

Instala MariaDB en Windows y realiza capturas de pantalla donde se muestre que te has conectado al SGBD utilizando tanto la interfaz de comandos como HeidiSQL.

Le muestro la captura de pantalla cuando entro en MariaDB utilizando la interfaz de comandos y usando la interfaz de HeidiSQL

```
c:\ mysql -u root -p
Setting environment for using XAMPP for Windows.
Admin@BREA c:/xampp
# mysql -u root -p
Enter password: ****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.14-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

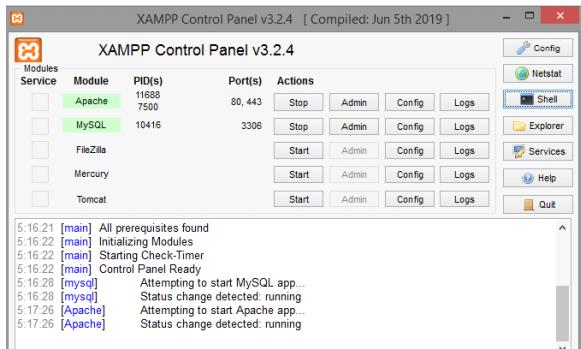
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| gasto    |
| information_schema |
| mysql    |
| performance_schema |
| phpmyadmin |
| test     |
| universidad |
+-----+
7 rows in set (0.171 sec)

MariaDB [(none)]>
```

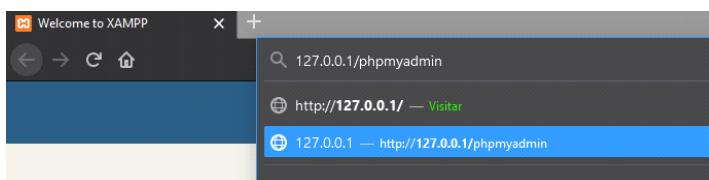
The screenshot shows the HeidiSQL application window. At the top, it says "MariaDB\universidad - HeidiSQL 11.0.0.5919". The menu bar includes Archivo, Editar, Buscar, Herramientas, Ir a, and Ayuda. Below the menu is a toolbar with various icons. The main area has a title bar "Filtro de bases de datos" and "Filtro de tablas". On the left, there's a tree view labeled "MariaDB" showing databases like gasto, information_schema, mysql, performance_schema, phpmyadmin, test, and universidad. The "universidad" database is expanded, showing a table named "asignatura" with 16,0 KiB size. To the right, there are tabs for "Base de datos: universidad", "Consulta", "Variables", "Estado", "Procesos", and "Estadísticas de Comandos". Under "Base de datos", there's a table with columns: Base de datos, Tamaño, Elem..., Últim..., Tablas, Vistas, Funci..., Proc..., Disp..., Event..., and Collation predeter... . The "universidad" database is listed with 16,0 KiB size, 1 element, 1 table, 0 views, 0 functions, 0 procedures, 0 triggers, and 0 events, using utf8mb4_spanish_ci collation.

2. Descarga e instala XAMPP para Windows, inicia los servicios necesarios e inicia sesión en MariaDB desde phpMyAdmin. Realiza capturas de pantalla que muestren la tarea pedida.

Para poder acceder a phpmyadmin , debemos activar Apache y mysql, y pulsamos Admin



Se abre una ventana y tecleamos la dirección de ip donde se encuentra nuestras bases de datos



Ponemos el usuario y la clave

The screenshot shows the phpMyAdmin login page. It features the phpMyAdmin logo and the text 'Bienvenido a phpMyAdmin'. There are two main sections: 'Idioma - Language' which shows 'Español - Spanish' selected, and 'Iniciar sesión' (Login) which contains fields for 'Usuario:' (root) and 'Contraseña:' (****). A 'Continuar' (Continue) button is at the bottom right.

Accedemos a phpMyAdmin

Bases de datos

Crear base de datos

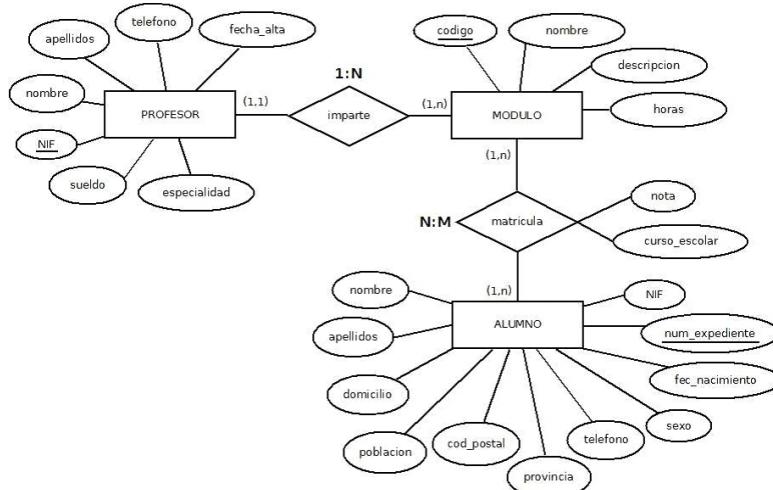
Nombre de la base de datos: utf8mb4_general_ci

Base de datos Cotejamiento Acción

- gasto utf8mb4_general_ci Seleccionar privilegios
- information_schema utf8_general_ci Seleccionar privilegios
- mysql utf8mb4_general_ci Seleccionar privilegios
- performance_schema utf8_general_ci Seleccionar privilegios
- phpmyadmin utf8_bin Seleccionar privilegios
- test latin1_swedish_ci Seleccionar privilegios
- universidad utf8mb4_spanish_ci Seleccionar privilegios

Total: 7

2. Despues de realizar el proceso de análisis y diseño de la BD para un centro de enseñanza, el diagrama E/R y el modelo relacional resultante son los siguientes:



MODULO (codigo, nombre, descripcion, horas, NIF_profesor)
 PROFESOR (NIF, nombre, apellido, telefono, sueldo, especialidad, fecha_alta)
 MATRICULA (expediente_alumno, cod_modulo, curso_escolar, nota)
 ALUMNO (num_expediente, NIF, nombre, apellido, domicilio, poblacion, cod_postal, provincia, telefono, sexo, fec_nacimiento)

A) Escribe las sentencias de creación de la BD y de las tablas en el orden apropiado utilizando el lenguaje SQL para que funcionen correctamente en MariaDB. Debes tener en cuenta lo siguiente:

- Elegir el tipo de dato y el tamaño o m(s adecuado para cada campo teniendo en cuenta los valores que éstos pueden almacenar.
- Recuerda nombrar adecuadamente los atributos, tablas y restricciones teniendo en cuenta las reglas que se explican en los contenidos de la unidad.
- Incluir en las sentencias de creación de tablas las siguientes restricciones utilizando el lenguaje SQL:
 - El campo **código** de la tabla **MODULO** identifica de forma única cada registro de la tabla y está formado por una cadena de **caracteres alfanuméricos de longitud fija**.
 - El campo **NIF** de la tabla **PROFESOR** identifica de forma única cada registro de la tabla y está formado por una cadena **de caracteres alfanuméricos de longitud fija**.
 - El campo **num_expediente** de la tabla **ALUMNO** identifica de forma única cada registro de la tabla y está formado por una cadena de **caracteres alfanuméricos de longitud fija**.
 - Los campos **expediente_alumno, cod_modulo y curso_escolar** de la tabla **MATRICULA** identifican de forma conjunta cada registro de la tabla y provienen de las entidades **ALUMNO** y **MODULO** que participan en la relación respectivamente.
 - El campo **NIF_profesor** de la tabla **MODULO** referencia a la tabla **PROFESOR**.
 - Se debe cumplir la **regla de integridad referencial** en todas las relaciones.
 - El **sueldo** de los profesores podrá estar comprendido entre **1500 y 2000 euros**.
 - El campo **horas** de la tabla **MODULO** no puede ser mayor de **300 horas**.
 - La **fecha de nacimiento** de los alumnos no puede contener valores nulos.
 - El campo **NIF** de la tabla **ALUMNO** no puede repetirse.

Solución:

Primero creo una base de datos, por ejemplo llamada UNIVERSIDAD, donde recogeremos las tablas de módulo, profesor, alumno y matricula.

CREATE DATABASE UNIVERSIDAD; // o bien, CREATE SCHEMA UNIVERSIDAD;

```
MariaDB [(none)]> CREATE SCHEMA UNIVERSIDAD;
Query OK, 1 row affected (0.035 sec)
```

Muestro las bases de datos creadas y selecciono universidad.

SHOW DATABASES;

USE UNIVERSIDAD;

```

MariaDB [none]> SHOW DATABASES;
+-----+
| Database |
+-----+
| gasto   |
| information_schema |
| mysql   |
| performance_schema |
| phpmyadmin |
| test    |
| universidad |
+-----+
7 rows in set (0.002 sec)

MariaDB [none]> USE UNIVERSIDAD;
Database changed

```

Primero creo la tabla de Profesor, ya que vamos a necesitar su clave primaria como clave foránea en la tabla módulo. Las condiciones que debe tener la tabla profesor, son: la clave primaria que será el nif es un tipo de dato alfanumérico de longitud fija, es decir, CHARACTER(9) que contiene 9 de longitud porque 8 le corresponderá al número y uno a la letra; También tendremos en cuenta el sueldo del profesor debe ser entre 1500 y 2000 euros.

Creo los nombre y apellidos del profesor una restricción not null, para que sea obligatorio ponerlo, el teléfono le pongo una longitud de 15 ya que hay algunos teléfonos que pueden traer prefijos como +34 ... etc, y considero el sueldo como tipo de dato un float con dos decimales de precisión para poner la restricción uso check en el intervalo indicado por el problema.

```

CREATE TABLE PROFESOR(
    NIF CHARACTER(9) PRIMARY KEY,
    NOMBRE VARCHAR(20) NOT NULL,
    APELLIDOS VARCHAR(40) NOT NULL,
    TELEFONO VARCHAR(15) NOT NULL,
    SUELDO FLOAT(4.2) CHECK (SUELDO>=1500.0 AND SUELDO<=2000.0),
    ESPECIALIDAD VARCHAR(40),
    FECHA_ALTA DATE);

```

```

MariaDB [UNIVERSIDAD]> CREATE TABLE PROFESOR(
->     NIF CHARACTER(9) PRIMARY KEY,
->     NOMBRE VARCHAR(20) NOT NULL,
->     APELLIDOS VARCHAR(40) NOT NULL,
->     TELEFONO VARCHAR(15),
->     SUELDO FLOAT(4.2) CHECK (SUELDO>=1500.0 AND SUELDO<=2000.0),
->     ESPECIALIDAD VARCHAR(40),
->     FECHA_ALTA DATE);
Query OK, 0 rows affected (0.342 sec)

```

Ahora creo la tabla MODULO, tengo en cuenta, la condición de que el número de horas del módulo debe ser inferior a 300 horas, que la clave primaria está formado por una longitud fija de 9 caracteres y que nif_profesor se trata de una clave foránea que es clave primaria en la tabla profesor.

```
CREATE TABLE MODULO(
    CODIGO CHARACTER(10) PRIMARY KEY,
    NOMBRE VARCHAR(30) NOT NULL,
    DESCRIPCION VARCHAR(255),
    HORAS INT(3) CHECK (HORAS<300),
    NIF_PROFESOR CHARACTER(9),
    FOREIGN KEY (NIF_PROFESOR) REFERENCES PROFESOR(NIF));
```

```
MariaDB [UNIVERSIDAD]> CREATE TABLE MODULO(
    -> CODIGO CHARACTER(10) PRIMARY KEY,
    -> NOMBRE VARCHAR(30) NOT NULL,
    -> DESCRIPCION VARCHAR(255),
    -> HORAS INT(3) CHECK (HORAS<300),
    -> NIF_PROFESOR CHARACTER(9),
    -> FOREIGN KEY (NIF_PROFESOR) REFERENCES PROFESOR(NIF);
Query OK, 0 rows affected (0.502 sec)
```

A continuación creo la tabla alumno, tengo un consideración lo siguiente: num_expediente es la clave primaria y además es alfanumérico con longitud fija, la fecha de nacimiento del alumno no puede contener valores nulos y el campo nif del alumno no puede repetirse, por lo tanto, será de tipo unique.

```
CREATE TABLE ALUMNO(
    NUM_EXPEDIENTE CHARACTER(10) PRIMARY KEY,
    NIF CHARACTER(9) UNIQUE,
    NOMBRE VARCHAR(20) NOT NULL,
    APELLIDOS VARCHAR(40) NOT NULL,
    DOMICILIO VARCHAR(50),
    POBLACION VARCHAR(30),
    COD_POSTAL INT(5),
    PROVINCIA VARCHAR(20),
    TELEFONO VARCHAR(15),
    SEXO CHARACTER(1),
    FEC_NACIMIENTO DATE NOT NULL);
```

```
MariaDB [universidad]> CREATE TABLE ALUMNO(
-> NUM_EXPEDIENTE CHARACTER(10) PRIMARY KEY,
-> NIF CHARACTER(9) UNIQUE,
-> NOMBRE VARCHAR(20) NOT NULL,
-> APELLIDOS VARCHAR(40) NOT NULL,
-> DOMICILIO VARCHAR(50),
-> POBLACION VARCHAR(30),
-> COD_POSTAL INT(5),
-> PROVINCIA VARCHAR(20),
-> TELEFONO VARCHAR(15),
-> SEXO CHARACTER(1),
-> FEC_NACIMIENTO DATE NOT NULL);
Query OK, 0 rows affected (0.335 sec)
```

Nos falta la tabla matricula donde num_expediente, cod_modulo y curso identifica a cada registro de la tabla:

```
CREATE TABLE MATRICULA(
EXPEDIENTE_ALUMNO CHARACTER(10),
COD_MODULO CHARACTER(10),
CURSO_ESCOLAR CHARACTER(10),
NOTA FLOAT(2.1),
PRIMARY KEY(EXPEDIENTE_ALUMNO,COD_MODULO,CURSO_ESCOLAR),
FOREIGN KEY (EXPEDIENTE_ALUMNO) REFERENCES ALUMNO (NUM_EXPEDIENTE),
FOREIGN KEY (COD_MODULO) REFERENCES MODULO (CODIGO));
```

```
MariaDB [universidad]> CREATE TABLE MATRICULA(
-> EXPEDIENTE_ALUMNO CHARACTER(10),
-> COD_MODULO CHARACTER(10),
-> CURSO_ESCOLAR CHARACTER(10),
-> NOTA FLOAT(2.1),
-> PRIMARY KEY(EXPEDIENTE_ALUMNO,COD_MODULO,CURSO_ESCOLAR),
-> FOREIGN KEY(EXPEDIENTE_ALUMNO) REFERENCES ALUMNO(NUM_EXPEDIENTE),
-> FOREIGN KEY(COD_MODULO) REFERENCES MODULO(CODIGO));
Query OK, 0 rows affected (0.389 sec)
```

B. Despues de crear las tablas, vamos a realizar algunas modificaciones sobre ellas. Escribe las sentencias SQL apropiadas para cada uno de los subapartados siguientes:

- Elimina la restriccin que establece el valor del campo sueldo entre 1500 y 2000 que se creó anteriormente.

```
ALTER TABLE PROFESOR MODIFY COLUMN SUELDO FLOAT(4.2);
```

```
MariaDB [UNIVERSIDAD]> ALTER TABLE PROFESOR MODIFY COLUMN SUELDO FLOAT(4.2);
Query OK, 0 rows affected (0.052 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Modifico esta errata de esta forma

```
MariaDB [UNIVERSIDAD]> ALTER TABLE PROFESOR MODIFY COLUMN SALARIO FLOAT(6,2);
Query OK, 0 rows affected (0.046 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Elimina el campo descripción de la tabla MODULO.

```
ALTER TABLE MODULO DROP COLUMN DESCRIPCION;
```

```
MariaDB [universidad]> ALTER TABLE MODULO DROP COLUMN DESCRIPCION;
Query OK, 0 rows affected (0.277 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Añade un campo en la tabla PROFESOR llamado cargo.

```
ALTER TABLE PROFESOR ADD COLUMN CARGO VARCHAR(20);
```

```
MariaDB [universidad]> ALTER TABLE PROFESOR ADD COLUMN CARGO VARCHAR(20);
Query OK, 0 rows affected (0.141 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Modifica la tabla MATRICULA para que el campo nota sólo pueda tomar valores comprendidos entre 0 y 10.

```
ALTER TABLE MATRICULA ADD CHECK (NOTA>=0 AND NOTA<=10);
```

```
MariaDB [universidad]> ALTER TABLE MATRICULA ADD CHECK (NOTA>=0 AND NOTA<=10);
Query OK, 0 rows affected (1.301 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Crea un índice con el campo nombre de la tabla ALUMNO.

```
CREATE INDEX INDICE_ALUMNO ON ALUMNO (NOMBRE);
```

```
MariaDB [universidad]> CREATE INDEX INDICE_ALUMNO ON ALUMNO (NOMBRE);
Query OK, 0 rows affected (0.363 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Renombra la tabla MODULO por ASIGNATURA.

```
ALTER TABLE MODULO RENAME TO ASIGNATURA;
```

```
MariaDB [universidad]> ALTER TABLE MODULO RENAME TO ASIGNATURA;
Query OK, 0 rows affected (0.646 sec)
```

- Modifica el nombre del campo sueldo por el de salario en la tabla PROFESOR.

```
MariaDB [UNIVERSIDAD]> ALTER TABLE PROFESOR CHANGE SUELDO SALARIO FLOAT(4.2);
Query OK, 0 rows affected (0.057 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- Desactiva temporalmente la restricción establecida en el campo horas de la tabla MODULO

Aquí como cree la tabla usando el CHECK en la misma linea, no la puse como constraint, entonces lo primero que voy hacer es modificar la columna y añadir el check como contricción.

```
MariaDB [UNIVERSIDAD]> ALTER TABLE ASIGNATURA MODIFY COLUMN HORAS INT(3);
Query OK, 0 rows affected (0.043 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Añado una restricción check a mi tabla llamada NUM_HORAS

```
MariaDB [UNIVERSIDAD]> ALTER TABLE ASIGNATURA ADD CONSTRAINT NUM_HORAS CHECK (HORAS<300);
Query OK, 0 rows affected (0.254 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Como es el único check de la tabla puedo usar esta opción:

```
MariaDB [UNIVERSIDAD]> SET CHECK_CONSTRAINT_CHECKS = 0;
Query OK, 0 rows affected (0.001 sec)
```

- El dominio del campo sexo sólo puede ser "H" (de Hombre) o "M" (de Mujer).

ALTER TABLE ALUMNO ADD CONSTRAINT GENERO CHECK (SEXO='H' OR SEXO='M');

```
MariaDB [UNIVERSIDAD]> ALTER TABLE ALUMNO ADD CONSTRAINT GENERO CHECK (SEXO='H' OR SEXO='M');
Query OK, 0 rows affected (0.269 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

4. Crea un documento de texto que contenga las sentencias DCL de SQL para dar respuesta a:

a) Crear un usuario “centro” con clave “centro”.

```
CREATE USER CENTRO IDENTIFIED BY "centro";
```

```
MariaDB [universidad]> CREATE USER CENTRO IDENTIFIED BY "centro";
Query OK, 0 rows affected (0.063 sec)
```

b) Mostrar los privilegios asignados al usuario “centro”.

```
SHOW GRANTS FOR CENTRO;
```

```
MariaDB [universidad]> SHOW GRANTS FOR CENTRO;
+-----+
| Grants for CENTRO@%          |
+-----+
| GRANT USAGE ON *.* TO 'CENTRO'@'%' IDENTIFIED BY PASSWORD '*DB645B53B86A8F0B22360D89D9D5CD7E31E68EDD' |
+-----+
1 row in set <0.002 sec>
```

c) ¿Tiene el usuario “centro” acceso a la BD creada en el ejercicio 3? ¿a qué se debe?

Si tiene acceso a la base de dato creada en el ejercicio 3 porque nos indica ON *.* que significa acceso global, por lo tanto, tiene acceso a todas las tablas de nuestras bases de datos.

d) Asignar permiso de consulta a la tabla MODULO. (que ahora se llama ASIGNATURA)

```
GRANT SELECT ON UNIVERSIDAD.ASIGNATURA TO CENTRO;
```

```
MariaDB [universidad]> GRANT SELECT ON UNIVERSIDAD.ASIGNATURA TO CENTRO;
Query OK, 0 rows affected <0.051 sec>
```

e) ¿Tiene ahora el usuario “centro” acceso a la BD creada en el ejercicio 3? ¿a qué se debe?

He entrado con la cuenta de CENTRO y no me da acceso a mi base de dato UNIVERSIDAD

```
MariaDB [(none)]> USE UNIVERSIDAD;
ERROR 1044 (42000): Access denied for user 'CENTRO'@'%' to database 'universidad'
```

f) Utilizando HeidiSQL intenta dar de alta un registro en la tabla MODULO utilizando el usuario “centro”. ¿Se puede? ¿razona la respuesta?

No se puede, porque nuestro usuario no tiene permiso para insertar registros, ni siquiera tiene acceso a la base de dato UNIVERSIDAD que es donde tenemos la tabla MODULO, que se llama esta tabla ahora ASIGNATURA, tras un apartado del ejercicio anterior.

g) Asigna permiso de inserción a todas las tablas de la BD creada en el ejercicio 3?

```

MariaDB [universidad]> SHOW TABLES;
+-----+
| Tables_in_universidad |
+-----+
| alumno
| asignatura
| matricula
| profesor
+-----+
4 rows in set (0.002 sec)

MariaDB [universidad]> GRANT INSERT ON UNIVERSIDAD.ALUMNO TO CENTRO;
Query OK, 0 rows affected (0.088 sec)

MariaDB [universidad]> GRANT INSERT ON UNIVERSIDAD.ASIGNATURA TO CENTRO;
Query OK, 0 rows affected (0.043 sec)

MariaDB [universidad]> GRANT INSERT ON UNIVERSIDAD.MATRICULA TO CENTRO;
Query OK, 0 rows affected (0.048 sec)

MariaDB [universidad]> GRANT INSERT ON UNIVERSIDAD.PROFESOR TO CENTRO;
Query OK, 0 rows affected (0.083 sec)

```

h) Mostrar los privilegios asignados al usuario “centro”. ¿Qué tablas se muestran al seleccionar la BD si se entra con el usuario “centro”?

```

MariaDB [universidad]> SHOW GRANTS FOR CENTRO;
+-----+
| Grants for CENTRO@%
|   :
+-----+
| GRANT USAGE ON *.* TO 'CENTRO'@'%' IDENTIFIED BY PASSWORD '*DB645B53B86A8F0B22
360D89D9D5CD7E31E68EDD'
| GRANT INSERT ON `universidad`.'alumno' TO 'CENTRO'@'%'
| GRANT INSERT ON `universidad`.'matricula' TO 'CENTRO'@'%'
| GRANT SELECT, INSERT ON `universidad`.'asignatura' TO 'CENTRO'@'%'
| GRANT INSERT ON `universidad`.'profesor' TO 'CENTRO'@'%'
+-----+
5 rows in set (0.001 sec)

```

Las tablas que se muestra cuando accedemos con la cuenta de CENTRO, son las tablas donde le hemos dado privilegio

```

MariaDB [none]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| test           |
| universidad   |
+-----+
3 rows in set (0.002 sec)

MariaDB [none]> use universidad;
Database changed
MariaDB [universidad]> show tables;
+-----+
| Tables_in_universidad |
+-----+
| alumno        |
| asignatura    |
| matricula     |
| profesor      |
+-----+
4 rows in set (0.002 sec)

```

i) Revoca todos los permisos asignados al usuario “centro” sobre la tabla “ALUMNO”.

```

MariaDB [none]> REVOKE ALL ON UNIVERSIDAD.ALUMNO FROM CENTRO;
Query OK, 0 rows affected (0.045 sec)

MariaDB [none]> SHOW GRANTS FOR CENTRO;
+-----+
| Grants for CENTRO@%          |
|                               |
+-----+
| GRANT USAGE ON *.* TO 'CENTRO'@'%' IDENTIFIED BY PASSWORD '*DB645B53B86A8F0B22360D89D9D5CD7E31E68EDD' |
| GRANT SELECT, INSERT ON `universidad`.`asignatura` TO 'CENTRO'@'%'                                |
| GRANT INSERT ON `universidad`.`matricula` TO 'CENTRO'@'%'                                     |
| GRANT INSERT ON `universidad`.`profesor` TO 'CENTRO'@'%'                                     |
+-----+
4 rows in set (0.001 sec)

```

Compruebo que al quitarle todos los privilegios sobre la tabla ALUMNO, usando la sentencia de mysql siguiente REVOKE ALL ON UNIVERSIDAD.ALUMNO FROM CENTRO; , el user CENTRO no tiene privilegios sobre la tabla ALUMNO.

j) Asigna el permiso de selección sobre las columnas “codigo” y “nombre” de la tabla ASIGNATURA al usuario “centro”.

```

MariaDB [none]> GRANT SELECT (CODIGO,NOMBRE) ON UNIVERSIDAD.ASIGNATURA TO CENTRO;
Query OK, 0 rows affected (0.056 sec)

```

k) Asigna el permiso para crear tablas al usuario “centro” sobre la BD creada en el ejercicio 3.

```

MariaDB [UNIVERSIDAD]> GRANT CREATE ON UNIVERSIDAD TO CENTRO;
Query OK, 0 rows affected (0.042 sec)

```

l) Cambia la clave al usuario “centro”, establece como nueva clave “centro1”.

```
MariaDB [UNIVERSIDAD]> ALTER USER CENTRO IDENTIFIED BY 'CENTRO2';
Query OK, 0 rows affected (0.060 sec)
```

m) Quitar todos los permisos asignados al usuario “centro”.

```
MariaDB [UNIVERSIDAD]> REVOKE ALL ON UNIVERSIDAD FROM CENTRO;
Query OK, 0 rows affected (0.039 sec)
```

n) Dar todos los permisos al usuario “centro” sobre la BD creada en el ejercicio 3, con la posibilidad de otorgar permisos a otros usuarios.

```
MariaDB [UNIVERSIDAD]> GRANT ALL ON UNIVERSIDAD TO CENTRO;
Query OK, 0 rows affected (0.041 sec)
```

o) Borrar el usuario “centro”.

```
MariaDB [UNIVERSIDAD]> DROP USER CENTRO;
Query OK, 0 rows affected (0.037 sec)
```

5. Creación de usuarios, permisos y roles (crea un documento de texto que contenga las sentencias DCL de SQL para dar respuesta a los siguientes apartados):

a. Conectar como usuario “root” al servidor de *BD MariaDB*:

- ✓ Crear una *BD* “hotel” y dentro de esta una tabla “reserva” con varios campos: idReserva, fecha, descripción,...
- ✓ Crear un usuario con nombre “pepe” y contraseña “pepe1234” y *BD* por defecto “hotel”.
No asignar ningún permiso.

Creo la base de dato

```
CREATE DATABASE HOTEL;
```

```
MariaDB [(none)]> CREATE DATABASE HOTEL;
Query OK, 1 row affected (0.007 sec)
```

Selecciono la base de dato USE HOTEL;

y creo la siguiente tabla reserva

```
MariaDB [HOTEL]> CREATE TABLE RESERVA(
-> IDRESERVA INT AUTO_INCREMENT PRIMARY KEY,
-> FECHA DATE,
-> DESCRIPCION VARCHAR(255),
-> PRECIO DECIMAL(6,2) CHECK (PRECIO>0),
-> DESCUENTO_PORCENTUAL DECIMAL(4,2) CHECK (DESCUENTO_PORCENTUAL>0));
Query OK, 0 rows affected (0.115 sec)
```

Creo un usuario PEPE con clave pepe1234

```
MariaDB [hotel]> GRANT USAGE ON hotel.* TO pepe IDENTIFIED BY 'pepe1234';
Query OK, 0 rows affected (0.004 sec)
```

b. Conectarse con el usuario “pepe” ¿qué puede hacer? ¿por qué? ¿se le ha asignado algún rol a “pepe”?

Se ha creado el usuario pepe, pero no puede acceder a nada y no tiene ningun permiso

```
Admin@DESKTOP-R7TAT5T c:\xampp
# mysql -u pepe -p
Enter password: *****
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 51
Server version: 10.4.17-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| test          |
+-----+
2 rows in set (0.002 sec)
```

No puede acceder a la base de dato hotel

```
MariaDB [(none)]> use hotel;
ERROR 1044 (42000): Access denied for user 'pepe'@'%' to database 'hotel'
```

No se le ha asignado ningun rol a nuestro usuario pepe aún.

c. Definir un rol “usuariosHotel”. Otorgarle la posibilidad de manipular los datos de las tablas de la *BD* “hotel”.

```

MariaDB [hotel]> CREATE ROLE usuariosHotel;
Query OK, 0 rows affected (0.003 sec)

MariaDB [hotel]> GRANT SELECT,INSERT,UPDATE,DELETE ON hotel.* TO usuariosHotel;
Query OK, 0 rows affected (0.003 sec)

MariaDB [hotel]> SHOW GRANTS FOR usuariosHotel;
+-----+
| Grants for usuariosHotel |
+-----+
| GRANT USAGE ON *.* TO `usuariosHotel` |
| GRANT SELECT, INSERT, UPDATE, DELETE ON `hotel`.* TO `usuariosHotel` |
+-----+
2 rows in set (0.001 sec)

```

d. Crear el usuario “juan” y asignarle el rol “usuariosHotel”.

```

MariaDB [hotel]> CREATE USER juan IDENTIFIED BY 'juan';
Query OK, 0 rows affected (0.003 sec)

MariaDB [hotel]> GRANT usuariosHotel TO juan;
Query OK, 0 rows affected (0.002 sec)

MariaDB [hotel]> SET DEFAULT ROLE usuariosHotel FOR juan;
Query OK, 0 rows affected (0.003 sec)

```

e. Modificar el rol “usuariosHotel” dejándole únicamente la posibilidad de consultar las reservas.
 Conectar como “juan” y tratar de insertar un nuevo registro. ¿se puede? Hacerlo con la cuenta de “root”.
 ¿se puede?

```

MariaDB [hotel]> REVOKE DELETE,UPDATE,INSERT ON hotel.* FROM usuariosHotel;
Query OK, 0 rows affected (0.003 sec)

```

Entramos con el usuario juan he intento insertar un valor

```

# mysql -u juan -p
Enter password: ****
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 53
Server version: 10.4.17-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| hotel    |
| information_schema |
| test     |
+-----+

MariaDB [hotel]> INSERT INTO reserva VALUES(1, '31-01-2021', 'Miguel Angel Brea grupo a 1DAM', 100, 5);
ERROR 1142 (42000): INSERT command denied to user 'juan'@'localhost' for table 'reserva'

```

Como era de esperar accion denegada, ya que nuestro usuario no puede insertar, solo puede seleccionar.
 Con el usuario root si podemos, ya que tenemos todos los privilegios.

```
MariaDB [hotel]> INSERT INTO reserva VALUES(1, '31-01-2021', 'Miguel Angel Brea grupo a 1DAM', 100, 5);
Query OK, 1 row affected, 1 warning (0.005 sec)
```

f. Borrar el rol “usuariosHotel”. ¿se puede?

```
MariaDB [hotel]> DROP ROLE usuariosHotel;
Query OK, 0 rows affected (0.004 sec)
```

Si se puede eliminar el role.