i686      3/5
  Installing : mysql-5.1.73-3.el6_5.i686           4/5
  Installing : mysql-server-5.1.73-3.el6_5.i686    5/5
  Verifying  : mysql-5.1.73-3.el6_5.i686           1/5
  Verifying  : perl-DBD-MySQL-4.013-3.el6.i686     2/5
  Verifying  : mysql-libs-5.1.73-3.el6_5.i686      3/5
  Verifying  : mysql-server-5.1.73-3.el6_5.i686   4/5
  Verifying  : perl-DBI-1.609-4.el6.i686           5/5

Installed:
  mysql-server.i686 0:5.1.73-3.el6_5

Dependency Installed:
  mysql.i686 0:5.1.73-3.el6_5      mysql-libs.i686 0:5.1.73-3.el6_5
  perl-DBD-MySQL.i686 0:4.013-3.el6 perl-DBI.i686 0:1.609-4.el6

Complete!
[root@localhost ~]#
```



# Instalación de MySQL (IV)

- En la instalación se crean:
  - El fichero `/etc/my.cnf`:
    - Fichero de configuración de MySQL.
    - Configura tanto opciones del cliente como del servidor.
  - El directorio `/var/lib/mysql`:
    - Directorio por defecto donde se almacenarán las bases de datos de MySQL.
    - Puede modificarse en el fichero de configuración.

# Instalación de MySQL (V)

- El servidor de MySQL puede arrancarse, pararse o rearrancarse de forma manual con el comando:

```
service mysqld {start|stop|restart}
```

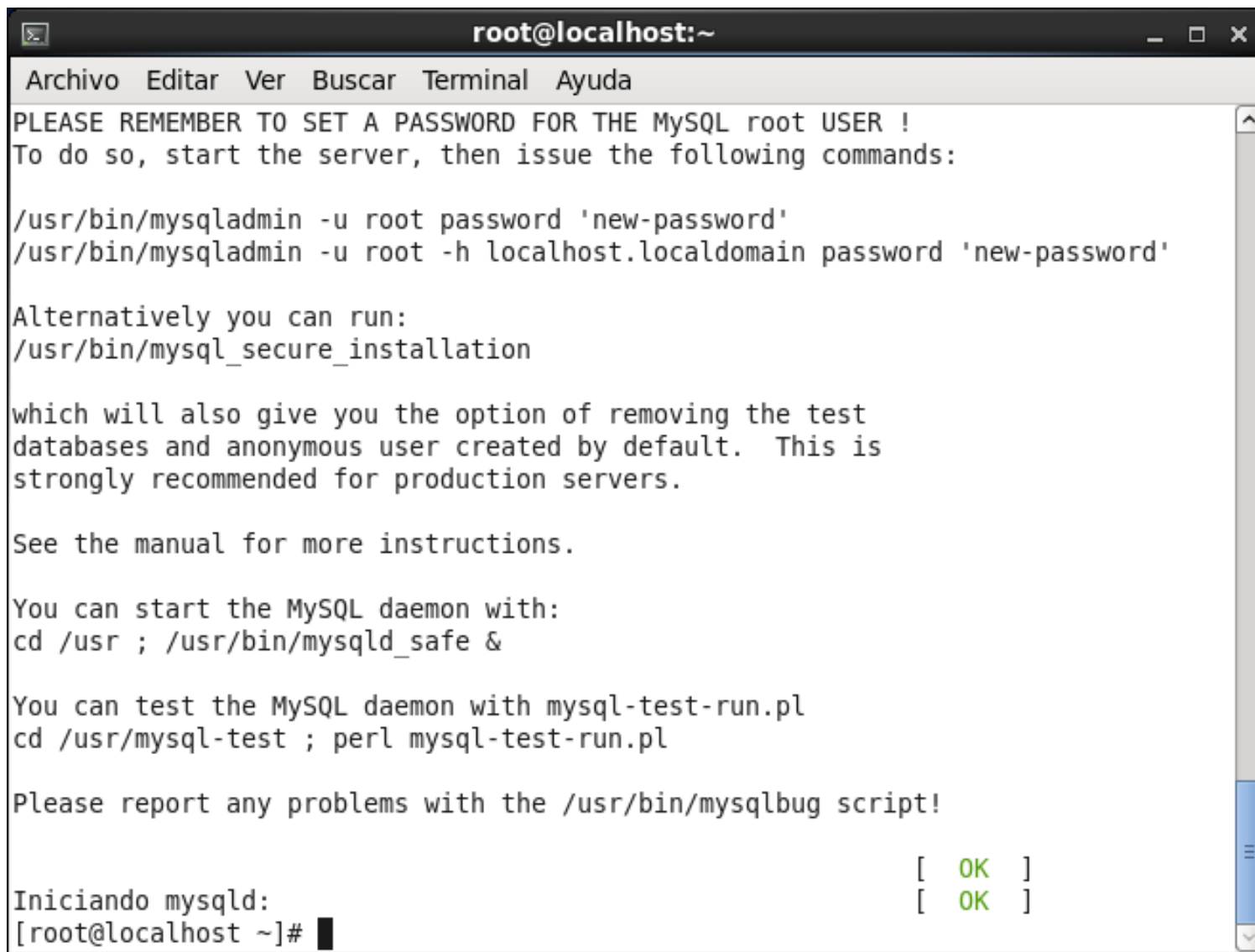
- Puede configurarse también para que arranque o se detenga con el sistema ejecutando el comando:

```
chkconfig mysqld on
```

- La primera vez que arranca crea las bases de datos del sistema.



# Instalación de MySQL (VI)

A terminal window titled 'root@localhost:~' with a menu bar containing 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal displays instructions for setting a password for the MySQL root user. It includes two commands: '/usr/bin/mysqladmin -u root password 'new-password'' and '/usr/bin/mysqladmin -u root -h localhost.localdomain password 'new-password''. It also mentions an alternative command '/usr/bin/mysql\_secure\_installation'. The text explains that this alternative command allows removing the test databases and anonymous user. It then provides instructions to start the MySQL daemon with 'cd /usr ; /usr/bin/mysqld\_safe &' and to test it with 'cd /usr/mysql-test ; perl mysql-test-run.pl'. A note asks to report problems with the '/usr/bin/mysqlbug' script. At the bottom, it shows 'Iniciando mysqld:' followed by two green 'OK' status messages in brackets. The prompt '[root@localhost ~]#' is visible at the very bottom.

```
root@localhost:~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:

/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h localhost.localdomain password 'new-password'

Alternatively you can run:
/usr/bin/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default. This is
strongly recommended for production servers.

See the manual for more instructions.

You can start the MySQL daemon with:
cd /usr ; /usr/bin/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd /usr/mysql-test ; perl mysql-test-run.pl

Please report any problems with the /usr/bin/mysqlbug script!

[ OK ]
[ OK ]

Iniciando mysqld:
[root@localhost ~]#
```

# Instalación de MySQL (VII)

- La instalación inicial del servidor de MySQL es insegura:
  - El administrador (**root**) no posee contraseña.
  - Existen usuarios anónimos creados.
  - El administrador (**root**) puede acceder de forma remota.
  - Existe una base de datos de prueba creada.
- Estos problemas de seguridad deben ser corregidos.

# Instalación de MySQL (VIII)

- Conectarse a la base de datos mediante el cliente de MySQL:

```
mysql -u root
```

- Poner contraseña al usuario root:

```
UPDATE mysql.user SET  
password=PASSWORD('mi_clave') WHERE user='root';
```

- Borrar el usuario anónimo:

```
DELETE FROM mysql.user WHERE user='';
```

- Limitar el acceso del usuario root al ordenador local:

```
DELETE FROM mysql.user WHERE user='root' AND  
host NOT IN ('localhost', '127.0.0.1', '::1');
```

# Instalación de MySQL (IX)

- Borrar la base de datos de test:

```
DROP DATABASE test;
```

- Por último, es necesario volver a releer los privilegios sobre las tablas con el comando:

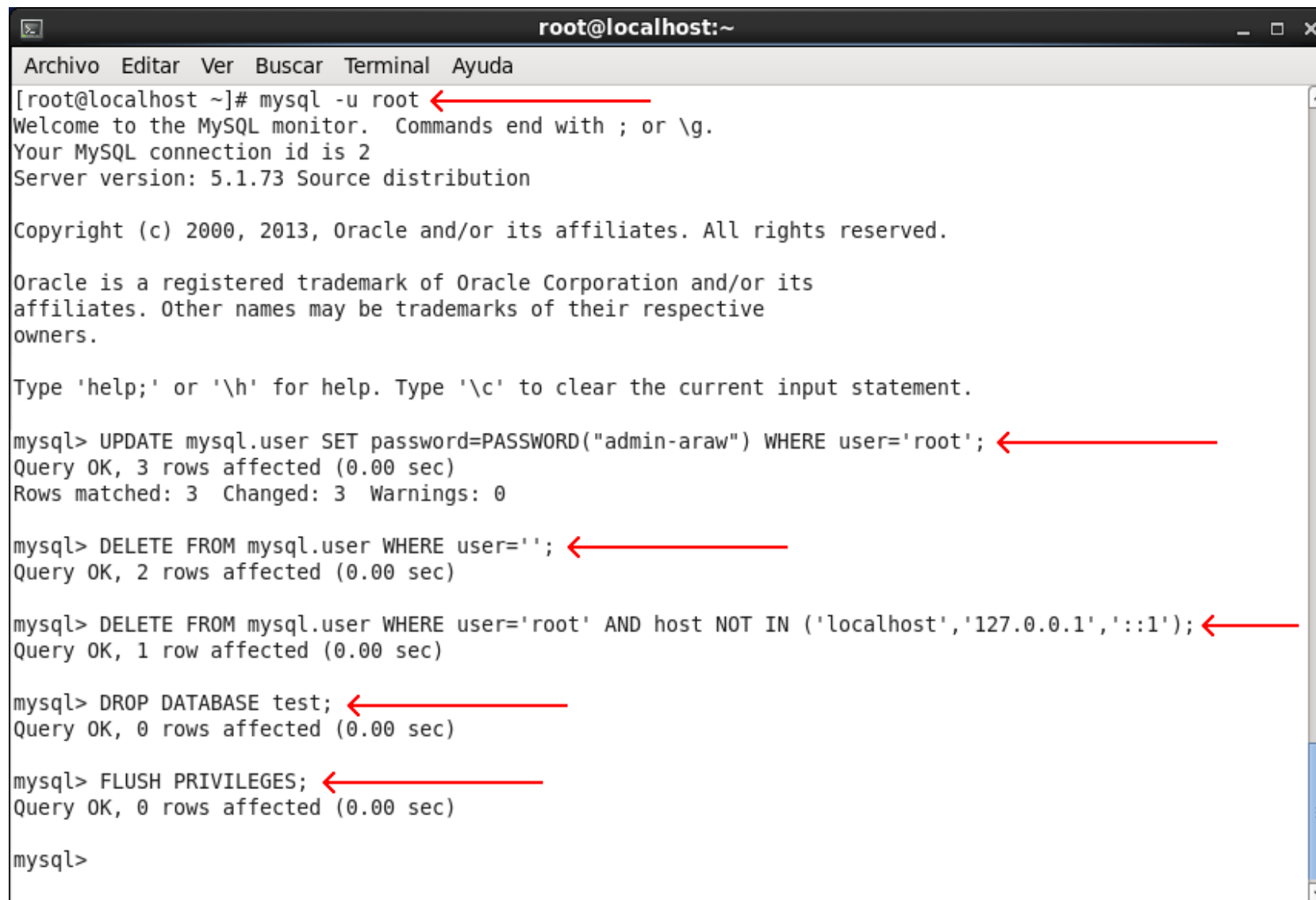
```
FLUSH PRIVILEGES;
```

- Los privilegios también se releen rearrancando el servidor.

- En versiones recientes de MySQL, en la instalación existe un sencillo script de shell que ejecuta todas estas acciones:

```
mysql_secure_installation
```

# Instalación de MySQL (X)



A screenshot of a terminal window titled 'root@localhost:~'. The window contains the following text:

```
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[root@localhost ~]# mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

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> UPDATE mysql.user SET password=PASSWORD("admin-araw") WHERE user='root';
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> DELETE FROM mysql.user WHERE user='';
Query OK, 2 rows affected (0.00 sec)

mysql> DELETE FROM mysql.user WHERE user='root' AND host NOT IN ('localhost','127.0.0.1','::1');
Query OK, 1 row affected (0.00 sec)

mysql> DROP DATABASE test;
Query OK, 0 rows affected (0.00 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

Red arrows point to the following commands in the terminal:

- `mysql -u root`
- `UPDATE mysql.user SET password=PASSWORD("admin-araw") WHERE user='root';`
- `DELETE FROM mysql.user WHERE user='';`
- `DELETE FROM mysql.user WHERE user='root' AND host NOT IN ('localhost','127.0.0.1','::1');`
- `DROP DATABASE test;`
- `FLUSH PRIVILEGES;`

# Configuración de MySQL (I)

- El fichero de configuración de MySQL esta formado por:
  - Secciones: Indican que aplicación utiliza los parámetros que se especifican a continuación.
    - Se especifican con [aplicación].
    - Terminan cuando empieza otra sección o al finalizar el fichero.
  - Parámetros: Variables y valores que se le asignan.
    - Las variables tienen valores por defecto que se utilizan si no son asignados en el fichero.



# Configuración de MySQL (II)

- Las variables existentes en MySQL y sus valores se obtienen:
  - Accediendo a la base de datos mediante el cliente de MySQL:

```
mysql -u root -p
```

- Ejecutando la sentencia:

```
SELECT * FROM  
information_schema.global_variables;
```

- En la versión instalada en CentOS 6.5 (MySQL 5.1) existen 277 variables.
  - Solo un conjunto pequeño de esas variables suelen ser modificadas.

# Configuración de MySQL (III)

- Por defecto, el fichero `/etc/my.cnf` contiene las siguientes líneas:

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to
# prevent assorted security risks
symbolic-links=0
```

```
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

# Configuración de MySQL (IV)

- Existen dos secciones:
  - [mysqld]: Configura valores del servidor de MySQL.
  - [mysqld\_safe]: Configura valores del script de arranque “seguro” de MySQL.
    - El script añade valores opcionales al arranque del servidor de MySQL.
- En ocasiones se suele configurar otra sección [mysql] para indicar valores por defecto que utiliza el cliente de MySQL.
  - Puede cambiarse [mysql] por el nombre del cliente si se utiliza otro cliente distinto.

# Configuración de MySQL (V)

- Los valores de la sección [mysqld] configuran:
  - **datadir**: Directorio donde se encuentran las bases de datos de MySQL.
  - **socket**: Localización del fichero de socket que se utiliza para la comunicación con los clientes locales.
  - **user**: Usuario con el que se ejecuta el servidor de MySQL.
  - **symbolic-links**: Deshabilita que MySQL permita el uso de enlaces simbólicos en el camino a las bases de datos.

# Configuración de MySQL (VI)

- Los valores de la sección [mysqld\_safe] configuran:
  - log-error: Indica la localización del fichero donde MySQL escribirá los logs de la aplicación.
  - pid-file: Localización del fichero que contiene el Process Identification (PID) del servidor de MySQL.

# Configuración de MySQL (VII)

- Existen otras variables de configuración que es conveniente conocer:
  - Permiten modificar el funcionamiento por defecto del servidor.
    - Compatibilidad con contraseñas antiguas.
    - Motor por defecto.
    - Sensibilidad de las tablas al contexto.
    - Puerto por defecto.
    - Soporte para UTF-8.
  - Permiten aumentar el rendimiento del servidor.
    - Resolución de nombres de anfitrión.
    - Número y timeout de las conexiones.
    - Tamaño de las memorias cache.
    - Logs especiales.



# Configuración de MySQL (VIII)

- Compatibilidad con contraseñas antiguas.
  - Los clientes de MySQL de las versiones 3 o anteriores manejaban las contraseñas de forma diferente a las versiones 4 o posteriores.
    - Los clientes antiguos de MySQL no pueden conectarse a los servidores más modernos.
  - Añadir la siguiente línea a la sección [mysqld]:

```
old_passwords=1
```

# Configuración de MySQL (IX)

- Motor por defecto.
  - MySQL crea las bases de datos por defecto con el motor MyISAM.
    - Puede modificarse de forma particular en la creación de cada tabla.
  - Puede modificarse el motor por defecto añadiendo a la sección [mysqld] el parámetro:

`default-storage-engine=InnoDB`

# Configuración de MySQL (X)

- Sensibilidad de las tablas al contexto.
  - Los nombres de las bases de datos y tablas son sensibles al contexto.
    - Problemas de compatibilidad entre Linux y Windows.
  - Se puede configurar la sensibilidad al contexto en la sección [mysqld] mediante el valor:

`lower_case_table_names=<valor>`

- <valor>=0: Sensibilidad al contexto. No debe usarse en sistemas operativos insensibles al contexto (Windows).
- <valor>=1: Insensible al contexto. Los nombres de bases de datos y tablas se almacenan en minúsculas.
- <valor>=2: Los nombres se almacenan como se indican en la creación, pero las comparaciones se realizan convirtiendo todo a minúsculas.

# Configuración de MySQL (XI)

- Configuración del puerto por defecto.
  - Por defecto, MySQL utiliza el puerto 3306 TCP para las conexiones de clientes remotos al servidor.
  - Puede cambiarse el puerto que utiliza el servidor en la sección [mysqld] con la opción:

`port=<puerto>`

- Si deseamos decirle a un cliente remoto que se conecte a este puerto, debemos poner en la sección [mysql] del cliente remoto la opción:

`port=<puerto>`

# Configuración de MySQL (XII)

- Uso de caracteres UTF-8:
  - En determinadas ocasiones es necesario utilizar como conjunto de caracteres a UTF-8.
  - Se debe configurar tanto en el cliente como en el servidor.
    - En caso contrario el cliente no funcionará correctamente.
  - Se debe añadir en la sección [mysql]:

```
default-character-set=utf8
```

- Y en la sección [mysqld]:

```
collation_server=utf8_unicode_ci  
character-set-server=utf8  
init_connect='SET collation_connection = utf8_general_ci'  
init_connect='SET NAMES utf8'
```

# Configuración de MySQL (XIII)

- Resolución de nombres de anfitrión.
  - MySQL utiliza el servicio de DNS para resolver las direcciones IP que se conectan al servidor y almacenar información sobre las mismas.
    - El interfaz de loopback nunca es resuelto.
  - Pueden existir problemas:
    - Gran número de clientes desde distintas IPs.
    - Servidores de DNS muy lentos en la resolución.
  - Es posible deshabilitar la resolución de nombres de anfitrión añadiendo a la sección [mysqld]:

`skip-name-resolve`



# Configuración de MySQL (XIV)

- Número de conexiones.
  - MySQL soporta un número máximo de conexiones.
    - Debe configurarse en función de las aplicaciones clientes y de las posibilidades del sistema.
  - Se pueden controlar las conexiones simultáneas y las conexiones máximas por usuario añadiendo en la sección [mysqld] los valores:

`max_connections=<conexiones totales>`

`max_user_connections=<conexiones por usuario>`

- Se pueden abrir `<conexiones totales>+1`. Esta última se reserva para el administrador.
- Si `<conexiones por usuario>` es 0 no existe límite.

# Configuración de MySQL (XV)

- Timeout de las conexiones.
  - Es necesario controlar el timeout de las conexiones.
    - Si una conexión no se utiliza debe cerrarse para dejar acceso a nuevas conexiones.
  - Existen dos timeouts:
    - Timeout de aplicaciones.
    - Timeout de conexiones interactivas (de usuarios).
  - Se pueden modificar con los timeouts añadiendo a la sección [mysqld] los valores:

```
wait_timeout=<tiempo>  
interactive_timeout=<tiempo>
```

- El interactive\_timeout debería ser mayor que el wait\_timeout.

# Configuración de MySQL (XVI)

- Tamaño de las memorias cache.
  - Optimizar el rendimiento de MySQL optimizando el uso de las memorias cache del sistema.
  - Las variables más importantes son:
    - `thread_cache_size`: Número de hilos que el servidor mantiene. Debe aumentarse si las conexiones/segundo son altas.
    - `table_cache`: Número de tablas que el servidor puede tener abiertas simultáneamente.
    - `query_cache_size`: Tamaño de la memoria para almacenar los resultados de consultas.
    - `tmp_table_size`: Tamaño máximo de las tablas que pueden mantenerse en memoria.
    - `key_buffer_size`: Tamaño de la memoria usada por las tablas MyISAM.

# Configuración de MySQL (XVII)

- Tamaño de las memorias cache.
  - Se puede cambiar su valor en la sección [mysqld] con:

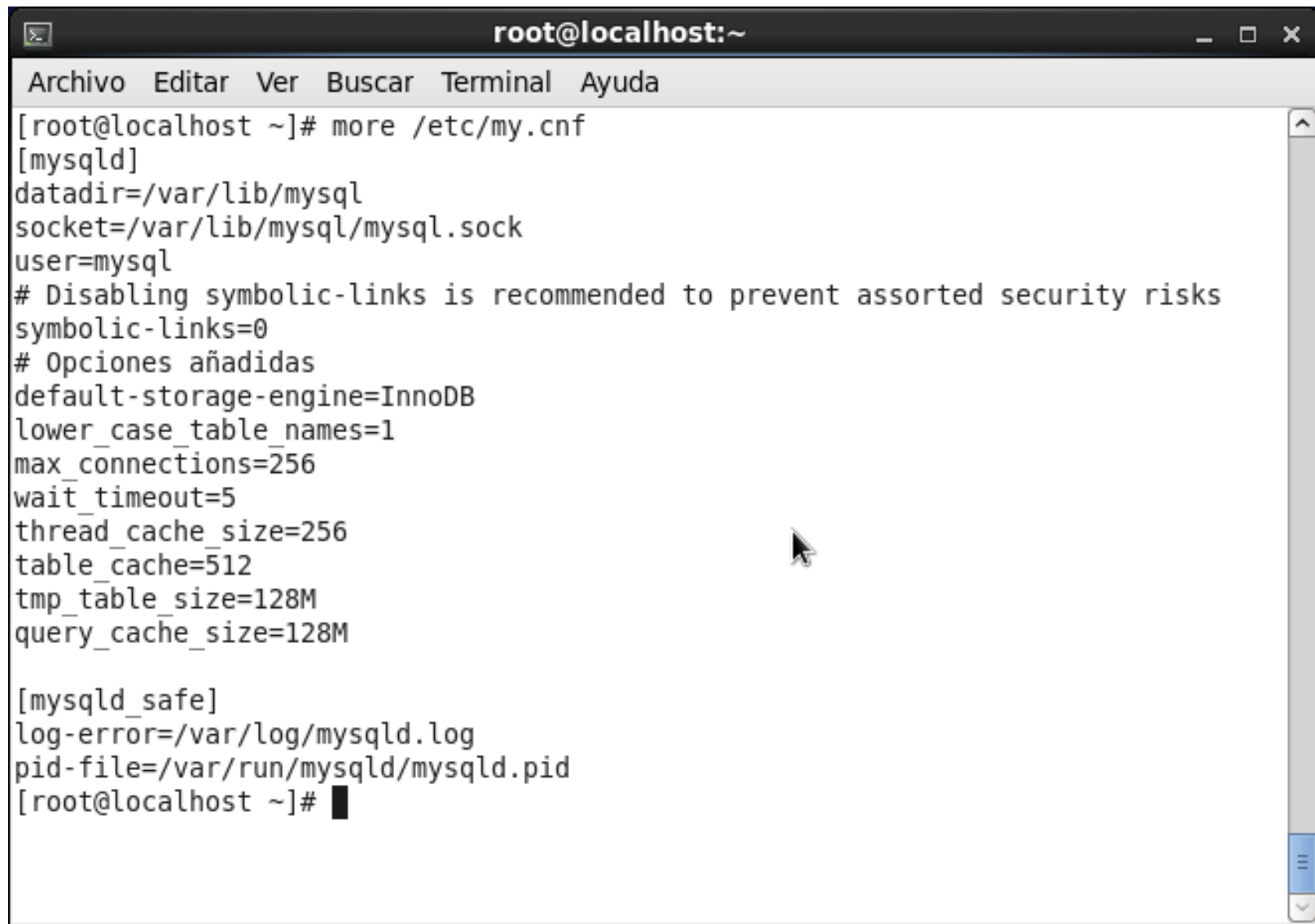
```
thread_cache_size=<valor>
table_cache=<valor>
query_cache_size=<tamaño>
tmp_table_size=<tamaño>
key_buffer_size=<tamaño>
```

# Configuración de MySQL (XVIII)

- Logs especiales.
  - En ocasiones algunas consultas ejecutadas al servidor pueden tardar mucho tiempo en responder.
    - Es posible almacenar dichas consultas para comprobar el motivo de la tardanza.
  - Se configuran en la sección [mysqld] con las variables:

```
long_query_time=<segundos>  
log-slow-queries=<fichero>
```

# Configuración de MySQL (XIX)



```
root@localhost:~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[root@localhost ~]# more /etc/my.cnf
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
# Opciones añadidas
default-storage-engine=InnoDB
lower_case_table_names=1
max_connections=256
wait_timeout=5
thread_cache_size=256
table_cache=512
tmp_table_size=128M
query_cache_size=128M

[mysqld_safe]
log-error=/var/log/mysql.log
pid-file=/var/run/mysql/mysql.pid
[root@localhost ~]#
```



# Bases de datos en MySQL (I)

- Inicialmente solo el usuario administrador tiene autorización para crear bases de datos.
  - Si otro usuario debe crear tablas, debe autorizarse que pueda crear nuevas tablas en el sistema:

```
GRANT CREATE ON *.* TO usuario;
```

- Una nueva base de datos se crea con la sentencia:

```
CREATE DATABASE nombre;
```

- Se borra con la sentencia:

```
DROP DATABASE nombre;
```

- Y se listan las bases de datos con la sentencia:

```
SHOW DATABASES;
```

# Bases de datos en MySQL (II)



The screenshot shows a terminal window titled 'root@localhost:~' with a menu bar containing 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal displays the following commands and output:

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
+-----+
2 rows in set (0.00 sec)
```

A red arrow points to the 'show databases;' command. The output shows two databases: 'information\_schema' and 'mysql'.

```
mysql> create database prueba;
Query OK, 1 row affected (0.00 sec)
```

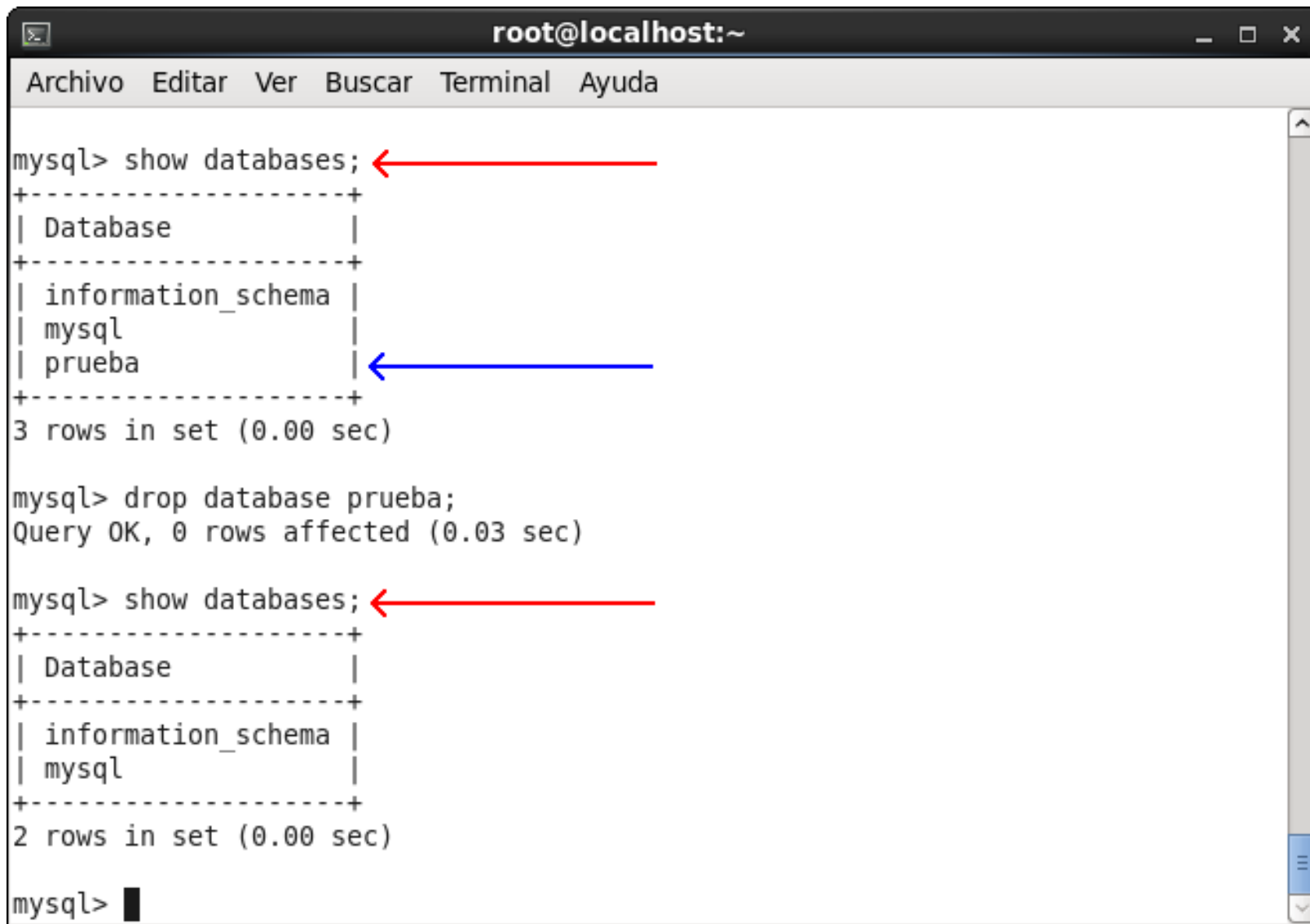
A red arrow points to the 'create database prueba;' command. The output indicates the query was successful.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
| prueba     |
+-----+
3 rows in set (0.00 sec)
```

A blue arrow points to the 'prueba' database in the output. The output now shows three databases: 'information\_schema', 'mysql', and 'prueba'.

```
mysql> █
```

# Bases de datos en MySQL (III)



The screenshot shows a terminal window titled 'root@localhost:~' with a menu bar containing 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal displays the following commands and output:

```
mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| mysql                   |
| prueba                  |
+-----+
3 rows in set (0.00 sec)
```

A red arrow points to the command 'show databases;', and a blue arrow points to the 'prueba' database in the output table.

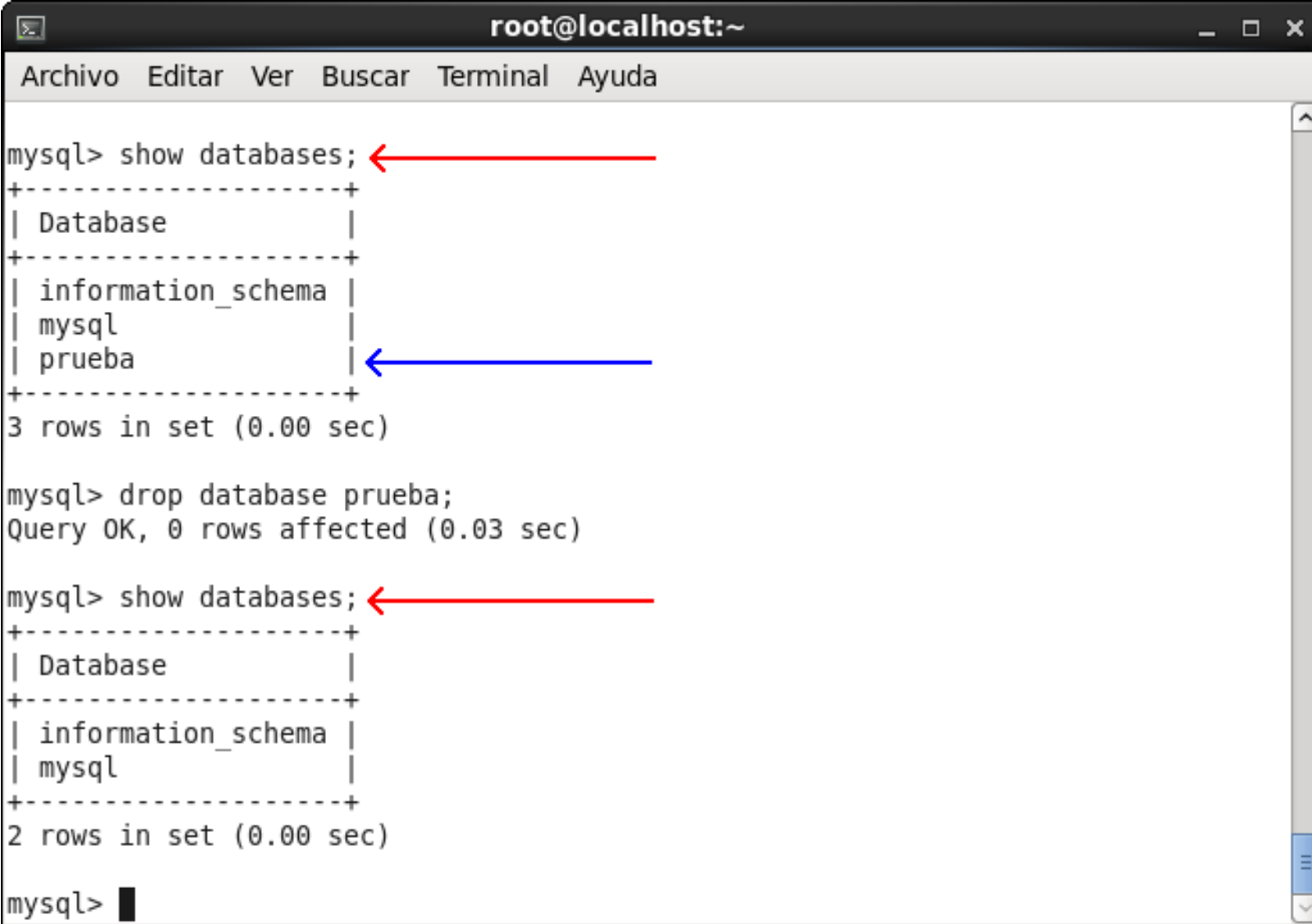
```
mysql> drop database prueba;
Query OK, 0 rows affected (0.03 sec)
```

Another red arrow points to the second 'show databases;' command.

```
mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| mysql                   |
+-----+
2 rows in set (0.00 sec)
```

The terminal ends with 'mysql>' and a cursor.

# Bases de datos en MySQL (III)



The screenshot shows a terminal window titled 'root@localhost:~' with a menu bar containing 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal displays the following commands and output:

```
mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| prueba                  |
+-----+
3 rows in set (0.00 sec)
```

A red arrow points to the command 'show databases;', and a blue arrow points to the 'prueba' database in the output table.

```
mysql> drop database prueba;
Query OK, 0 rows affected (0.03 sec)
```

Another red arrow points to the second 'show databases;' command.

```
mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
+-----+
2 rows in set (0.00 sec)
```

The terminal ends with 'mysql>' and a cursor.

# Tablas en MySQL (I)

- Las tablas en MySQL se crean dentro de bases de datos existentes.
- El comando que se utiliza es:

```
CREATE database.nombre_tabla(elementos) ENGINE =  
{InnoDB|MyISAM};
```

- Si no se especifica un motor se utiliza el motor por defecto definido en la configuración de MySQL.
- Si deseamos utilizar una base de datos en particular podemos ejecutar el comando:

```
USE database;
```

- Y a partir de este momento si no especificamos la base de datos se utilizará la indicada.

# Tablas en MySQL (II)

- Es muy importante tener en cuenta el tipo de motor seleccionado en la tabla:
  - El motor InnoDB comprueba la integridad referencial en la inserción de elementos.
  - El motor MyISAM no realiza ninguna comprobación referencial.



# Tablas en MySQL (III)

- Creamos dos bases de datos (bd1 y db2) y dentro de cada una de ellas dos tablas con los motores MyISAM e InnoDB respectivamente.

```
CREATE DATABASE bd1;
```

```
USE bd1;
```

```
CREATE TABLE tabla1(a INTEGER NOT NULL, PRIMARY  
KEY(a)) ENGINE=MyISAM;
```

```
CREATE TABLE tabla2(b INTEGER NOT NULL, FOREIGN  
KEY (b) REFERENCES tabla1(a)) ENGINE=MyISAM;
```

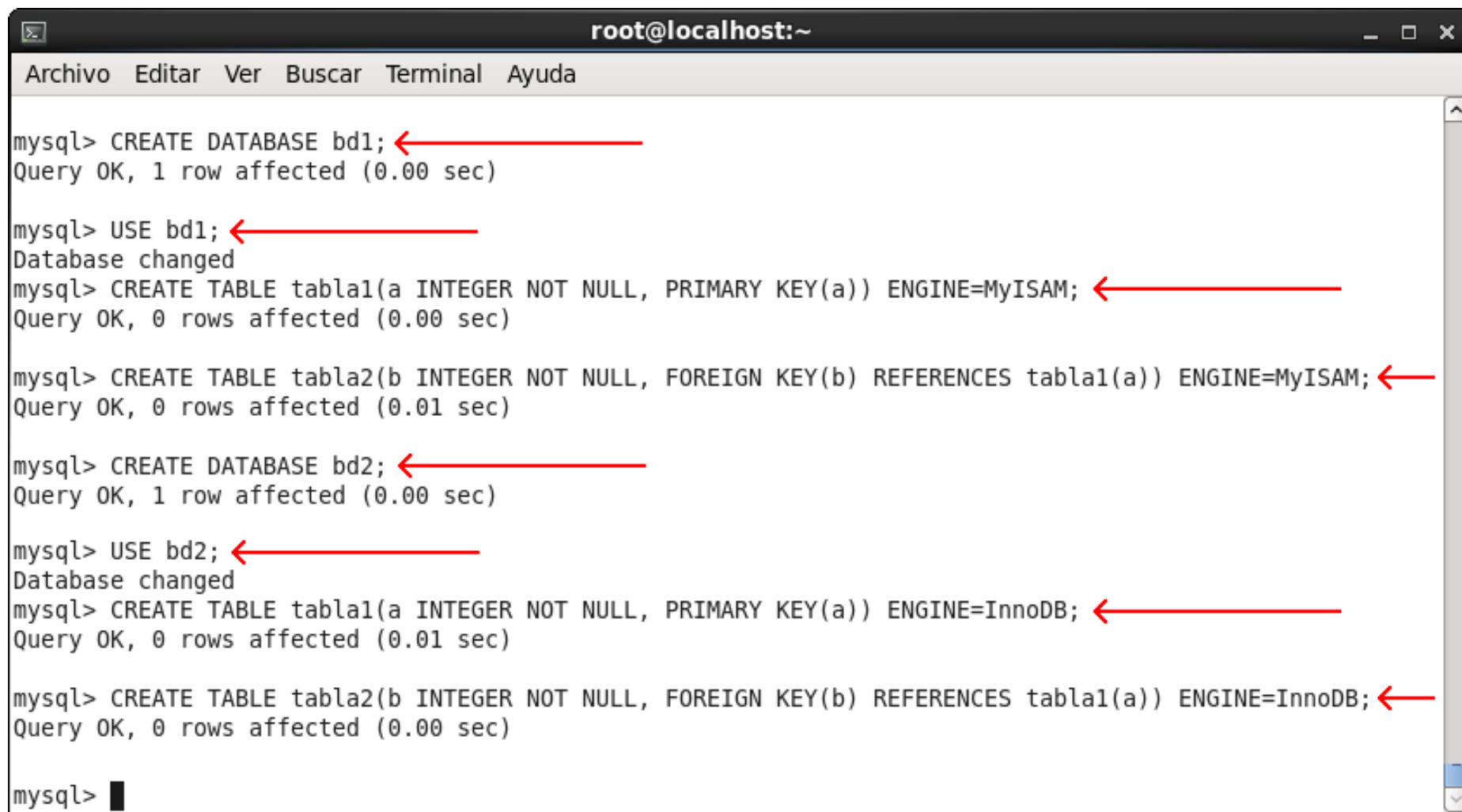
```
CREATE DATABASE bd2;
```

```
USE bd2;
```

```
CREATE TABLE tabla1(a INTEGER NOT NULL, PRIMARY  
KEY(a)) ENGINE=InnoDB;
```

```
CREATE TABLE tabla2(b INTEGER NOT NULL, FOREIGN  
KEY (b) REFERENCES tabla1(a)) ENGINE=InnoDB;
```

# Tablas en MySQL (IV)



```
root@localhost:~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
  
mysql> CREATE DATABASE bd1; ←  
Query OK, 1 row affected (0.00 sec)  
  
mysql> USE bd1; ←  
Database changed  
mysql> CREATE TABLE tabla1(a INTEGER NOT NULL, PRIMARY KEY(a)) ENGINE=MyISAM; ←  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> CREATE TABLE tabla2(b INTEGER NOT NULL, FOREIGN KEY(b) REFERENCES tabla1(a)) ENGINE=MyISAM; ←  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> CREATE DATABASE bd2; ←  
Query OK, 1 row affected (0.00 sec)  
  
mysql> USE bd2; ←  
Database changed  
mysql> CREATE TABLE tabla1(a INTEGER NOT NULL, PRIMARY KEY(a)) ENGINE=InnoDB; ←  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> CREATE TABLE tabla2(b INTEGER NOT NULL, FOREIGN KEY(b) REFERENCES tabla1(a)) ENGINE=InnoDB; ←  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> █
```

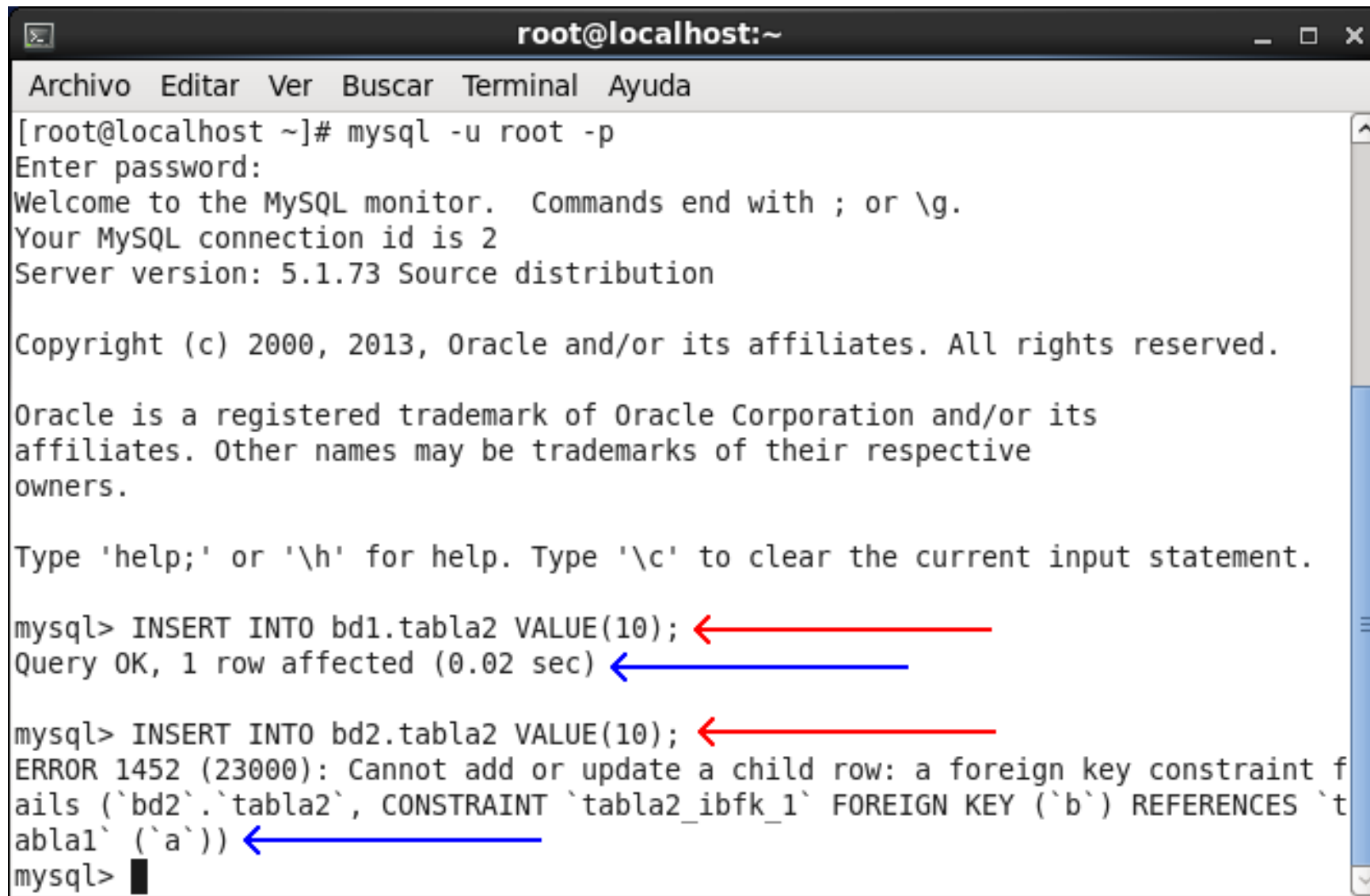
# Tablas en MySQL (V)

- Ejecutamos ahora los comandos:

```
INSERT INTO bd1.tabla2 VALUES(10);  
INSERT INTO bd2.tabla2 VALUES(10);
```

- Podemos comprobar que:
  - El motor MyISAM (bd1) no comprueba la integridad referencial.
  - El motor InnoDB (bd2) comprueba la integridad referencial y no permite la inserción del elemento.

# Tablas en MySQL (VI)



```
root@localhost:~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
[root@localhost ~]# mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 2  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.  
  
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> INSERT INTO bd1.tabla2 VALUE(10);  
Query OK, 1 row affected (0.02 sec)  
  
mysql> INSERT INTO bd2.tabla2 VALUE(10);  
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint f  
ails (`bd2`.`tabla2`, CONSTRAINT `tabla2_ibfk_1` FOREIGN KEY (`b`) REFERENCES `t  
abla1` (`a`))  
mysql>
```

The screenshot shows a terminal window titled 'root@localhost:~'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The user runs the command 'mysql -u root -p' and enters a password. The MySQL monitor displays the welcome message, connection ID (2), and server version (5.1.73 Source distribution). It also shows copyright information and usage instructions. The user then executes two SQL commands. The first command, 'INSERT INTO bd1.tabla2 VALUE(10);', is successful, and the second command, 'INSERT INTO bd2.tabla2 VALUE(10);', results in an error: 'ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`bd2`.`tabla2`, CONSTRAINT `tabla2\_ibfk\_1` FOREIGN KEY (`b`) REFERENCES `tabla1` (`a`))'. Red and blue arrows point to the respective SQL commands in the terminal output.

# Usuarios en MySQL (I)

- Los usuarios existentes en MySQL se listan con el comando:

```
SELECT * FROM mysql.user;
```

- Proporciona información de todos los campos.
- Es excesiva en la mayoría de casos.

```
SELECT user,host,password FROM mysql.user;
```

- Proporciona el nombre del usuario, el ordenador desde el que esta autorizado a conectarse y la contraseña cifrada.

# Usuarios en MySQL (II)

- Los usuarios se crean en MySQL con la sentencia:

```
CREATE USER 'usuario'[@'ordenador'] IDENTIFIED  
BY clave;
```





- El parámetro 'ordenador' es opcional.
  - Indica el ordenador desde el que puede conectarse el usuario.
  - Si no se especifica el usuario puede conectarse desde cualquier ordenador (valor '%').
- Un usuario se borra con la sentencia:

```
DROP USER 'usuario'[@'ordenador'];
```

- ES necesario tener en cuenta que solo se borra el usuario que accede desde ese ordenador.



# Usuarios en MySQL (III)

```
root@localhost:~  
Archivo  Editor  Ver  Buscar  Terminal  Ayuda  
mysql> select user,host,password from mysql.user;   
+-----+-----+-----+  
| user | host      | password  
+-----+-----+-----+  
| root | localhost | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22 |  
| root | 127.0.0.1 | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22 |  
+-----+-----+-----+  
2 rows in set (0.00 sec)  
  
mysql> create user usuario identified by 'mi_clave';   
Query OK, 0 rows affected (0.00 sec)  
  
mysql> select user,host,password from mysql.user;   
+-----+-----+-----+  
| user      | host      | password  
+-----+-----+-----+  
| root      | localhost | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22 |  
| root      | 127.0.0.1 | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22 |  
| usuario   | %         | *7E3CEF4AE3E97137EF162C0354FBEF51608C1DA6 |   
+-----+-----+-----+  
3 rows in set (0.00 sec)  
  
mysql> █
```



# Usuarios en MySQL (IV)

```
root@localhost:~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
mysql> select user,host,password from mysql.user;  
+-----+-----+-----+  
| user      | host      | password  
+-----+-----+-----+  
| root      | localhost | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22  
| root      | 127.0.0.1 | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22  
| usuario   | %         | *7E3CEF4AE3E97137EF162C0354FBEF51608C1DA6  
+-----+-----+-----+  
3 rows in set (0.00 sec)  
  
mysql> create user 'otro_usuario'@'localhost' identified by 'otra_clave';  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> select user,host,password from mysql.user;  
+-----+-----+-----+  
| user      | host      | password  
+-----+-----+-----+  
| root      | localhost | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22  
| root      | 127.0.0.1 | *CA59F888C1BED14A60696A3DEFA5A8EF3F0F7C22  
| otro_usuario | localhost | *7F3D2EE9EC9EE8639D3565AF13E8502C537C613F  
| usuario   | %         | *7E3CEF4AE3E97137EF162C0354FBEF51608C1DA6  
+-----+-----+-----+  
4 rows in set (0.00 sec)  
  
mysql> █
```

# Usuarios en MySQL (V)

- Cuando un usuario se crea no tiene ningún tipo de permiso sobre ninguna tabla.
  - Es necesario asignarle permisos sobre las tablas deseadas.

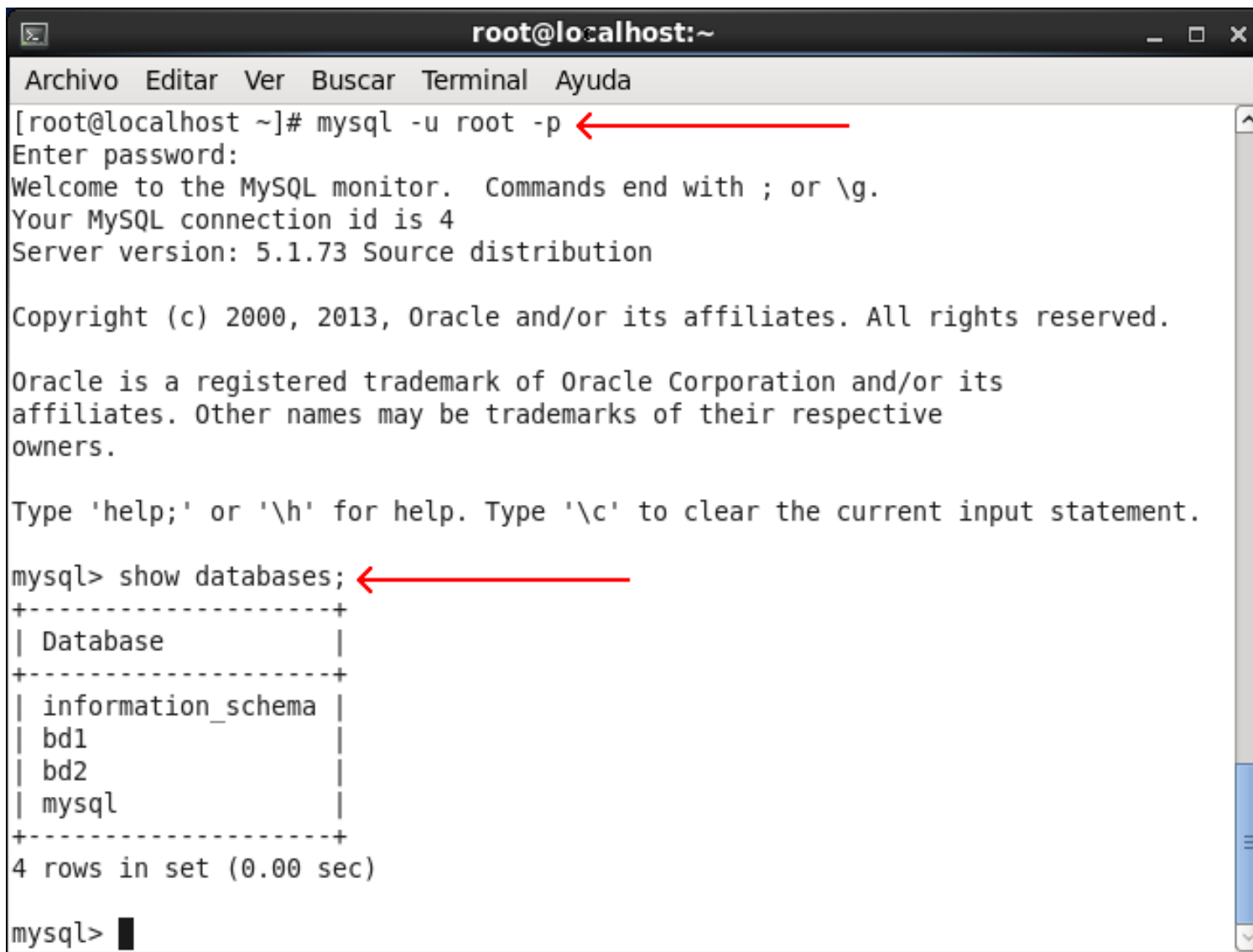
```
GRANT permiso[,permiso] ON basedatos.tabla TO  
usuario;
```

- basedatos: Base de datos sobre la que se asignan los permisos o \* para todas las bases de datos..
- tabla: Tabla sobre la que se asignan los permisos o \* para todas las tablas de la base de datos.
- permiso: Permiso que se otorga al usuario sobre la base de datos y tabla.

# Usuarios en MySQL (VI)

Permiso	Descripción
SELECT	Consultar la tabla.
INSERT	Insertar nuevos datos (elementos) en la tabla.
UPDATE	Modificar datos (elementos) de la tabla.
DELETE	Borrar datos (elementos) de la tabla.
CREATE	Crear nuevas tablas en la base de datos.
ALTER	Modificar la estructura de la tabla, añadiendo nuevos atributos (columnas).
DROP	Eliminar tablas existentes en la base de datos.
INDEX	Crear índices utilizando un campo de la tabla. Crear índice permite que una búsqueda por ese campo se efectúe de forma más rápida.
LOCK TABLES	Bloquear el acceso a la tabla siempre que se tenga permiso de consulta sobre la misma.
ALL PRIVILEGES	Asignar todos los permisos anteriores.

# Usuarios en MySQL (VII)



```
root@localhost:~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

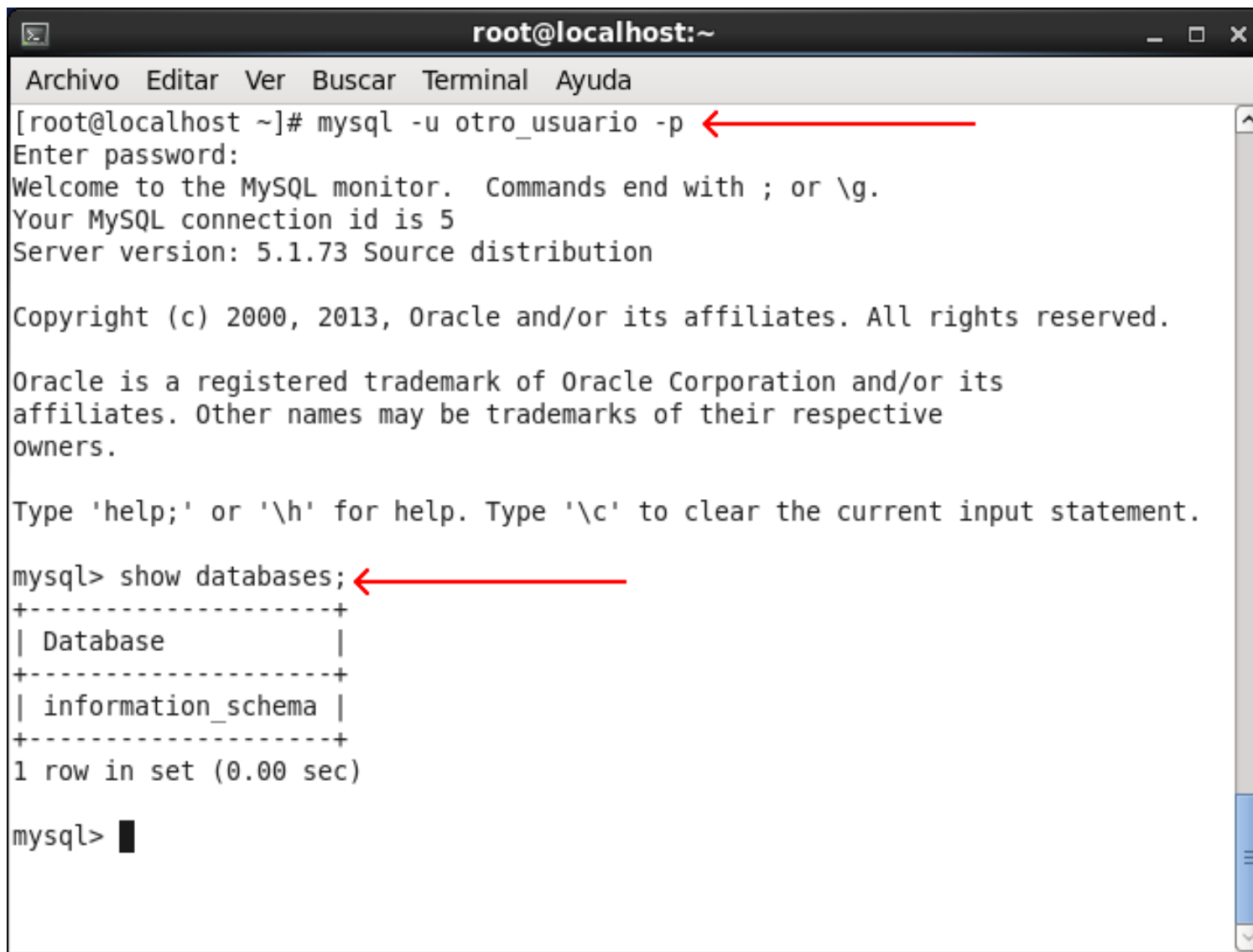
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| bd1                      |
| bd2                      |
| mysql                    |
+-----+
4 rows in set (0.00 sec)

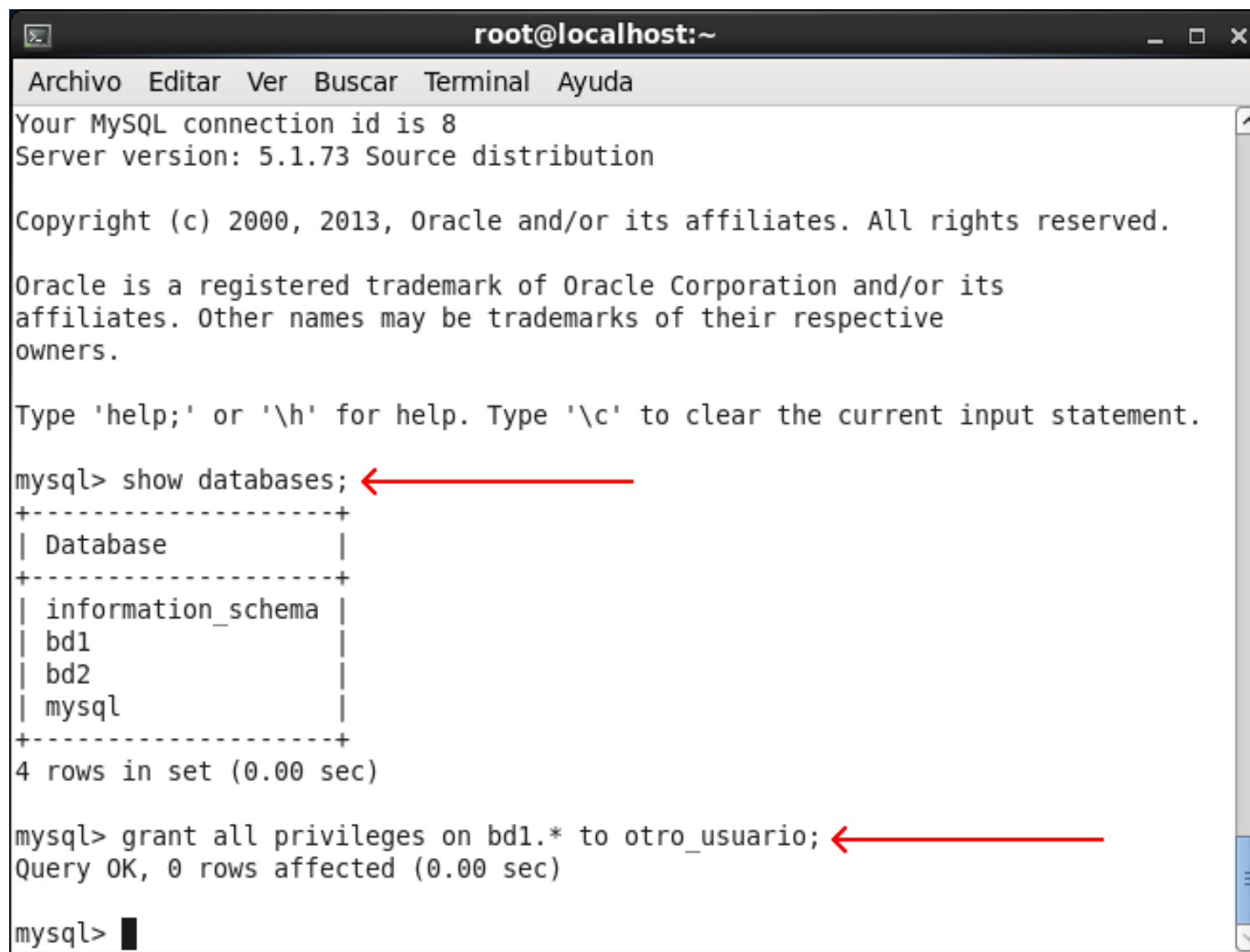
mysql>
```

# Usuarios en MySQL (VIII)

A terminal window titled 'root@localhost:~' with a menu bar containing 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal shows the execution of the 'mysql -u otro\_usuario -p' command, which prompts for a password and then displays the MySQL monitor welcome message. The user then enters 'show databases;' and the output shows a single row for 'information\_schema'.

```
root@localhost:~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
[root@localhost ~]# mysql -u otro_usuario -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 5  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.  
  
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> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
+-----+  
1 row in set (0.00 sec)  
  
mysql> 
```

# Usuarios en MySQL (IX)

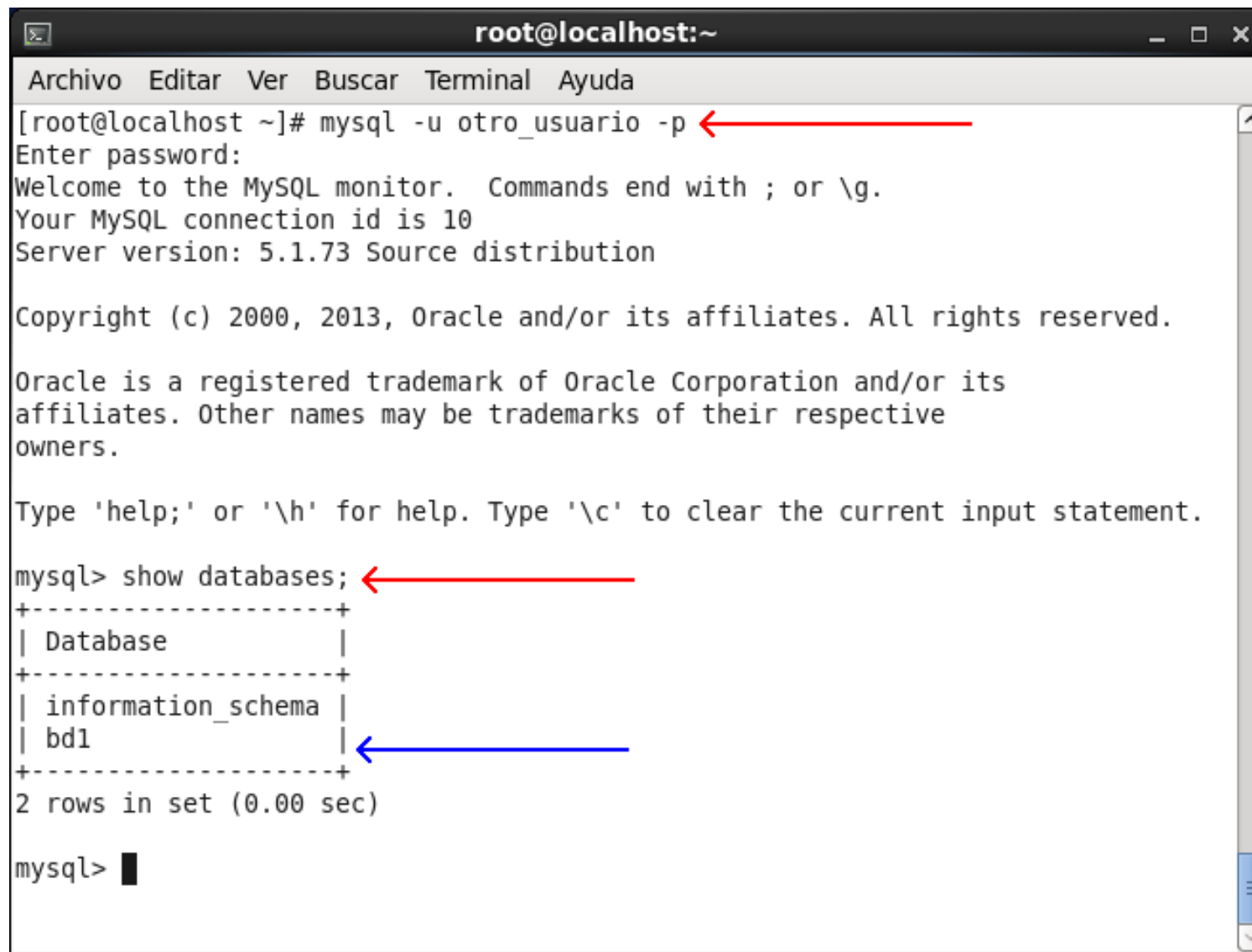


The screenshot shows a terminal window titled 'root@localhost:~'. The terminal displays the MySQL command-line interface. It starts with the prompt 'mysql>'. The user enters 'show databases;', which is highlighted by a red arrow. The output shows a list of databases: information\_schema, bd1, bd2, and mysql. Below the list, it says '4 rows in set (0.00 sec)'. The user then enters 'grant all privileges on bd1.\* to otro\_usuario;', which is also highlighted by a red arrow. The output is 'Query OK, 0 rows affected (0.00 sec)'. The prompt 'mysql>' is shown again at the bottom.

```
root@localhost:~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
Your MySQL connection id is 8  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.  
  
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> show databases; ←  
+-----+  
| Database |  
+-----+  
| information_schema |  
| bd1 |  
| bd2 |  
| mysql |  
+-----+  
4 rows in set (0.00 sec)  
  
mysql> grant all privileges on bd1.* to otro_usuario; ←  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> █
```



# Usuarios en MySQL (X)



The screenshot shows a terminal window titled 'root@localhost:~'. The menu bar includes 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal content shows the execution of the command 'mysql -u otro\_usuario -p', indicated by a red arrow. It prompts for a password and displays the MySQL monitor welcome message, including the connection ID (10) and server version (5.1.73). Below this, copyright and trademark notices are shown. The command 'mysql> show databases;' is entered, indicated by another red arrow. The output is a table with two rows: 'information\_schema' and 'bd1', with a blue arrow pointing to the second row. The terminal concludes with '2 rows in set (0.00 sec)' and a new prompt 'mysql>'.

```
root@localhost:~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[root@localhost ~]# mysql -u otro_usuario -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

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> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| bd1                      |
+-----+
2 rows in set (0.00 sec)

mysql>
```



# Copias de seguridad en MySQL (I)

- De forma general, es conveniente hacer copias de seguridad de la base de datos.
  - Permiten restaurar la base de datos ante cualquier fallo.
  - Es posible migrar la base de datos de servidor.
- Las copias de seguridad se realizan con el comando `mysqldump`.
- Es posible hacer copia de:
  - Una única tabla de una base de datos.
  - Una base de datos.
  - Varias bases de datos.
  - Todas las bases de datos existentes.

# Copias de seguridad en MySQL (II)

- Una única tabla:

```
mysqldump -u root -p basedatos tabla > tabla.sql
```

- Una base de datos:

```
mysqldump -u root -p basedatos > basedatos.sql
```

- Varias bases de datos:

```
mysqldump -u root -p --databases basedatos >  
varias_basedatos.sql
```

- Todas las bases de datos:

```
mysqldump -u root -p --all-databases > todas.sql
```

# Copias de seguridad en MySQL (III)

- El fichero de copia de seguridad de la base de datos es un fichero de texto:

```
...  
DROP TABLE IF EXISTS `user`;  
/*!40101 SET @saved_cs_client      = @@character_set_client */;  
/*!40101 SET character_set_client = utf8 */;  
CREATE TABLE `user` (  
  `Host` char(60) COLLATE utf8_bin NOT NULL DEFAULT '',  
  `User` char(16) COLLATE utf8_bin NOT NULL DEFAULT '',  
  `Password` char(41) CHARACTER SET latin1 COLLATE latin1_bin NOT NULL  
DEFAULT '  
,  
  `Select_priv` enum('N','Y') CHARACTER SET utf8 NOT NULL DEFAULT 'N',  
  `Insert_priv` enum('N','Y') CHARACTER SET utf8 NOT NULL DEFAULT 'N',  
  `Update_priv` enum('N','Y') CHARACTER SET utf8 NOT NULL DEFAULT 'N',  
  `Delete_priv` enum('N','Y') CHARACTER SET utf8 NOT NULL  
...  

```

# Copias de seguridad en MySQL (IV)

- La restauración de la base de datos se realiza siempre de la misma forma:
  - Crear desde MySQL la base de datos donde se desea restaurar la copia:

```
CREATE DATABASE restaura_basedatos;
```

- Volcar la copia realizada en la base de datos:

```
mysql -u root -p restaura_basedatos <  
fichero_copia.sql
```