

IntelliJ - SQLite

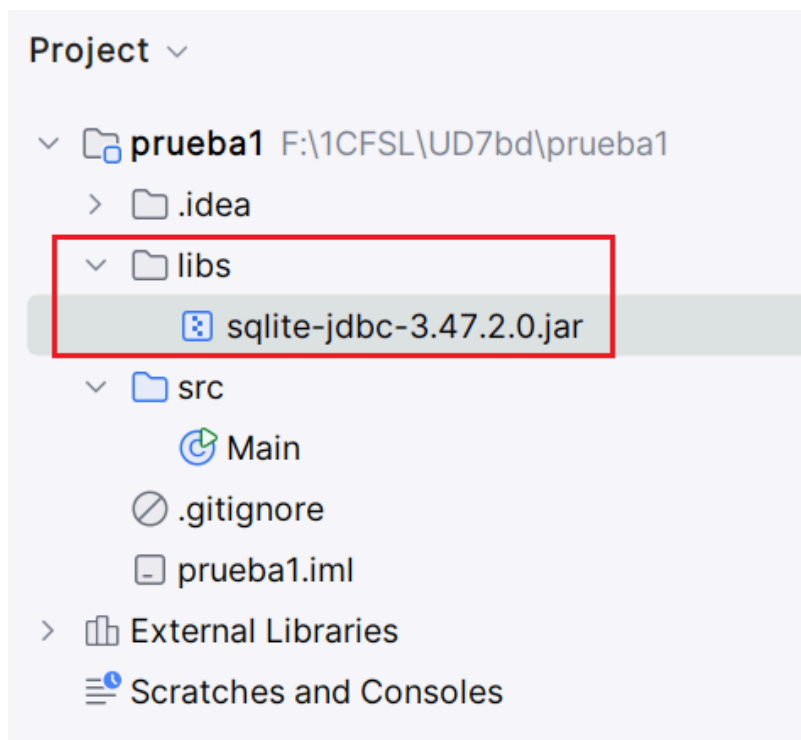
ÍNDICE

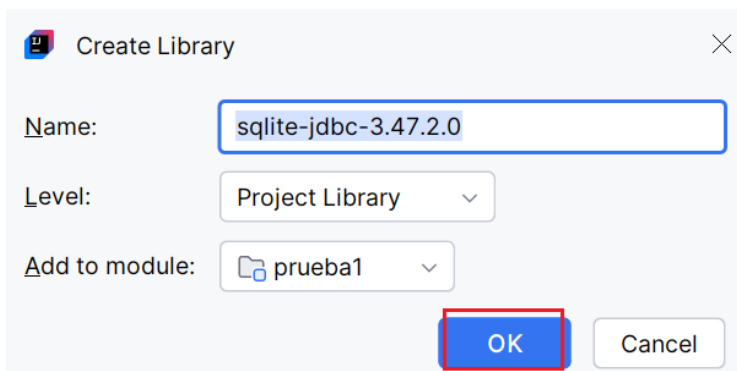
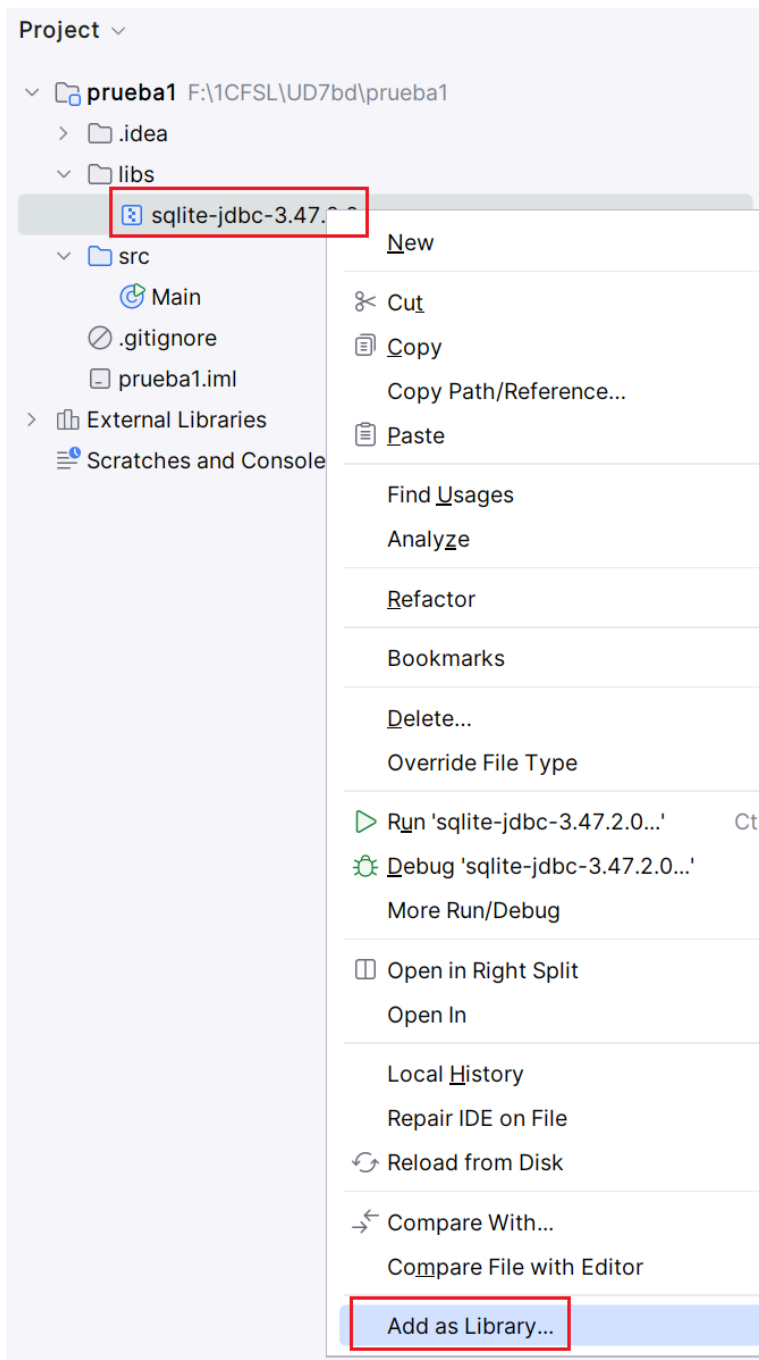
1. sqlite-jdbc	1
2. DB Browser for SQLite	3
3. Conexión a la BD	0

1. sqlite-jdbc

Para poder acceder a una base de datos sqlite a través de Java hay que añadir la dependencia de SQLite a nuestro proyecto. Para ello hay que realizar los siguientes pasos:

- Descarga el archivo .jar desde: <https://github.com/xerial/sqlite-jdbc/releases>
- Coloca el archivo .jar en tu proyecto (por ejemplo, en una **carpeta libs**).
- Agrega el .jar a las dependencias del proyecto:
 - Haz clic derecho en el archivo .jar en IntelliJ.
 - Selecciona Add as Library.





2. DB Browser for SQLite

Siempre que desarrollamos aplicaciones que acceden a una base de datos es recomendable disponer de un visor / editor adecuado al sistema gestor de bases de datos con el que estamos trabajando para poder realizar comprobaciones. Con la herramienta **DB Browser for SQLite** podemos crear, modificar y consultar la estructura de una base de datos.

Se puede descargar desde el siguiente enlace <https://sqlitebrowser.org/dl/>

Para Linux puedes descargar la Appliance que no necesita instalación desde: <https://sqlitebrowser.org/dl/#linux>

(<https://download.sqlitebrowser.org/DB.Browser.for.SQLite-v3.13.1-x86.64-v2.Appliance>)

Una vez descargada, para poder ejecutarla con doble clic, sigue estos pasos:

- Haz clic con el botón derecho y entra en *Propiedades*.
- Entra en la pestaña *Permisos*.
- Marca la casilla *Permite la ejecución del fichero como programa*.

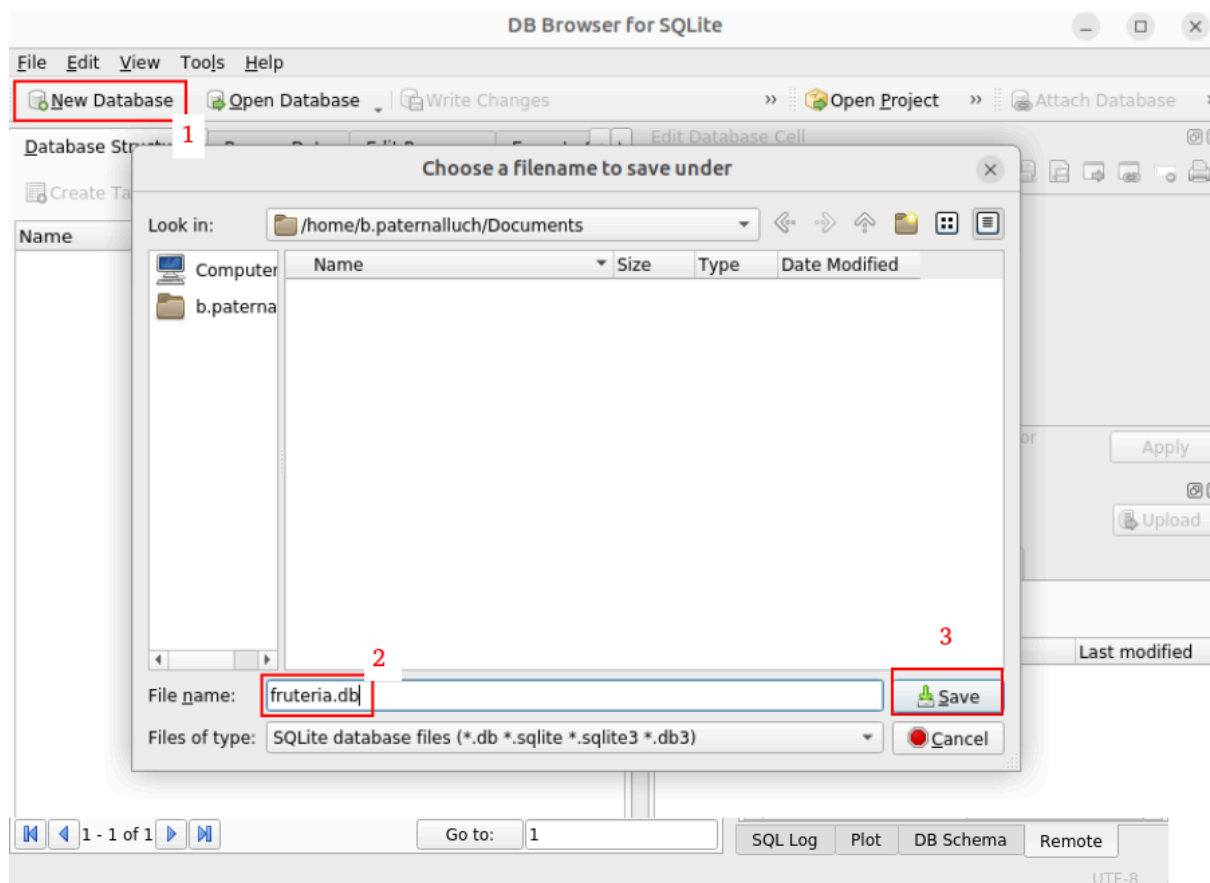
Propietats de: DB.Browser.for.SQLite-v3.13.1-x86.64-v2.Appliance

Bàsic	Emblemes	Permisos	Obre amb	Notes	Llista de control d'accés	Atributs d
Propietari:	b.paternalluch - PATERNA LLUCH					
Accés:	Lectura i escriptura					
Grup:	Domain Users					
Accés:	Cap					
Altres						
Accés:	Cap					
Execució:	<input checked="" type="checkbox"/> Permet l'execució del fitxer com a programa					
Context SELinux:	desconegut					
Últim canvi:	dijous, 6 de febrer de 2025, 17:25:18					

A continuación se describen los pasos a seguir para crear una base de datos llamada **fruteria.db** con una tabla llamada **productos** con la siguiente estructura:

Campo	Tipo de Dato	Descripción
id	INTEGER (PK, AUTOINCREMENT)	Identificador único.
nombre	CHAR(15)	Nombre.
precio	REAL	Precio (con decimales).
procedencia	CHAR(3)	Código de país de procedencia.

Crear la base de datos:



Crear la tabla y los campos

Edit table definition

Table: **productos** ¹

Advanced

Fields | Index Constraints | Foreign Keys | Check Constraints

Add ² Remove Move to top Move up Move down Move to bottom

Name	Type	NN	PK	AI	U	Default	Check	Collation	F
id	INTEGER	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
nombre	CHAR(15)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
precio	REAL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
procedencia	CHAR(3)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				

³

```
1 CREATE TABLE "productos" (  
2     "id" INTEGER,  
3     "nombre" CHAR(15),  
4     "precio" REAL,  
5     "procedencia" CHAR(3),  
6     PRIMARY KEY("id" AUTOINCREMENT)  
7 );
```

⁴

Cancel OK

DB Browser for SQLite - /home/b.paternalluch/Documents/fruteria.db

File Edit View Tools Help

New Database Open Database **Write Changes** Open Project Attach Database

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Create Table Create Index Print Refresh

Name	Type	Sche
Tables (2)		
productos	CREA	
sqlite_sequence	CREA	
Indices (0)		
Views (0)		
Triggers (0)		

Edit Database Cell

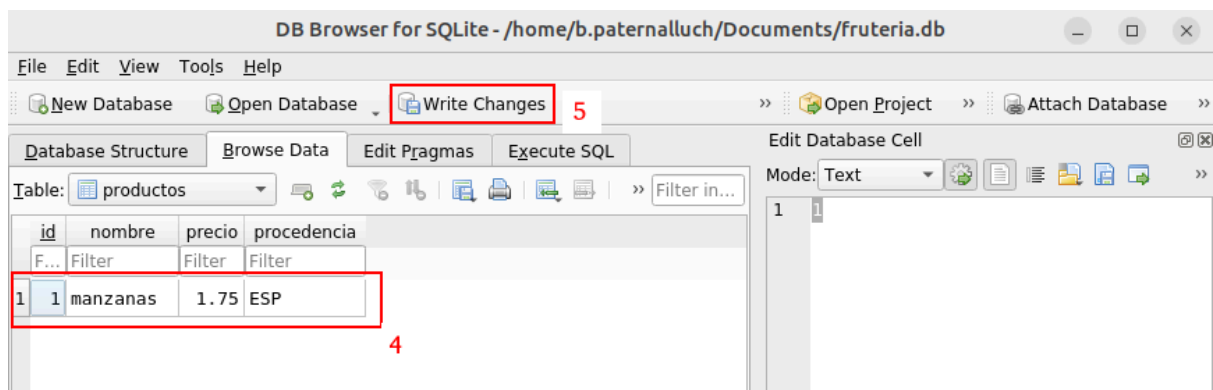
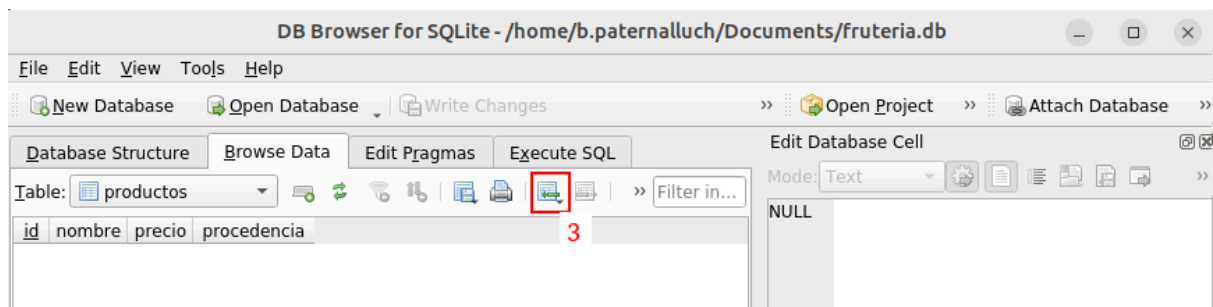
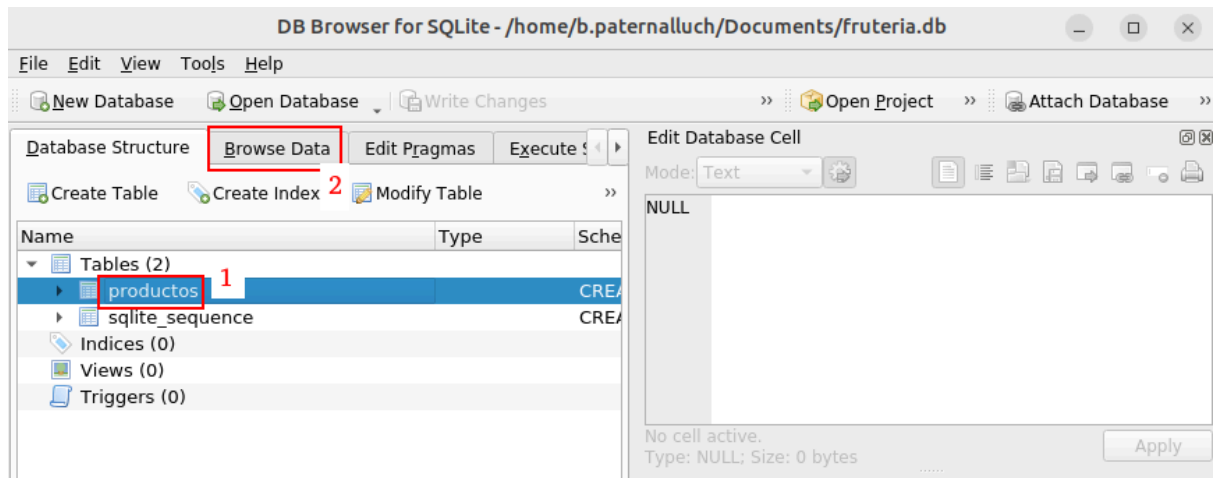
Mode: Text

NULL

No cell active.
Type: NULL | Size: 0 bytes

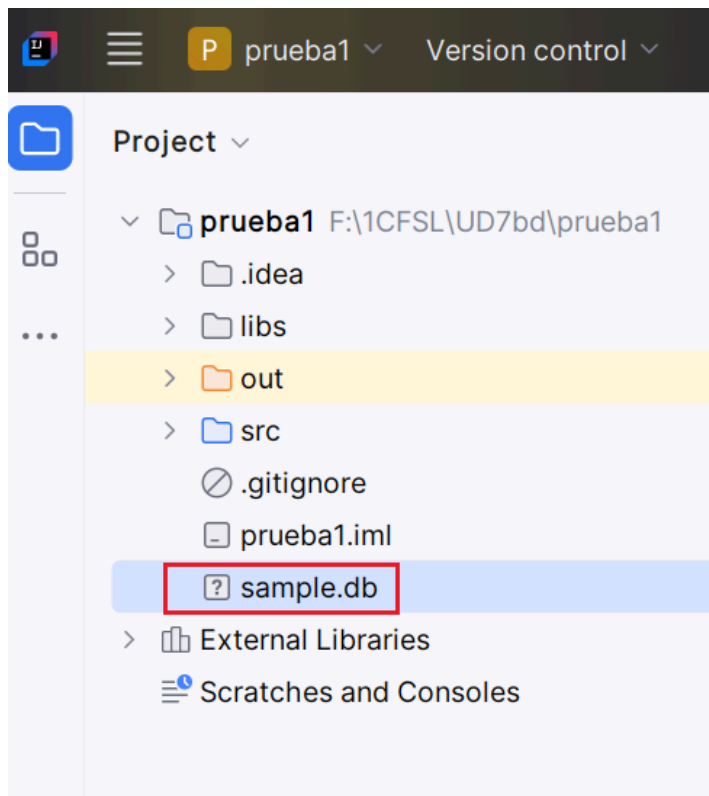
Apply

Añadir información a la tabla:



3. Conexión a la BD

Una vez añadida la dependencia de SQLite, el siguiente paso es copiar la base de datos a la carpeta del proyecto.



Por último hay que programar la conexión a la base de datos. Puedes tomar el siguiente código como ejemplo base de tu main para cualquier programa que desarrolles que necesite acceder a una base de datos:


```

import java.sql.Connection;
import java.sql.DriverManager;

public class Main {

    public static void main(String[] args) {
        String url = "jdbc:sqlite:sample.db"; // Ruta de la base de datos
        Connection conn = null;
        try {
            conn = DriverManager.getConnection(url); // Establecer conexión
            System.out.println("La conexión con la BD se ha establecido correctamente.");

            //programación  Líneas de código del programa

        } catch (Exception e) {
            System.out.println(e.getMessage()); // Si ocurre error de conexión
        } finally {
            try {
                if (conn != null) {
                    conn.close();
                    System.out.println("Conexión cerrada.");
                }
            } catch (Exception ex) {
                System.out.println(ex.getMessage()); // Si ocurre error al cerrar la conexión
            }
        }
    }
}

```