

# SQLite

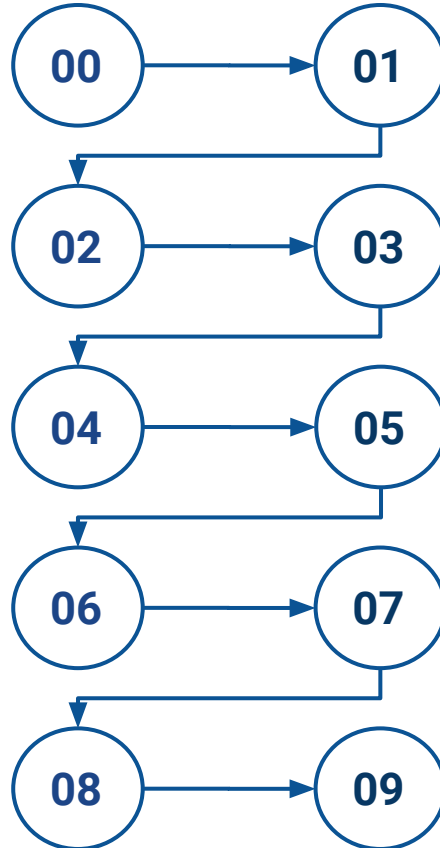
¿Qué es un BD?

SQL

Conectar

INSERT

UPDATE



¿Qué es SQLite?

Tipo de datos

Crear tabla

SELECT

DELETE

# ¿Qué es un BD?

Una base de datos permite guardar información de forma persistente. Todo se agrupa en tablas, y cada tabla contiene sus propias columnas.

E6		<i>fx</i>
	A	B
1	<b>nombre</b>	<b>apellido</b>
2	Marty	Macfly
3	Beef	Tannen
4	Emmet	Brown

En una base de datos la hoja sería la tabla, y cada columna sería un campo de esa tabla.

# ¿Qué es SQLite?

SQLite es un gestor de bases de datos relacionales.

Es Open Source y se ha instalado por defecto con Python, es decir forma parte de la biblioteca estándar, no tenemos que instalar ningún módulo con pip.

# ¿Qué es SQLite?

Este gestor de base de datos tiene por objetivo ser parte de la misma aplicación con la que colabora, es decir no cumple los conceptos de cliente y servidor.

# ¿Cuándo usar SQLite?

Si una aplicación necesita almacenar información local con cierta estructura el empleo de SQLite es una excelente opción.



Firefox usa **SQLite** para almacenar los favoritos, el historial, las cookies etc.

Las BD relacionales utilizan un lenguaje llamado SQL (Structured Query Language) mediante el cual se puede gestionar la base de datos.

Formado por:

- Lenguaje de definición de datos (DDL)
- Lenguaje de manipulación de datos (DML)

Se encarga de la modificación de la estructura de los objetos de la base de datos.

- CREATE (Crear)
- ALTER (Alterar)
- DROP (Eliminar)
- TRUNCATE (Truncar)

Permite llevar a cabo las tareas de consulta o manipulación de los datos

- SELECT (Seleccionar)
- INSERT (Insertar)
- UPDATE (Actualizar)
- DELETE (Borrar)



# Tipo de datos SQLite

<b>NULL</b>	Es un valor NULO.
<b>INTEGER</b>	Entero con signo.
<b>REAL</b>	Valor de coma flotante.
<b>TEXT</b>	Cadena de texto
<b>BLOB</b>	Son bytes almacenados exactamente como se ingresaron.

# Conectar con SQLite

Para poder trabajar con bases de datos de tipo SQLite debemos primero importar el módulo 'sqlite3' , luego conectar con la base de datos, después operar y por último no olvidar de cerrar la conexión.

# Conectar con SQLite

```
import sqlite3
conexion=sqlite3.connect("bd_btf.db")
...
conexion.close()
```

ó

```
import sqlite3
with sqlite3.connect("bd_btf.db") as conexion:
    ...
```

# Crear tabla en SQLite

Para poder crear una tabla en la base de datos:

```
import sqlite3
with sqlite3.connect("CLASE_sqlite/bd_btf.db") as conexion:
    try:
        sentencia = ''' create table personajes
                        (
                            id integer primary key autoincrement,
                            nombre text,
                            apellido text,
                            anio real
                        )
                    '''
        conexion.execute(sentencia)
        print("Se creo la tabla personajes")
    except sqlite3.OperationalError:
        print("La tabla personajes ya existe")
```

# Insert con SQLite

Para poder insertar filas en una tabla:

```
import sqlite3

with sqlite3.connect("CLASE_sqlite/bd_btf.db") as conexion:
    try:
        conexion.execute("insert into personajes(nombre,apellido,anio)
        values (?,?,?)", ("Marty", "MacFly", "1968"))
        conexion.execute("insert into personajes(nombre,apellido,anio)
        values (?,?,?)", ("Emmet", "Brown", "1914"))
        conexion.commit()# Actualiza los datos realmente en la tabla
    except:
        print("Error")
```

# Select con SQLite

Para poder recuperar todas las filas de una tabla:

```
with sqlite3.connect("CLASE_sqlite/bd_btf.db") as conexion:  
    cursor=conexion.execute("SELECT * FROM personajes")  
    for fila in cursor:  
        print(fila)
```

# Select con SQLite

Para poder recuperar una fila de la tabla donde el id es 1:

```
id = "1"

with sqlite3.connect("CLASE_sqlite/bd_btfd.db") as conexion:

    sentencia = "SELECT * FROM personajes WHERE id=?"

    cursor=conexion.execute(sentencia, (id,))

    for fila in cursor:

        print(fila)
```

# Update con SQLite

Para poder actualizar una fila de la tabla donde el id es 1:

```
id = "1"

with sqlite3.connect("CLASE_sqlite/bd_btfd.db") as conexion:
    sentencia = "UPDATE personajes SET nombre = 'XX' WHERE id=?"
    cursor=conexion.execute(sentencia, (id,))
    filas=cursor.fetchall()
    for fila in filas:
        print(fila)
```



# Delete con SQLite

Para poder eliminar la o las filas de la tabla donde el id es 1:

```
id = "1"

with sqlite3.connect("CLASE_sqlite/bd_btbf.db") as conexion:

    sentencia = "DELETE FROM personajes WHERE id=?"

    cursor=conexion.execute(sentencia, (id,))
```

YAPA

The background features a series of overlapping chevron shapes pointing to the right. The colors transition from a dark charcoal blue on the left to a light sky blue on the right. A prominent teal-colored chevron is layered over the darker shades in the center.

# DB Browser for SQLite

DB Browser for SQLite es una herramienta la cual permite crear, diseñar y editar archivos de bases de datos compatibles con SQLite.

<https://sqlitebrowser.org/dl/>

# JOIN con SQLite

Un join es una operación que relaciona dos o más tablas para obtener un resultado que incluya datos (campos y registros) de ambas; las tablas participantes se combinan según los campos comunes.

# INNER JOIN

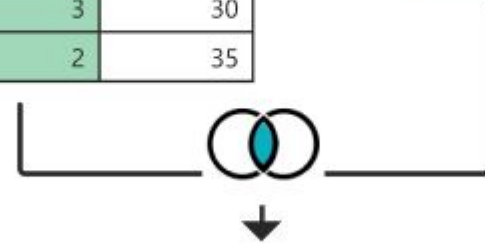
**SELECT \***  
**FROM A**  
**INNER JOIN B**  
**ON A.CountryID = B.ID**

**A**

Date	CountryID	Units
1/1/2020	1	40
1/2/2020	1	25
1/3/2020	3	30
1/4/2020	2	35

**B**

ID	Country
3	Panama
4	Spain



Date	CountryID	Units	Country
1/3/2020	3	30	Panama