

BÁO CÁO THỰC HÀNH

Môn học: Lập trình mạng căn bản

Buổi báo cáo: Lab 03

Tên chủ đề: SQLITE

GVHD: Phan Xuân Thiện

Ngày thực hiện: 25/10/2024

THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT118.P13.1

STT	Họ và tên	MSSV	Email
1	Phạm Thị Thanh Vinh	22521680	22521680@gm.uit.edu.vn

1. ĐÁNH GIÁ KHÁC:

Nội dung	Kết quả
Tổng thời gian thực hiện bài thực hành trung bình	2 ngày
Mã nguồn (Github):	https://github.com/Pttvinh253/Lab_NT118.git
Ý kiến (nếu có) + Khó khăn + Đề xuất ...	
Điểm tự đánh giá	10/10

Contents

1. Quản lý Contact 2
2. Quản lý danh sách sinh viên, thêm, xem, xóa, sửa, nhấn vào mỗi 1 sinh viên sẽ rat rang chi tiết 1 sinh viên có sử dụng RecyclerView và SQLite. 5

BÁO CÁO CHI TIẾT

1. Quản lý Contact

- Tạo 1 lớp Contact có Constructor, Getter, Setter cho class Contact để format kiểu dữ liệu:

```
package com.example.lab03;

public class Contact { 17 usages
    private int id; 2 usages
    private String name; 2 usages
    private String phoneNumber; 2 usages

    public Contact(String name, String phoneNumber) { 6 usages
        this.name = name;
        this.phoneNumber = phoneNumber;
    }

    public void setId(int id) { this.id = id; }

    public int getId() { return id; }

    public String getName() { return name; }

    public String getPhoneNumber() { return phoneNumber; }
```

- Tạo 1 lớp DatabaseHandler kế thừa từ SQLiteOpenHelper
 - Override hàm onCreate() để tạo table

```
// Creating Tables
@Override
public void onCreate(SQLiteDatabase db) {
    String CREATE_CONTACTS_TABLE = "CREATE TABLE " +
        TABLE_CONTACTS + "("
        + KEY_ID + " INTEGER PRIMARY KEY," + KEY_NAME + " TEXT,"
        + KEY_PH_NO + " TEXT" + ")";
    db.execSQL(CREATE_CONTACTS_TABLE);
}
```

- Override hàm onUpgrade() hủy table cũ, tạo table mới

```
// Upgrading database
@Override no usages
public void onUpgrade(SQLiteDatabase db, int oldVersion, int
    newVersion) {
    // Drop older table if existed
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_CONTACTS);
    // Create tables again
    onCreate(db);
}
```

- Thêm các phương thức để insert, select, update và delete contact

```
// Adding new contact
public void addContact(Contact contact) { 4 usages
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(KEY_NAME, contact.getName());
    values.put(KEY_PH_NO, contact.getPhoneNumber());

    // Inserting Row
    db.insert(TABLE_CONTACTS, nullColumnHack: null, values);
    db.close(); // Closing database connection
}

// Getting single contact
public Contact getContact(int id) { no usages
    SQLiteDatabase db = this.getReadableDatabase();

    Cursor cursor = db.query(TABLE_CONTACTS, new String[] { KEY_ID,
        KEY_NAME, KEY_PH_NO }, "selection: KEY_ID + "=?",
        new String[] { String.valueOf(id) }, null, null, null, null, null);
    if (cursor != null)
        cursor.moveToFirst();

    Contact contact = new Contact(cursor.getString(1), cursor.getString(2));
    contact.setId(cursor.getInt(0)); // Set ID
    cursor.close();
    return contact;
}

// Getting All Contacts
public List<Contact> getAllContacts() { 1 usage
    List<Contact> contactList = new ArrayList<>();
    // Select All Query
    String selectQuery = "SELECT * FROM " + TABLE_CONTACTS;

    SQLiteDatabase db = this.getWritableDatabase();
    Cursor cursor = db.rawQuery(selectQuery, selectionArgs: null);

    // Looping through all rows and adding to list
    if (cursor.moveToFirst()) {
        do {
            Contact contact = new Contact(cursor.getString(1), cursor.getString(2));
            contact.setId(cursor.getInt(0)); // Set ID
            contactList.add(contact);
        } while (cursor.moveToNext());
    }
    cursor.close();
    return contactList;
}
```



```
// Updating single contact
public int updateContact(Contact contact) { no usages
    SQLiteDatabase db = this.getWritableDatabase();

    ContentValues values = new ContentValues();
    values.put(KEY_NAME, contact.getName());
    values.put(KEY_PH_NO, contact.getPhoneNumber());

    // Updating row
    return db.update(TABLE_CONTACTS, values, whereClause: KEY_ID + " = ?",
        new String[] { String.valueOf(contact.getId()) });
}

// Deleting single contact
public void deleteContact(Contact contact) { 1 usage
    SQLiteDatabase db = this.getWritableDatabase();
    db.delete(TABLE_CONTACTS, whereClause: KEY_ID + " = ?",
        new String[] { String.valueOf(contact.getId()) });
    db.close();
}
```

- MainActivity.java

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_main);
    ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
        Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
        v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
        return insets;
    });

    contactListView = findViewById(R.id.contact_list_view);
    db = new DatabaseHandler(this);

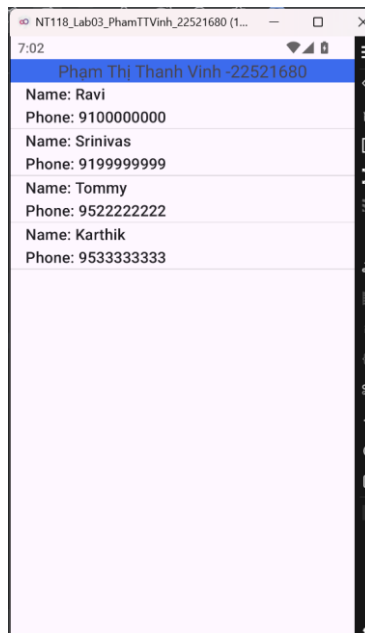
    // Inserting Contacts
    Log.d(tag, "Insert: ", msg, "Inserting ..");
    db.addContact(new Contact(name: "Ravi", phoneNumber: "9100000000"));
    db.addContact(new Contact(name: "Srinivas", phoneNumber: "9199999999"));
    db.addContact(new Contact(name: "Tommy", phoneNumber: "9522222222"));
    db.addContact(new Contact(name: "Karthik", phoneNumber: "9533333333"));

    // Load and display contacts
    loadContacts();

    // Set up long-click listener for deleting contacts
    contactListView.setOnItemClickListener((parent, view, position, id) -> {
        Contact contact = contactList.get(position);
        db.deleteContact(contact); // Xóa contact khỏi database
        Toast.makeText(context, MainActivity.this, text, "Deleted: " + contact.getName(), Toast.LENGTH_SHORT).show();
        loadContacts(); // Tải lại danh sách sau khi xóa
        return true;
    });
}
```

- Kết quả hiển thị hiện tại của ứng dụng (xem tại Log Cat)

```
Android tag: name: tag: reading:
Reading: com.example.lab03 E Reading all contacts..
Name: com.example.lab03 E Id: 1 ,Name: Ravi, Phone: 9100000000
Name: com.example.lab03 E Id: 2 ,Name: Srinivas, Phone: 9199999999
Name: com.example.lab03 E Id: 3 ,Name: Tommy, Phone: 9522222222
Name: com.example.lab03 E Id: 4 ,Name: Karthik, Phone: 9533333333
```



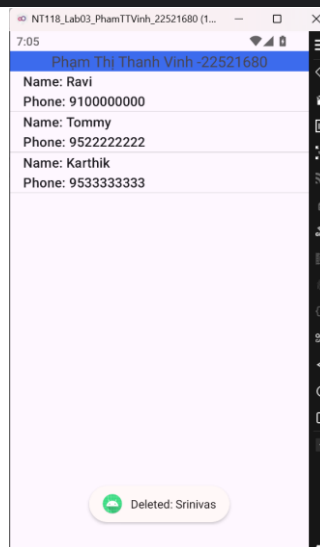
- Sau khi nhấn giữ 1 item thì nó sẽ bị xóa kèm thêm Toast thông báo đã xóa Contact nào đồng thời sẽ thực hiện lại hàm loadContacts rồi in ra danh sách các Contact mới:

```

tag:name: tag:reading:
Reading: com.example.Lab03
Name: com.example.Lab03
Name: com.example.Lab03
Name: com.example.Lab03
Name: com.example.Lab03
Reading: com.example.Lab03
Name: com.example.Lab03
Name: com.example.Lab03
Name: com.example.Lab03

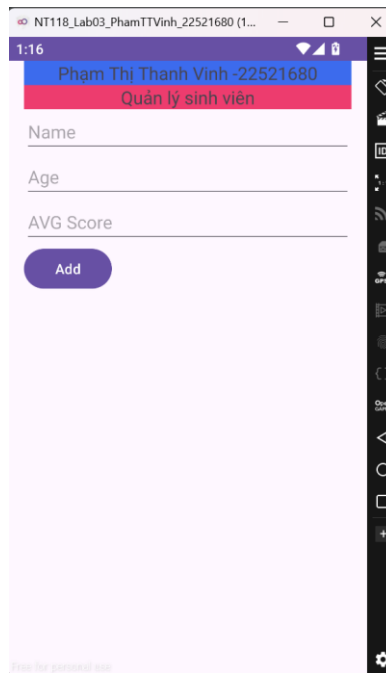
E Reading all contacts..
E Id: 1 ,Name: Ravi, Phone: 9100000000
E Id: 2 ,Name: Srinivas, Phone: 9199999999
E Id: 3 ,Name: Tommy, Phone: 9522222222
E Id: 4 ,Name: Karthik, Phone: 9533333333
E Reading all contacts..
E Id: 1 ,Name: Ravi, Phone: 9100000000
E Id: 3 ,Name: Tommy, Phone: 9522222222
E Id: 4 ,Name: Karthik, Phone: 9533333333

```

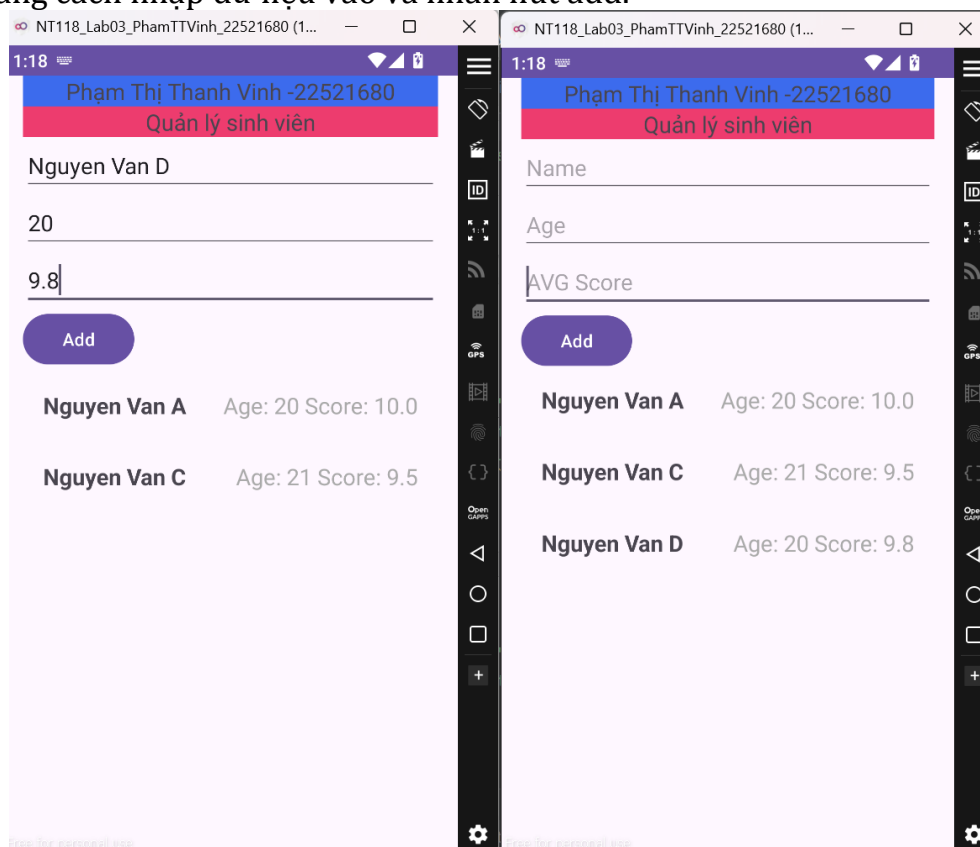


2. Quản lý danh sách sinh viên, thêm, xem, xóa, sửa, nhấn vào mỗi 1 sinh viên sẽ rat rang chi tiết 1 sinh viên có sử dụng Recyclerview và SQLite.

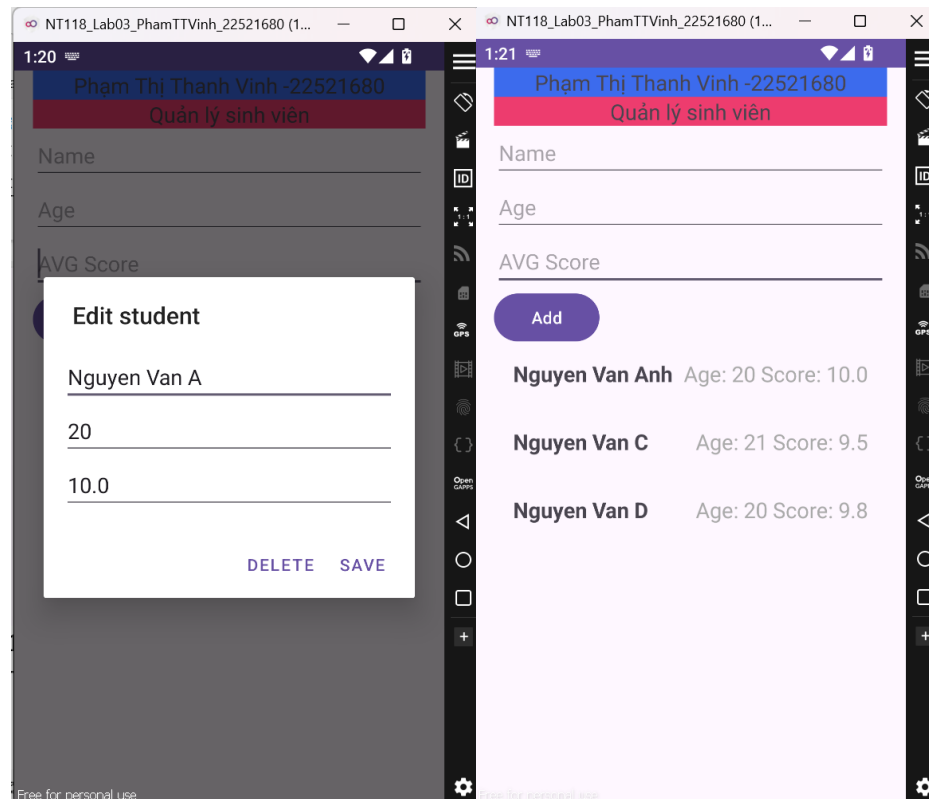
- Logic tương tự bài 1.
- Giao diện khi chạy:



- Test bằng cách nhập dữ liệu vào và nhấn nút add.



- Khi nhấn vào 1 item thì nó sẽ hiện tùy chọn chỉnh sửa, sau đó có 2 nút là **save** hoặc **delete** để mình có thể chọn:
 - + Test với sửa dữ liệu trước:



+ Test với nút xóa dữ liệu:

