Examen práctico 4.

Sqlite:

1. Versión que estoy utilizando.

- Con el comando sqlite3 –version en el cmd de Windows podemos comprobar la versión.

```
C:\Users\Angela\Desktop\sqlite-tools-win32-x86-3400100>sqlite3 --version
3.40.1 2022-12-28 14:03:47 df5c253c0b3dd24916e4ec7cf77d3db5294cc9fd45ae7b9c5e82ad8197f38a24
```

- También desde el cmd podemos ejecutar el comando sqlite3 que conecta y muestra la versión.

```
C:\Users\Angela\Desktop\sqlite-tools-win32-x86-3400100>sqlite3
SQLite version 3.40.1 2022-12-28 14:03:47
Enter ".help" for usage hints.
Connected to a transient in-memory database.
Use ".open FILENAME" to reopen on a persistent database.
sqlite>
```

No es necesario ejecutar los comandos en la cmd ya que en la propia consola de sqlite podemos ver la versión arriba de todo.

2. Realizar la consulta de cuánto es 20 por 50

```
sqlite> select 20*50;
1000
```

3. Realiza la importación del archivo de direcciones a tu BD sqlite

Creamos la base de datos y la tabla donde vamos a importar que en este caso la vamos a llamar Contactos:

```
sqlite> .open operaciones.db
sqlite> CREATE TABLE Contactos (
   ...>   nombre TEXT,
   ...>   apellido TEXT,
   ...>   direccion TEXT,
   ...>   ciudad TEXT,
   ...>   estado TEXT,
   ...>   cp TEXT
```

Realizamos la importación y comprobamos que se realizó:

```
sqlite> .import --csv /Users/Angela/Desktop/addresses.csv Contactos
sqlite> SELECT * FROM Contactos;
John|Doe|120 jefferson st.|Riverside| NJ| 08075
Jack|McGinnis|220 hobo Av.|Phila| PA|09119
John "Da Man"|Repici|120 Jefferson St.|Riverside| NJ|08075
Stephen|Tyler|7452 Terrace "At the Plaza" road|SomeTown|SD| 91234
|Blankman||SomeTown| SD| 00298
Joan "the bone", Anne|Jet|9th, at Terrace plc|Desert City|CO|00123
sqlite>
```

MariaDB

1. Qué BD tienes instaladas en el sistema.

Con el comando show databases; podemos ver las que tenemos

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| prueba             |
| sys                |
+--------------------+
5 rows in set (0.001 sec)
```

2. Crea una base de datos que se llame ejemplo, con una tabla que se llame tabla1 con dos campos. Exporta la base de datos y envíamela.

Como se puede ver en la imagen una vez creada la base de datos con el comando CREATE DATABASE ejemplo; se crea la tabla1, a mayores añadí valores para comprobar que funcionase.

```
MariaDB [(none)]> USE ejemplo;
Database changed
MariaDB [ejemplo]> CREATE TABLE tabla1 (id INT, libro VARCHAR(50));
Query OK, 0 rows affected (0.036 sec)

MariaDB [ejemplo]> SELECT * FROM tabla1;
Empty set (0.063 sec)

MariaDB [ejemplo]> INSERT INTO tabla1 (id,libro) VALUES (1, 'El puerto escondido');
Query OK, 1 row affected (0.012 sec)

MariaDB [ejemplo]> SELECT * FROM tabla1;
+------+--------------------+
| id   | libro              |
+------+--------------------+
|    1 | El puerto escondido |
+------+--------------------+
1 row in set (0.008 sec)
```

Diferentes opciones para exportar:

```
C:\Users\Angela\Desktop>mysqldump -u root -p ejemplo > ejemplo.sql
Enter password: ********

C:\Users\Angela\Desktop>mysqldump -u root -p ejemplo > ejemplo.dump
Enter password: ********
```

Postgres

Crea una BD que se llame direcciones y muéstrame cuántas BD están creadas

Primero creamos la BD con el comando CREATE DATABASE direcciones;

Después comprobamos que bases de datos tenemos:

```
postgres=# \l
                                                  Listado de base de datos
   Nombre    | Due±o   | Codificaci¾n |      Collate      |      Ctype        | configuraci¾n ICU | Proveedor de locale |      Privilegios
-------------+---------+--------------+-------------------+-------------------+-------------------+---------------------+-----------------------
 direcciones | postgres | UTF8         | Spanish_Spain.1252 | Spanish_Spain.1252 |                   | libc                |
 postgres    | postgres | UTF8         | Spanish_Spain.1252 | Spanish_Spain.1252 |                   | libc                |
 prueba      | postgres | UTF8         | Spanish_Spain.1252 | Spanish_Spain.1252 |                   | libc                |
 template0   | postgres | UTF8         | Spanish_Spain.1252 | Spanish_Spain.1252 |                   | libc                | =c/postgres          +
             |          |              |                    |                    |                   |                     | postgres=CTc/postgres
 template1   | postgres | UTF8         | Spanish_Spain.1252 | Spanish_Spain.1252 |                   | libc                | =c/postgres          +
             |          |              |                    |                    |                   |                     | postgres=CTc/postgres
(5 filas)
```

Nos conectamos a la base de datos direcciones:

```
postgres=# \c direcciones;
Ahora está conectado a la base de datos «direcciones» con el usuario «postgres».
direcciones=#
```

Creamos la tabla y comprobamos si está bien realizada:

```
direcciones=# CREATE TABLE Contactos (nombre VARCHAR(20), apellido VARCHAR(20), direccion VARCHAR(40), ciudad VARCHAR(10), estado VARCHAR(15), CP VARCHAR(10));
CREATE TABLE
direcciones=# \dt
        Listado de relaciones
 Esquema |  Nombre   | Tipo  |  Dueño
---------+-----------+-------+----------
 public  | contactos | tabla | postgres
(1 fila)


direcciones=#
```

Al hacer la importación he tenido que aumentar el tamaño de los VARCHAR como se puede ver en la siguiente imagen:

```
direcciones=# copy Contactos(nombre, apellido, direccion, ciudad, estado, CP) FROM 'C:\addresses.csv' DELIMITER ',' CSV HEADER;
ERROR:  el valor es demasiado largo para el tipo character varying(20)
CONTEXTO:  COPY contactos, line 6, column nombre: "Joan "the bone", Anne"
direcciones=# ALTER TABLE Contactos ALTER COLUMN nombre TYPE VARCHAR(50);
ALTER TABLE
direcciones=# copy Contactos(nombre, apellido, direccion, ciudad, estado, CP) FROM 'C:\addresses.csv' DELIMITER ',' CSV HEADER;
ERROR:  el valor es demasiado largo para el tipo character varying(10)
CONTEXTO:  COPY contactos, line 6, column ciudad: "Desert City"
direcciones=# ALTER TABLE Contactos ALTER COLUMN ciudad TYPE VARCHAR(50);
ALTER TABLE
direcciones=# copy Contactos(nombre, apellido, direccion, ciudad, estado, CP) FROM 'C:\addresses.csv' DELIMITER ',' CSV HEADER;
COPY 5
direcciones=#
```

Comprobamos que se realizó la importación:

```
COPY 5
direcciones=# SELECT * FROM Contactos;
       nombre         | apellido |           direccion             |   ciudad    | estado |   cp
----------------------+----------+---------------------------------+-------------+--------+--------
 Jack                 | McGinnis | 220 hobo Av.                    | Phila       |  PA    | 09119
 John "Da Man"        | Repici   | 120 Jefferson St.               | Riverside   |  NJ    | 08075
 Stephen              | Tyler    | 7452 Terrace "At the Plaza" road | SomeTown    | SD     |  91234
                      | Blankman |                                 | SomeTown    |  SD    |  00298
 Joan "the bone", Anne | Jet      | 9th, at Terrace plc             | Desert City | CO     | 00123
(5 filas)


direcciones=#
```