**1. Viết một ứng dụng MultiThread sử dụng Message. Giao diện:**

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, thiết kế

Nội dung do AI tạo ra có thể không chính xác.Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, Điện thoại di động

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

**<MainActivity>**

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.appcompat.app.AppCompatActivity;  
  
import java.util.Random;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ProgressBar pbFirst, pbSecond;  
 private TextView tvMsgWorking, tvMsgReturned;  
 private boolean isRunning;  
 private static final int *MAX\_SEC* = 20; // demo  
 private int intTest = 0;  
 private Thread bgThread;  
 private Handler handler;  
 private Button btnStart;  
  
 private static final int *MSG\_UPDATE* = 1001;  
 private static final int *MSG\_DONE* = 1002;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 findViewByIds();  
 initVariables();  
 btnStart.setOnClickListener(v -> {  
 if (bgThread == null || !bgThread.isAlive()) {  
 initBgThread();  
 bgThread.start();  
 }  
 });  
 }  
  
 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;  
 pbFirst.setMax(*MAX\_SEC*);  
 pbFirst.setProgress(0);  
  
 // Ẩn lúc đầu  
 pbFirst.setVisibility(View.*GONE*);  
 pbSecond.setVisibility(View.*GONE*);  
  
 // Spinner luôn indeterminate  
 pbSecond.setIndeterminate(true);  
  
 handler = new Handler(Looper.*getMainLooper*()) {  
 @Override public void handleMessage(Message msg) {  
 switch (msg.what) {  
 case *MSG\_UPDATE*: {  
 Bundle b = msg.getData();  
 int sec = b.getInt("sec");  
 int rand = b.getInt("rand");  
 int gval = b.getInt("gval");  
  
 tvMsgReturned.setText(getString(R.string.*returned\_by\_bg\_thread*)  
 + " sec:" + sec + ", Thread value:" + rand + ", Global value seen:" + gval);  
  
 pbFirst.setProgress(sec); // thanh ngang tăng dần  
 tvMsgWorking.setText(getString(R.string.*working*) + sec);  
 break;  
 }  
 case *MSG\_DONE*: {  
 tvMsgReturned.setText(getString(R.string.*done\_background\_thread\_has\_been\_stopped*));  
 tvMsgWorking.setText(getString(R.string.*done*));  
  
 // Ẩn cả hai khi xong  
 pbFirst.setVisibility(View.*GONE*);  
 pbSecond.setVisibility(View.*GONE*);  
 btnStart.setVisibility(View.*VISIBLE*);  
 isRunning = false;  
 break;  
 }  
 }  
 }  
 };  
 }  
  
  
 private void initBgThread() {  
 isRunning = true;  
 bgThread = new Thread(() -> {  
 Random r = new Random();  
 for (int i = 1; i <= *MAX\_SEC* && isRunning; i++) {  
 intTest++;  
 int rand = r.nextInt(101); // 0..100  
 Message m = handler.obtainMessage(*MSG\_UPDATE*);  
 Bundle b = new Bundle();  
 b.putInt("sec", i);  
 b.putInt("rand", rand);  
 b.putInt("gval", intTest);  
 m.setData(b);  
 handler.sendMessage(m);  
 try { Thread.*sleep*(1000); } catch (InterruptedException ignored) {}  
 }  
 handler.sendEmptyMessage(*MSG\_DONE*);  
 });  
 }  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 // (khởi tạo thread khi nhấn Start ở onClick)  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 isRunning = false;  
 }  
}

<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:orientation="vertical" android:padding="@dimen/margin\_base"  
 android:layout\_width="match\_parent" android:layout\_height="match\_parent">  
  
 <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:indeterminate="false"  
 android:max="20"  
 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"  
 tools:ignore="UselessLeaf">  
 </FrameLayout>  
  
  
 <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"  
 style="?android:attr/progressBarStyleLarge"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:indeterminate="true"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="@dimen/margin\_base" />  
  
</LinearLayout>

**2.Viết ứng dụng MultiThread sử dụng Post**

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

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

<PortActivity>

package com.example.lab5;  
  
import android.os.Bundle;  
import android.os.Handler;  
import android.os.Looper;  
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 PostActivity extends AppCompatActivity {  
  
 private ProgressBar pbWaiting;  
 private TextView tvTopCaption;  
 private EditText etInput;  
 private Button btnExecute;  
  
 private int globalValue, accum;  
 private long startTime;  
 private static final String *PATIENCE* =  
 "Some important data is being collected now.\nPlease be patient...wait...";  
  
 private Handler handler;  
 private Runnable fgRunnable, bgRunnable;  
 private Thread testThread;  
  
 private volatile boolean alive = true;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_post*);  
  
 findViewByIds();  
 initVariables();  
  
 // Runnable chạy trên main (tăng 100 mỗi lần và cập nhật UI)  
 fgRunnable = () -> {  
 if (!alive) return;  
 globalValue += 100;  
 tvTopCaption.setText(*PATIENCE* + "\nGV=" + globalValue);  
 };  
  
 // Runnable chạy nền (mỗi 1s tăng 1, post về main để cập nhật)  
 bgRunnable = () -> {  
 while (alive && (System.*currentTimeMillis*() - startTime) < 10000) {  
 globalValue += 1;  
 handler.post(fgRunnable); // tương tác UI an toàn  
 try { Thread.*sleep*(1000); } catch (InterruptedException ignored) {}  
 }  
 handler.post(() -> tvTopCaption.setText(getString(R.string.*bg\_work\_is\_over*)));  
 };  
  
 testThread = new Thread(bgRunnable);  
 testThread.start();  
  
 btnExecute.setOnClickListener(v -> {  
 String data = etInput.getText().toString().trim();  
 Toast.*makeText*(this, data.isEmpty() ? "(empty)" : data, Toast.*LENGTH\_SHORT*).show();  
 });  
 }  
  
 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*());  
 pbWaiting.setMax(10);  
 // Thanh tiến trình tự tăng minh họa  
 new Thread(() -> {  
 for (int i = 1; i <= 10 && alive; i++) {  
 int progress = i;  
 handler.post(() -> pbWaiting.setProgress(progress));  
 try { Thread.*sleep*(1000); } catch (InterruptedException ignored) {}  
 }  
 }).start();  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 alive = false;  
 }  
}

<activity\_post.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:orientation="vertical" android:padding="@dimen/margin\_base"  
 android:layout\_width="match\_parent" android:layout\_height="match\_parent">  
  
 <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:importantForAutofill="no"  
 android:textSize="@dimen/text\_medium"  
 tools:ignore="TextFields" />  
  
 <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>

**3.Viết ứng dụng sử dụng Asynctask ( Hiện nay không khuyến khích dùng nữa nhưng Cô vẫn để vào để các biết)**

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

Nội dung do AI tạo ra có thể không chính xác.Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

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

**<MainActivity.java>**

package com.example.lab5;  
  
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 Button btnQuickJob, btnSlowJob;  
 private TextView tvStatus;  
 private SlowTask slowTask;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 findViewByIds();  
  
 // Khởi tạo SlowTask  
 slowTask = new SlowTask(MainActivity.this, tvStatus);  
  
 // Xử lý nút Quick Job  
 btnQuickJob.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyy HH:mm:ss");  
 tvStatus.setText(sdf.format(new Date()));  
 }  
 });  
  
 // Xử lý nút Slow Job  
 btnSlowJob.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 slowTask.execute();  
 }  
 });  
 }  
  
 private void findViewByIds() {  
 btnQuickJob = findViewById(R.id.*btn\_quick\_job*);  
 btnSlowJob = findViewById(R.id.*btn\_slow\_job*);  
 tvStatus = findViewById(R.id.*tv\_status*);  
 }  
}

**<SlowTask.java>**

package com.example.lab5;  
  
import android.app.ProgressDialog;  
import android.content.Context;  
import android.os.AsyncTask;  
import android.util.Log;  
import android.widget.TextView;  
  
public class SlowTask extends AsyncTask<String, Long, Void> {  
 private ProgressDialog pdWaiting;  
 private Context context;  
 private Long startTime;  
 private TextView tvStatus;  
  
 public SlowTask(Context context, TextView tvStatus) {  
 this.context = context;  
 this.tvStatus = tvStatus;  
 }  
  
 @Override  
 protected void onPreExecute() {  
 super.onPreExecute();  
  
 // Có thể thao tác trên UI thread tại đây  
 pdWaiting = new ProgressDialog(context);  
 startTime = System.*currentTimeMillis*();  
 tvStatus.setText("Start time: " + startTime);  
 pdWaiting.setMessage(context.getString(R.string.*please\_wait*));  
 pdWaiting.show();  
 }  
  
 @Override  
 protected Void doInBackground(String... params) {  
 // Thực hiện tác vụ chạy lâu (SLOW job)  
 try {  
 for (Long i = 0L; i < 3L; i++) {  
 Thread.*sleep*(2000); // giả lập tiến trình 2 giây  
 publishProgress(i); // gửi dữ liệu cho UI  
 }  
 } catch (Exception e) {  
 Log.*e*("SlowJob", e.getMessage());  
 }  
 return null;  
 }  
  
 @Override  
 protected void onProgressUpdate(Long... values) {  
 super.onProgressUpdate(values);  
 tvStatus.append("\nWorking..." + values[0]);  
 }  
  
 @Override  
 protected void onPostExecute(Void aVoid) {  
 super.onPostExecute(aVoid);  
  
 // Có thể sử dụng UI thread tại đây  
 if (pdWaiting.isShowing())  
 pdWaiting.dismiss();  
  
 // Hiển thị thông báo hoàn thành  
 tvStatus.append("\nEnd Time: " + System.*currentTimeMillis*());  
 tvStatus.append("\nDone!");  
 }  
  
}

**<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:orientation="vertical" android:padding="@dimen/margin\_base"  
 android:layout\_width="match\_parent" android:layout\_height="match\_parent">  
  
 <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"  
 tools:ignore="ButtonStyle" />  
  
 <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\_marginLeft="@dimen/margin\_base"  
 tools:ignore="ButtonStyle,RtlHardcoded" />  
 </LinearLayout>  
</LinearLayout>

**4) Tự viết ứng dụng AsyncTask phát nhạc**

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

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

**<MusicActivity.java>**

package com.example.lab5;  
  
import android.os.Bundle;  
import android.widget.Button;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MusicActivity extends AppCompatActivity {  
  
 private Button btnPlay;  
 private MusicTask currentTask;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_music*);  
  
 btnPlay = findViewById(R.id.*btn\_play*);  
  
 btnPlay.setOnClickListener(v -> {  
 if (currentTask != null) currentTask.cancel(true);  
 currentTask = new MusicTask(this);  
 });  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 if (currentTask != null) {  
 currentTask.cancel(true);  
 currentTask = null;  
 }  
 }  
}

**<MusicTask.java>**

package com.example.lab5;  
  
import android.app.Activity;  
import android.app.ProgressDialog;  
import android.media.MediaPlayer;  
import android.os.AsyncTask;  
  
import java.lang.ref.WeakReference;  
  
@SuppressWarnings("deprecation")  
public class MusicTask extends AsyncTask<Void, Void, MediaPlayer> {  
  
 private final WeakReference<Activity> activityRef;  
 private ProgressDialog dialog;  
  
 public MusicTask(Activity activity) {  
 this.activityRef = new WeakReference<>(activity);  
 }  
  
 @Override  
 protected void onPreExecute() {  
 Activity act = activityRef.get();  
 if (act == null || act.isFinishing()) return;  
  
 dialog = new ProgressDialog(act);  
 dialog.setMessage("Preparing music...");  
 dialog.setCancelable(false);  
 dialog.show();  
 }  
  
 @Override  
 protected MediaPlayer doInBackground(Void... voids) {  
 Activity act = activityRef.get();  
 if (act == null || act.isFinishing()) return null;  
  
 // Tạo MediaPlayer từ raw resource (file bạn đặt ở res/raw/sample.wav)  
 // Lưu ý: KHÔNG đổi tên resource, để đúng là "sample"  
 return MediaPlayer.*create*(act, R.raw.*sample*);  
 }  
  
 @Override  
 protected void onPostExecute(MediaPlayer mp) {  
 Activity act = activityRef.get();  
 if (act == null || act.isFinishing()) {  
 if (mp != null) mp.release();  
 return;  
 }  
  
 if (dialog != null && dialog.isShowing()) dialog.dismiss();  
  
 if (mp != null) {  
 // Giải phóng tài nguyên khi phát xong  
 mp.setOnCompletionListener(player -> {  
 try { player.stop(); } catch (Exception ignored) {}  
 player.release();  
 });  
 mp.start();  
 }  
 }  
  
 @Override  
 protected void onCancelled(MediaPlayer mp) {  
 if (dialog != null && dialog.isShowing()) dialog.dismiss();  
 if (mp != null) {  
 try { mp.stop(); } catch (Exception ignored) {}  
 mp.release();  
 }  
 }  
}

**<activity\_music.xml>**

<?xml version="1.0" encoding="utf-8"?>  
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:padding="16dp">  
  
 <Button  
 android:id="@+id/btn\_play"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Play with AsyncTask"  
 android:layout\_gravity="center" />  
</FrameLayout>

Link github để tải sample.wav

<https://github.com/tranphatthinh/duanPMMNM.git>

Tải file về máy app/res/raw/sample.wav

**5. Viết lại ứng dụng bài 4 sử dụng RxJava, Rx Android**

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

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

**build.gradle (Module)**

dependencies {

implementation 'io.reactivex.rxjava3:rxjava:3.1.8'

implementation 'io.reactivex.rxjava3:rxandroid:3.0.2'

}

**<MusicActivity.java>**

package com.example.lab5;  
  
import android.media.MediaPlayer;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.Toast;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import io.reactivex.rxjava3.android.schedulers.AndroidSchedulers;  
import io.reactivex.rxjava3.core.Observable;  
import io.reactivex.rxjava3.disposables.CompositeDisposable;  
import io.reactivex.rxjava3.schedulers.Schedulers;  
  
public class MusicActivity extends AppCompatActivity {  
  
 private Button btnPlayRx;  
 private final CompositeDisposable bag = new CompositeDisposable();  
 private MediaPlayer mp;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_music*);  
  
 btnPlayRx = findViewById(R.id.*btn\_play\_rx*);  
  
 btnPlayRx.setOnClickListener(v -> {  
 bag.clear();  
 bag.add(  
 Observable.*fromCallable*(() -> {  
 // chuẩn bị media ở background  
 return MediaPlayer.*create*(getApplicationContext(), R.raw.*sample*);  
 })  
 .subscribeOn(Schedulers.*io*())  
 .observeOn(AndroidSchedulers.*mainThread*())  
 .doOnSubscribe(d -> Toast.*makeText*(this, "Preparing...", Toast.*LENGTH\_SHORT*).show())  
 .subscribe(player -> {  
 mp = player;  
 if (mp != null) {  
 mp.setOnCompletionListener(MediaPlayer::release);  
 mp.start();  
 }  
 }, err -> Toast.*makeText*(this, "Error: " + err.getMessage(), Toast.*LENGTH\_SHORT*).show())  
 );  
 });  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 if (mp != null) {  
 try { mp.stop(); } catch (Exception ignored) {}  
 mp.release();  
 mp = null;  
 }  
 bag.clear();  
 }  
}

<activity\_music.xml>

<?xml version="1.0" encoding="utf-8"?>  
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent" android:layout\_height="match\_parent"  
 android:padding="16dp">  
  
 <Button  
 android:id="@+id/btn\_play\_rx"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Play with RxJava"  
 android:layout\_gravity="center"/>  
</FrameLayout>