Práctica Número 3. Unidad II

1. **Crear tabla Estudiante**

```
sqlite> CREATE TABLE Estudiante(
(x1...> matricula int not null primary key,
(x1...> name text not null,
(x1...> paterno text not null,
(x1...> materno text null,
(x1...> carrera text not null)
   ...> ;
sqlite> .tables
Estudiante  usuario
sqlite>
```

2. **Insertar 5 registros de compañeros**

```
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('1222100503', 'Alan', 'Aguilar', 'Fuentes', 'Redes Digitales');
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('1222100486', 'Angel', 'Mancilla', 'Ortiz', 'Redes Digitales');
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('1222100895', 'Monserrat', 'Orduña', 'Redes Digitales');
Parse error: 4 values for 5 columns
sqlite> VALUES ('1222100895', 'Monserrat', 'Orduña', ' ', 'Redes Digitales');
1222100895|Monserrat|Orduña| |Redes Digitales
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('1222100895', 'Monserrat', 'Orduña', ' ', 'Redes Digitales');
```

```
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('1222100815', 'Sharon', 'Olvera', 'Ibarra', 'Redes Digitales');
sqlite>
```

3. **Listar todos los estudiantes, pero, solo matricula, nombre**

```
sqlite> SELECT matricula, name FROM Estudiante;
1222100487|Kevin
1222100503|Alan
1222100486|Angel
1222100895|Monserrat
1222100815|Sharon
```

4. **Insertar registro**
   - **Matricula: 100210046**
   - **Nombre: Abraham**
   - **Paterno: López**
   - **Materno: Juárez**
   - **Carrera: Redes Digitales**

```
sqlite> INSERT INTO Estudiante(matricula, name, paterno, materno, carrera)
   ...> VALUES ('100210046', 'Abraham', 'Lopez', 'Juarez', 'Redes Digitales');
sqlite>
```

```
100210046|Abraham
```

5. **Modificar el registro Abraham**
   Abraham Josue

```
sqlite> UPDATE Estudiante
   ...> SET
   ...> name = 'Abraham Josue'
   ...> WHERE
   ...> matricula = '100210046';
sqlite> SELECT matricula, name FROM Estudiante;
1222100487|Kevin
1222100503|Alan
1222100486|Angel
1222100895|Monserrat
1222100815|Sharon
100210046|Abraham Josue
sqlite>
```

6. **Eliminar al menos dos registros**

```
sqlite> DELETE FROM Estudiante
   ...> WHERE
   ...> name = 'Sharon';
sqlite> DELETE FROM Estudiante
   ...> WHERE
   ...> name = 'Alan';
sqlite> SELECT matricula, name FROM Estudiante;
1222100487|Kevin
1222100486|Angel
1222100895|Monserrat
100210046|Abraham Josue
sqlite>
```

7. **Implementar método GET con Python**

Sharon Michelle Olvera Ibarra

```python
import json
import sqlite3
from flask import Flask, request, jsonify


app = Flask(__name__)

def get_db_connection():
    conn = sqlite3.connect('crud.db')
    conn.row_factory = sqlite3.Row
    return conn

@app.route('/', methods=['GET'])
def query_records():
    name = request.args.get('name')
    registros = []
    conn = get_db_connection()
    cursor = conn.cursor()
    cursor.execute('SELECT * FROM Estudiante')
    data = cursor.fetchall()
    for reg in data:
        registros.append(dict(reg))
    conn.close()
    return jsonify(json.dumps(registros))

app.run(debug=True)
```