

Lab bài 7 – SQLITE

1. MỤC TIÊU

Sau khi hoàn thành bài thực hành, bạn có thể nắm được các vấn đề sau:

- Nắm được khái niệm và ý nghĩa sử dụng của SQLite
- Biết cách làm việc với SQLite trong ứng dụng Android

2. THỜI GIAN THỰC HIỆN: 20'

3. YÊU CẦU

Viết một ứng dụng Android đơn giản sử dụng SQLite cho phép người dùng có thể lấy thông tin của một sinh viên, cũng như thay đổi thông tin về tên của sinh viên đó.

4. CÁC BƯỚC THỰC HIỆN

Bước 1

Tạo project với các thông tin sau:

- Project Name: testSQLite
- Package Name: sqlite.tensv.lab

Bước 1

Tạo lớp Student, đại diện cho các đối tượng sinh viên

```
package funix.prm.lab;

public class Student {
    private int id;
    private String name;
    private String address;
    private String phone_number;
    public Student(int id, String name, String address, String phone_number) {
        super();
        this.id = id;
        this.name = name;
        this.address = address;
        this.phone_number = phone_number;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getAddress() {
        return address;
    }
    public String getPhone_number() {
        return phone_number;
    }
}
```

Bước 2

Tạo file MyDbHelper để triển khai SQLite cho ứng dụng

```

package funix.prm.lab;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class MyDBHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "schoolManager";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "students";

    private static final String KEY_ID = "id";
    private static final String KEY_NAME = "name";
    private static final String KEY_ADDRESS = "address";
    private static final String KEY_PHONE_NUMBER = "phone_number";

    public MyDBHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String create_students_table = String.format(
            "CREATE TABLE %s(%s INTEGER PRIMARY KEY, %s TEXT, %s TEXT, %s TEXT)",
            TABLE_NAME, KEY_ID, KEY_NAME, KEY_ADDRESS, KEY_PHONE_NUMBER);
        db.execSQL(create_students_table);
    }

```

```

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        String drop_students_table = String.format("DROP TABLE IF EXISTS %s", TABLE_NAME);
        db.execSQL(drop_students_table);

        onCreate(db);
    }

    public void addStudent(Student student) {
        SQLiteDatabase db = this.getWritableDatabase();

        ContentValues values = new ContentValues();
        values.put(KEY_NAME, student.getName());
        values.put(KEY_ADDRESS, student.getAddress());
        values.put(KEY_PHONE_NUMBER, student.getPhone_number());

        db.insert(TABLE_NAME, nullColumnHack: null, values);
        db.close();
    }

    public Student getStudent(int studentId) {
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_NAME, null, selection: KEY_ID + " = ?",
            new String[]{String.valueOf(studentId)}, null, null, null, null);
        if (cursor != null)
            cursor.moveToFirst();
        Student student = new Student(cursor.getInt(0), cursor.getString(1),
            cursor.getString(2), cursor.getString(3));
        return student;
    }

```

```

    public void updateStudent(Student student) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_NAME, student.getName());
        values.put(KEY_ADDRESS, student.getAddress());
        values.put(KEY_PHONE_NUMBER, student.getPhone_number());

        db.update(TABLE_NAME, values, whereClause: KEY_ID + " = ?",
            new String[] {String.valueOf(student.getId())});
        db.close();
    }
}

```

Bước 3

Tạo activity có tên MainActivity để xử lý logic cho ứng dụng

```

package funix.prm.lab;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        final TextView tvName = (TextView) findViewById(R.id.tvName);
        Button btnGetName = (Button) findViewById(R.id.btnGetName);
        Button btnUpdate = (Button) findViewById(R.id.btnUpdateName);
        final MyDBHelper mDB = new MyDBHelper(getApplicationContext());

        Student student = new Student( id: 1, name: "Nguyễn Mạnh Hung",
            address: "Hà Nội", phone_number: "0973527072");
        mDB.addStudent(student);

        btnGetName.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String name = mDB.getStudent( studentId: 1).getName();
                tvName.setText(name);
            }
        });
    }
}

```

```

        btnUpdate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Student student = mDB.getStudent( studentId: 1);
                student.setName("Nguyen Van A");
                mDB.updateStudent(student);

                String name = mDB.getStudent( studentId: 1).getName();
                tvName.setText(name);
            }
        });
    }
}

```

Bước 4

Tạo file layout có tên activity_main để tạo giao diện cho MainActivity

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/tvName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="100dp"
        android:textSize="20sp"
        android:textStyle="bold" />

    <Button
        android:id="@+id/btnGetName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Get Name" />

    <Button
        android:id="@+id/btnUpdateName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Update Name" />
</LinearLayout>

```

Bước 5

Cấu hình MainActivity trên AndroidManifest

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="funix.prm.lab">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="PRM391xLabs"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>

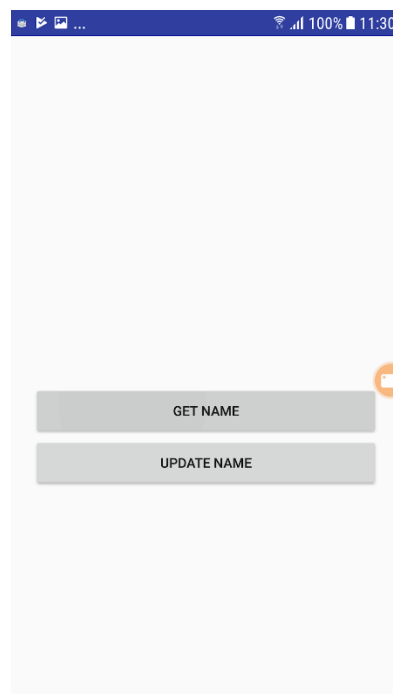
```

5. KIỂM TRA KẾT QUẢ

Chạy ứng dụng trên AVD hoặc bất cứ thiết bị Android nào mà bạn có. Thực hiện các thao tác sau để kiểm tra kết quả

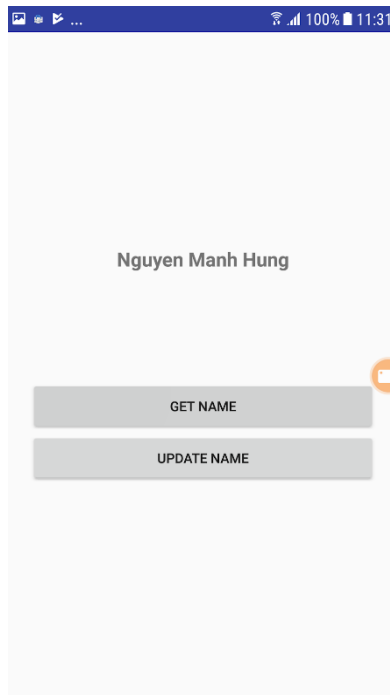
Case 1

Khởi tạo ứng dụng, màn hình sau xuất hiện và đồng thời thông tin của sinh viên đã được insert vào trong csdl



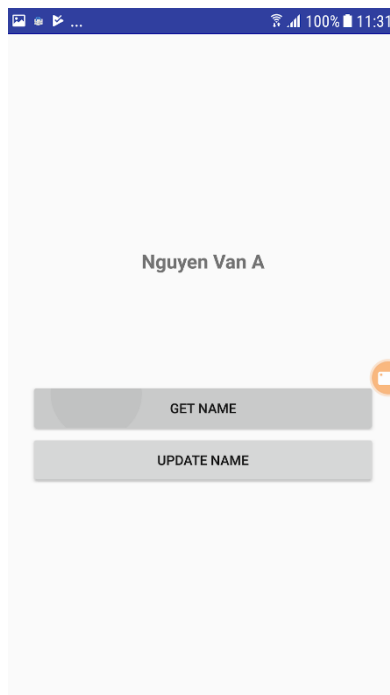
Case 2

Chọn vào nút "GET NAME", thông tin tên của sinh viên đã được insert trước đó xuất hiện trên màn hình



Case 3

Chọn vào nút "Update Name", thông tin tên của sinh viên sẽ được cập nhật và hiển thị lại trên màn hình



Case 4

Thoát ứng dụng và khởi động lại ứng dụng. Chọn nút "GET NAME" thông tin trên màn hình hiển thị giống **Case 3**