

MYSQL:

Gestión de usuarios

UF1845: Bases de datos



Ver los usuarios del sistema:

```
USE mysql;
```

```
SELECT * FROM user;
```

O más fácil:

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

Crear usuario

CREATE USER *nombre@host* IDENTIFIED BY *password*;

La parte @host es opcional y se refiere a desde donde se permite la conexión. Valores posibles:

- 127,0,0,1 o localhost ,
- % (cualquier dirección), que equivale a no poner nada,
- Una Ip concreta.

Ejemplos:

CREATE USER *admin0* IDENTIFIED BY '123abc';

CREATE USER *admin1@localhost* IDENTIFIED BY '123abc';

CREATE USER *admin2@127.0.0.1* IDENTIFIED BY '123abc';

CREATE USER *admin3@'%'* IDENTIFIED BY '123abc';

Eliminar usuario

DROP USER *usuario*;

Ejemplo:

DROP USER *admin1@localhost*;



Asignar permisos a los usuarios

GRANT *permisos* **ON** *bd.tablas* **TO** *usuario*;

Los permisos se pueden especificar individualmente o asignar todos los posibles mediante **ALL PRIVILEGES**

Ejemplos:

GRANT *select* **ON** *universidad.** **TO** *admin1@localhost*;

GRANT *select, update* **ON** *universidad.**
TO *admin2@localhost*;

Los permisos son acumulativos.

Ejemplos:

GRANT *select* **ON** *universidad.** **TO** *admin3@localhost*;

GRANT *update* **ON** *universidad.** **TO** *admin3@localhost*;

Y con ello *admin3@localhost* tiene los permisos de select y update



Permisos que se pueden asignar

CREATE permite crear nuevas tablas o bases de datos.

DROP permite eliminar tablas o bases de datos.

DELETE permite eliminar registros de tablas.

INSERT permite insertar registros en tablas.

SELECT permite leer registros en las tablas.

UPDATE permite actualizar registros en las tablas.

GRANT OPTION permite eliminar permisos de usuarios.

SHOW DATABASE Permite listar las bases de datos existentes.



Mostrar los permisos de un usuario

SHOW GRANTS FOR *usuario*;

Ejemplo:

SHOW GRANTS FOR *admin1@localhost*;

Eliminar permisos

REVOKE *permisos* ON *bd.tablas* TO *usuario*;

Los permisos se retiran de forma similar a como se asignan

Ejemplos:

REVOKE select ON *universidad. TO *admin1@localhost*;**

REVOKE select, update ON *universidad. TO *admin2@localhost*;**

Actualizar la tabla de usuarios y permisos

FLUSH PRIVILEGES;



Creación de roles

Puede ser conveniente asignar siempre los mismos permisos a usuarios diferentes. En ese caso se pueden especificar los permisos a un rol, es decir a un perfil o categoría, al que luego se adscriben los usuarios individualmente.

Primero se crean los roles:

CREATE ROLE *rol*;

Ejemplo:

CREATE ROLE '*rl_lectura*';

Se pueden crear varios roles a la vez:

CREATE ROLE '*rl_lectura*', '*rl_escritura*';



Asignación de permisos a los roles

La asignación es similar a como se hacía con los usuarios:

GRANT *permisos* **ON** *bd.tabla* **TO** *rol*;

Ejemplos :

GRANT *select* **ON** *universidad.** **TO** *'rl_lectura'*;

GRANT *insert, update, delete* **ON** *universidad.** **TO** *'rl_escritura'*;

Para ver los permisos asignados usamos GRANTS (en plural):

SHOW GRANTS FOR *'rl_lectura'*;

SHOW GRANTS FOR *'rl_escritura'*;

Retirar permisos a los roles

REVOKE *permisos* **ON** *bd.tabla* **TO** *rol*;

Ejemplos :

REVOKE *select* **ON** *universidad.** **TO** *'rl_lectura'*;



Asignación roles a usuarios

GRANT *rol* **TO** *usuario*;

Ejemplos :

GRANT *'rl_lectura'* **TO** *admin1@localhost*;

GRANT *insert, update, delete* **ON** *universidad.** **TO** *'rl_escritura'*;

Para ver los permisos asignados usamos GRANTS (en plural):

SHOW GRANTS FOR *'rl_lectura'*;

SHOW GRANTS FOR *'rl_escritura'*;

Retirar roles los usuarios

REVOKE *rol* **TO** *usuario*;

Ejemplo:

REVOKE *'rl_lectura'* **TO** *admin1@localhost*;