Yêu cầu 1: Tạo 1 CSDL SQLite, thêm vào CSDL danh sách các user, sau đó lấy dữ liệu từ CSDL hiện lên listview

Bước 1: Ta code nội dung trong file activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <ListView  
 android:id="@+id/lv\_user"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
</LinearLayout>

Bước 2: Tạo 1 file tên là item\_user.xml và gõ lệnh code sau

<?xml version="1.0" encoding="utf-8"?>  
<TextView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/tv\_user"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:padding="16dp"  
 android:textSize="18sp"/>

Bước 3: Tạo 1 class tên là DatabaseHelper gồm 2 phương thức onCreate() và onUpgrade() để quản lý việc tạo CSDL và version của CSDL

package com.example.a1150070035\_huynhngocphuc\_bai3.helper;  
  
import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
public class DatabaseHelper extends SQLiteOpenHelper {  
 private static final String *DATABASE\_CREATE* =  
 "CREATE TABLE users (\_id INTEGER PRIMARY KEY AUTOINCREMENT, " +  
 "name TEXT NOT NULL);";  
  
 public DatabaseHelper(Context context, String name,  
 SQLiteDatabase.CursorFactory factory, int version) {  
 super(context, name, factory, version);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL(*DATABASE\_CREATE*);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS users");  
 onCreate(db);  
 }  
}

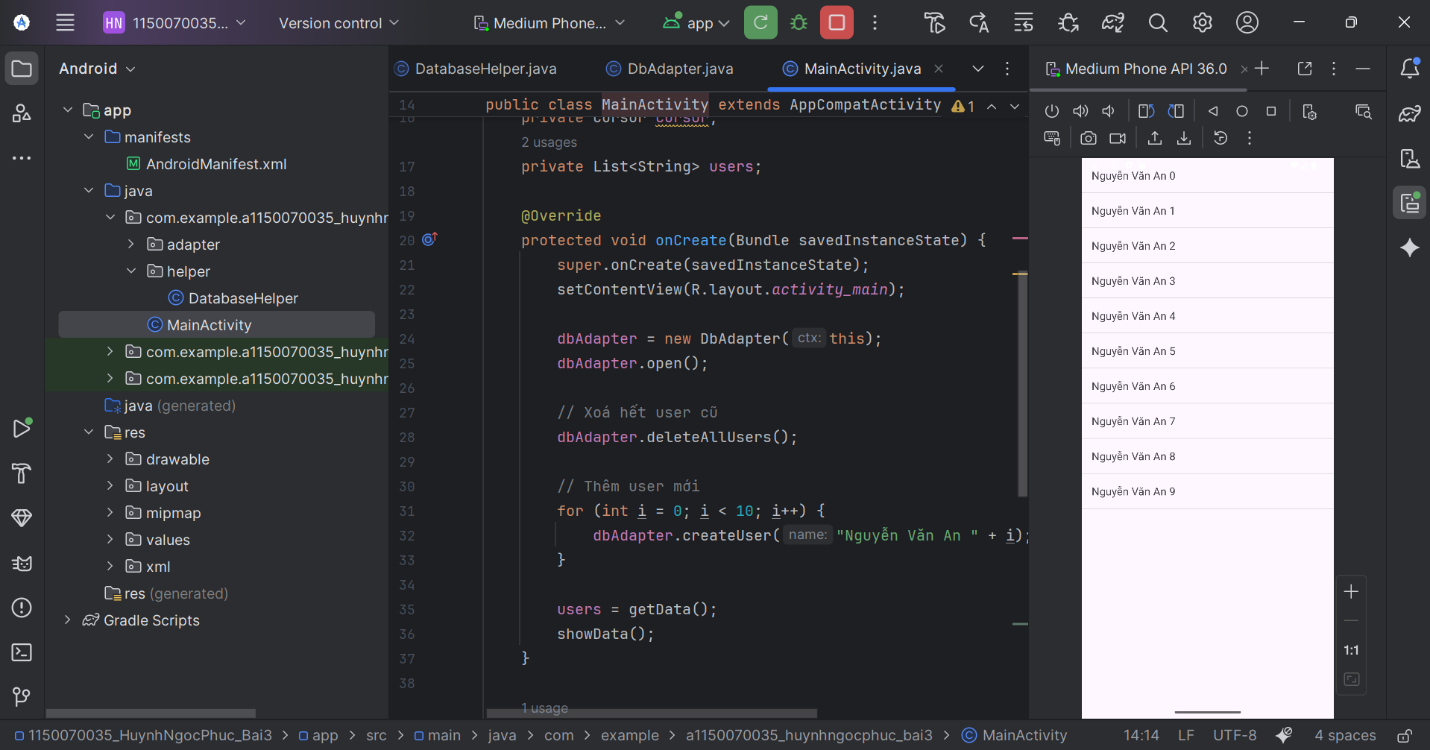
Bước 4: Add DbAdapter trong Java để quản lý các thao tác cơ bản với cơ sở dữ liệu SQLite, bao gồm các chức năng mở, đóng, thêm, xóa và lấy dữ liệu người dùng.

package com.example.a1150070035\_huynhngocphuc\_bai3.adapter;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.helper.DatabaseHelper;  
  
public class DbAdapter {  
 public static final String *KEY\_ID* = "\_id";  
 public static final String *KEY\_NAME* = "name";  
  
 private DatabaseHelper dbHelper;  
 private SQLiteDatabase sqLiteDatabase;  
  
 private static final String *DATABASE\_NAME* = "Database\_Demo";  
 private static final String *DATABASE\_TABLE* = "users";  
 private static final int *DATABASE\_VERSION* = 1;  
  
 private final Context context;  
  
 public DbAdapter(Context ctx) {  
 this.context = ctx;  
 }  
  
 public DbAdapter open() {  
 dbHelper = new DatabaseHelper(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 sqLiteDatabase = dbHelper.getWritableDatabase();  
 return this;  
 }  
  
 public void close() {  
 dbHelper.close();  
 }  
  
 public long createUser(String name) {  
 ContentValues inititalValues = new ContentValues();  
 inititalValues.put(*KEY\_NAME*, name);  
 return sqLiteDatabase.insert(*DATABASE\_TABLE*, null, inititalValues);  
 }  
  
 public boolean deleteAllUsers() {  
 return sqLiteDatabase.delete(*DATABASE\_TABLE*, null, null) > 0;  
 }  
  
 public Cursor getAllUsers() {  
 return sqLiteDatabase.query(*DATABASE\_TABLE*,  
 new String[]{*KEY\_ID*, *KEY\_NAME*},  
 null, null, null, null, null);  
 }  
}

Bước 5: Trong mục MainActivity

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.database.Cursor;  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.adapter.DbAdapter;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private DbAdapter dbAdapter;  
 private Cursor cursor;  
 private List<String> users;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 dbAdapter = new DbAdapter(this);  
 dbAdapter.open();  
  
 // Xoá hết user cũ  
 dbAdapter.deleteAllUsers();  
  
 // Thêm user mới  
 for (int i = 0; i < 10; i++) {  
 dbAdapter.createUser("Nguyễn Văn An " + i);  
 }  
  
 users = getData();  
 showData();  
 }  
  
 private List<String> getData() {  
 List<String> users = new ArrayList<>();  
 cursor = dbAdapter.getAllUsers();  
 while (cursor.moveToNext()) {  
 users.add(cursor.getString(cursor.getColumnIndexOrThrow(DbAdapter.*KEY\_NAME*)));  
 }  
 cursor.close();  
 return users;  
 }  
  
 private void showData() {  
 ListView lvUser = findViewById(R.id.*lv\_user*);  
 ArrayAdapter<String> userAdapter =  
 new ArrayAdapter<>(MainActivity.this, R.layout.*item\_user*, users);  
 lvUser.setAdapter(userAdapter);  
 }  
}

Bước 6: Ta sẽ hiện ra màn hình như sau



Yêu cầu 2: Quản lý Contact

Bước 1: Tạo 1 lớp tên là Contact

package com.example.a1150070035\_huynhngocphuc\_bai3.model;  
  
public class Contact {  
 private int id;  
 private String name;  
 private String phoneNumber;  
  
 // Constructors  
 public Contact() {}  
  
 public Contact(int id, String name, String phoneNumber) {  
 this.id = id;  
 this.name = name;  
 this.phoneNumber = phoneNumber;  
 }  
  
 public Contact(String name, String phoneNumber) {  
 this.name = name;  
 this.phoneNumber = phoneNumber;  
 }  
  
 // Getter - Setter  
 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 getPhoneNumber() {  
 return phoneNumber;  
 }  
  
 public void setPhoneNumber(String phoneNumber) {  
 this.phoneNumber = phoneNumber;  
 }  
}

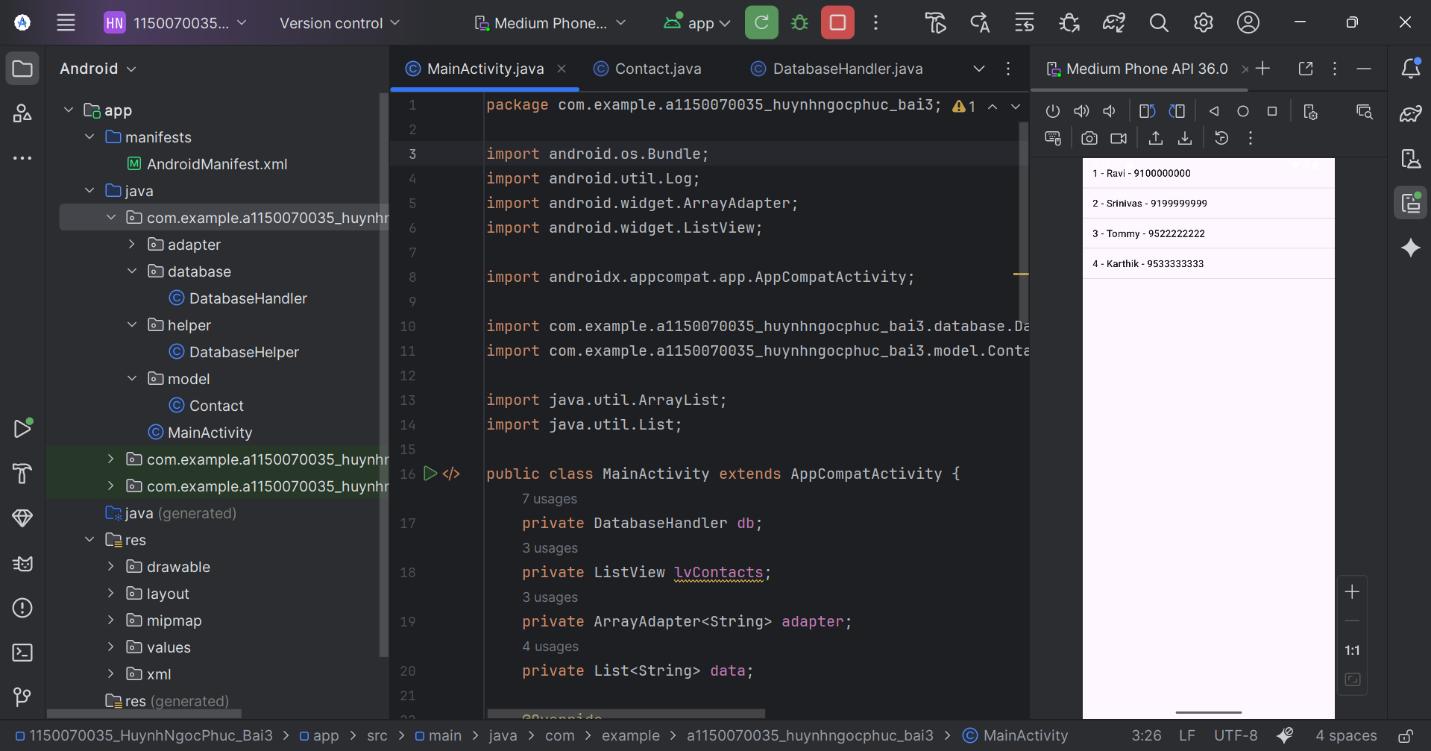
Bước 2: Tạo 1 lớp DatabaseHandler kế thừa từ SQLiteOpenHelper

package com.example.a1150070035\_huynhngocphuc\_bai3.database;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Contact;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class DatabaseHandler extends SQLiteOpenHelper {  
  
 // Database Info  
 private static final int *DATABASE\_VERSION* = 1;  
 private static final String *DATABASE\_NAME* = "contactsManager";  
 private static final String *TABLE\_CONTACTS* = "contacts";  
  
 // Column names  
 private static final String *KEY\_ID* = "id";  
 private static final String *KEY\_NAME* = "name";  
 private static final String *KEY\_PH\_NO* = "phone\_number";  
  
 public DatabaseHandler(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 }  
  
 // Create table  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 String CREATE\_CONTACTS\_TABLE = "CREATE TABLE " + *TABLE\_CONTACTS* + "("  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT,"  
 + *KEY\_NAME* + " TEXT,"  
 + *KEY\_PH\_NO* + " TEXT" + ")";  
 db.execSQL(CREATE\_CONTACTS\_TABLE);  
 }  
  
 // Upgrade table  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_CONTACTS*);  
 onCreate(db);  
 }  
  
 // Thêm contact  
 public void addContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, contact.getName());  
 values.put(*KEY\_PH\_NO*, contact.getPhoneNumber());  
  
 db.insert(*TABLE\_CONTACTS*, null, values);  
 db.close();  
 }  
  
 // Lấy 1 contact theo ID  
 public Contact getContact(int id) {  
 SQLiteDatabase db = this.getReadableDatabase();  
  
 Cursor cursor = db.query(*TABLE\_CONTACTS*, new String[]{*KEY\_ID*, *KEY\_NAME*, *KEY\_PH\_NO*},  
 *KEY\_ID* + "=?", new String[]{String.*valueOf*(id)}, null, null, null);  
  
 if (cursor != null)  
 cursor.moveToFirst();  
  
 Contact contact = new Contact(  
 cursor.getInt(0),  
 cursor.getString(1),  
 cursor.getString(2)  
 );  
  
 cursor.close();  
 return contact;  
 }  
  
 // Lấy tất cả contact  
 public List<Contact> getAllContacts() {  
 List<Contact> contactList = new ArrayList<>();  
 String selectQuery = "SELECT \* FROM " + *TABLE\_CONTACTS*;  
  
 SQLiteDatabase db = this.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, null);  
  
 if (cursor.moveToFirst()) {  
 do {  
 Contact contact = new Contact();  
 contact.setId(cursor.getInt(0));  
 contact.setName(cursor.getString(1));  
 contact.setPhoneNumber(cursor.getString(2));  
 contactList.add(contact);  
 } while (cursor.moveToNext());  
 }  
 cursor.close();  
 return contactList;  
 }  
  
 // Update contact  
 public int updateContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, contact.getName());  
 values.put(*KEY\_PH\_NO*, contact.getPhoneNumber());  
  
 return db.update(*TABLE\_CONTACTS*, values, *KEY\_ID* + " = ?",  
 new String[]{String.*valueOf*(contact.getId())});  
 }  
  
 // Xóa contact  
 public void deleteContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 db.delete(*TABLE\_CONTACTS*, *KEY\_ID* + " = ?",  
 new String[]{String.*valueOf*(contact.getId())});  
 db.close();  
 }  
}

Bước 3: Ở mục MainActivity ta sửa lại như sau

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.database.DatabaseHandler;  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Contact;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private DatabaseHandler db;  
 private ListView lvContacts;  
 private ArrayAdapter<String> adapter;  
 private List<String> data;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 lvContacts = findViewById(R.id.*lv\_user*); // tái sử dụng listview cũ  
 db = new DatabaseHandler(this);  
  
 // Xoá contact cũ (nếu muốn clean DB)  
 // getWritableDatabase().execSQL("DELETE FROM contacts");  
  
 // Thêm contact mẫu  
 db.addContact(new Contact("Ravi", "9100000000"));  
 db.addContact(new Contact("Srinivas", "9199999999"));  
 db.addContact(new Contact("Tommy", "9522222222"));  
 db.addContact(new Contact("Karthik", "9533333333"));  
  
 // Lấy toàn bộ contact  
 List<Contact> contacts = db.getAllContacts();  
  
 // Đưa dữ liệu vào List<String>  
 data = new ArrayList<>();  
 for (Contact c : contacts) {  
 String line = c.getId() + " - " + c.getName() + " - " + c.getPhoneNumber();  
 data.add(line);  
 Log.*d*("Contact", line);  
 }  
  
 // Adapter  
 adapter = new ArrayAdapter<>(this, android.R.layout.*simple\_list\_item\_1*, data);  
 lvContacts.setAdapter(adapter);  
  
 // Xoá contact khi long click  
 lvContacts.setOnItemLongClickListener((parent, view, position, id) -> {  
 Contact toDelete = contacts.get(position);  
 db.deleteContact(toDelete);  
 contacts.remove(position);  
 data.remove(position);  
 adapter.notifyDataSetChanged();  
 return true;  
 });  
 }  
}

Bước 4: Ta sẽ có kết quả như này



Yêu cầu 3: 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 dung Recyclerview và SQLite. Thiết kế giao diện theo cảm nhận của mỗi bạn.

Bước 1: Tạo 1 model Student

package com.example.a1150070035\_huynhngocphuc\_bai3.model;  
  
public class Student {  
 private int id;  
 private String name;  
 private String email;  
 private String phone;  
  
 public Student() {}  
  
 public Student(String name, String email, String phone) {  
 this.name = name;  
 this.email = email;  
 this.phone = phone;  
 }  
  
 public Student(int id, String name, String email, String phone) {  
 this.id = id;  
 this.name = name;  
 this.email = email;  
 this.phone = phone;  
 }  
  
 // Getter - Setter  
 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 getEmail() { return email; }  
 public void setEmail(String email) { this.email = email; }  
  
 public String getPhone() { return phone; }  
 public void setPhone(String phone) { this.phone = phone; }  
}

Bước 2: Tạo DatabaseHandler cho Student tên là StudentDatabaseHandler

package com.example.a1150070035\_huynhngocphuc\_bai3.database;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Student;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class StudentDatabaseHandler extends SQLiteOpenHelper {  
 private static final int *DATABASE\_VERSION* = 1;  
 private static final String *DATABASE\_NAME* = "studentManager";  
 private static final String *TABLE\_STUDENTS* = "students";  
  
 private static final String *KEY\_ID* = "id";  
 private static final String *KEY\_NAME* = "name";  
 private static final String *KEY\_EMAIL* = "email";  
 private static final String *KEY\_PHONE* = "phone";  
  
 public StudentDatabaseHandler(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 String CREATE\_STUDENTS\_TABLE = "CREATE TABLE " + *TABLE\_STUDENTS* + "("  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT,"  
 + *KEY\_NAME* + " TEXT,"  
 + *KEY\_EMAIL* + " TEXT,"  
 + *KEY\_PHONE* + " TEXT" + ")";  
 db.execSQL(CREATE\_STUDENTS\_TABLE);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_STUDENTS*);  
 onCreate(db);  
 }  
  
 // Thêm sinh viên  
 public void addStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, student.getName());  
 values.put(*KEY\_EMAIL*, student.getEmail());  
 values.put(*KEY\_PHONE*, student.getPhone());  
 db.insert(*TABLE\_STUDENTS*, null, values);  
 db.close();  
 }  
  
 // Lấy tất cả sinh viên  
 public List<Student> getAllStudents() {  
 List<Student> studentList = new ArrayList<>();  
 String selectQuery = "SELECT \* FROM " + *TABLE\_STUDENTS*;  
  
 SQLiteDatabase db = this.getReadableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, null);  
  
 if (cursor.moveToFirst()) {  
 do {  
 Student student = new Student();  
 student.setId(cursor.getInt(0));  
 student.setName(cursor.getString(1));  
 student.setEmail(cursor.getString(2));  
 student.setPhone(cursor.getString(3));  
 studentList.add(student);  
 } while (cursor.moveToNext());  
 }  
 cursor.close();  
 return studentList;  
 }  
  
 // Update sinh viên  
 public int updateStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, student.getName());  
 values.put(*KEY\_EMAIL*, student.getEmail());  
 values.put(*KEY\_PHONE*, student.getPhone());  
 return db.update(*TABLE\_STUDENTS*, values, *KEY\_ID* + "=?",  
 new String[]{String.*valueOf*(student.getId())});  
 }  
  
 // Xóa sinh viên  
 public void deleteStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 db.delete(*TABLE\_STUDENTS*, *KEY\_ID* + "=?",  
 new String[]{String.*valueOf*(student.getId())});  
 db.close();  
 }  
}

Bước 3: Ở layout ta tạo activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical"  
 android:padding="16dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:id="@+id/btnAdd"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Thêm sinh viên" />  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/recyclerViewStudents"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingTop="8dp"/>  
</LinearLayout>

Bước 4: Tạo 1 layout tên là dialog\_student.xml để chỉnh sửa

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <EditText  
 android:id="@+id/edtName"  
 android:hint="Tên sinh viên"  
 android:inputType="textPersonName"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"/>  
  
 <EditText  
 android:id="@+id/edtEmail"  
 android:hint="Email"  
 android:inputType="textEmailAddress"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"/>  
  
 <EditText  
 android:id="@+id/edtPhone"  
 android:hint="Số điện thoại"  
 android:inputType="phone"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"/>  
</LinearLayout>

Bước 5: Ở mục MainActivity sửa lại

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.adapter.StudentAdapter;  
import com.example.a1150070035\_huynhngocphuc\_bai3.database.StudentDatabaseHandler;  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Student;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private StudentDatabaseHandler db;  
 private StudentAdapter adapter;  
 private List<Student> students;  
 private RecyclerView recyclerView;  
 private Button btnAdd;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 db = new StudentDatabaseHandler(this);  
 recyclerView = findViewById(R.id.*recyclerViewStudents*);  
 btnAdd = findViewById(R.id.*btnAdd*);  
  
 // Load dữ liệu từ DB  
 students = new ArrayList<>(db.getAllStudents());  
  
 // Adapter + RecyclerView  
 adapter = new StudentAdapter(this, students, new StudentAdapter.OnStudentClickListener() {  
 @Override  
 public void onStudentClick(Student student, int position) {  
 showEditStudentDialog(student, position); // nhấn để sửa  
 }  
  
 @Override  
 public void onStudentLongClick(Student student, int position) {  
 db.deleteStudent(student);  
 students.remove(position); // xoá chính xác theo index  
 adapter.notifyItemRemoved(position);  
 Toast.*makeText*(MainActivity.this, "Đã xoá " + student.getName(), Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 recyclerView.setLayoutManager(new LinearLayoutManager(this));  
 recyclerView.setAdapter(adapter);  
  
 // Nút thêm sinh viên  
 btnAdd.setOnClickListener(v -> showAddStudentDialog());  
 }  
  
 // Dialog thêm sinh viên  
 private void showAddStudentDialog() {  
 View dialogView = getLayoutInflater().inflate(R.layout.*dialog\_student*, null);  
  
 EditText edtName = dialogView.findViewById(R.id.*edtName*);  
 EditText edtEmail = dialogView.findViewById(R.id.*edtEmail*);  
 EditText edtPhone = dialogView.findViewById(R.id.*edtPhone*);  
  
 new AlertDialog.Builder(this)  
 .setTitle("Thêm sinh viên")  
 .setView(dialogView)  
 .setPositiveButton("Thêm", (dialog, which) -> {  
 String name = edtName.getText().toString().trim();  
 String email = edtEmail.getText().toString().trim();  
 String phone = edtPhone.getText().toString().trim();  
  
 if (!name.isEmpty()) {  
 Student student = new Student(name, email, phone);  
 db.addStudent(student);  
  
 // Reload list  
 students.clear();  
 students.addAll(db.getAllStudents());  
 adapter.notifyDataSetChanged();  
  
 Toast.*makeText*(this, "Đã thêm sinh viên", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Tên không được bỏ trống", Toast.*LENGTH\_SHORT*).show();  
 }  
 })  
 .setNegativeButton("Hủy", null)  
 .show();  
 }  
  
 // Dialog sửa sinh viên  
 private void showEditStudentDialog(Student student, int position) {  
 View dialogView = getLayoutInflater().inflate(R.layout.*dialog\_student*, null);  
  
 EditText edtName = dialogView.findViewById(R.id.*edtName*);  
 EditText edtEmail = dialogView.findViewById(R.id.*edtEmail*);  
 EditText edtPhone = dialogView.findViewById(R.id.*edtPhone*);  
  
 // Gán dữ liệu cũ  
 edtName.setText(student.getName());  
 edtEmail.setText(student.getEmail());  
 edtPhone.setText(student.getPhone());  
  
 new AlertDialog.Builder(this)  
 .setTitle("Sửa sinh viên")  
 .setView(dialogView)  
 .setPositiveButton("Cập nhật", (dialog, which) -> {  
 String name = edtName.getText().toString().trim();  
 String email = edtEmail.getText().toString().trim();  
 String phone = edtPhone.getText().toString().trim();  
  
 if (!name.isEmpty()) {  
 student.setName(name);  
 student.setEmail(email);  
 student.setPhone(phone);  
  
 db.updateStudent(student);  
 students.set(position, student); // update list tại vị trí  
 adapter.notifyItemChanged(position);  
  
 Toast.*makeText*(this, "Đã cập nhật", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Tên không được bỏ trống", Toast.*LENGTH\_SHORT*).show();  
 }  
 })  
 .setNegativeButton("Hủy", null)  
 .show();  
 }  
}

Bước 6: Ta sẽ có kết quả này

Yêu cầu 1: Tạo 1 CSDL SQLite, thêm vào CSDL danh sách các user, sau đó lấy dữ liệu từ CSDL hiện lên listview

Bước 1: Ta code nội dung trong file activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <ListView  
 android:id="@+id/lv\_user"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
</LinearLayout>

Bước 2: Tạo 1 file tên là item\_user.xml và gõ lệnh code sau

<?xml version="1.0" encoding="utf-8"?>  
<TextView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/tv\_user"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:padding="16dp"  
 android:textSize="18sp"/>

Bước 3: Tạo 1 class tên là DatabaseHelper gồm 2 phương thức onCreate() và onUpgrade() để quản lý việc tạo CSDL và version của CSDL

package com.example.a1150070035\_huynhngocphuc\_bai3.helper;  
  
import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
public class DatabaseHelper extends SQLiteOpenHelper {  
 private static final String *DATABASE\_CREATE* =  
 "CREATE TABLE users (\_id INTEGER PRIMARY KEY AUTOINCREMENT, " +  
 "name TEXT NOT NULL);";  
  
 public DatabaseHelper(Context context, String name,  
 SQLiteDatabase.CursorFactory factory, int version) {  
 super(context, name, factory, version);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL(*DATABASE\_CREATE*);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS users");  
 onCreate(db);  
 }  
}

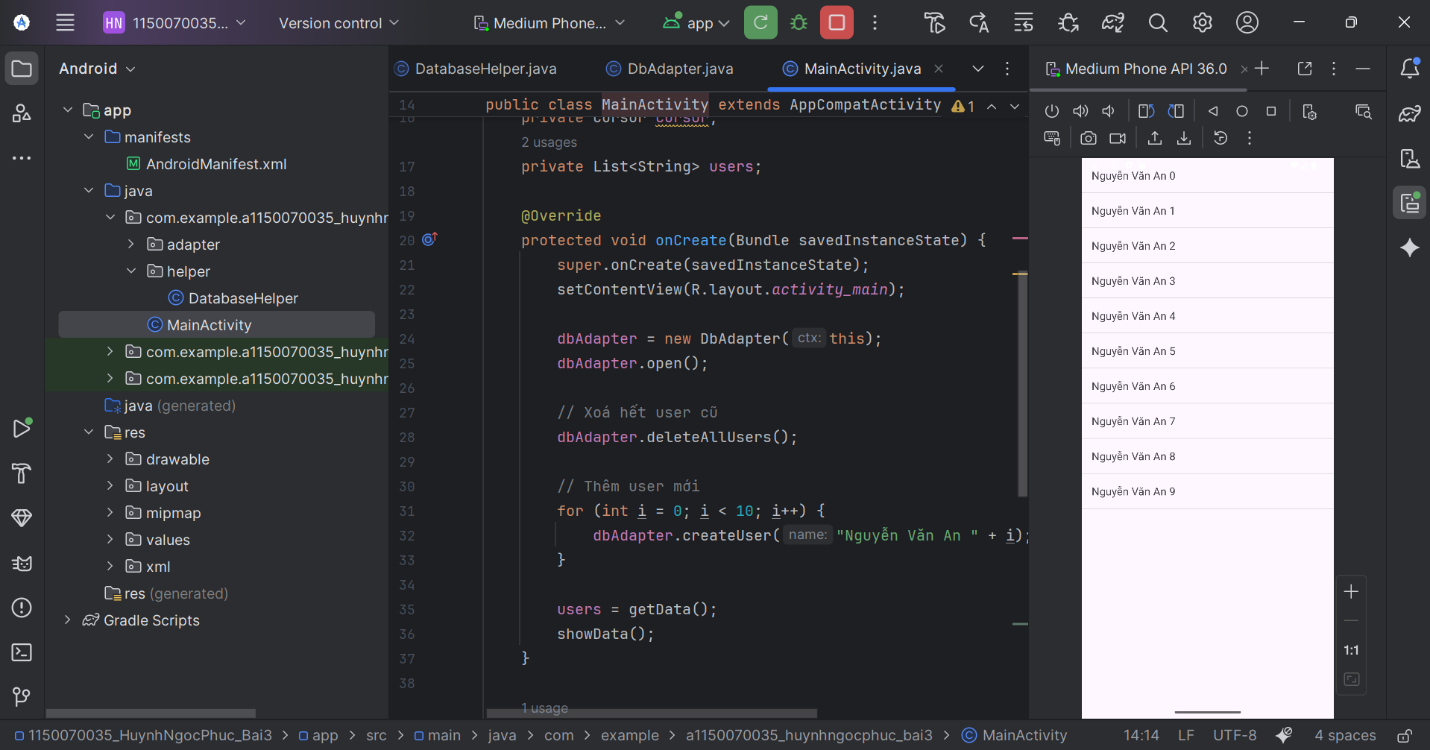
Bước 4: Add DbAdapter trong Java để quản lý các thao tác cơ bản với cơ sở dữ liệu SQLite, bao gồm các chức năng mở, đóng, thêm, xóa và lấy dữ liệu người dùng.

package com.example.a1150070035\_huynhngocphuc\_bai3.adapter;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.helper.DatabaseHelper;  
  
public class DbAdapter {  
 public static final String *KEY\_ID* = "\_id";  
 public static final String *KEY\_NAME* = "name";  
  
 private DatabaseHelper dbHelper;  
 private SQLiteDatabase sqLiteDatabase;  
  
 private static final String *DATABASE\_NAME* = "Database\_Demo";  
 private static final String *DATABASE\_TABLE* = "users";  
 private static final int *DATABASE\_VERSION* = 1;  
  
 private final Context context;  
  
 public DbAdapter(Context ctx) {  
 this.context = ctx;  
 }  
  
 public DbAdapter open() {  
 dbHelper = new DatabaseHelper(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 sqLiteDatabase = dbHelper.getWritableDatabase();  
 return this;  
 }  
  
 public void close() {  
 dbHelper.close();  
 }  
  
 public long createUser(String name) {  
 ContentValues inititalValues = new ContentValues();  
 inititalValues.put(*KEY\_NAME*, name);  
 return sqLiteDatabase.insert(*DATABASE\_TABLE*, null, inititalValues);  
 }  
  
 public boolean deleteAllUsers() {  
 return sqLiteDatabase.delete(*DATABASE\_TABLE*, null, null) > 0;  
 }  
  
 public Cursor getAllUsers() {  
 return sqLiteDatabase.query(*DATABASE\_TABLE*,  
 new String[]{*KEY\_ID*, *KEY\_NAME*},  
 null, null, null, null, null);  
 }  
}

Bước 5: Trong mục MainActivity

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.database.Cursor;  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.adapter.DbAdapter;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private DbAdapter dbAdapter;  
 private Cursor cursor;  
 private List<String> users;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 dbAdapter = new DbAdapter(this);  
 dbAdapter.open();  
  
 // Xoá hết user cũ  
 dbAdapter.deleteAllUsers();  
  
 // Thêm user mới  
 for (int i = 0; i < 10; i++) {  
 dbAdapter.createUser("Nguyễn Văn An " + i);  
 }  
  
 users = getData();  
 showData();  
 }  
  
 private List<String> getData() {  
 List<String> users = new ArrayList<>();  
 cursor = dbAdapter.getAllUsers();  
 while (cursor.moveToNext()) {  
 users.add(cursor.getString(cursor.getColumnIndexOrThrow(DbAdapter.*KEY\_NAME*)));  
 }  
 cursor.close();  
 return users;  
 }  
  
 private void showData() {  
 ListView lvUser = findViewById(R.id.*lv\_user*);  
 ArrayAdapter<String> userAdapter =  
 new ArrayAdapter<>(MainActivity.this, R.layout.*item\_user*, users);  
 lvUser.setAdapter(userAdapter);  
 }  
}

Bước 6: Ta sẽ hiện ra màn hình như sau



Yêu cầu 2: Quản lý Contact

Bước 1: Tạo 1 lớp tên là Contact

package com.example.a1150070035\_huynhngocphuc\_bai3.model;  
  
public class Contact {  
 private int id;  
 private String name;  
 private String phoneNumber;  
  
 // Constructors  
 public Contact() {}  
  
 public Contact(int id, String name, String phoneNumber) {  
 this.id = id;  
 this.name = name;  
 this.phoneNumber = phoneNumber;  
 }  
  
 public Contact(String name, String phoneNumber) {  
 this.name = name;  
 this.phoneNumber = phoneNumber;  
 }  
  
 // Getter - Setter  
 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 getPhoneNumber() {  
 return phoneNumber;  
 }  
  
 public void setPhoneNumber(String phoneNumber) {  
 this.phoneNumber = phoneNumber;  
 }  
}

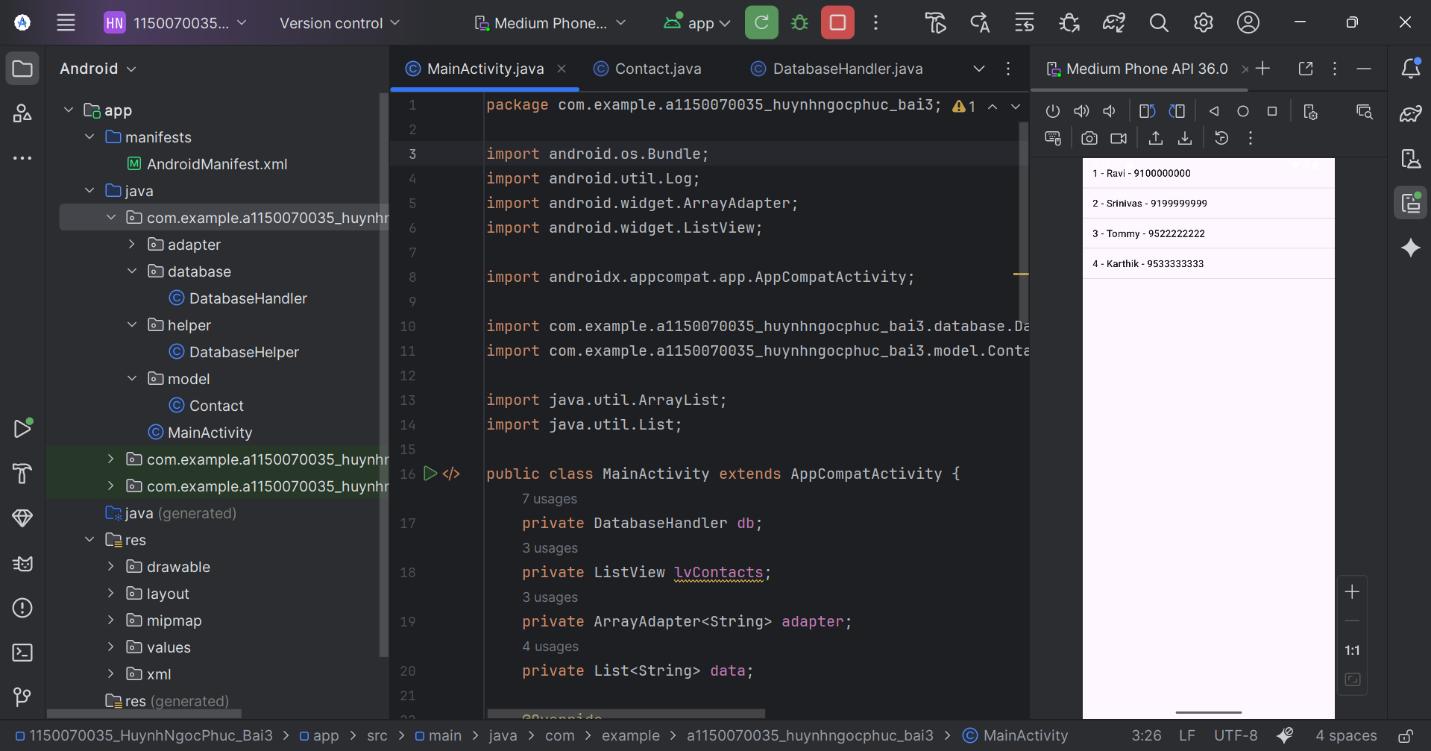
Bước 2: Tạo 1 lớp DatabaseHandler kế thừa từ SQLiteOpenHelper

package com.example.a1150070035\_huynhngocphuc\_bai3.database;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Contact;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class DatabaseHandler extends SQLiteOpenHelper {  
  
 // Database Info  
 private static final int *DATABASE\_VERSION* = 1;  
 private static final String *DATABASE\_NAME* = "contactsManager";  
 private static final String *TABLE\_CONTACTS* = "contacts";  
  
 // Column names  
 private static final String *KEY\_ID* = "id";  
 private static final String *KEY\_NAME* = "name";  
 private static final String *KEY\_PH\_NO* = "phone\_number";  
  
 public DatabaseHandler(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 }  
  
 // Create table  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 String CREATE\_CONTACTS\_TABLE = "CREATE TABLE " + *TABLE\_CONTACTS* + "("  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT,"  
 + *KEY\_NAME* + " TEXT,"  
 + *KEY\_PH\_NO* + " TEXT" + ")";  
 db.execSQL(CREATE\_CONTACTS\_TABLE);  
 }  
  
 // Upgrade table  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_CONTACTS*);  
 onCreate(db);  
 }  
  
 // Thêm contact  
 public void addContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, contact.getName());  
 values.put(*KEY\_PH\_NO*, contact.getPhoneNumber());  
  
 db.insert(*TABLE\_CONTACTS*, null, values);  
 db.close();  
 }  
  
 // Lấy 1 contact theo ID  
 public Contact getContact(int id) {  
 SQLiteDatabase db = this.getReadableDatabase();  
  
 Cursor cursor = db.query(*TABLE\_CONTACTS*, new String[]{*KEY\_ID*, *KEY\_NAME*, *KEY\_PH\_NO*},  
 *KEY\_ID* + "=?", new String[]{String.*valueOf*(id)}, null, null, null);  
  
 if (cursor != null)  
 cursor.moveToFirst();  
  
 Contact contact = new Contact(  
 cursor.getInt(0),  
 cursor.getString(1),  
 cursor.getString(2)  
 );  
  
 cursor.close();  
 return contact;  
 }  
  
 // Lấy tất cả contact  
 public List<Contact> getAllContacts() {  
 List<Contact> contactList = new ArrayList<>();  
 String selectQuery = "SELECT \* FROM " + *TABLE\_CONTACTS*;  
  
 SQLiteDatabase db = this.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, null);  
  
 if (cursor.moveToFirst()) {  
 do {  
 Contact contact = new Contact();  
 contact.setId(cursor.getInt(0));  
 contact.setName(cursor.getString(1));  
 contact.setPhoneNumber(cursor.getString(2));  
 contactList.add(contact);  
 } while (cursor.moveToNext());  
 }  
 cursor.close();  
 return contactList;  
 }  
  
 // Update contact  
 public int updateContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, contact.getName());  
 values.put(*KEY\_PH\_NO*, contact.getPhoneNumber());  
  
 return db.update(*TABLE\_CONTACTS*, values, *KEY\_ID* + " = ?",  
 new String[]{String.*valueOf*(contact.getId())});  
 }  
  
 // Xóa contact  
 public void deleteContact(Contact contact) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 db.delete(*TABLE\_CONTACTS*, *KEY\_ID* + " = ?",  
 new String[]{String.*valueOf*(contact.getId())});  
 db.close();  
 }  
}

Bước 3: Ở mục MainActivity ta sửa lại như sau

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.ArrayAdapter;  
import android.widget.ListView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.database.DatabaseHandler;  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Contact;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private DatabaseHandler db;  
 private ListView lvContacts;  
 private ArrayAdapter<String> adapter;  
 private List<String> data;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 lvContacts = findViewById(R.id.*lv\_user*); // tái sử dụng listview cũ  
 db = new DatabaseHandler(this);  
  
 // Xoá contact cũ (nếu muốn clean DB)  
 // getWritableDatabase().execSQL("DELETE FROM contacts");  
  
 // Thêm contact mẫu  
 db.addContact(new Contact("Ravi", "9100000000"));  
 db.addContact(new Contact("Srinivas", "9199999999"));  
 db.addContact(new Contact("Tommy", "9522222222"));  
 db.addContact(new Contact("Karthik", "9533333333"));  
  
 // Lấy toàn bộ contact  
 List<Contact> contacts = db.getAllContacts();  
  
 // Đưa dữ liệu vào List<String>  
 data = new ArrayList<>();  
 for (Contact c : contacts) {  
 String line = c.getId() + " - " + c.getName() + " - " + c.getPhoneNumber();  
 data.add(line);  
 Log.*d*("Contact", line);  
 }  
  
 // Adapter  
 adapter = new ArrayAdapter<>(this, android.R.layout.*simple\_list\_item\_1*, data);  
 lvContacts.setAdapter(adapter);  
  
 // Xoá contact khi long click  
 lvContacts.setOnItemLongClickListener((parent, view, position, id) -> {  
 Contact toDelete = contacts.get(position);  
 db.deleteContact(toDelete);  
 contacts.remove(position);  
 data.remove(position);  
 adapter.notifyDataSetChanged();  
 return true;  
 });  
 }  
}

Bước 4: Ta sẽ có kết quả như này



Yêu cầu 3: 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 dung Recyclerview và SQLite. Thiết kế giao diện theo cảm nhận của mỗi bạn.

Bước 1: Tạo 1 model Student

package com.example.a1150070035\_huynhngocphuc\_bai3.model;  
  
public class Student {  
 private int id;  
 private String name;  
 private String email;  
 private String phone;  
  
 public Student() {}  
  
 public Student(String name, String email, String phone) {  
 this.name = name;  
 this.email = email;  
 this.phone = phone;  
 }  
  
 public Student(int id, String name, String email, String phone) {  
 this.id = id;  
 this.name = name;  
 this.email = email;  
 this.phone = phone;  
 }  
  
 // Getter - Setter  
 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 getEmail() { return email; }  
 public void setEmail(String email) { this.email = email; }  
  
 public String getPhone() { return phone; }  
 public void setPhone(String phone) { this.phone = phone; }  
}

Bước 2: Tạo DatabaseHandler cho Student tên là StudentDatabaseHandler

package com.example.a1150070035\_huynhngocphuc\_bai3.database;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Student;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class StudentDatabaseHandler extends SQLiteOpenHelper {  
 private static final int *DATABASE\_VERSION* = 1;  
 private static final String *DATABASE\_NAME* = "studentManager";  
 private static final String *TABLE\_STUDENTS* = "students";  
  
 private static final String *KEY\_ID* = "id";  
 private static final String *KEY\_NAME* = "name";  
 private static final String *KEY\_EMAIL* = "email";  
 private static final String *KEY\_PHONE* = "phone";  
  
 public StudentDatabaseHandler(Context context) {  
 super(context, *DATABASE\_NAME*, null, *DATABASE\_VERSION*);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 String CREATE\_STUDENTS\_TABLE = "CREATE TABLE " + *TABLE\_STUDENTS* + "("  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT,"  
 + *KEY\_NAME* + " TEXT,"  
 + *KEY\_EMAIL* + " TEXT,"  
 + *KEY\_PHONE* + " TEXT" + ")";  
 db.execSQL(CREATE\_STUDENTS\_TABLE);  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_STUDENTS*);  
 onCreate(db);  
 }  
  
 // Thêm sinh viên  
 public void addStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, student.getName());  
 values.put(*KEY\_EMAIL*, student.getEmail());  
 values.put(*KEY\_PHONE*, student.getPhone());  
 db.insert(*TABLE\_STUDENTS*, null, values);  
 db.close();  
 }  
  
 // Lấy tất cả sinh viên  
 public List<Student> getAllStudents() {  
 List<Student> studentList = new ArrayList<>();  
 String selectQuery = "SELECT \* FROM " + *TABLE\_STUDENTS*;  
  
 SQLiteDatabase db = this.getReadableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, null);  
  
 if (cursor.moveToFirst()) {  
 do {  
 Student student = new Student();  
 student.setId(cursor.getInt(0));  
 student.setName(cursor.getString(1));  
 student.setEmail(cursor.getString(2));  
 student.setPhone(cursor.getString(3));  
 studentList.add(student);  
 } while (cursor.moveToNext());  
 }  
 cursor.close();  
 return studentList;  
 }  
  
 // Update sinh viên  
 public int updateStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put(*KEY\_NAME*, student.getName());  
 values.put(*KEY\_EMAIL*, student.getEmail());  
 values.put(*KEY\_PHONE*, student.getPhone());  
 return db.update(*TABLE\_STUDENTS*, values, *KEY\_ID* + "=?",  
 new String[]{String.*valueOf*(student.getId())});  
 }  
  
 // Xóa sinh viên  
 public void deleteStudent(Student student) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 db.delete(*TABLE\_STUDENTS*, *KEY\_ID* + "=?",  
 new String[]{String.*valueOf*(student.getId())});  
 db.close();  
 }  
}

Bước 3: Ở layout ta tạo activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical"  
 android:padding="16dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:id="@+id/btnAdd"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Thêm sinh viên" />  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/recyclerViewStudents"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingTop="8dp"/>  
</LinearLayout>

Bước 4: Tạo 1 layout tên là dialog\_student.xml để chỉnh sửa

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <EditText  
 android:id="@+id/edtName"  
 android:hint="Tên sinh viên"  
 android:inputType="textPersonName"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"/>  
  
 <EditText  
 android:id="@+id/edtEmail"  
 android:hint="Email"  
 android:inputType="textEmailAddress"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"/>  
  
 <EditText  
 android:id="@+id/edtPhone"  
 android:hint="Số điện thoại"  
 android:inputType="phone"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"/>  
</LinearLayout>

Bước 5: Ở mục MainActivity sửa lại

package com.example.a1150070035\_huynhngocphuc\_bai3;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import com.example.a1150070035\_huynhngocphuc\_bai3.adapter.StudentAdapter;  
import com.example.a1150070035\_huynhngocphuc\_bai3.database.StudentDatabaseHandler;  
import com.example.a1150070035\_huynhngocphuc\_bai3.model.Student;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
 private StudentDatabaseHandler db;  
 private StudentAdapter adapter;  
 private List<Student> students;  
 private RecyclerView recyclerView;  
 private Button btnAdd;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 db = new StudentDatabaseHandler(this);  
 recyclerView = findViewById(R.id.*recyclerViewStudents*);  
 btnAdd = findViewById(R.id.*btnAdd*);  
  
 // Load dữ liệu từ DB  
 students = new ArrayList<>(db.getAllStudents());  
  
 // Adapter + RecyclerView  
 adapter = new StudentAdapter(this, students, new StudentAdapter.OnStudentClickListener() {  
 @Override  
 public void onStudentClick(Student student, int position) {  
 showEditStudentDialog(student, position); // nhấn để sửa  
 }  
  
 @Override  
 public void onStudentLongClick(Student student, int position) {  
 db.deleteStudent(student);  
 students.remove(position); // xoá chính xác theo index  
 adapter.notifyItemRemoved(position);  
 Toast.*makeText*(MainActivity.this, "Đã xoá " + student.getName(), Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 recyclerView.setLayoutManager(new LinearLayoutManager(this));  
 recyclerView.setAdapter(adapter);  
  
 // Nút thêm sinh viên  
 btnAdd.setOnClickListener(v -> showAddStudentDialog());  
 }  
  
 // Dialog thêm sinh viên  
 private void showAddStudentDialog() {  
 View dialogView = getLayoutInflater().inflate(R.layout.*dialog\_student*, null);  
  
 EditText edtName = dialogView.findViewById(R.id.*edtName*);  
 EditText edtEmail = dialogView.findViewById(R.id.*edtEmail*);  
 EditText edtPhone = dialogView.findViewById(R.id.*edtPhone*);  
  
 new AlertDialog.Builder(this)  
 .setTitle("Thêm sinh viên")  
 .setView(dialogView)  
 .setPositiveButton("Thêm", (dialog, which) -> {  
 String name = edtName.getText().toString().trim();  
 String email = edtEmail.getText().toString().trim();  
 String phone = edtPhone.getText().toString().trim();  
  
 if (!name.isEmpty()) {  
 Student student = new Student(name, email, phone);  
 db.addStudent(student);  
  
 // Reload list  
 students.clear();  
 students.addAll(db.getAllStudents());  
 adapter.notifyDataSetChanged();  
  
 Toast.*makeText*(this, "Đã thêm sinh viên", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Tên không được bỏ trống", Toast.*LENGTH\_SHORT*).show();  
 }  
 })  
 .setNegativeButton("Hủy", null)  
 .show();  
 }  
  
 // Dialog sửa sinh viên  
 private void showEditStudentDialog(Student student, int position) {  
 View dialogView = getLayoutInflater().inflate(R.layout.*dialog\_student*, null);  
  
 EditText edtName = dialogView.findViewById(R.id.*edtName*);  
 EditText edtEmail = dialogView.findViewById(R.id.*edtEmail*);  
 EditText edtPhone = dialogView.findViewById(R.id.*edtPhone*);  
  
 // Gán dữ liệu cũ  
 edtName.setText(student.getName());  
 edtEmail.setText(student.getEmail());  
 edtPhone.setText(student.getPhone());  
  
 new AlertDialog.Builder(this)  
 .setTitle("Sửa sinh viên")  
 .setView(dialogView)  
 .setPositiveButton("Cập nhật", (dialog, which) -> {  
 String name = edtName.getText().toString().trim();  
 String email = edtEmail.getText().toString().trim();  
 String phone = edtPhone.getText().toString().trim();  
  
 if (!name.isEmpty()) {  
 student.setName(name);  
 student.setEmail(email);  
 student.setPhone(phone);  
  
 db.updateStudent(student);  
 students.set(position, student); // update list tại vị trí  
 adapter.notifyItemChanged(position);  
  
 Toast.*makeText*(this, "Đã cập nhật", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Tên không được bỏ trống", Toast.*LENGTH\_SHORT*).show();  
 }  
 })  
 .setNegativeButton("Hủy", null)  
 .show();  
 }  
}

Bước 6: Ta sẽ có kết quả này

