# Android SQLite Database

SQLite Database là CSDL quan hệ được tích hợp sẵn trên Android OS.

Dữ liệu được lưu trong thư mục data/data/APP\_NAME/databases/FILENAME

* DATA: là đường dẫn có thể get bởi câu lệnh Environment.getDataDirectory()
* APP\_NAME: tên ứng dụng
* FILENAME: tên CSDL được tạo trong ứng dụng

Tại sao dùng SQLite DB?

* SQLite hỗ trợ lưu các đối tượng dưới dạng bảng (table) và sau đó truy xuất lấy ra một cách nhanh chóng theo cú pháp SQL.
* Việc lưu trữ đối tượng là khá khó đối với Shared Preferences, ta phải chuyển nó sang String rồi lưu lại dưới dạng Key – Value. Với kiểu lưu Internal hay External thì rất khó.

SQLiteDB được sử dụng khi nào?

Yêu cầu bài toán lưu trữ dữ liệu dưới dạng Object, dạng text… xuống thiết bị để có thể lấy ra bất kỳ lúc nào.

Ví dụ: ứng dụng ZingMp3 lưu trữ danh sách bài hát yêu thích: tên, lyric, ca sĩ,…

Ứng dụng Alarm Clock lưu trữ danh sách các báo thức đã cài đặt.

Tuy nhiên SQLite chỉ đáp ứng với dữ liệu không quá lớn vài ngàn dòng. Nếu quá nặng thì ứng dụng sẽ chạy rất chậm.

Làm việc với SQLite database:

+ Cần tạo mới một class mở rộng từ class **SQLiteOpenHelper**. Class này thực hiện các chức năng **CRUD** (Create, Read, Update, Delete) trên database. Có 2 phương thức phải ghi đè là:

* onCreate() được gọi một lần khi app được cài đặt. Bảng SQL được khởi tạo trong method này.
* onUpgrade() được gọi khi một bản update được released. Cần phải sửa DATABASE\_VERSION để thực thi phương thức này khi có sự thay đổi trong TABLE structure. Chú ý, workaround bảo vệ data bảng SQL cũ trong device tránh mất mát.

+ class **SQLiteDatabase** là class cung cấp các phương thức thao tác với CSDL như

* execSQL(): truy vấn trực tiếp CSDL bằng câu lệnh SQL
* rawQuery(): chấp nhận một câu lệnh lựa chọn như một đầu vào
* query(): cung cấp cấu trúc cho lệnh SQL đơn giản
* insert()
* update()
* delete()

+ **Cursor**: Một truy vấn CSDL SQLite luôn trả về một đối tượng Cursor.

* moveToFirst()
* moveToNext()
* isAfterLast()
* getCount()
* getLong(columnIndex)
* getString(columnIndex)

***Chú ý phải đóng DB và Cursor sau khi sử dụng.***

Ví dụ một số câu lệnh:

|  |
| --- |
| sqLiteDatabase.execSQL(Movie.CREATE\_TABLE); |
| sqLiteDatabase.execSQL("DROP TABLE IF EXISTS " + Movie.TABLE\_NAME); |
| SQLiteDatabase db = this.getWritableDatabase();  // insert data  long id = db.insert(Movie.TABLE\_NAME, null, values); |
| // update row  return db.update(Movie.TABLE\_NAME, values, Movie.COLUMN\_ID + " = ?",  new String[]{String.valueOf(movie.getId())}); |
| // delete row  db.delete(Movie.TABLE\_NAME, Movie.COLUMN\_ID + " = ?",  new String[]{String.valueOf(movie.getId())}); |
| Cursor cursor = db.query(Movie.TABLE\_NAME,  new String[]{Movie.COLUMN\_ID, Movie.COLUMN\_TITLE, Movie.COLUMN\_GENRE, Movie.COLUMN\_YEAR, Movie.COLUMN\_TIMESTAMP},  Movie.COLUMN\_ID + "=?",  new String[]{String.valueOf(id)}, null, null, null, null); |
| // Select All Query  String selectQuery = "SELECT \* FROM " + Movie.TABLE\_NAME + " ORDER BY " +  Movie.COLUMN\_TIMESTAMP + " DESC";  Cursor cursor = db.rawQuery(selectQuery, null); |

Sử dụng Android SQLite DB

Class MovieDatabaseHelper extends SQLiteOpenHelper

+ insertMovie()

+ getMovie()

+ getAllMovies()

+ getMoviesCount()

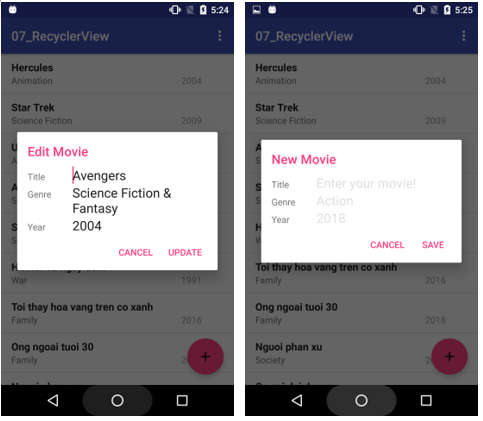
+ updateMovie()

+ deleteMovie()

Minh Họa đối tượng Movie(title, genre, year)

Github: <https://github.com/buinguyentung/android_views_tutorial>

Commit: SQLite:003:implement local Movie database



# Tham khảo:

[1] https://developer.android.com/training/data-storage/sqlite

[2] https://www.androidhive.info/2011/11/android-sqlite-database-tutorial/

[3] https://thangcoder.com/lap-trinh-android/hoc-lap-trinh-android-nang-cao/sqlite-trong-lap-trinh-android