TEMA 09 – GESTIÓN DE USUARIOS Y PERMISOS

Creación de usuarios en MySQL

- CREATE USER user IDENTIFIED BY ‘password’;

```
-- ejemplos
CREATE USER 'user1'@'localhost' IDENTIFIED BY '12345678' PASSWORD EXPIRE;
CREATE USER 'user2'@'localhost' IDENTIFIED BY '12345678' PASSWORD EXPIRE INTERVAL 30 DAY;
CREATE USER 'user3'@'localhost' IDENTIFIED BY '12345678' ACCOUNT LOCK;
```

- Opciones para gestionar la caducidad de la contraseña:
  - ‘PASSWORD EXPIRE’: obliga a cambiar la contraseña en el próximo inicio de sesión
  - ‘PASSWORD EXPIRE INTERVAL n DAY’: obliga a cambiar la contraseña cada n días
  - ’PASSWORD EXPIRE NEVER’: la contraseña nunca caduca
  - ‘PASSWORD EXPIRE DEFAULT’: la contraseña caduca en el nº de días que indica la variable del sistema default_password_lifetime
    - 360 días en las versiones inferiores a 5.7.11 y no caduca a partir de esa
  - ‘ACCOUNT LOCK’: podemos crear un usuario con la cuenta bloqueada
- Podemos limitar los recursos con las siguientes opciones:
  - ‘MAX_QUERIES_PER_HOUR’
  - ‘MAX_UPDATES_PER_HOUR’
  - ‘MAX_CONNECTIONS_PER_HOUR’
  - ‘MAX_USER_CONNECTIONS’: máximo de conexiones simultáneas que se permiten a cada usuario, por defecto ilimitadas

```
CREATE USER 'user1@localhost' IDENTIFIED BY '12345678'
WITH MAX_QUERIES_PER_HOUR 100 MAX_USER_CONNECTIONS 10;
```

Modificación de usuarios

- ‘ALTER USER [IF EXISTS] user;’
- Para bloquear usuario: ‘ALTER USER [IF EXISTS] user ACCOUNT LOCK’
- Para modificar la contraseña: ‘ALTER USER [IF EXISTS] user IDENTIFIED BY ‘NuevaPass’’
- Para modificar el nombre: ‘RENAME USER user@localhost YO usuario@localhost;’
- Para borrar usuario: ‘DROP USER usuario@localhost;’

Gestión de permisos

- Permisos más habituales que se pueden asignar a los usuarios:
  - ALL PRIVILEGES: acceder a todas las bases de datos asignadas en el sistema

    - CREATE: crear nuevas tablas o bases de datos
    - DROP: borrar tablas o bases de datos
    - DELETE: eliminar registros de las tablas
    - INSERT: insertar registros en las tablas
    - SELECT: leer registros en las tablas
    - UPDATE: actualizar registros en las tablas
    - GRANT OPTION: permite remover privilegios de usuarios
    - CREATE USER: permite gestionar los usuarios
    - EXECUTE: permite ejecutar procedimientos y funciones
- Orden GRANT. Asignar permisos
    - 'GRANT permiso ON [nombreBaseDatos].[nombreTabla] TO usuario'
    - *.* hace referencia a todas las bases de datos y tablas del sistema
    - Basedatos.* hace referencia a totas las tablas de la base de datos indicada
    - Basedatos.tabla solo hacer referencia a la tabla de la base de datos
    - Una vez asignados los privilegios no es necesario ejecutar la orden FLUSH PRIVILEGES
    - Solo es necesaria esta instrucción si modificamos los valores utilizando un update, delete o insert
- Mostrar privilegios
    - 'SHOW GRANTS'
    - Los cambios a nivel global tienen efecto cuando el usuario vuelve a conectarse
    - Los cambios a nivel de base de datos cuando el usuario vuelve a seleccionarla
    - Los cambios a nivel de tabla y columna tienen un efecto inmediato
- Orden REVOKE. Quitar privilegios
    - 'REVOKE permiso ON [nombreBaseDatos].[nombredetabla] FROM usuario'
- Opción WITH GRAN OPTION
    - Habilitamos a ese usuario para que pueda otorgar ese mismo permiso a otros usuarios

```
GRANT SELECT,INSERT ON sakila.* TO user1@localhost WITH GRANT OPTION;
```

Gestión de roles

- Conjunto de privilegios que se pueden otorgar a un usuario o a otro Rol
- Simplifica el trabajo de DBA
- Pueden ser de nivel global, de base de datos o de tabla y columnas
- 'CREATE ROLE [IF NOT EXISTS] nombrerol;'
- Para asignar y retirar privilegios a los roles utilizamos la misma orden GRANT
- REVOKE para eliminarlos
- Un usuario puede tener diferentes roles asignados
- En el último paso seleccionamos cual es el rol por defecto que va a utilizar con la instrucción SET DEFAULT ROLE

```
CREATE ROLE IF NOT EXISTS consultaSakila;
```