

SQLITE

Concepto y Funciones



SQLITE

es un sencillo sistema de gestión de bases de datos de tipo relacional

SQLite no funciona como otros gestores que necesitan de un servidor de base de datos ejecutándose en un proceso separado al que se le hacen peticiones.

La librería SQLite es tan pequeña que se enlaza con los programas y éstos hacen llamadas directamente a los procedimientos y funciones disponibles para interactuar con las bases de datos; siendo este modo de trabajo más eficiente que el basado en comunicar peticiones a procesos externos

TABLAS

Una tabla es el objeto principal de una base de datos por ser el lugar donde se almacena la información a gestionar.

Una base de datos suele contener un conjunto de tablas y por sus datos, algunas de ellas, pueden estar relacionadas entre sí.

Las tablas se utilizan para organizar la información y se componen de filas y columnas.

Un registro es cada una de las filas en que se divide la tabla y un campo es cada una de las columnas de la tabla que, normalmente, contienen datos de diferentes tipos.

Tipos de Datos

- NULL: Se refiere a los valores nulos o NULL: `typeof(NULL) -> null`
- INTEGER: Números enteros: `(100) -> integer`
- REAL: Números reales: `(100.10) -> real`
- TEXT: Texto: `('100') -> text`
- BLOB: Binario: `(x'100') -> blob`

Fechas

Las fechas en SQLITE3 se pueden almacenar en un campo como textos, números reales o números enteros.

Crear Tablas

CREATE se utiliza para crear las tablas de una base de datos. En primer lugar, a cada tabla se le asigna un nombre. Después, se define la lista de campos indicando para cada uno de ellos el tipo de dato que va a contener (TEXT, INTEGER, REAL, TEXT y BLOB).

Además, cada campo puede incluir en su definición algunas restricciones básicas como NOT NULL para que no contengan valores nulos o UNIQUE para que no se acepte un mismo dato en diferentes filas o registros.

También se puede cargar un valor predeterminada con DEFAULT

Clave Primaria

Una clave primaria es un campo o combinación de campos que identifica de forma única a cada fila de una tabla.

```
campo1 INTEGER PRIMARY KEY,
```

```
CREATE TABLE usuarios ( id INTEGER PRIMARY KEY  
AUTOINCREMENT,  
cuenta_usuario TEXT UNIQUE, nombre TEXT NOT  
NULL, id_dpto INTEGER,  
ecorreo TEXT NOT NULL, FOREIGN KEY (id_dpto)  
REFERENCES dptos(id_dpto)
```

```
CREATE TABLE dptos (id_dpto INTEGER PRIMARY  
KEY AUTOINCREMENT,  
denom TEXT NOT NULL UNIQUE)
```

Agregar registro

La sentencia INSERT permite insertar registros en una tabla.

```
INSERT INTO tabla1 (campo1, campo2) VALUES  
    (dato1, dato2)
```

```
INSERT INTO  
asignatura(dni,nombre,apellido) VALUES  
    (?, ?, ?) ", (DNI, nombre, apellido)
```

Consulta

La sentencia **SELECT** permite consultar los datos almacenados en una base de datos.

```
SELECT campo1, campo2 FROM tabla1;
```

```
SELECT * FROM tabla1;
```

```
SELECT campo1, campo2 FROM tabla1 ORDER BY  
campo1 ASC, campo2 DESC;
```

Actualizar datos

La sentencia UPDATE se utiliza para actualizar datos en los registros o filas de una tabla.

Se utiliza siempre que se quiera agregar o modificar un dato en un registro o varios registros

Actualizar un campo en registros que cumplan una condición:

```
UPDATE tabla1 SET campo1 = 'dato1' WHERE campo2 = 3;
```

Actualizar varios campos en registros que cumplan una condición:

```
UPDATE tabla1 SET campo1 = 'dato1', campo2 = 'dato2'  
WHERE campo3 = 3;
```

Eliminar registros

La sentencia DELETE sirve para borrar registros de las tablas de una base de datos.

Borrar los registros que cumplan una condición:

```
DELETE FROM tabla1 WHERE campo1 > 10;
```

Borrar todos los registros:

```
DELETE FROM tabla1;
```



Modificar tabla

`ALTER TABLE` en SQLite permite al usuario renombrar una tabla o añadir una nueva columna a una tabla existente. No es posible renombrar una columna, eliminarla o añadir o eliminar limitaciones de una tabla.

Renombrar una tabla:

```
ALTER TABLE tabla1 RENAME TO tabla1_anterior;
```

Añadir campo o columna:

```
ALTER TABLE tabla1 ADD COLUMN campo4 INTEGER;
```

La nueva columna no puede ser UNIQUE o PRIMARY KEY. Si la nueva columna tiene una restricción NOT NULL es necesario especificar un valor predeterminado que no sea el valor NULL:

```
CREATE TABLE tabla1 (... campo4 INTEGER DEFAULT 1, ...);
```


Borrar contenido de un tabla

Para borrar una tabla, utiliza el comando DROP
TABLE

```
DROP TABLE tabla1;
```