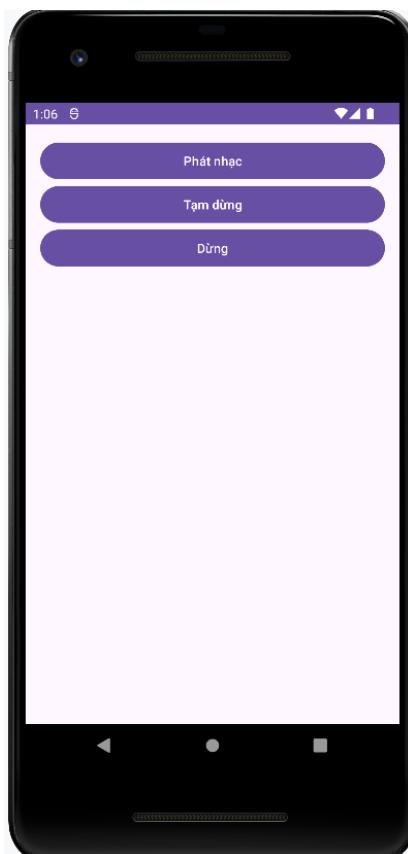


MỘT SỐ BÀI TẬP THAM KHẢO

A. Nhóm bài tập về phát nhạc/ âm thanh:

Bài 1: Phát nhạc từ tập tin nội bộ (MediaPlayer)

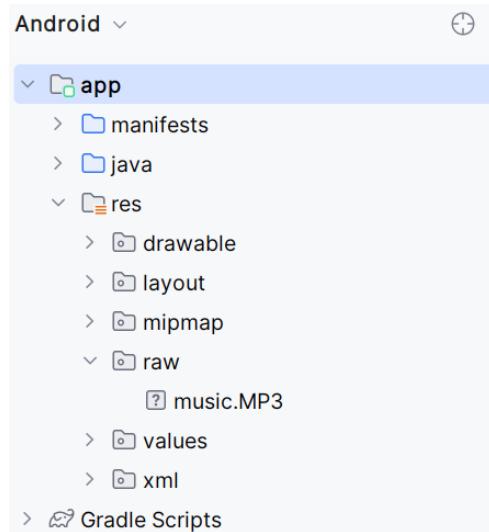
- Tạo ứng dụng có nút Phát, Tạm dừng, Dừng để điều khiển nhạc.
- Nhạc được phát từ tệp trong thư mục res/raw.



Hình 1. Bài tập 2.1

Hướng dẫn thực hiện

Chuẩn bị file âm thanh một file mp3 vào thư mục res/raw/ (nếu chưa có, tạo thư mục raw trong res). Đổi tên tập tin thành music.mp3 (hoặc tên khác).



Tạo giao diện trong activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <Button
        android:id="@+id/btnPlay"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Phát nhạc" />
    <Button
        android:id="@+id/btnPause"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Tạm dừng" />
    <Button
        android:id="@+id/btnStop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Dừng" />
</LinearLayout>
```

Xử lý phát nhạc trong MainActivity.java

```

import android.media.MediaPlayer;
import android.os.Bundle;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    9 usages
    MediaPlayer mediaPlayer;

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

        Button btnPlay = findViewById(R.id.btnPlay);
        Button btnPause = findViewById(R.id.btnPause);
        Button btnStop = findViewById(R.id.btnStop);

        mediaPlayer = MediaPlayer.create(context: this, R.raw.music);
    }
}

```

Viết xử lý cho các button

```

btnPlay.setOnClickListener(view -> {
    if (!mediaPlayer.isPlaying()) {
        mediaPlayer.start();
    }
});

btnPause.setOnClickListener(view -> {
    if (mediaPlayer.isPlaying()) {
        mediaPlayer.pause();
    }
});

btnStop.setOnClickListener(view -> {
    if (mediaPlayer.isPlaying()) {
        mediaPlayer.stop();
        mediaPlayer.release();
        mediaPlayer = MediaPlayer.create(context: this, R.raw.music);
    }
});

```

Chạy chương trình. Nhấn "Phát nhạc" để nghe nhạc, "Tạm dừng" để dừng tạm thời, "Dừng" để dừng hẳn và phát lại từ đầu.

Bài 2: Phát nhạc từ URL (Streaming Audio)

- Ứng dụng phát nhạc từ một URL trực tuyến.

- Có nút Phát, Dừng.



Hình 2. Bài tập 2.2

Hướng dẫn thực hiện

Thêm quyền internet trong AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET"/>
```

Tạo giao diện trong activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <Button
        android:id="@+id	btnPlay"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Phát nhạc từ URL" />

    <Button
        android:id="@+id	btnStop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Dừng" />
</LinearLayout>
```

Xử lý phát nhạc từ URL trong MainActivity.java

```
import android.media.MediaPlayer;
import android.os.Bundle;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {
    9 usages
    MediaPlayer mediaPlayer;
    // Thay bằng URL nhạc
    1 usage
    String audioUrl = "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3";
```

Thay URL bằng link nhạc tùy chọn.

Trong onCreate, viết xử lý cho các Button

```

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

    Button btnPlay = findViewById(R.id.btnPlay);
    Button btnStop = findViewById(R.id.btnStop);

    btnPlay.setOnClickListener(view -> {
        if (mediaPlayer == null) {
            mediaPlayer = new MediaPlayer();
            try {
                mediaPlayer.setDataSource(audioUrl);
                mediaPlayer.prepareAsync();
                mediaPlayer.setOnPreparedListener(MediaPlayer::start);
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    });

    btnStop.setOnClickListener(view -> {
        if (mediaPlayer != null) {
            mediaPlayer.stop();
            mediaPlayer.release();
            mediaPlayer = null;
        }
    });
}

```

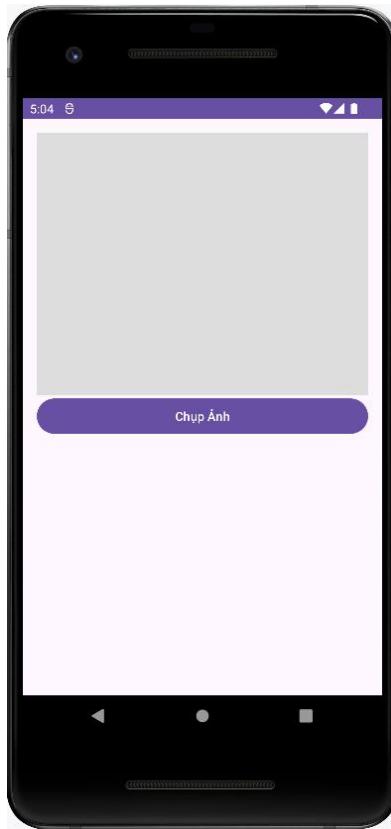
Chạy chương trình. Nhạc sẽ phát từ URL khi nhấn **"Phát nhạc từ URL"** và dừng khi nhấn **"Đừng"**.

B. Nhóm bài tập về hình ảnh

Bài 1: Chụp ảnh bằng Camera và hiển thị lên ImageView

- Ứng dụng có nút **Chụp ảnh**.

- Mở camera để chụp ảnh.
- Hiển thị ảnh lên ImageView.



Hình 3. Bài tập 3.1

Hướng dẫn thực hiện

Thêm quyền camera trong AndroidManifest.xml

```
<uses-permission android:name="android.permission.CAMERA"/>
<uses-feature android:name="android.hardware.camera" android:required="true"/>
```

Tạo giao diện trong activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
```

```
<ImageView  
    android:id="@+id/imgPreview"  
    android:layout_width="match_parent"  
    android:layout_height="300dp"  
    android:scaleType="centerCrop"  
    android:background="#ddd"/>  
  
<Button  
    android:id="@+id	btnCapture"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Chụp Ảnh" />  
</LinearLayout>
```

Xử lý chụp ảnh trong MainActivity.java

```
import android.app.Activity;  
import android.content.Intent;  
import android.graphics.Bitmap;  
import android.os.Bundle;  
import android.provider.MediaStore;  
import android.widget.Button;  
import android.widget.ImageView;  
import androidx.activity.result.ActivityResult;  
import androidx.activity.result.contract.ActivityResultContracts;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
    2 usages  
    ImageView imgPreview;  
    2 usages  
    ActivityResultLauncher<Intent> cameraLauncher;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        imgPreview = findViewById(R.id.imgPreview);  
        Button btnCapture = findViewById(R.id.btnCapture);
```

```

// Đăng ký launcher để xử lý kết quả trả về từ camera
cameraLauncher = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    result -> {
        if (result.getResultCode() == Activity.RESULT_OK && result.getData() != null) {
            Bitmap imageBitmap = (Bitmap) result.getData().getExtras().get("data");
            imgPreview.setImageBitmap(imageBitmap);
        }
    }
);

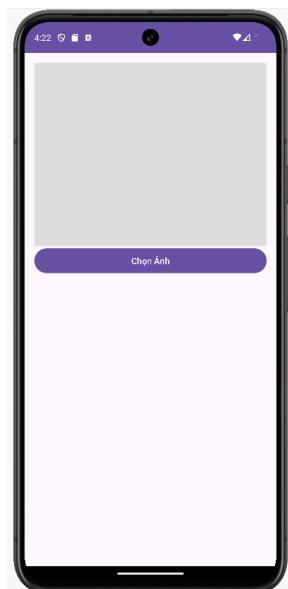
// Mở camera khi nhấn nút
btnCapture.setOnClickListener(view -> {
    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    cameraLauncher.launch(intent);
});
}
}

```

Chạy chương trình. Nhấn "**Chụp Ảnh**" → Mở Camera → Chụp ảnh → Hiển thị lên ImageView.

Bài 2: Chọn ảnh từ thư viện và hiển thị lên ImageView

- Ứng dụng có nút **Chọn ảnh**.
- Người dùng chọn ảnh từ thư viện.
- Hiển thị ảnh lên ImageView.



Hình 4. Bài tập 3.2

Hướng dẫn thực hiện

Thêm quyền đọc bộ nhớ (AndroidManifest.xml)

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
```

Tạo giao diện (activity_main.xml)

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <ImageView
        android:id="@+id/imgSelected"
        android:layout_width="match_parent"
        android:layout_height="300dp"
        android:scaleType="centerCrop"
        android:background="#ddd"/>
    <Button
        android:id="@+id/btnSelect"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Chọn Ảnh" />
</LinearLayout>
```

Xử lý chọn ảnh từ thư viện (MainActivity.java)

```
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.widget.Button;
import android.widget.ImageView;
import androidx.activity.result.ActivityResultLauncher;
import androidx.activity.result.contract.ActivityResultContracts;
import androidx.appcompat.app.AppCompatActivity;
```

```

public class MainActivity extends AppCompatActivity {
    2 usages
    ImageView imgSelected;
    2 usages
    ActivityResultLauncher<Intent> imagePickerLauncher;

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

        imgSelected = findViewById(R.id.imgSelected);
        Button btnSelect = findViewById(R.id.btnSelect);

        // Khởi tạo launcher để nhận kết quả chọn ảnh
        imagePickerLauncher = registerForActivityResult(
            new ActivityResultContracts.StartActivityForResult(),
            result -> {
                if (result.getResultCode() == RESULT_OK && result.getData() != null) {
                    Uri imageUri = result.getData().getData();
                    imgSelected.setImageURI(imageUri);
                }
            }
        );
    }

    // Mở trình chọn ảnh khi nhấn nút
    btnSelect.setOnClickListener(view -> {
        Intent intent = new Intent(Intent.ACTION_PICK);
        intent.setType("image/*");
        imagePickerLauncher.launch(intent);
    });
}
}

```

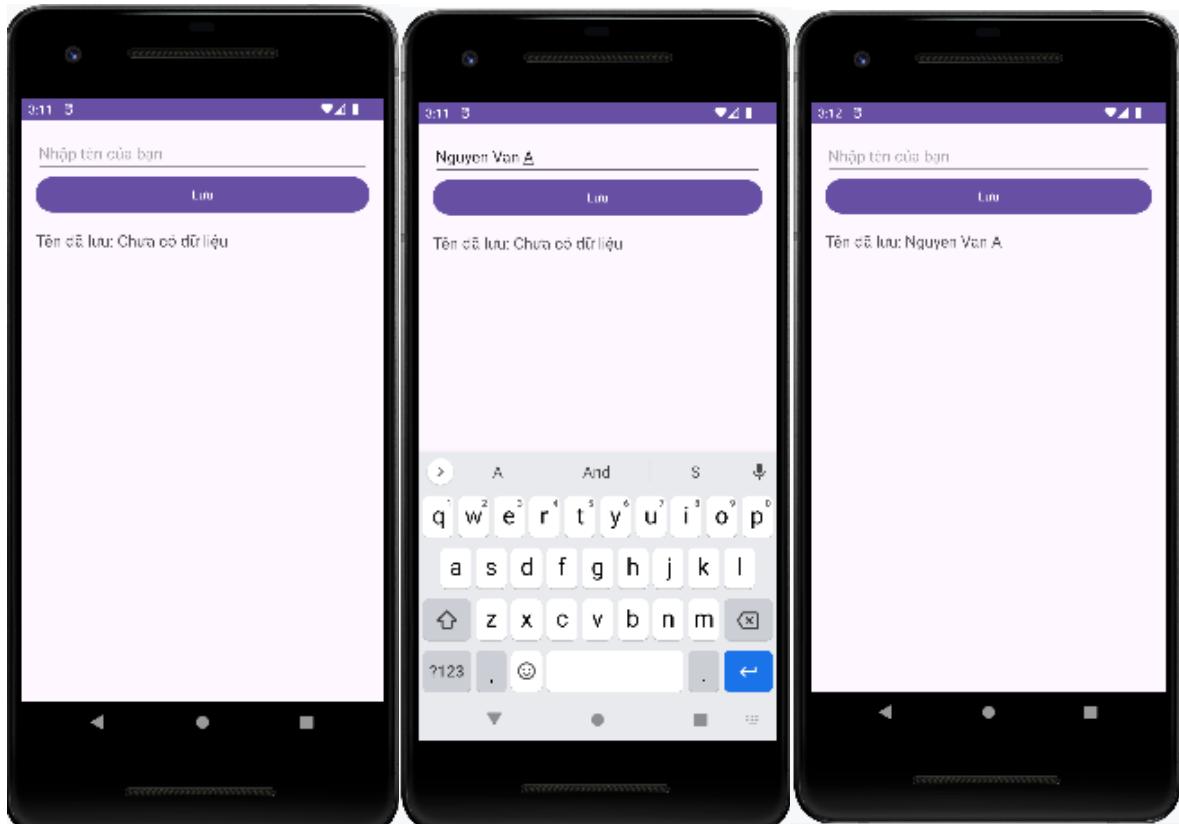
Chạy chương trình. Nhấn "**Chọn Ảnh**" → Mở thư viện ảnh → Chọn ảnh → Hiển thị lên ImageView.

C. Nhóm bài tập về lưu trữ dữ liệu

Bài 1: Lưu và đọc dữ liệu với SharedPreferences

Tạo một ứng dụng đơn giản cho phép người dùng nhập tên và lưu vào SharedPreferences.

Khi mở lại ứng dụng, tên sẽ được tự động hiển thị.



Hình 5. Bài tập 1.1

Hướng dẫn thực hiện

Xây dựng giao diện trong activity_main.xml, bao gồm

- Một EditText để nhập tên.
- Một Button để lưu dữ liệu.
- Một TextView để hiển thị dữ liệu đã lưu.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <EditText
        android:id="@+id edtName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nhập tên của bạn" />
    <Button
        android:id="@+id btnSave"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Lưu" />
    <TextView
        android:id="@+id txtResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:paddingTop="16dp"/>
</LinearLayout>
```

Xử lý lưu và đọc dữ liệu trong MainActivity.java

```
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    2 usages
    EditText edtName;
    2 usages
    Button btnSave;
    3 usages
    TextView txtResult;
    3 usages
    SharedPreferences sharedPreferences;
```

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

    edtName = findViewById(R.id.edtName);
    btnSave = findViewById(R.id.btnSave);
    txtResult = findViewById(R.id.txtResult);

    sharedpreferences = getSharedPreferences( name: "UserData", MODE_PRIVATE);
```

Sử dụng đối tượng SharedPreferences:

```
// Đọc dữ liệu khi mở lại app
String savedName = sharedpreferences.getString( key: "name", defaultValue: "Chưa có dữ liệu");
txtResult.setText("Tên đã lưu: " + savedName);

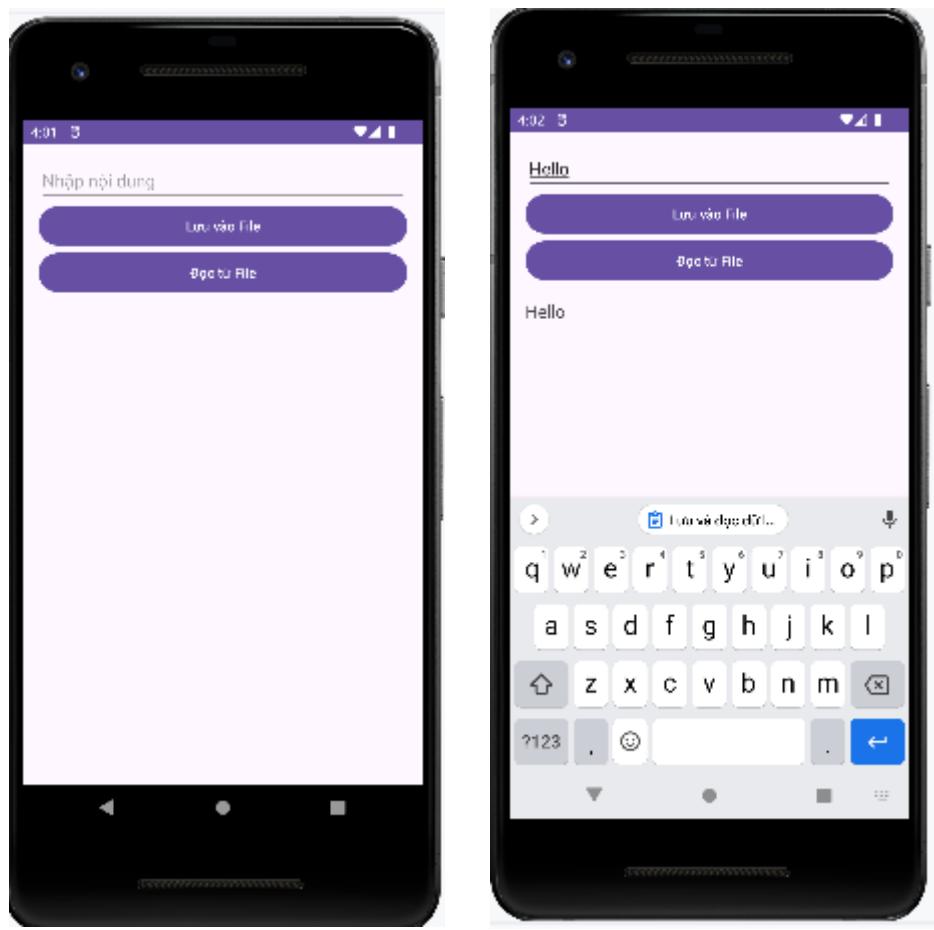
btnSave.setOnClickListener(view -> {
    String name = edtName.getText().toString();
    SharedPreferences.Editor editor = sharedpreferences.edit();
    editor.putString("name", name);
    editor.apply();

    txtResult.setText("Tên đã lưu: " + name);
});
```

Chạy thử chương trình. Khi nhập tên vào EditText và nhấn "Lưu", dữ liệu sẽ được lưu. Khi mở lại ứng dụng, tên sẽ hiển thị từ SharedPreferences.

Bài 2: Lưu và đọc dữ liệu từ file nội bộ

Tạo ứng dụng cho phép nhập nội dung và lưu vào một file trên bộ nhớ trong. Khi mở lại ứng dụng, nội dung sẽ được hiển thị.



Hình 6. Bài tập 1.2

Hướng dẫn thực hiện

Tạo giao diện trong activity_main.xml gồm

- Một EditText để nhập nội dung.
- Hai Button để lưu và đọc dữ liệu từ file.
- Một TextView để hiển thị nội dung đọc được.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <EditText
        android:id="@+id/edtContent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nhập nội dung" />
    <Button
        android:id="@+id	btnSave"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Lưu vào File" />
    <Button
        android:id="@+id	btnRead"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Đọc từ File" />
    <TextView
        android:id="@+id/txtResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:paddingTop="16dp"/>
</LinearLayout>
```

Xử lý lưu và đọc dữ liệu trong MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import java.io.*;
```

```
public class MainActivity extends AppCompatActivity {  
    2 usages  
    EditText edtContent;  
    2 usages  
    Button btnSave, btnRead;  
    2 usages  
    TextView txtResult;  
    2 usages  
    final String FILE_NAME = "data.txt";  
  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        edtContent = findViewById(R.id.edtContent);  
        btnSave = findViewById(R.id.btnSave);  
        btnRead = findViewById(R.id.btnRead);  
        txtResult = findViewById(R.id.txtResult);  
  
        btnSave.setOnClickListener(view -> saveToFile(edtContent.getText().toString()));  
        btnRead.setOnClickListener(view -> txtResult.setText(readFromFile()));  
    }  
}
```

Xử lý lưu file

```
private void saveToFile(String data) {  
    try (FileOutputStream fos = openFileOutput(FILE_NAME, MODE_PRIVATE);  
         OutputStreamWriter writer = new OutputStreamWriter(fos)) {  
        writer.write(data);  
    } catch (IOException e) {  
        e.printStackTrace();  
    }  
}
```

Xử lý đọc file

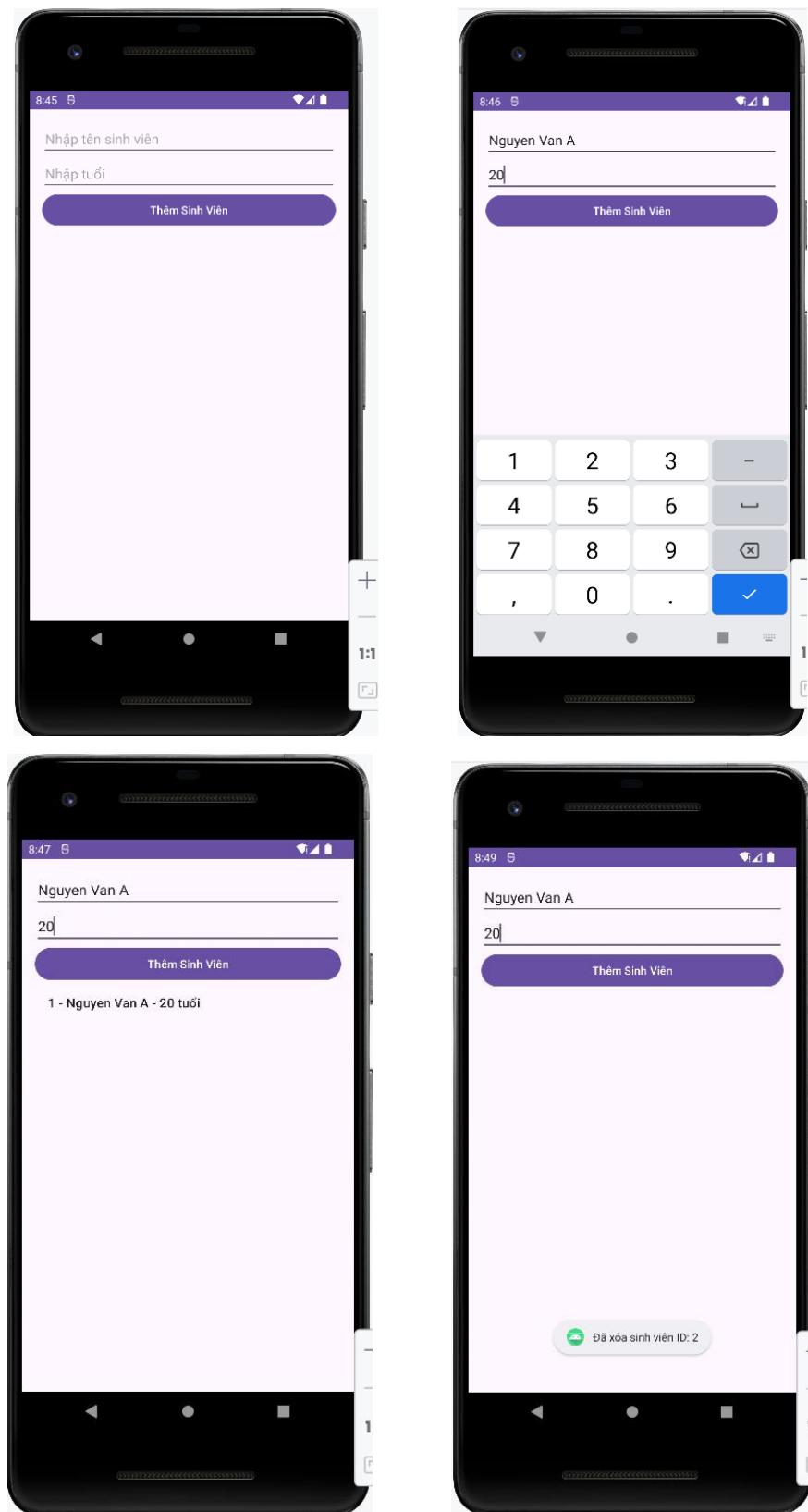
```
private String readFromFile() {  
    StringBuilder stringBuilder = new StringBuilder();  
    try (FileInputStream fis = openFileInput(FILE_NAME);  
        InputStreamReader reader = new InputStreamReader(fis);  
        BufferedReader br = new BufferedReader(reader)) {  
  
        String line;  
        while ((line = br.readLine()) != null) {  
            stringBuilder.append(line).append("\n");  
        }  
    } catch (IOException e) {  
        e.printStackTrace();  
    }  
    return stringBuilder.toString();  
}
```

Chạy chương trình. Khi nhập nội dung vào EditText và nhấn "Lưu vào File", nội dung sẽ được lưu vào file. Khi nhấn "Đọc từ File", nội dung sẽ được đọc và hiển thị lên TextView.

Bài 3: Lưu và truy xuất dữ liệu với SQLite

Tạo ứng dụng quản lý danh sách sinh viên bằng **SQLite**. Người dùng có thể:

- Thêm sinh viên (tên, tuổi).
- Xem danh sách sinh viên.
- Xóa một sinh viên theo ID.



Hình 7. Bài tập 1.3

Hướng dẫn thực hiện

Tạo giao diện trong activity_main.xml. Giao diện gồm:

- 2 EditText để nhập tên và tuổi.
- 1 Button để thêm sinh viên.
- 1 ListView để hiển thị danh sách.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <EditText
        android:id="@+id/edtName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nhập tên sinh viên" />
    <EditText
        android:id="@+id/edtAge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nhập tuổi"
        android:inputType="number" />
    <Button
        android:id="@+id/btnAdd"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Thêm Sinh Viên" />
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

Tạo lớp hỗ trợ SQLite trong DatabaseHelper.java. Lớp này giúp quản lý CSDL SQLite:

- Tạo bảng students.
- Thêm dữ liệu vào bảng.

- Lấy danh sách sinh viên.
- Xóa sinh viên theo ID.

```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;

2 usages
public class DatabaseHelper extends SQLiteOpenHelper {
    1 usage
    private static final String DATABASE_NAME = "student_db";
    1 usage
    private static final int DATABASE_VERSION = 1;

    5 usages
    private static final String TABLE_NAME = "students";
    2 usages
    private static final String COLUMN_ID = "id";
    2 usages
    private static final String COLUMN_NAME = "name";
    2 usages
    private static final String COLUMN_AGE = "age";

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

    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME + " (" +
            COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            COLUMN_NAME + " TEXT, " +
            COLUMN_AGE + " INTEGER)";
        db.execSQL(createTable);
    }
}

```

```
10 usages
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
    onCreate(db);
}
```

Thao tác thêm mới 1 sinh viên

```
public void addStudent(String name, int age) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COLUMN_NAME, name);
    values.put(COLUMN_AGE, age);

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

Thao tác lấy danh sách sinh viên

```
public ArrayList<String> getAllStudents() {
    ArrayList<String> studentList = new ArrayList<>();
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery(sql: "SELECT * FROM " + TABLE_NAME, selectionArgs: null);

    if (cursor.moveToFirst()) {
        do {
            int id = cursor.getInt(columnIndex: 0);
            String name = cursor.getString(columnIndex: 1);
            int age = cursor.getInt(columnIndex: 2);
            studentList.add(id + " - " + name + " - " + age + " tuổi");
        } while (cursor.moveToNext());
    }
    cursor.close();
    return studentList;
}
```

Thao tác xóa một sinh viên (theo mã sinh viên)

```

    public void deleteStudent(int id) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_NAME, whereClause: COLUMN_ID + " = ?", new String[]{String.valueOf(id)});
        db.close();
    }
}

```

Xử lý thêm, hiển thị và xóa dữ liệu trong MainActivity.java

```

public class MainActivity extends AppCompatActivity {
    2 usages
    EditText edtName, edtAge;
    2 usages
    Button btnAdd;
    3 usages
    ListView listView;
    4 usages
    DatabaseHelper databaseHelper;
    3 usages
    ArrayList<String> studentList;
    2 usages
    ArrayAdapter<String> adapter;

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

        edtName = findViewById(R.id.edtName);
        edtAge = findViewById(R.id.edtAge);
        btnAdd = findViewById(R.id.btnAdd);
        listView = findViewById(R.id.listView);

        databaseHelper = new DatabaseHelper(context: this);
        loadStudentList();

        btnAdd.setOnClickListener(view -> {
            String name = edtName.getText().toString();
            int age = Integer.parseInt(edtAge.getText().toString());
            databaseHelper.addStudent(name, age);
            loadStudentList();
        });
    }
}

```

```

listView.setOnItemClickListener((parent, view, position, id) -> {
    String selectedItem = studentList.get(position);
    String[] parts = selectedItem.split( regex: " - ");
    int studentId = Integer.parseInt(parts[0]);

    databaseHelper.deleteStudent(studentId);
    loadStudentList();
    Toast.makeText( context: MainActivity.this, text: "Đã xóa sinh viên ID: " +
        studentId, Toast.LENGTH_SHORT).show();
});
}

3 usages
private void loadStudentList() {
    studentList = databaseHelper.getAllStudents();
    adapter = new ArrayAdapter<>( context: this, android.R.layout.simple_list_item_1, studentList);
    listView.setAdapter(adapter);
}

```

Chạy chương trình và xem kết quả.

- Nhập tên và tuổi → Nhận "Thêm Sinh Viên".
- Danh sách hiển thị ID - Tên - Tuổi.
- Nhấn vào sinh viên bất kỳ để xóa khỏi danh sách.