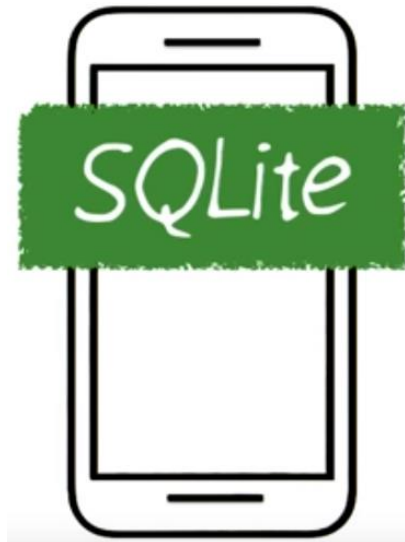


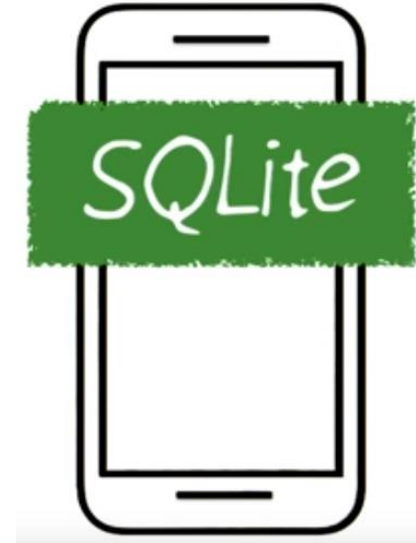
SQLite

Por: Javier Vasquez Cruz



SQLite

- Versión ligera de una base de datos típica.
- Existen otras opciones pero tienen características mas complicadas (MySQL).
- No requiere servidor.
- Los datos se guardan en un archivo de texto localmente en el dispositivo.
- La biblioteca de SQLite es gratuita y de código abierto.
- Esta integrado con Android.
- No requiere descargar ni instalar algo extra.



SQLite

- Puedes usar la biblioteca SQLite directamente en la computadora.



Instalación

Instalación:

- Veamos los pasos para instalar SQLite en su computadora.

Windows:

Para instalar SQLite en una máquina con Windows, necesitará

1. Descargue los archivos Zip correctos.
 2. Crea una carpeta llamada C:\sqlite para extraerlos.
 3. Agregue C:\sqlite a su variable de entorno PATH.
 4. Ejecute el comando sqlite3 en su símbolo del sistema.
- Vamos a profundizar un poco más en esos pasos.
 - Para descargar los archivos correctos, querrá ir [aquí](#) y descargar sqlite-tools-win32-*. Zip
 - Una vez que haya descargado este zip, abra el explorador de Windows y navegue hasta su unidad C: \. Una vez allí, crea una carpeta llamada sqlite. Copie el archivo .zip que acaba de descargar en esta carpeta, luego extráigalos.
 - Una vez que haya hecho eso, deberá agregar C:\sqlite a la variable de entorno PATH de su sistema. Si nunca has hecho esto antes, sigue [estos pasos](#) para hacerlo.
 - Finalmente, deberá abrir su programa de solicitud de comando y escribir en sqlite3, luego presionar enter.
 - Aquí hay un [video que](#) describe los pasos anteriores para la instalación de Windows.

```
C:\Windows\System32\cmd.exe - sqlite3 test.db
```

```
Microsoft Windows [Versión 10.0.16299.371]  
(c) 2017 Microsoft Corporation. Todos los derechos reservados.  
  
C:\mydir>sqlite3 test.db  
SQLite version 3.23.1 2018-04-10 17:39:29  
Enter ".help" for usage hints.  
sqlite>
```

SQLite

- Crear la Base de datos:

```
C:\sqlite>sqlite3
SQLite version 3.23.1 2018-04-10 17:39:29
Enter ".help" for usage hints.
Connected to a transient in-memory database.
Use ".open FILENAME" to reopen on a persistent database.
sqlite> .open pets.db
sqlite> verificar que se creo el archivo....
```

- Para salir de SQLite: .quit
- Ayuda: .help
- Para abrir la base de datos: sqlite3 nombre_base_dedatos.db

Creando la base de datos

_id	nombre	raza	genero	peso
Entero	Texto	Texto	Entero	Entero

Cada valor almacenado en una base de datos SQLite (o manipulado por el motor de base de datos) tiene una de las siguientes clases de almacenamiento:

- **NULL**. The value is a NULL value.
- **INTEGER**. The value is a signed integer, stored in 1, 2, 3, 4, 6, or 8 bytes depending on the magnitude of the value.
- **REAL**. The value is a floating point value, stored as an 8-byte IEEE floating point number.
- **TEXT**. The value is a text string, stored using the database encoding (UTF-8, UTF-16BE or UTF-16LE).
- **BLOB**. The value is a blob of data, stored exactly as it was input.

Peso: No considera decimales.

Genero: 3 opciones (macho, hembra y desconocido).

Nuevas razas son agregadas siempre.

Tipos de datos SQLite
Click

Crear Tablas

- Crear Tablas:

```
CREATE TABLE <table_name> (<column_name_1> <data_type_1>,  
                             <column_name_2> <data_type_2>,
```

cmd C:\WINDOWS\system32\cmd.exe - sqlite3

```
sqlite> CREATE TABLE headphones(_id INTEGER,name TEXT,price INTEGER)  
...> ;  
sqlite> CREATE TABLE headphones(_id INTEGER,name TEXT,price INTEGER);  
Error: table headphones already exists  
sqlite> .tables  
headphones  
sqlite> .schema headphones  
CREATE TABLE headphones(_id INTEGER,name TEXT,price INTEGER);  
sqlite> PRAGMA TABLE_INFO(headphones);  
0|_id|INTEGER|0||0  
1|name|TEXT|0||0  
2|price|INTEGER|0||0  
sqlite>
```

- Para ver Tablas: .tables
- Para ver esquema de tabla: .schema tabla
- Información de un tabla: PRAGMA TABLE_INFO(moto);
- Eliminar Tablas: DROP TABLE tabla;

Ejemplo: Crear Tabla

Explicación:

- `CREATE TABLE nombre (_id INTEGER, nombreTEXT, raza TEXT, genero INTEGER, peso INTEGER);`
- Entre paréntesis está la lista que define cada columna en la tabla y su tipo de datos correspondiente. Recuerde finalizar el comando con un punto y coma.

CRUD

CREATE

READ

UPDATE

DELETE

Lectura de datos: SELECT

SELECT <columns> **FROM** <table_name>

- Ejemplo:

SELECT <columns> <table_name>
 * **FROM** pets;

Ordenando una selección:

- Ejemplo:

```
SELECT name, weight FROM pets WHERE gender == 1 ORDER BY  
weight DESC;
```

Crear o Insertar datos: INSERT

```
INSERT INTO <table_name> (<columns_name_1>, <columns_name_2>,...)  
                VALUES (<value_1>, <value_2>,...);
```

- Ejemplo:

```
INSERT INTO  pets  (_id, name, breed, gender, weight)  
                VALUES (1, "Tommy", "Pomeranian", 1, 4);
```

RESTRICCIONES

PRIMARY KEY

AUTO INCREMENT

NOT NULL

DEFAULT <value>

Ejemplo:

```
CREATE TABLE mascota (_id INTEGER PRIMARY KEY AUTOINCREMENT,  
nombre TEXT NOT NULL,  
raza TEXT,  
genero INTEGER NOT NULL,  
peso INTEGER NOT NULL DEFAULT 0);
```

Inserta algunas filas a la tabla....

ACTUALIZAR DATOS

- UPDATE tabla SET campo=valor WHERE _id==valor;

Ejemplo:

```
UPDATE mascota SET nombre="pecas" WHERE _id==2;
```

ELIMINAR DATOS

- Borrar tabla: `DELETE FROM tabla;`
- Borrar un dato: `DELETE FROM tabla WHERE id=<valor>;`

Documentación SQLite

Caso de estudio

Asilo para mascotas