

Consultas SQL intermedias

Consulta con WHERE

Una vez contamos con algún campo que nos sirva de identificador único, podemos realizar consultas específicas utilizando la cláusula `WHERE`:

Código

```
import sqlite3

conexion = sqlite3.connect('usuarios_autoincremental.db')
cursor = conexion.cursor()

# Recuperamos un registro de la tabla de usuarios
cursor.execute("SELECT * FROM usuarios WHERE id=1")

usuario = cursor.fetchone()
print(usuario)

conexion.close()
```

Resultado

```
(1, '11111111A', 'Hector', 27, 'hector@ejemplo.com')
```

También podemos buscar sólo algunos campos específicos utilizando el DNI:

Código

```
import sqlite3

conexion = sqlite3.connect('usuarios_autoincremental.db')
cursor = conexion.cursor()

# Recuperamos un registro de la tabla de usuarios
cursor.execute("SELECT nombre, edad, email FROM usuarios " \
               "WHERE dni='22222222B'")

usuario = cursor.fetchone()
print(usuario)

conexion.close()
```

Resultado

```
('Mario', 51, 'mario@ejemplo.com')
```

Modificación con UPDATE

De forma similar al SELECT podemos utilizar la cláusula:

```
UPDATE tabla
SET columna1 = valor1, columna2 = valor2..., columnaN = valorN
WHERE [condicion]
```

Código

```
import sqlite3

conexion = sqlite3.connect('usuarios_autoincremental.db')
cursor = conexion.cursor()

# Actualizamos un registro
cursor.execute("UPDATE usuarios SET nombre='Hector Costa' " \
              "WHERE dni='11111111A'")

# Ahora lo consultamos de nuevo
cursor.execute("SELECT * FROM usuarios WHERE dni='11111111A'")
usuario = cursor.fetchone()
print(usuario)

conexion.commit()
conexion.close()
```

Resultado

```
(1, '11111111A', 'Hector Costa', 27, 'hector@ejemplo.com')
```

Importantísimo: No olvidar la cláusula WHERE o podéis acabar actualizando todos los registros

Borrado con DELETE

Finalmente, para borrar un registro a partir de su id o campo único:

```
DELETE FROM tabla WHERE [condicion]
```

Código

```
import sqlite3

conexion = sqlite3.connect('usuarios_autoincremental.db')
cursor = conexion.cursor()
```

```

# Creamos un registro de prueba
cursor.execute("INSERT INTO usuarios VALUES " \
              "(null, '55555555E', 'Fernando', 31, 'fernando@ejemplo.com')")

# Consultamos los usuarios
for usuario in cursor.execute("SELECT * FROM usuarios"):
    print(usuario)

# Ahora lo borramos
cursor.execute("DELETE FROM usuarios WHERE dni='55555555E'")

print() # Espacio en blanco

# Consultamos de nuevo los usuarios
for usuario in cursor.execute("SELECT * FROM usuarios"):
    print(usuario)

conexion.commit()
conexion.close()

```

Resultado

```

(1, '11111111A', 'Hector Costa', 27, 'hector@ejemplo.com')
(2, '22222222B', 'Mario', 51, 'mario@ejemplo.com')
(3, '33333333C', 'Mercedes', 38, 'mercedes@ejemplo.com')
(4, '44444444D', 'Juan', 19, 'juan@ejemplo.com')
(5, '55555555E', 'Fernando', 31, 'fernando@ejemplo.com')

(1, '11111111A', 'Hector Costa', 27, 'hector@ejemplo.com')
(2, '22222222B', 'Mario', 51, 'mario@ejemplo.com')
(3, '33333333C', 'Mercedes', 38, 'mercedes@ejemplo.com')
(4, '44444444D', 'Juan', 19, 'juan@ejemplo.com')

```

No te olvides el WHERE

En SQL es posible realizar actualizaciones y borrados en masa, pero las dos últimas son un poco peligrosas. Sin embargo realizarlas es tan sencillo como olvidarnos la cláusula WHERE en el UPDATE o el DELETE.

Código

```

import sqlite3

conexion = sqlite3.connect('usuarios_autoincremental.db')
cursor = conexion.cursor()

# Borramos sin el WHERE
cursor.execute("DELETE FROM usuarios")

# Consultamos de nuevo los usuarios

```

```
usuarios = cursor.execute("SELECT * FROM usuarios").fetchall()
print(usuarios)

conexion.commit()
conexion.close()
```

Resultado

```
[]
```

Más por aprender

SQL es un lenguaje muy extenso con muchísimas posibilidades.

Como esta unidad no deja de ser una introducción, te animo a seguir aprendiendo por tu cuenta conceptos tan importantes como:

- **Consultas avanzadas:** or, and, like, join
- **Funciones simples:** count, group by, distinct
- **Funciones avanzadas:** sum, avg, min, max
- **Manejo de fechas:** date, year, month, day
- **Relaciones y claves foráneas**