

como trabajar bases de datos sqlite con python

Para trabajar con bases de datos SQLite en Python, se utiliza el módulo integrado `sqlite3`, que no requiere instalación adicional y permite crear, conectar, manipular y consultar bases de datos SQLite de forma sencilla.

1. Conexión a la base de datos

Se crea o abre una base de datos SQLite con:

```
import sqlite3

conexion = sqlite3.connect("mi_base_de_datos.db")
```

Si el archivo no existe, SQLite lo crea automáticamente^{[1] [2] [3]}.

2. Crear tablas

Para crear tablas se ejecuta un comando SQL con `execute`. Para evitar errores si la tabla ya existe, se usa `CREATE TABLE IF NOT EXISTS`:

```
conexion.execute("""
CREATE TABLE IF NOT EXISTS articulos (
    codigo INTEGER PRIMARY KEY AUTOINCREMENT,
    descripcion TEXT,
    precio REAL
)
""")
conexion.commit()
```

Se recomienda usar `commit()` para guardar los cambios^{[1] [3]}.

3. Insertar datos

Para insertar registros se usa `execute` con parámetros para evitar inyección SQL:

```
conexion.execute("""
INSERT INTO articulos (descripcion, precio) VALUES (?, ?)
""", ("Producto A", 10.5))
conexion.commit()
```

4. Consultar datos

Para leer datos se crea un cursor y se ejecuta una consulta SELECT:

```
cursor = conexion.cursor()
cursor.execute("SELECT codigo, descripcion, precio FROM articulos")

for fila in cursor.fetchall():
    print(fila)
```

5. Actualizar y eliminar datos

Ejemplo para actualizar:

```
conexion.execute("""
UPDATE articulos SET precio = ? WHERE codigo = ?
""", (12.0, 1))
conexion.commit()
```

Ejemplo para eliminar:

```
conexion.execute("""
DELETE FROM articulos WHERE codigo = ?
""", (1,))
conexion.commit()
```

6. Manejo de excepciones

Para evitar errores al crear tablas que ya existen, se puede usar `try/except` o la cláusula `IF NOT EXISTS` en SQL ^[1].

7. Cerrar la conexión

Al finalizar, siempre cerrar la conexión:

```
conexion.close()
```

8. Recursos y tutoriales recomendados

- Tutorial escrito con ejemplos claros: Tutoriales Programación Ya ^[1]
- Tutorial en video en español explicando desde crear tablas hasta consultas y actualizaciones: YouTube Dimas ^[4]
- Documentación oficial de Python para sqlite3 con ejemplos avanzados: Python Docs ^{[2] [3]}

- Curso completo con proyectos reales en Udemy para profundizar: Udemy^[5]

Esta es la forma estándar y recomendada de trabajar con SQLite en Python, usando el módulo `sqlite3` para crear bases de datos, tablas, insertar, consultar, actualizar y eliminar datos de manera sencilla y eficiente.



1. <https://www.tutorialesprogramacionya.com/pythonya/detalleconcepto.php?punto=87&codigo=88&inicio=75>
2. <https://docs.python.org/es/3.8/library/sqlite3.html>
3. <https://docs.python.org/es/3.13/library/sqlite3.html>
4. <https://www.youtube.com/watch?v=uB0928SOTEQ>
5. <https://www.udemy.com/course/python-y-sqlite-desde-cero-a-hero/>