

DAW/DAM. UD 7. USUARIOS Y EXTENSIONES. ACTIVIDADES NO EVALUABLES PARTE 1 (SOLUCIONADO)

DAW/DAM. Bases de datos (BD)

UD 7. USUARIOS Y EXTENSIONES

Parte 1. Gestión de usuarios en MySQL. Prácticas no evaluables (solucionado)

Abelardo Martínez y Pau Miñana

Basado y modificado de Sergio Badal (www.sergiobadal.com) y Raquel Torres.

Curso 2023-2024

Aspectos a tener en cuenta

Importante

Estas actividades son opcionales y no evaluables pero es recomendable hacerlas para un mejor aprendizaje de la asignatura.

Si buscas las soluciones por Internet o preguntas al oráculo de ChatGPT, te estarás engañando a ti mismo. Ten en cuenta que **ChatGPT no es infalible ni todopoderoso.**

Es una gran herramienta para agilizar el trabajo una vez se domina una materia, pero usarlo como atajo en el momento de adquirir habilidades y conocimientos básicos perjudica gravemente tu aprendizaje. Si lo utilizas para obtener soluciones o asesoramiento respecto a las tuyas, revisa cuidadosamente las soluciones propuestas igualmente. Intenta resolver las actividades utilizando los recursos que hemos visto y la documentación extendida que encontrarás en el “Aula Virtual”.

Recomendaciones

Importante

- **No uses NUNCA tildes, ni eñes, ni espacios, ni caracteres no alfanuméricos** (salvo el guión bajo) **en los metadatos** (nombres de elementos de una base de datos).
- Sé coherente con el uso de mayúsculas/minúsculas.

1. BD Jardinería

Actividad no evaluable

Utiliza la base de datos **DBJardineria** creada en unidades anteriores. Accede a ella desde la consola con un usuario con permisos de superadministrador y ejecuta las sentencias necesarias para conseguir estos hitos.

1.1. Ejercicio

- Crea un nuevo usuario superadministrador llamado '**glucas**' y con la clave 123456. Dale todos los privilegios posibles cuando acceda desde la misma máquina donde esté instalada la base de datos. ¡Cuidado, este usuario debe poder conceder privilegios!
- Sal de la consola (quit) y vuelve a entrar con ese nuevo usuario. Comprueba que está en la tabla donde se almacenan los pares **usuario+host**, y cambia su clave por 'starwars'.

Si tienes problemas con SYSTEM_USER, ejecuta esta orden desde tu usuario root:

```
GRANT SYSTEM_USER ON *.* TO root;
```

Solución

Conexión a MySQL con usuario superadministrador (root):

```
terminal> mysql -u root -p
```

```
CREATE USER IF NOT EXISTS 'glucas'@'localhost' IDENTIFIED BY '123456'; --
GRANT ALL PRIVILEGES ON *.* TO 'glucas'@'localhost' WITH GRANT OPTION; --
FLUSH PRIVILEGES;
```

```
quit;
```

Conexión a MySQL con usuario 'glucas':

```
terminal> mysql -u glucas -p
```

```
SELECT user, host  
FROM mysql.user  
WHERE user = 'glucas';  
  
ALTER USER 'glucas'@'localhost' IDENTIFIED BY 'starwars';
```

1.2. Ejercicio

- Accediendo desde la misma cuenta 'glucas', crea tres usuarios nuevos llamados 'yoda', 'r2d2' y 'darthvader', con las claves que quieras de manera que podamos gestionar los accesos de los tres según accedan desde localhost, o desde cualquier IP.
- Después, dale permisos de lectura (SELECT) a todos ellos sobre todas las bases de datos y todas las tablas si acceden desde fuera y de lectura y actualización (UPDATE) si lo hacen desde la máquina donde está la base de datos.
- Quítale el permiso de actualización a 'darthvader'.
- Muestra luego los permisos de cada uno de ellos, para los dos tipos de acceso, para asegurarte de que lo has dejado todo como se pedía.

Solución

Conexión a MySQL con usuario 'glucas':

```
terminal> mysql -u glucas -p
```

```
CREATE USER IF NOT EXISTS 'yoda'@'localhost' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'yoda'@'%' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'r2d2'@'localhost' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'r2d2'@'%' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'darthvader'@'localhost' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'darthvader'@'%' IDENTIFIED BY '123456'; --

GRANT SELECT, UPDATE ON *.* TO 'yoda'@'localhost'; --
GRANT SELECT ON *.* TO 'yoda'@'%'; --
GRANT SELECT, UPDATE ON *.* TO 'r2d2'@'localhost'; --
GRANT SELECT ON *.* TO 'r2d2'@'%'; --
```

```
GRANT SELECT, UPDATE ON *.* TO 'darthvader'@'localhost'; --  
GRANT SELECT ON *.* TO 'darthvader'@'%'; --
```

```
REVOKE UPDATE ON *.* FROM 'darthvader'@'%'; --
```

```
FLUSH PRIVILEGES;
```

```
SHOW GRANTS FOR 'yoda'@'localhost'; --  
SHOW GRANTS FOR 'yoda'@'%'; --  
SHOW GRANTS FOR 'r2d2'@'localhost'; --  
SHOW GRANTS FOR 'r2d2'@'%'; --  
SHOW GRANTS FOR 'darthvader'@'localhost'; --  
SHOW GRANTS FOR 'darthvader'@'%';
```


1.3. Ejercicio

- Crea un nuevo usuario llamado 'jedi', que pueda acceder todas a las tablas de la base de datos **db_empresa** para leer, actualizar, insertar y borrar cuando accede desde la misma máquina que la base de datos, y únicamente a la tabla **producto** de la base de datos **db_tienda_friki** cuando accede desde cualquier lugar.
- Muestra luego los permisos de cada uno de ellos, para los dos tipos de acceso, para asegurarte de que lo has dejado todo como se pedía.

Solución

```
CREATE USER IF NOT EXISTS 'jedi'@'localhost' IDENTIFIED BY '123456'; --
CREATE USER IF NOT EXISTS 'jedi'@'%' IDENTIFIED BY '123456'; --
GRANT SELECT, UPDATE, DELETE, INSERT ON db_empresa.* TO 'jedi'@'localhost'; --
GRANT SELECT, UPDATE, DELETE, INSERT ON db_tienda_friki.producto TO 'jedi'@'%'

SHOW GRANTS FOR 'jedi'@'localhost'; --
SHOW GRANTS FOR 'jedi'@'%';
```

1.4. Ejercicio

- Lista todos los pares usuario+host que has creado.
- Revoca todos los permisos de 'yoda', 'r2d2', 'darthvader' y 'jedi' que has concedido dependiendo de sus accesos, en orden inverso a su concesión.
- Elimina los usuarios+host.

Solución

```
SELECT user, host
FROM mysql.user
WHERE user = 'yoda' OR user = 'r2d2' OR user = 'darthvader' OR user = 'jedi';

REVOKE SELECT, UPDATE ON *.* FROM 'yoda'@'localhost'; --
REVOKE SELECT ON *.* FROM 'yoda'@'%'; --
REVOKE SELECT, UPDATE ON *.* FROM 'r2d2'@'localhost'; --
REVOKE SELECT ON *.* FROM 'r2d2'@'%'; --
REVOKE SELECT ON *.* FROM 'darthvader'@'localhost'; --
REVOKE SELECT ON *.* FROM 'darthvader'@'%'; --
REVOKE SELECT, UPDATE, DELETE, INSERT ON db_tienda_friki.producto FROM 'jedi'@'localhost';
REVOKE SELECT, UPDATE, DELETE, INSERT ON db_empresa.* FROM 'jedi'@'localhost';
FLUSH PRIVILEGES;

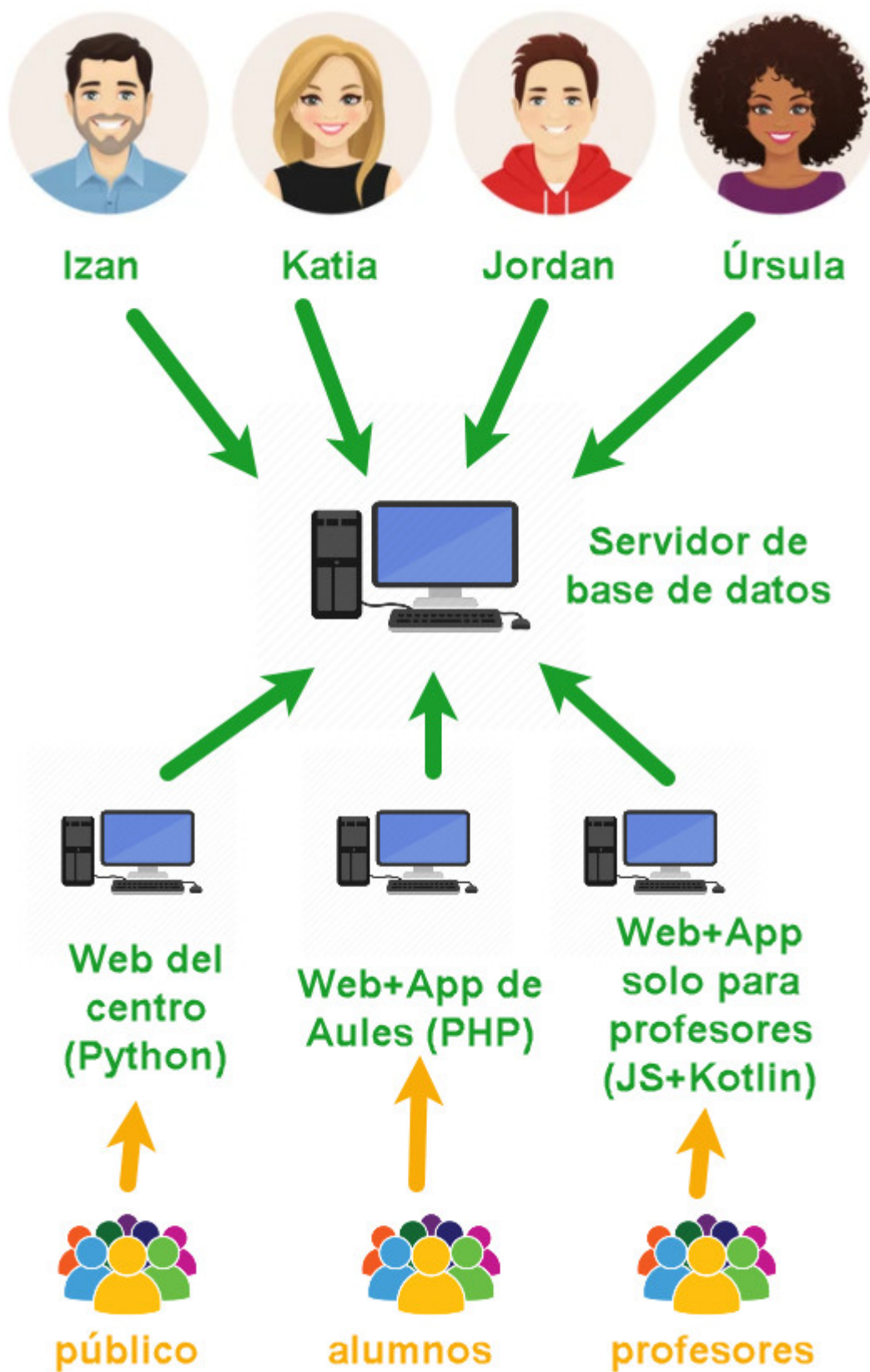
DROP USER 'yoda'@'localhost'; --
DROP USER 'yoda'@'%'; --
DROP USER 'r2d2'@'localhost'; --
DROP USER 'r2d2'@'%'; --
DROP USER 'darthvader'@'localhost'; --
DROP USER 'darthvader'@'%'; --
```

```
DROP USER 'jedi'@'localhost'; --  
DROP USER 'jedi'@'%';
```

2. BD Centro educativo

Actividad no evaluable

Queremos crear los usuarios y permisos necesarios para modelar el acceso a la base de datos de un centro educativo que tiene su propia base de datos, a la que acceden varios técnicos, la web del centro, la web+App de Aules y una web+App creadas para profesores.



a) Direcciones IP

- Servidor de base de datos. IP 252.258.25.4

- Departamento de Informática. IP 252.258.25.2
- Los técnicos se conectan desde sus casas con las siguientes IPs:
 - Izan 214.28.5.4
 - Katia 224.18.5.40
 - Jordan 224.8.5.22
 - Úrsula 224.8.53.52.
- Web del centro. Está alojada en la IP 252.258.25.1
- App y web de profesores. IP 252.258.25.3
- Aules. IP 211.11.12.13

b) Base de datos

El servidor de BD tiene esta información:

- BD alumnado. Tablas: estudiantes, materias
- BD profesorado. Tablas: docentes, documentos
- BD sistema. Tablas: eventos, matriculas, estudios

Debes ejecutar el script que se muestra debajo para crearla y poder gestionar los usuarios.

```
DROP DATABASE IF EXISTS alumnado;
DROP DATABASE IF EXISTS profesorado;
DROP DATABASE IF EXISTS sistema;

CREATE DATABASE alumnado;
CREATE DATABASE profesorado;
CREATE DATABASE sistema;

USE alumnado;
CREATE TABLE estudiantes (id INT PRIMARY KEY, nombre VARCHAR(50));
CREATE TABLE materias (id INT PRIMARY KEY, nombre VARCHAR(50));
CREATE TABLE notas (id INT PRIMARY KEY, nombre VARCHAR(50));

USE profesorado;
CREATE TABLE docentes (id INT PRIMARY KEY, nombre VARCHAR(50));
```

```
CREATE TABLE documentos (id INT PRIMARY KEY, nombre VARCHAR(50));
```

```
USE sistema;
```

```
CREATE TABLE eventos (id INT PRIMARY KEY, nombre VARCHAR(50));
```

```
CREATE TABLE matriculas (id INT PRIMARY KEY, nombre VARCHAR(50));
```

```
CREATE TABLE estudios (id INT PRIMARY KEY, nombre VARCHAR(50));
```

2.1. Ejercicio

Ejecuta las sentencias necesarias para los requisitos que se piden.

CONSEJO: Usa la clave '123' en todos los perfiles de usuario para hacer el ejercicio más sencillo.

- Cada técnico debe poder acceder desde su casa (con cualquier *software* que considere o accediendo por consola de manera remota) a cualquier tabla para lectura (SELECT), y a cualquier tabla para SELECT, INSERT, UPDATE y DELETE desde el Dpto. de Informática.
- Solo uno de ellos, Úrsula, debe poder conectarse desde el propio servidor (con permiso total) y dar permisos de todo tipo a todos los usuarios y, solo uno de ellos, Jordan, debe poder conectarse desde cualquier PC del mundo a la tabla sistema.eventos solo para lectura.
- Respecto a la web del centro, se podrá conectar usando Python solo a la tabla estudios de la BD sistema y solo en modo lectura.
- Respecto a Aules, se podrá conectar usando PHP a todas las tablas de la BD alumnado para SELECT, INSERT, UPDATE y DELETE.
- Respecto a la web/App para profesores, se podrá conectar usando JS y/o Kotlin a todas las tablas de la BD profesorado para SELECT, INSERT, UPDATE y DELETE.
- El equipo directivo cambia de opinión a última hora, y decide quitar el acceso de borrado de Aules a la BD alumnado y de la web/App de profesores a la BD profesorado.
- Izan cambia la IP de su casa a la 214.28.5.6.
- Úrsula cambia su clave para cuando se conecta desde el servidor de la BD: 983422nsd78

Solución


```

--
-- acceso desde IPs particulares
DROP USER IF EXISTS izan@214.28.5.4;
DROP USER IF EXISTS katia@224.18.5.40;
DROP USER IF EXISTS jordan@224.8.5.22;
DROP USER IF EXISTS ursula@224.8.53.52;

CREATE USER izan@214.28.5.4 IDENTIFIED BY '123';
CREATE USER katia@224.18.5.40 IDENTIFIED BY '123';
CREATE USER jordan@224.8.5.22 IDENTIFIED BY '123';
CREATE USER ursula@224.8.53.52 IDENTIFIED BY '123';

GRANT SELECT ON *.* TO izan@214.28.5.4, katia@224.18.5.40, jordan@224.8.5.22,

-- acceso desde localhost
DROP USER IF EXISTS ursula@localhost;
CREATE USER ursula@localhost IDENTIFIED BY '123';
GRANT ALL PRIVILEGES ON *.* TO ursula@localhost WITH GRANT OPTION;

-- acceso al departamento de Informática
DROP USER IF EXISTS izan@252.258.25.2;
DROP USER IF EXISTS katia@252.258.25.2;
DROP USER IF EXISTS jordan@252.258.25.2;
DROP USER IF EXISTS ursula@252.258.25.2;

CREATE USER izan@252.258.25.2 IDENTIFIED BY '123';
CREATE USER katia@252.258.25.2 IDENTIFIED BY '123';
CREATE USER jordan@252.258.25.2 IDENTIFIED BY '123';
CREATE USER ursula@252.258.25.2 IDENTIFIED BY '123';

GRANT SELECT, INSERT, UPDATE, DELETE ON *.* TO izan@252.258.25.2, katia@252.258.25.2,

-- acceso universal
DROP USER IF EXISTS jordan@'%'; --
CREATE USER jordan@'%' IDENTIFIED BY '123'; --

```

```

GRANT SELECT ON sistema.eventos TO jordan@'%';

-- acceso web del centro
DROP USER IF EXISTS webcentro@252.258.25.1;
CREATE USER webcentro@252.258.25.1 IDENTIFIED BY '123';
GRANT SELECT ON sistema.estudios TO webcentro@252.258.25.1;

-- acceso aules
DROP USER IF EXISTS aules@211.11.12.13;
CREATE USER aules@211.11.12.13 IDENTIFIED BY '123';
GRANT SELECT, INSERT, UPDATE, DELETE ON alumnado.* TO aules@211.11.12.13;

-- acceso web/app profesores
DROP USER IF EXISTS webprofes@252.258.25.3;
CREATE USER webprofes@252.258.25.3 IDENTIFIED BY '123';
GRANT SELECT, INSERT, UPDATE, DELETE ON profesorado.* TO webprofes@252.258.25.3;

-- quitar acceso a aules y webprofes
REVOKE DELETE ON profesorado.* FROM webprofes@252.258.25.3;
REVOKE DELETE ON alumnado.* FROM aules@211.11.12.13;

-- Izan cambia su IP, por lo que nos toca eliminar su perfil y volverlo a crear
-- estudiar si se puede modificar la ip de un perfil ?;
DROP USER IF EXISTS izan@214.28.5.4;
DROP USER IF EXISTS izan@214.28.5.6;
CREATE USER izan@214.28.5.6 IDENTIFIED BY '123';
GRANT SELECT ON *.* TO izan@214.28.5.6;

-- Úrsula cambia su clave
ALTER USER ursula@localhost IDENTIFIED BY '983422nsd78';

FLUSH PRIVILEGES;

```

3. Bibliografía

- MySQL 8.0 Reference Manual. <https://dev.mysql.com/doc/refman/8.0/en/>
- Oracle Database Documentation. <https://docs.oracle.com/en/database/oracle/oracle-database/index.html>
- MySQL Tutorial. <https://www.w3schools.com/mysql/>
- Mysql Gestión de permisos. https://wiki.cifprodolfoucha.es/index.php?title=Mysql_Gesti%C3%B3n_de_permisos#ROLES
- Tema 1: Lenguaje DCL: Usuarios y permisos en MYSQL. http://linkiafp.duckdns.org/M10%20-%20Administraci%C3%B3n%20de%20sistemas%20gestores%20de%20bases%20de%20UF1/Temario/ASIX_M10_T01_IMPR.pdf



Obra publicada con [Licencia Creative Commons Reconocimiento Compartir igual 4.0](https://creativecommons.org/licenses/by-sa/4.0/)