

Python

SQLite



Rogelio Ferreira Escutia

Profesor / Investigador
Tecnológico Nacional de México
Campus Morelia



Surgimiento y características de SQLite

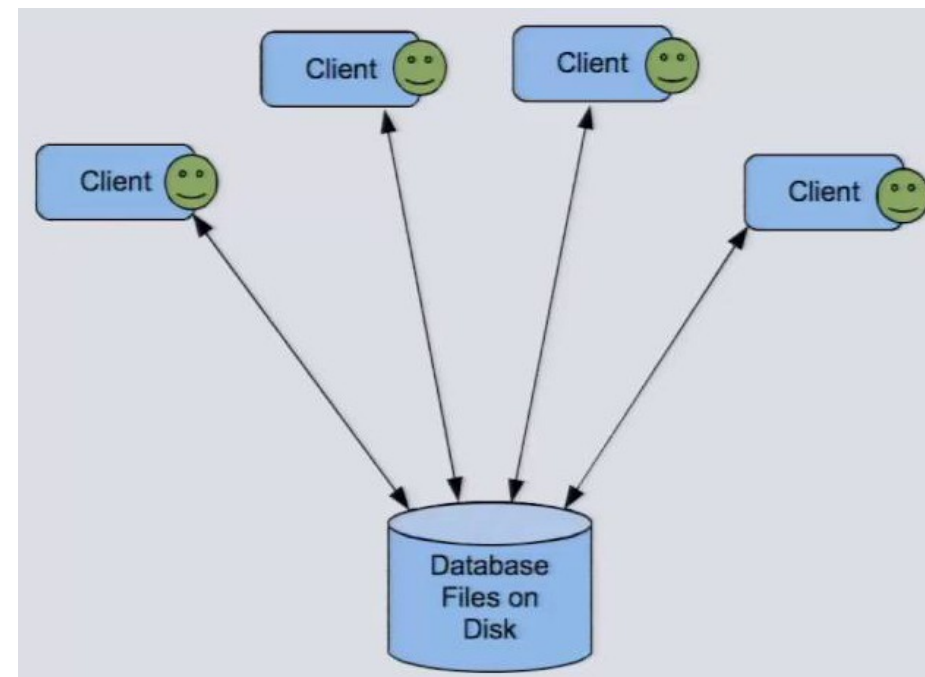
Surgimiento de SQLite

- **SQLite es un sistema de gestión de bases de datos relacional compatible con ACID, contenida en una relativamente pequeña biblioteca escrita en C.**
- **SQLite es un proyecto de dominio público creado por D. Richard Hipp.**



Características de SQLite

- A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica, en lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo.
- En su versión 3, SQLite permite bases de datos de hasta 2 Terabytes de tamaño, y también permite la inclusión de campos tipo BLOB



SQLite con Python

SQLite - Biblioteca

- Importar la biblioteca (usando la versión 3):

```
import sqlite3
```

SQLite – Base de datos

- **Crear una conexión con una base de datos “informacion.db”.**
- **Si la base de datos existe, la abre.**
- **Si la base de datos no existe, la crea y guarda el archivo en la misma ruta donde se ejecuta el código Python.**

```
conexion = sqlite3.connect('informacion.db')
```

SQLite – Objeto Cursor

- **Crear un objeto cursor para referenciarlo al hacer operaciones sobre la base de datos:**

```
c = conexion.cursor()
```


SQLite – Crear una tabla

- Creamos una tabla “agenda” con los siguientes campos y tipos de datos:
 - id: entero (llave índice)
 - nombre: texto
 - telefono: texto
 - edad: entero
 - peso: real

```
c.execute("CREATE TABLE agenda (id integer PRIMARY KEY,  
nombre text, telefono text, edad int, peso real)")
```

SQLite – Insertar un registro

- Insertamos 2 registros en la base de datos:

```
c.execute("INSERT INTO agenda VALUES(1, 'Juan', '443-312-15-70', 18, 74.5)")
```

```
c.execute("INSERT INTO agenda VALUES(2, 'Ana', '443-845-93-34', 25, 63.8)")
```

SQLite – Guardar cambios

- Usamos el método `commit ()` para guardar todos los cambios que hemos hecho hasta el momento, y asegurar que queden guardados en la base de datos física:

```
conexion.commit()
```

SQLite – Leyendo los registros

- Hacemos un “SELECT” para leer y extraer todos los registros de la base de datos:

```
c.execute( 'SELECT * FROM agenda' )
```

SQLite – Impresión de Registros

- Se usa el método `fetchall()` del objeto `cursor` para almacenar los valores en una variable y luego se hace un ciclo para recorrer cada valor e imprimirlo:

```
registros = c.fetchall()
for ciclo in registros:
    print(ciclo)
```

SQLite – Código Completo

```
1 # Importar SQLite versión 3
2 import sqlite3
3 print('\n\n*** SQLite ***')
4 # Crear una conexión con una base de datos
5 print('\nAbriendo la base de datos "informacion.db"')
6 conexion = sqlite3.connect('informacion.db')
7 # Crear un objeto cursor para referenciarlo al hacer operaciones sobre la base de datos
8 c = conexion.cursor()
9 # Creando una tabla
10 print('Creando la tabla: "agenda.db"')
11 c.execute("CREATE TABLE agenda (id integer PRIMARY KEY, nombre text, telefono text, edad int, peso real)")
12 # Insertando 2 registros en la tabla
13 print('Insertando un registro en la tabla: "agenda.db"')
14 c.execute("INSERT INTO agenda VALUES(1, 'Juan', '443-312-15-70', 18, 74.5)")
15 c.execute("INSERT INTO agenda VALUES(2, 'Ana', '443-845-93-34', 25, 63.8)")
16 # Guardando las operaciones realizadas
17 conexion.commit()
18 # Leyendo todos los registros
19 print('Leyendo todos los registros de la tabla: "agenda.db"')
20 c.execute('SELECT * FROM agenda')
21 # Ciclo para imprimir todos los registros
22 print('Imprimiendo los registros de la tabla: "agenda.db"')
23 registros = c.fetchall()
24 for ciclo in registros:
25     print(ciclo)
```

SQLite – Salida en Consola

- La salida en consola de la ejecución de todas las líneas anteriores sería la siguiente:

```
*** SQLite ***
```

```
Abriendo la base de datos "informacion.db"
```

```
Creando la tabla: "agenda.db"
```

```
Insertando un registro en la tabla: "agenda.db"
```

```
Leyendo todos los registros de la tabla: "agenda.db"
```

```
Imprimiendo los registros de la tabla: "agenda.db"
```

```
(1, 'Juan', '443-312-15-70', 18, 74.5)
```

```
(2, 'Ana', '443-845-93-34', 25, 63.8)
```

SQLite – Modificación de un registro

- Si queremos modificar la edad de “Juan” (de 18 a 45 años) requerimos ejecutar la sentencia UPDATE sobre la tabla “agenda” de la siguiente manera:

```
c.execute('UPDATE agenda SET edad = 45 where nombre = "Juan"')  
conexion.commit()
```


SQLite – Borrar un registro

- Si queremos eliminar el registro “Juan” de nuestra tabla, ejecutamos la sentencia “DELETE” de la siguiente manera:

```
c.execute('DELETE FROM agenda WHERE nombre = "Juan"')  
conexion.commit()
```



Rogelio Ferreira Escutia

Profesor / Investigador
Tecnológico Nacional de México
Campus Morelia



rogelio.fe@morelia.tecnm.mx



rogeplus@gmail.com



xumarhu.net



[@rogeplus](https://twitter.com/rogeplus)



[https://www.youtube.com/
channel/UC0on88n3LwTKxJb8T09sGjg](https://www.youtube.com/channel/UC0on88n3LwTKxJb8T09sGjg)



[rogelioferreiraescutia](https://www.linkedin.com/in/rogelioferreiraescutia)

