**Bài tập lí thuyết lab5**

**Bài 1: Viết ứng dụng MultiThread sử dụng Message**

**Bước 1:** Tạo **project** mới **–** đặt tên **lab5**

**Bước 2:** Ở **value** tạo file **dimens.xml**

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Nội dung do AI tạo ra có thể không chính xác.**

\* code **dimens.xml**

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <dimen name="margin\_base">5dp</dimen>  
 <dimen name="text\_small">14sp</dimen>  
 <dimen name="text\_medium">16sp</dimen>  
 <dimen name="text\_medium\_large">18sp</dimen>  
 <dimen name="text\_large">20sp</dimen>  
</resources>

**\*** code **strings.xml**

<resources>  
 <string name="start">Start</string>  
 <string name="returned\_by\_bg\_thread">Returned by background thread: \n\n</string>  
 <string name="done\_background\_thread\_has\_been\_stopped">Done \nBackground thread has been stopped</string>  
 <string name="done">Done</string>  
 <string name="working">Working...</string>  
 <string name="global\_value\_seen">\n global value seen by all thread</string>  
</resources>

**Bước 3: Ở layout**

**\*** code **activity\_main.xml** như sau

<?xml version="1.0" encoding="utf-8"?>  
<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:orientation="vertical"  
 android:padding="@dimen/margin\_base"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/tv\_working"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="@dimen/text\_medium\_large"  
 android:textStyle="bold" />  
  
 <ProgressBar  
 android:id="@+id/pb\_first"  
 style="?android:attr/progressBarStyleHorizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="@dimen/margin\_base" />  
  
 <TextView  
 android:id="@+id/tv\_return"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="@dimen/text\_medium" />  
  
 <FrameLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <Button  
 android:id="@+id/btn\_start"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="@string/start" />  
  
 <ProgressBar  
 android:id="@+id/pb\_second"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="@dimen/margin\_base" />  
 </FrameLayout>  
</LinearLayout>

**Bước 4:** Trong **MainActivity.java**

**\*** code như sau

package com.example.lab5;  
  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.Looper;  
import android.os.Message;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ProgressBar;  
import android.widget.TextView;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import java.util.Random;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ProgressBar pbFirst, pbSecond;  
 private TextView tvMsgWorking, tvMsgReturned;  
 private Button btnStart;  
  
 private boolean isRunning;  
 private int MAX\_SEC = 20;  
 private int intTest;  
 private Thread bgThread;  
 private Handler handler;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 findViewByIds();  
 initVariables();  
  
 btnStart.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 isRunning = true;  
 pbFirst.setVisibility(View.*VISIBLE*);  
 pbSecond.setVisibility(View.*VISIBLE*);  
 btnStart.setVisibility(View.*GONE*);  
 bgThread.start();  
 }  
 });  
 }  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 initBgThread();  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 isRunning = false;  
 }  
  
 private void findViewByIds() {  
 pbFirst = findViewById(R.id.*pb\_first*);  
 pbSecond = findViewById(R.id.*pb\_second*);  
 tvMsgWorking = findViewById(R.id.*tv\_working*);  
 tvMsgReturned = findViewById(R.id.*tv\_return*);  
 btnStart = findViewById(R.id.*btn\_start*);  
 }  
  
 private void initVariables() {  
 isRunning = false;  
 intTest = 1;  
 pbFirst.setMax(MAX\_SEC);  
 pbFirst.setProgress(0);  
  
 pbFirst.setVisibility(View.*GONE*);  
 pbSecond.setVisibility(View.*GONE*);  
  
 handler = new Handler(Looper.*getMainLooper*()) {  
 @Override  
 public void handleMessage(@NonNull Message msg) {  
 super.handleMessage(msg);  
 String returnedValue = (String) msg.obj;  
 tvMsgReturned.setText(getString(R.string.*returned\_by\_bg\_thread*) + returnedValue);  
 pbFirst.incrementProgressBy(2);  
  
 if (pbFirst.getProgress() == MAX\_SEC) {  
 tvMsgReturned.setText(getString(R.string.*done\_background\_thread\_has\_been\_stopped*));  
 tvMsgWorking.setText(getString(R.string.*done*));  
 pbFirst.setVisibility(View.*GONE*);  
 pbSecond.setVisibility(View.*GONE*);  
 btnStart.setVisibility(View.*VISIBLE*);  
 isRunning = false;  
 } else {  
 tvMsgWorking.setText(getString(R.string.*working*) + pbFirst.getProgress());  
 }  
 }  
 };  
 }  
  
 private void initBgThread() {  
 bgThread = new Thread(new Runnable() {  
 @Override  
 public void run() {  
 try {  
 for (int i = 0; i < MAX\_SEC && isRunning; i++) {  
 Thread.*sleep*(1000);  
  
 Random rnd = new Random();  
 String data = "Thread value: " + (int) rnd.nextInt(101);  
 data += getString(R.string.*global\_value\_seen*) + " " + intTest;  
 intTest++;  
  
 if (isRunning) {  
 Message msg = handler.obtainMessage(1, (String) data);  
 handler.sendMessage(msg);  
 }  
 }  
 } catch (Throwable t) {  
 // Just finish the background thread  
 }  
 }  
 });  
 }  
}

**Bài 2: Viết ứng dụng MultiThread sử dụng Post**

**Bước 1:** Ở file **strings.xml** cập nhật lại code như sau

<resources>  
 <string name="bg\_work\_is\_over">Background work is over!</string>  
 <string name="execute">Execute</string>  
 <string name="enter\_some\_data\_here">Enter some data here</string>  
</resources>

**Bước 2**: ở **layout** thiết kế giao diện **activity\_main.xml**

\* code **activity\_main.xml** như sau

<?xml version="1.0" encoding="utf-8"?>  
<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:orientation="vertical"  
 android:padding="@dimen/margin\_base"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/tv\_top\_caption"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="@dimen/text\_medium" />  
  
 <ProgressBar  
 android:id="@+id/pb\_waiting"  
 style="?android:attr/progressBarStyleHorizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="@dimen/margin\_base" />  
  
 <EditText  
 android:id="@+id/et\_input"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/enter\_some\_data\_here"  
 android:textSize="@dimen/text\_medium" />  
  
 <Button  
 android:id="@+id/btn\_execute"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="@string/execute" />  
</LinearLayout>

Bước 3: Trong file MainActivity.java code như sau

package com.example.lab5;  
  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.Looper;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ProgressBar;  
import android.widget.TextView;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ProgressBar pbWaiting;  
 private TextView tvTopCaption;  
 private EditText etInput;  
 private Button btnExecute;  
  
 private int globalValue, accum;  
 private long startTime;  
 private final String PATIENCE = "Some important data is being collected now.\nPlease be patient...wait...";  
 private Handler handler;  
 private Runnable fgRunnable, bgRunnable;  
 private Thread testThread;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 findViewByIds();  
 initVariables();  
  
 btnExecute.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String input = etInput.getText().toString();  
 Toast.*makeText*(MainActivity.this, "You said: " + input, Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 testThread = new Thread(bgRunnable);  
 testThread.start();  
 pbWaiting.setMax(100);  
 }  
  
 private void findViewByIds() {  
 tvTopCaption = findViewById(R.id.*tv\_top\_caption*);  
 pbWaiting = findViewById(R.id.*pb\_waiting*);  
 etInput = findViewById(R.id.*et\_input*);  
 btnExecute = findViewById(R.id.*btn\_execute*);  
 }  
  
 private void initVariables() {  
 globalValue = 0;  
 accum = 0;  
 startTime = System.*currentTimeMillis*();  
 handler = new Handler(Looper.*getMainLooper*());  
  
 fgRunnable = new Runnable() {  
 @Override  
 public void run() {  
 try {  
 int progressStep = 5;  
 double totalTime = (System.*currentTimeMillis*() - startTime) / 1000.0;  
 // Sử dụng synchronized để đảm bảo an toàn khi truy cập biến globalValue từ nhiều luồng  
 synchronized (this) {  
 globalValue += 100;  
 }  
  
 tvTopCaption.setText(PATIENCE + totalTime + " - " + globalValue);  
 pbWaiting.incrementProgressBy(progressStep);  
 accum += progressStep;  
  
 if (accum >= pbWaiting.getMax()) {  
 tvTopCaption.setText(getString(R.string.*bg\_work\_is\_over*));  
 pbWaiting.setVisibility(View.*GONE*);  
 }  
 } catch (Exception e) {  
 Log.*e*("fgRunnable", e.getMessage());  
 }  
 }  
 };  
  
 bgRunnable = new Runnable() {  
 @Override  
 public void run() {  
 try {  
 for (int i = 0; i < 20; i++) {  
 Thread.*sleep*(1000);  
 synchronized (this) {  
 globalValue += 1;  
 }  
 handler.post(fgRunnable);  
 }  
 } catch (Exception e) {  
 Log.*e*("bgRunnable", e.getMessage());  
 }  
 }  
 };  
 }  
}

**Bài 3: Viết ứng dụng sử dụng AsyncTask**

**Bước 1:** Ở file **strings.xml** code như sau

<resources>  
 <string name="quick\_job">Quick Job</string>  
 <string name="slow\_job">Slow Job</string>  
 <string name="please\_wait">Some SLOW job is being done. Please wait...</string>  
</resources>

**Bước 2: Thiết kế giao diện activity\_main.xml như sau**

**\* code**

<?xml version="1.0" encoding="utf-8"?>  
<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:orientation="vertical"  
 android:padding="@dimen/margin\_base"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/tv\_status"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="@dimen/text\_medium" />  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="@dimen/margin\_base"  
 android:orientation="horizontal">  
  
 <Button  
 android:id="@+id/btn\_quick\_job"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/quick\_job" />  
  
 <Button  
 android:id="@+id/btn\_slow\_job"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/slow\_job"  
 android:layout\_marginStart="@dimen/margin\_base"/>  
 </LinearLayout>  
</LinearLayout>

**Ảnh có chứa ảnh chụp màn hình, văn bản, thiết kế

Nội dung do AI tạo ra có thể không chính xác.**

**Bước 3:** Lập trình trong **MainActivity.java** và **class SlowTask**

**\*** code **MainActivity.java**

package com.example.lab5;  
  
import android.app.ProgressDialog;  
import android.content.Context;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
import java.text.SimpleDateFormat;  
import java.util.Date;  
import java.util.Locale;  
  
public class MainActivity extends AppCompatActivity {  
  
 private TextView tvStatus;  
 private Button btnQuickJob, btnSlowJob;  
 private SlowTask slowTask;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 findViewByIds();  
 slowTask = new SlowTask(this, tvStatus);  
  
 btnQuickJob.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss", Locale.*getDefault*());  
 String currentDateandTime = sdf.format(new Date());  
 tvStatus.setText(currentDateandTime);  
 }  
 });  
  
 btnSlowJob.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // Cần tạo instance mới mỗi lần thực thi  
 new SlowTask(MainActivity.this, tvStatus).execute();  
 }  
 });  
 }  
  
 private void findViewByIds() {  
 tvStatus = findViewById(R.id.*tv\_status*);  
 btnQuickJob = findViewById(R.id.*btn\_quick\_job*);  
 btnSlowJob = findViewById(R.id.*btn\_slow\_job*);  
 }  
}  
  
class SlowTask extends AsyncTask<String, String, String> {  
 private ProgressDialog pd;  
 private Context context;  
 private TextView tvStatus;  
  
 public SlowTask(Context context, TextView tvStatus) {  
 this.context = context;  
 this.tvStatus = tvStatus;  
 }  
  
 @Override  
 protected void onPreExecute() {  
 super.onPreExecute();  
 pd = new ProgressDialog(context);  
 pd.setMessage(context.getString(R.string.*please\_wait*));  
 pd.setCancelable(false);  
 pd.show();  
 }  
  
 @Override  
 protected String doInBackground(String... params) {  
 try {  
 for (int i = 0; i < 5; i++) {  
 Thread.*sleep*(2000);  
 publishProgress("i = " + i);  
 }  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 return "Done!";  
 }  
  
 @Override  
 protected void onProgressUpdate(String... values) {  
 super.onProgressUpdate(values);  
 tvStatus.append("\n" + values[0]);  
 }  
  
 @Override  
 protected void onPostExecute(String result) {  
 super.onPostExecute(result);  
 if (pd.isShowing()) {  
 pd.dismiss();  
 }  
 tvStatus.append("\n" + result);  
 }  
}

**Bài 4: Ứng dụng phát nhạc bằng AsyncTask**

**Bước 1:** Trong **Manifest.xml** thêm thẻ

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

**Bước 2:** Cập nhật **file strings.xml**

<resources>  
 <string name="app\_name">Music Player</string>  
 <string name="play\_music">Play Music</string>  
 <string name="status\_default">Nhấn nút để bắt đầu phát nhạc</string>  
 <string name="status\_preparing">Đang chuẩn bị phát nhạc...</string>  
 <string name="status\_playing">Đang phát...</string>  
 <string name="status\_error">Lỗi khi phát nhạc.</string>  
</resources>

**Bước 3:** Trongfile **activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<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:orientation="vertical"  
 android:gravity="center"  
 android:padding="16dp"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/tv\_status"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/status\_default"  
 android:textSize="18sp"  
 android:layout\_marginBottom="24dp"/>  
  
 <Button  
 android:id="@+id/btn\_play\_music"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/play\_music" />  
  
</LinearLayout>

**Ảnh có chứa văn bản, Phông chữ, ảnh chụp màn hình, biểu tượng

Nội dung do AI tạo ra có thể không chính xác.**

**Bước 4:** Tạo 1 file **java** mới đặt tên **PlayMusicTask.java**

**Ảnh có chứa văn bản, Phông chữ, ảnh chụp màn hình, Đồ họa

Nội dung do AI tạo ra có thể không chính xác.**

package com.example.lab5;  
  
import android.app.ProgressDialog;  
import android.content.Context;  
import android.media.AudioManager;  
import android.media.MediaPlayer;  
import android.os.AsyncTask;  
import android.widget.TextView;  
import android.widget.Toast;  
import java.io.IOException;  
  
public class PlayMusicTask extends AsyncTask<String, Void, Boolean> {  
  
 private Context context;  
 private TextView tvStatus;  
 private MediaPlayer mediaPlayer;  
 private ProgressDialog progressDialog;  
  
 public PlayMusicTask(Context context, TextView tvStatus) {  
 this.context = context;  
 this.tvStatus = tvStatus;  
 }  
  
 */\*\*  
 \* Chạy trên UI thread trước khi doInBackground() bắt đầu.  
 \* Dùng để khởi tạo giao diện, ví dụ như hiển thị ProgressDialog.  
 \*/* @Override  
 protected void onPreExecute() {  
 super.onPreExecute();  
 tvStatus.setText(R.string.*status\_preparing*);  
 progressDialog = new ProgressDialog(context);  
 progressDialog.setMessage("Loading music...");  
 progressDialog.setCancelable(false);  
 progressDialog.show();  
 }  
  
 */\*\*  
 \* Chạy trên một background thread.  
 \* Thực hiện các tác vụ tốn thời gian ở đây (ví dụ: tải và chuẩn bị nhạc).  
 \* Không được cập nhật UI trực tiếp từ đây.  
 \*/* @Override  
 protected Boolean doInBackground(String... params) {  
 String musicUrl = params[0];  
 mediaPlayer = new MediaPlayer();  
 mediaPlayer.setAudioStreamType(AudioManager.*STREAM\_MUSIC*);  
  
 try {  
 mediaPlayer.setDataSource(musicUrl);  
 mediaPlayer.prepare(); // Bước này có thể tốn thời gian  
 mediaPlayer.start();  
 return true; // Trả về true nếu thành công  
 } catch (IOException e) {  
 e.printStackTrace();  
 return false; // Trả về false nếu có lỗi  
 }  
 }  
  
 */\*\*  
 \* Chạy trên UI thread sau khi doInBackground() hoàn thành.  
 \* Dùng để cập nhật UI với kết quả trả về từ doInBackground().  
 \*/* @Override  
 protected void onPostExecute(Boolean success) {  
 super.onPostExecute(success);  
 if (progressDialog.isShowing()) {  
 progressDialog.dismiss();  
 }  
  
 if (success) {  
 tvStatus.setText(R.string.*status\_playing*);  
 } else {  
 tvStatus.setText(R.string.*status\_error*);  
 Toast.*makeText*(context, "Không thể phát nhạc.", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
}

**Bước 5:** Trong **MainActivity.java**

package com.example.lab5;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 private TextView tvStatus;  
 private Button btnPlayMusic;  
  
 // URL của một file nhạc mp3 online để test  
 private final String MUSIC\_URL = "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Ánh xạ các view từ layout  
 tvStatus = findViewById(R.id.*tv\_status*);  
 btnPlayMusic = findViewById(R.id.*btn\_play\_music*);  
  
 // Thiết lập sự kiện click cho button  
 btnPlayMusic.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 // Tạo một instance mới của PlayMusicTask và thực thi nó  
 PlayMusicTask playMusicTask = new PlayMusicTask(MainActivity.this, tvStatus);  
 playMusicTask.execute(MUSIC\_URL);  
 }  
 });  
 }  
}

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Nội dung do AI tạo ra có thể không chính xác.**

**Bài 5: Viết lại ứng dụng phát nhạc bằng RxJava**

**Bước 1:** Thêm thư viện **RxJava** và **RxAndroid** vào dự án

- Mở **build.gradle( Module:app)** thêm các thư viện sau – bấm **Syn Now**

*implementation*("io.reactivex.rxjava3:rxandroid:3.0.0")  
*implementation*("io.reactivex.rxjava3:rxjava:3.0.0")

**Bước 2:** Lập trình **MainActivity.java** với **RxJava**

**\*** code như sau

package com.example.lab5;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.media.AudioManager;  
import android.media.MediaPlayer;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import io.reactivex.rxjava3.android.schedulers.AndroidSchedulers;  
import io.reactivex.rxjava3.core.Observable;  
import io.reactivex.rxjava3.disposables.Disposable;  
import io.reactivex.rxjava3.schedulers.Schedulers;  
  
public class MainActivity extends AppCompatActivity {  
  
 private TextView tvStatus;  
 private Button btnPlayMusic;  
  
 private final String MUSIC\_URL = "https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3";  
 private Disposable musicDisposable; // Đối tượng để quản lý luồng RxJava  
 private MediaPlayer mediaPlayer;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 tvStatus = findViewById(R.id.*tv\_status*);  
 btnPlayMusic = findViewById(R.id.*btn\_play\_music*);  
  
 btnPlayMusic.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 playMusicWithRxJava(MUSIC\_URL);  
 }  
 });  
 }  
  
 private void playMusicWithRxJava(String url) {  
 tvStatus.setText(getString(R.string.*status\_preparing*));  
 btnPlayMusic.setEnabled(false); // Vô hiệu hóa nút trong khi xử lý  
  
 // 1. Tạo một Observable để thực hiện công việc nền  
 musicDisposable = Observable.*fromCallable*(() -> {  
 // Công việc này chạy trên luồng nền (background thread)  
 mediaPlayer = new MediaPlayer();  
 mediaPlayer.setAudioStreamType(AudioManager.*STREAM\_MUSIC*);  
 mediaPlayer.setDataSource(url);  
 mediaPlayer.prepare(); // Tác vụ tốn thời gian  
 mediaPlayer.start();  
 return true; // Trả về true nếu thành công  
 })  
 // 2. Chỉ định luồng để chạy tác vụ  
 .subscribeOn(Schedulers.*io*()) // Chạy công việc trên một luồng I/O  
 // 3. Chỉ định luồng để nhận kết quả và cập nhật UI  
 .observeOn(AndroidSchedulers.*mainThread*()) // Nhận kết quả trên luồng chính (UI thread)  
 // 4. Đăng ký để lắng nghe kết quả  
 .subscribe(  
 success -> { // onNext: Được gọi khi tác vụ thành công và trả về kết quả  
 tvStatus.setText(getString(R.string.*status\_playing*));  
 btnPlayMusic.setEnabled(true);  
 },  
 error -> { // onError: Được gọi khi có lỗi xảy ra  
 tvStatus.setText(getString(R.string.*status\_error*));  
 Toast.*makeText*(MainActivity.this, "Lỗi: " + error.getMessage(), Toast.*LENGTH\_SHORT*).show();  
 btnPlayMusic.setEnabled(true);  
 }  
 );  
 }  
  
 @Override  
 protected void onDestroy() {  
 super.onDestroy();  
 // Rất quan trọng: Hủy bỏ subscription và giải phóng tài nguyên khi Activity bị hủy  
 if (musicDisposable != null && !musicDisposable.isDisposed()) {  
 musicDisposable.dispose();  
 }  
 if (mediaPlayer != null) {  
 mediaPlayer.release();  
 mediaPlayer = null;  
 }  
 }  
}

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Nội dung do AI tạo ra có thể không chính xác.**