

 Base de données SQLite	
Sommaire :	
I - Introduction.....	1
II - Installation.....	1
III - Administration de SQLite.....	1
III.1. En ligne de commande.....	1
III.2. Avec SQLite Manager.....	2
III.3. Avec SQLite Database Browser.....	2
IV - Accès à SQLite avec différents langages.....	3
IV.1. En langage C.....	3
IV.2. En langage php.....	4
IV.3. En langage Java.....	5
V - Accès à SQLite à distance.....	6

I - Introduction

SQLite est une bibliothèque écrite en C qui propose un moteur de base de données relationnelle accessible par le langage SQL.

Contrairement aux serveurs de bases de données traditionnels, comme MySQL ou PostgreSQL, sa particularité est de ne pas reproduire le schéma habituel client-serveur mais d'être directement intégrée aux programmes. L'intégralité de la base de données (déclarations, tables, index et données) est stockée dans un unique fichier indépendant de la plateforme.

La version **3** de **SQLite**, est publiée en 2004 et ajoute, entre autres, la régionalisation (avec le support des collations et de l'Unicode) et la déclaration des types. Les principaux types sont les suivants : INTEGER, BOOL, REAL, DOUBLE, FLOAT, CHAR, TEXT, VARCHAR, BLOB, NUMERIC, DATETIME.

II - Installation

Dans un environnement Linux, il suffit d'installer les paquets suivants :

```
sudo apt-get install sqlite3
sudo apt-get install libsqlite3-0 libsqlite3-dev
sudo apt-get install php5-sqlite
```

III - Administration de SQLite

III.1. En ligne de commande

```
// creation de la base test.db
user@host:~$ sqlite3 test.db
```

```
// creation de la table collecte
```

```
sqlite> create table collecte (id integer primary key, temp VARCHAR( 250 ), date DATETIME);
```

```
// Insertion de valeurs dans la table collecte
```

```
sqlite>insert into collecte (temp,date) values('23.1','2015-08-01 08:20:51');
```

```
sqlite>insert into collecte (temp,date) values('25.2','2015-08-01 08:21:52');
```

```
// Listage de la table collecte
```

```
sqlite> select * from collecte;
```

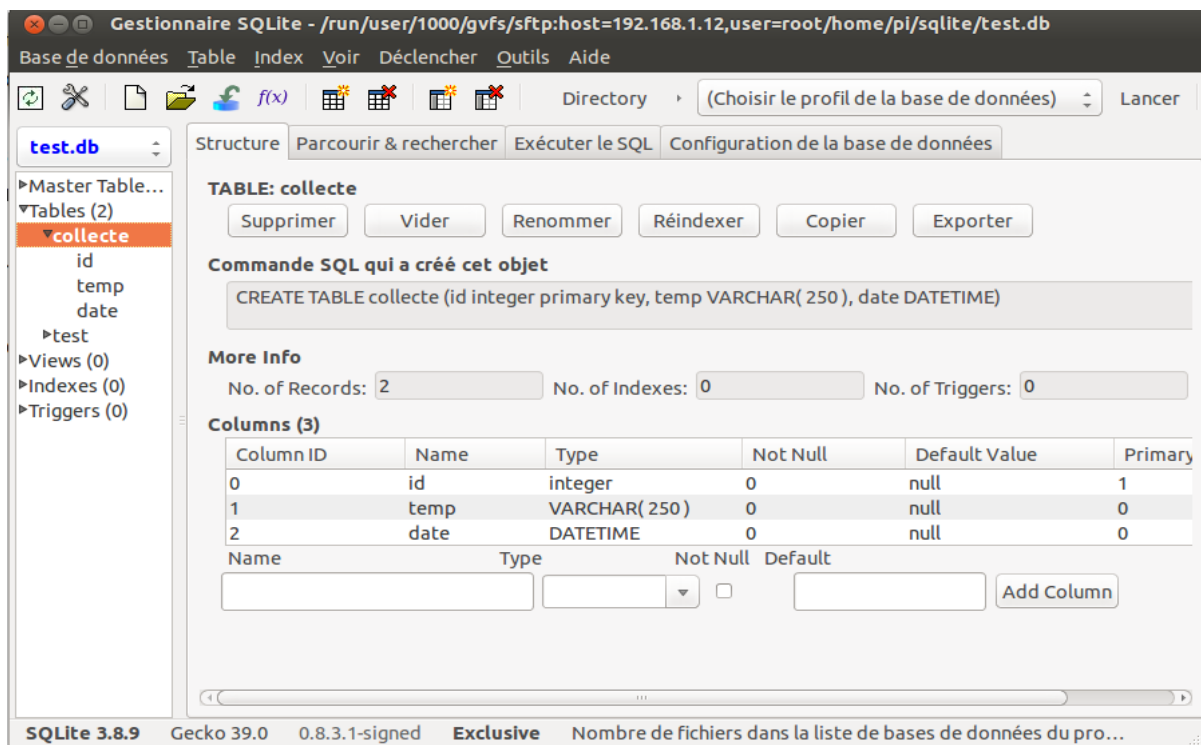
```
// Sortie de l'administration
```

```
sqlite> .exit
```

```
user@host:~$
```

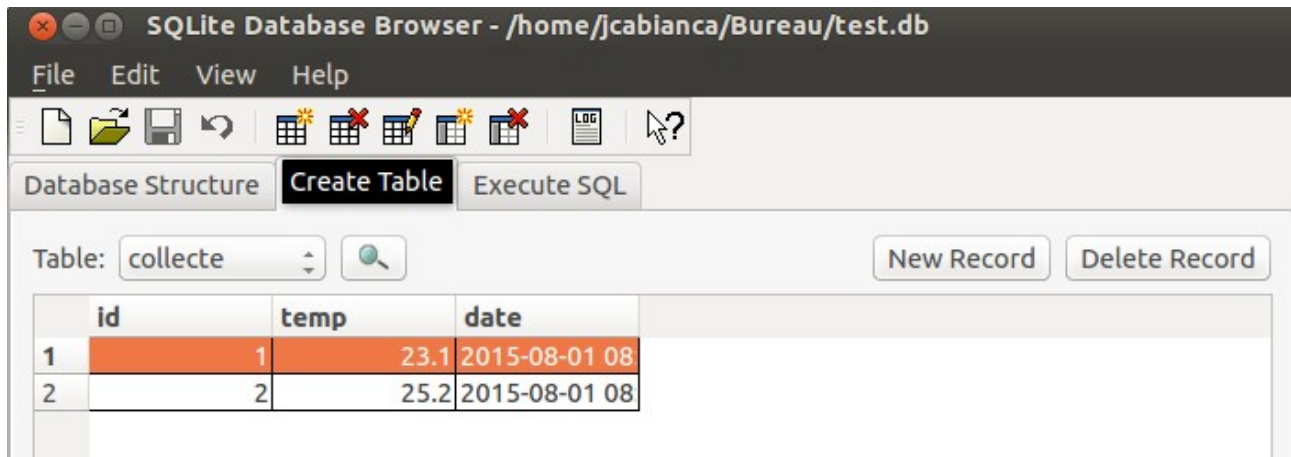
III.2. Avec SQLite Manager

SQLite Manager est un plugin de **Firefox** qu'il suffit d'installer. Le plugin est accessible via le menu **Développement** de **Firefox**.



III.3. Avec SQLite Database Browser

Sous **Linux Ubuntu**, il faut installer le paquet **sqlitebrowser**. Le logiciel sera disponible dans le menu **Programmation**.



IV - Accès à SQLite avec différents langages

IV.1. En langage C

Fichier source `sqlite.c` :

```
#include <stdio.h>
#include <sqlite3.h>

static int callback(void *NotUsed, int argc, char **argv, char **azColName){
    int i;
    for(i=0; i<argc; i++){
        printf("%s = %s\n", azColName[i], argv[i] ? argv[i] : "NULL");
    }
    printf("\n");
    return 0;
}

int main(int argc, char **argv){
    sqlite3 *db;
    char *zErrMsg = 0;
    int rc;
    if( argc!=3 ){
        fprintf(stderr, "Usage: %s DATABASE SQL-STATEMENT\n", argv[0]);
        return(1);
    }
    rc = sqlite3_open(argv[1], &db);
    if( rc ){
        fprintf(stderr, "Can't open database: %s\n", sqlite3_errmsg(db));
        sqlite3_close(db);
        return(1);
    }
    rc = sqlite3_exec(db, argv[2], callback, 0, &zErrMsg);
    if( rc!=SQLITE_OK ){
        fprintf(stderr, "SQL error: %s\n", zErrMsg);
        sqlite3_free(zErrMsg);
    }
    sqlite3_close(db);
    return 0;
}
```

Compilation :

```
gcc sqlite.c -o sqlite -lsqlite3
```

Exécution :

```
./sqlite test.db "select * from collecte"
```

Résultat :

```
id = 1
temp = 23.1
date = 2015-08-01 08:20:51
id = 2
temp = 25.2
date = 2015-08-01 08:21:52
```

IV.2. En langage php

Fichier source sqlite.php :

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
date_default_timezone_set('Europe/Paris');

try{
    //$pdo = new PDO('sqlite:'.dirname(__FILE__).'/test.db');
    $pdo = new PDO('sqlite:/home/pi/sqlite/test.db');
    $pdo->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "<br>Acces à la base de données SQLite : <br>";

    echo "<br>Lecture de données dans la table collecte : <br>";
    // Lire des données avec SELECT
    $stmt = $pdo->prepare("SELECT * FROM collecte");
    $stmt->execute(array());
    $result = $stmt->fetchAll();
    print_r($result);

    echo "<br>Insertion de données dans la table collecte : <br>";
    $date = date("Y-m-d H:i:s");
    echo "<br>Date : ".$date;
    echo "<br>";
    // Ecrire des données avec INSERT
    $stmt = $pdo->prepare("INSERT INTO collecte (temp, date) VALUES (:temp, :date)");
    $result = $stmt->execute(array(
        'temp'      => "21.3",
        'date'      => $date
    ));

    // Ou bien
    $insert = "INSERT INTO collecte (temp, date) VALUES (:temp, :date)";
    $stmt->bindValue(':temp', "21.3");
    $stmt->bindParam(':date', date("Y-m-d H:i:s"));
    $stmt->execute();
}
```

```
$pdo = null;
}
catch(Exception $e) {
    echo "Impossible d'accéder à la base de données SQLite : ".$e->getMessage();
    die();
}
?>
```

Résultat :

Acces à la base de données SQLite :
Lecture de données dans la table collecte :
Array ([0] => Array ([id] => 1 [temp] => 23.1 [date] => 2015-08-01 08:20:51) [1]
=> Array ([id] => 2 [temp] => 25.2 [date] => 2015-08-01 08:21:52))

Remarque :

Pour créer une table ou insérer un enregistrement, il faut que l'utilisateur ait les mêmes droits sur le dossier que sur le fichier de la base de données.

IV.3. En langage Java**Fichier source sqlite.java :**

```
package test.sqlite;

import java.sql.Connection;
import java.sql.DatabaseMetaData;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class TestSqlite {
    public static void main(String[] args) {
        try {
            Class.forName("org.sqlite.JDBC");
            //String dbURL = "jdbc:sqlite:product.db";
            String dbURL = "jdbc:sqlite:/home/jcabanca/Bureau/test.db";
            Connection conn = DriverManager.getConnection(dbURL);
            Statement statement = conn.createStatement();
            System.out.println("Connexion a " + dbURL + " avec succès");
            if (conn != null) {
                System.out.println("Connected to the database");
                DatabaseMetaData dm = (DatabaseMetaData) conn.getMetaData();
                System.out.println("Driver name: " + dm.getDriverName());
                System.out.println("Driver version: " + dm.getDriverVersion());
                System.out.println("Product name: " + dm.getDatabaseProductName());
                System.out.println("Product version: " + dm.getDatabaseProductVersion());
                ResultSet resultat = null;
                String requete="select * from collecte";
                try {
                    resultat = statement.executeQuery(requete);
                    try {
                        while (resultat.next()) {
```

```
        System.out.println("Temp : "+resultat.getString("temp"));
        System.out.println("Date : "+resultat.getString("date"));
    }
}
catch (SQLException e) {
    e.printStackTrace();
}
} catch (SQLException e) {
    e.printStackTrace();
    System.out.println("Erreur dans la requete : " + requete);
}
statement.close();
conn.close();
}
} catch (ClassNotFoundException ex) {
    ex.printStackTrace();
} catch (SQLException ex) {
    ex.printStackTrace();
}
}
}
```

Driver JDBC :

Il faut télécharger le driver **sqlite-jdbc-3.8.7.jar** et l'inclure au projet.

Résultat :

```
Connexion a jdbc:sqlite:/home/jcabianca/Bureau/test.db avec succès
Connected to the database
Driver name: SQLiteJDBC
Driver version: native
Product name: SQLite
Product version: 3.8.7
Temp : 23.1
Date : 2015-08-01 08:20:51
Temp : 25.2
Date : 2015-08-01 08:21:52
```

V - Accès à SQLite à distance

Pour accéder à une base de données distante via **ssh**, on peut utiliser le chemin suivant :

```
sudo apt-get install sshfs
mkdir /home/user/Bureau/bdd
sshfs root@192.168.1.12:/home/pi/sqlite ~/Bureau/bdd // pour monter la ressource
dbURL = "jdbc:sqlite:/home/user/Bureau/bdd/test.db"; // Pour Java
$pdo = new PDO('sqlite:/home/user/Bureau/bdd/test.db'); // Pour Php
fusermount -u /home/user/Bureau/bdd // pour démonter la ressource
```