****

**UNIVERSITI MALAYSIA TERENGGANU**

**FACULTY OF COMPUTER SCIENCE AND MATHEMATICS**

**­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**CSM3123 - NATIVE MOBILE PROGRAMMING**

**BANCHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONORS**

**LAB 1 (Task 1,2,3)**

**SEMESTER 5 2024/2025**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**PREPARED FOR:**

**DR RABIEI B MAMAT**

**PREPARED BY:**

**NUR EZREENA SHUHADA BT EMRAN**

**S66467**

**TASK 1**

***Activity***

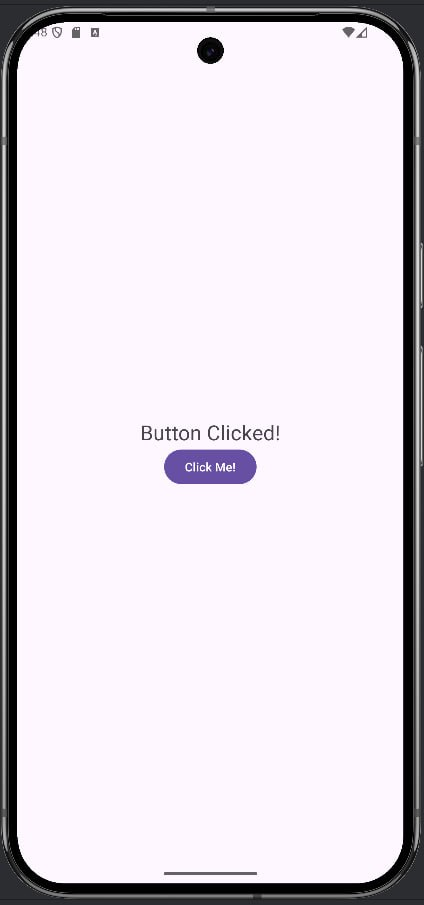
Activity\_main.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">  <LinearLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:gravity="center">  <TextView  android:id="@+id/textView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Hello World!"  android:textSize="24sp"/>  <Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Click Me!" />  </LinearLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

MainActivity.java

|  |
| --- |
| package com.example.myfirstappjava;  import android.os.Bundle;  import android.view.View;  import android.widget.Button;  import android.widget.TextView;  import androidx.activity.EdgeToEdge;  import androidx.appcompat.app.AppCompatActivity;  import androidx.core.graphics.Insets;  import androidx.core.view.ViewCompat;  import androidx.core.view.WindowInsetsCompat;  public class MainActivity extends AppCompatActivity {  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  EdgeToEdge.enable(this);  setContentView(R.layout.activity\_main);  TextView textView = findViewById(R.id.textView);  Button button = findViewById(R.id.button);  button.setOnClickListener(new View.OnClickListener(){  public void onClick(View v){  textView.setText("Button Clicked!");  }  });  }  } |

**Output:**



***HANDS ON Activities***

Activity\_main.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">   <LinearLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:gravity="center">   <TextView  android:id="@+id/textView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Hello,World!"  android:textSize="24sp"  android:textColor="@color/black" />   <Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Click Me!" />  <Button  android:id="@+id/buttonNavigate"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Go to Second Activity" />  </LinearLayout>   </androidx.constraintlayout.widget.ConstraintLayout> |

MainActivity.java

|  |
| --- |
| package com.example.myfirstappjava;  import android.content.Intent; import android.graphics.Color; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.TextView; import android.widget.Toast;  import androidx.activity.EdgeToEdge; import androidx.appcompat.app.AppCompatActivity; import androidx.core.graphics.Insets; import androidx.core.view.ViewCompat; import androidx.core.view.WindowInsetsCompat;  public class MainActivity extends AppCompatActivity {  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  TextView textView = findViewById(R.id.*textView*);   Button button = findViewById(R.id.*button*);  button.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  textView.setText("Button Clicked!");  textView.setTextColor(Color.*RED*);   Toast.*makeText*(MainActivity.this, "Button was clicked!", Toast.*LENGTH\_SHORT*).show();   }   });   Button buttonNav = findViewById(R.id.*buttonNavigate*);  buttonNav.setOnClickListener(new View.OnClickListener(){  public void onClick(View v){  Intent intent = new Intent(MainActivity.this, SecondActivity.class);  intent.putExtra("EXTRA\_MESSAGE","Hello from MainActivity!");  startActivity(intent);  }  });  } } |

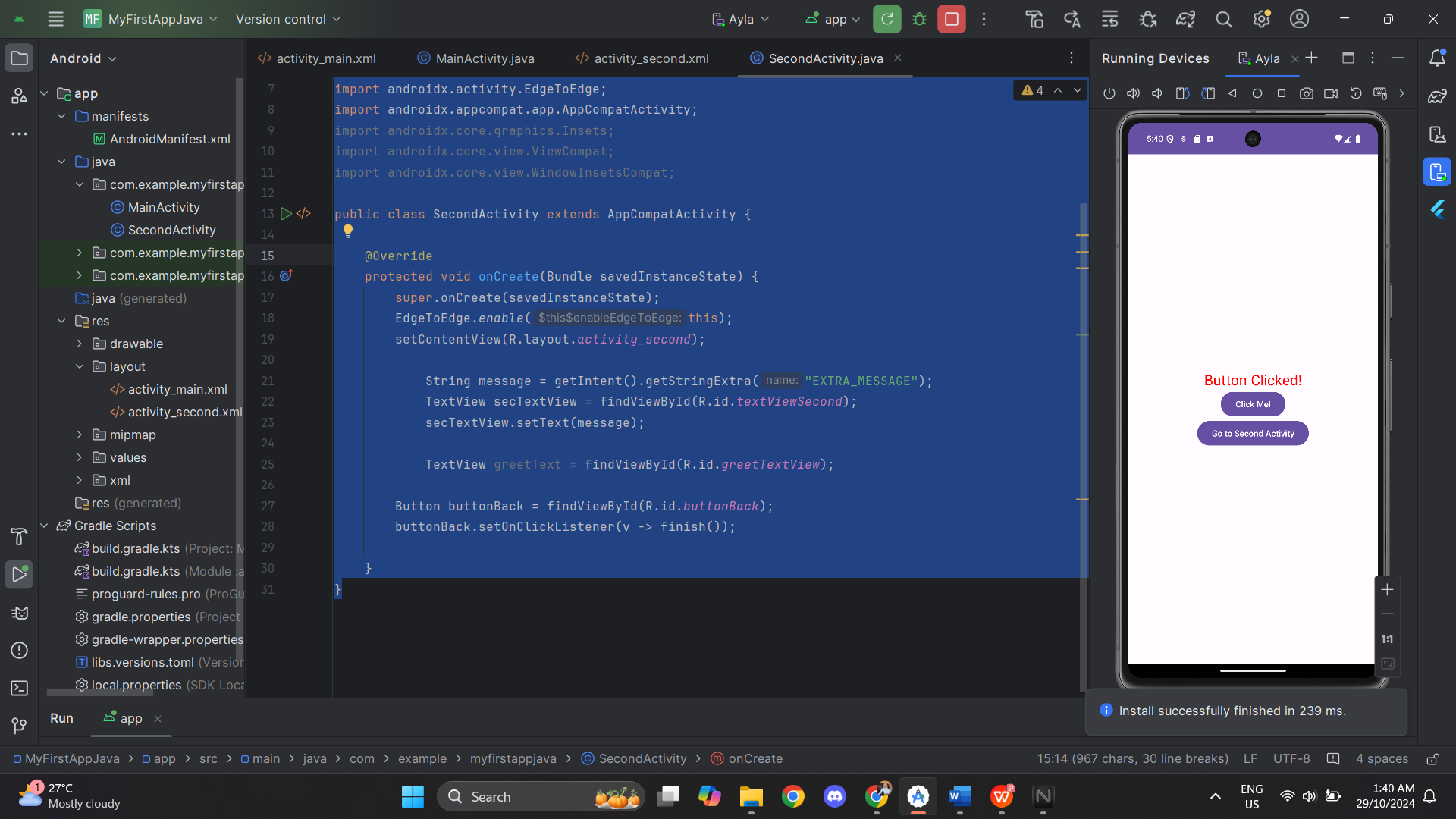
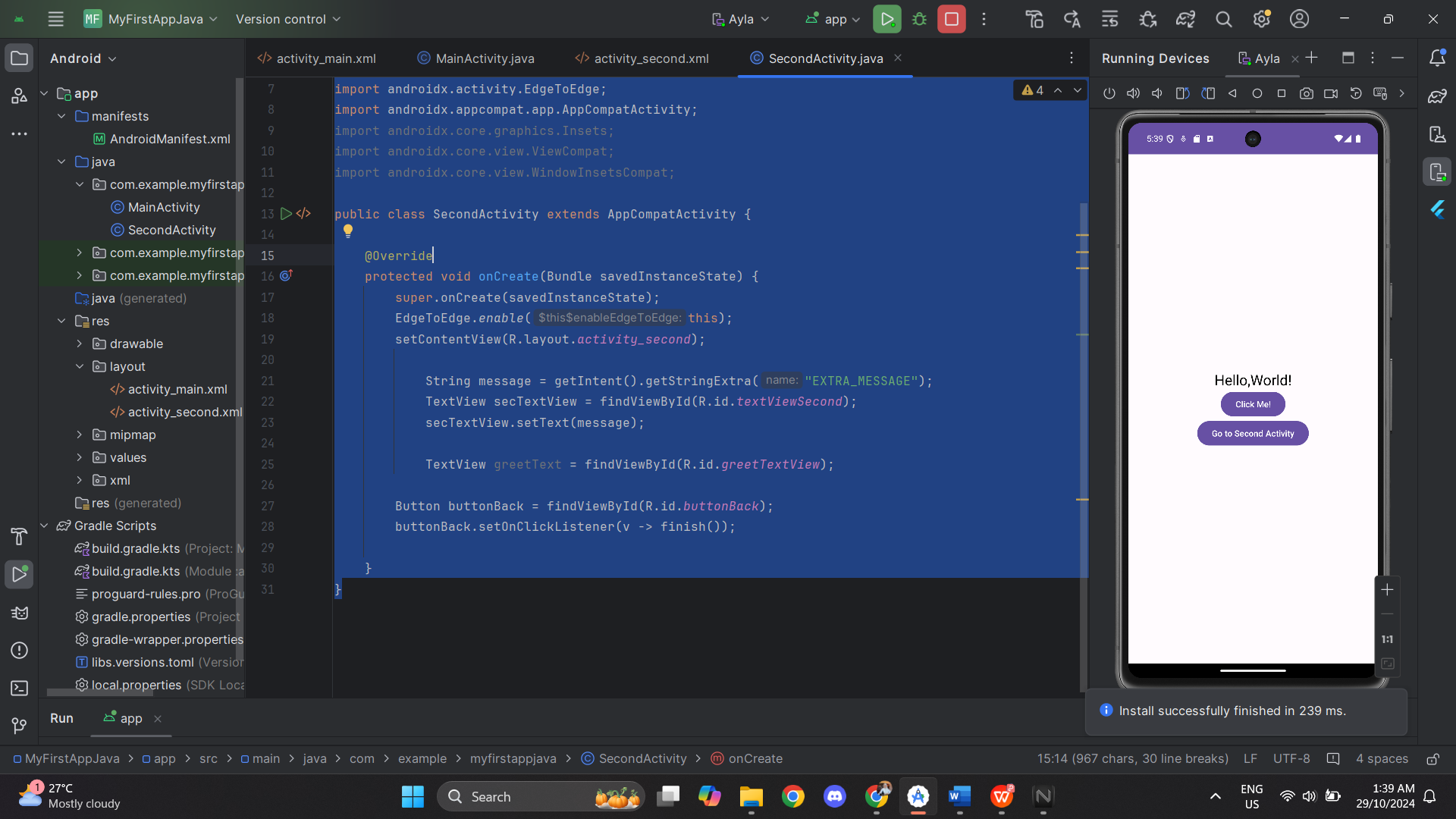
Activity\_second.xml

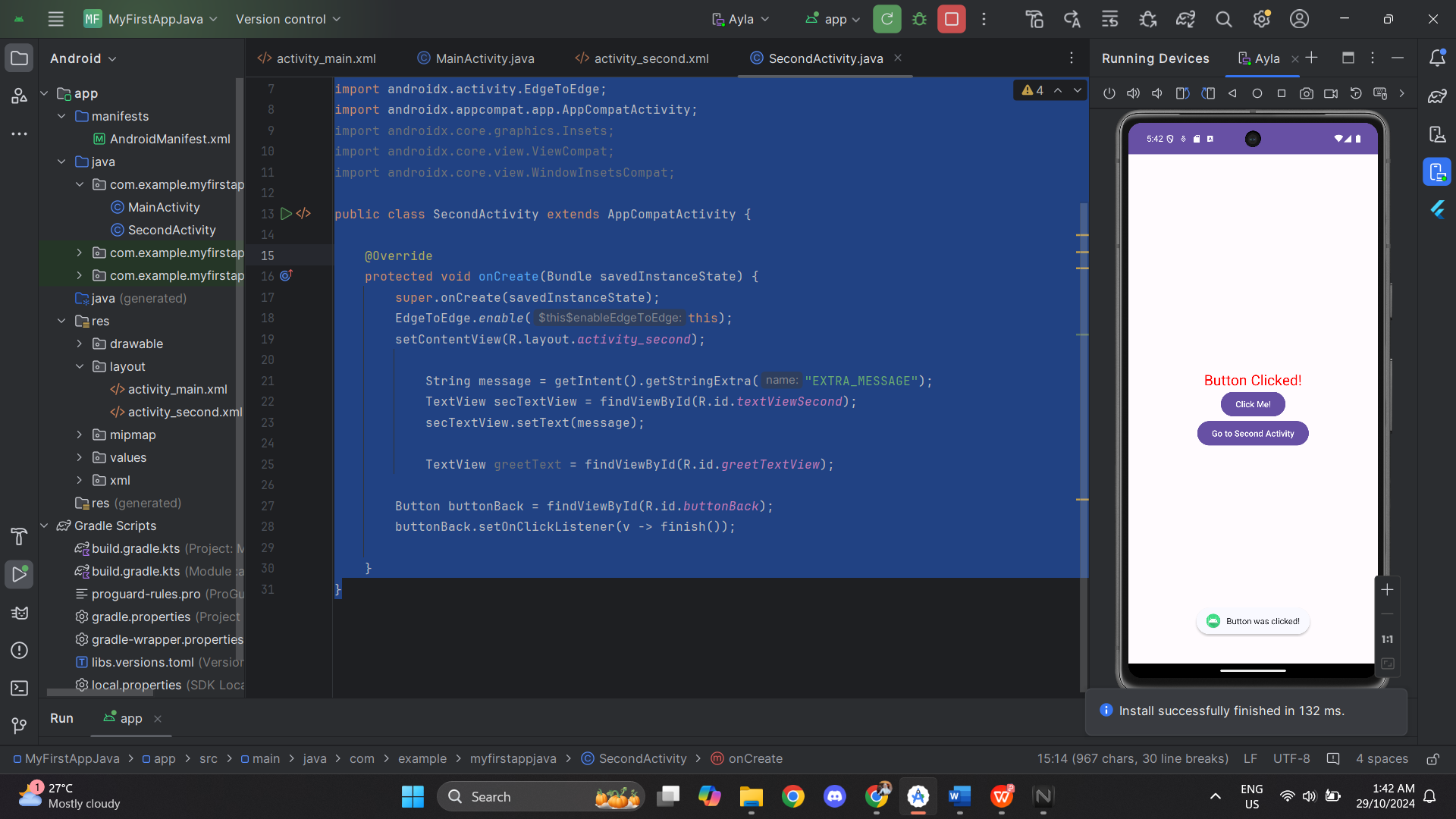
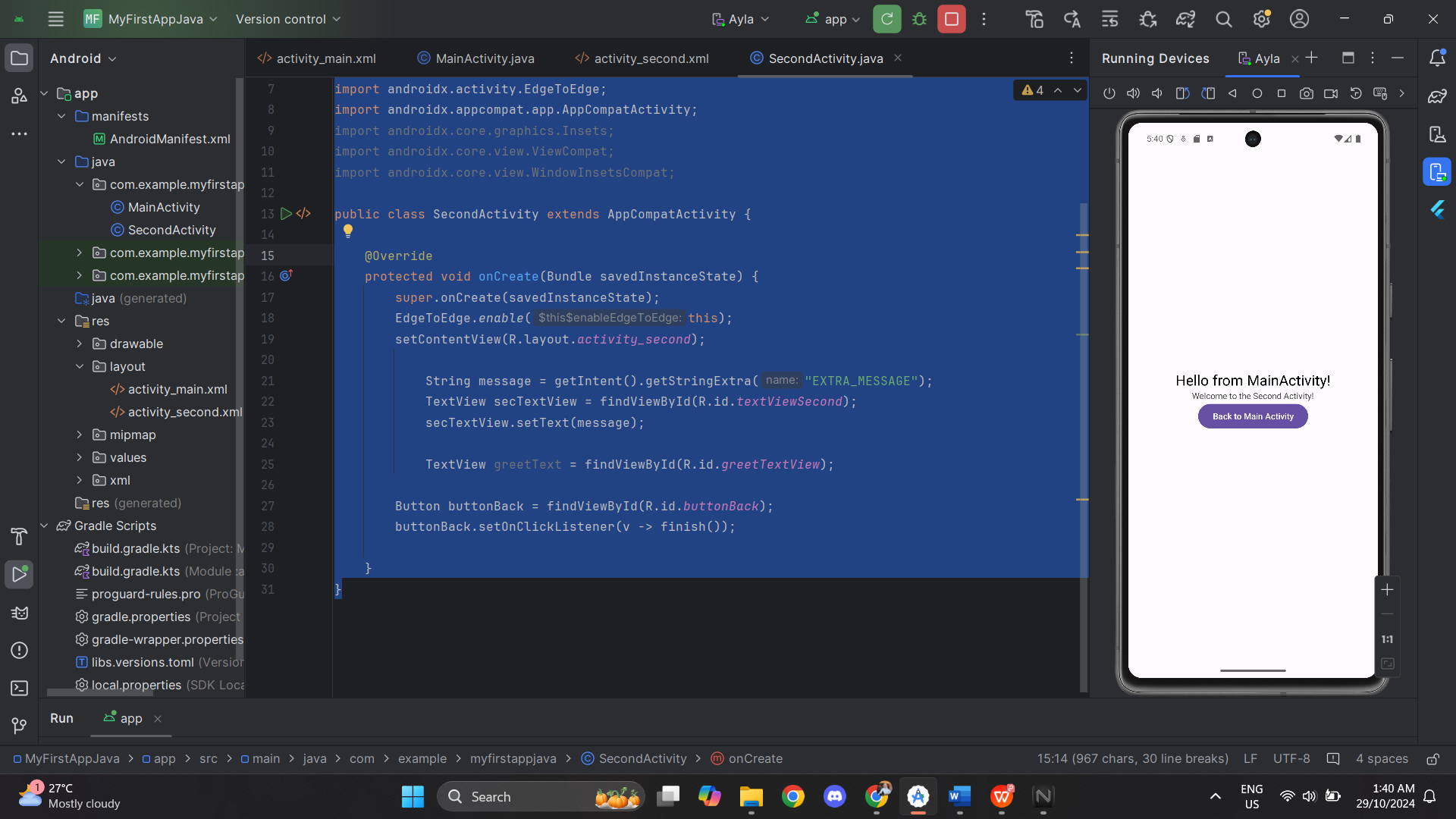
|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".SecondActivity">    <LinearLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:gravity="center">   <TextView  android:id="@+id/textViewSecond"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:textSize="24sp"  android:textColor="@color/black" />   <TextView  android:id="@+id/greetTextView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Welcome to the Second Activity!"/>   <Button  android:id="@+id/buttonBack"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Back to Main Activity" />   </LinearLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

SecondActivity.java

|  |
| --- |
| package com.example.myfirstappjava;  import android.os.Bundle; import android.widget.TextView; import android.widget.Button;  import androidx.activity.EdgeToEdge; import androidx.appcompat.app.AppCompatActivity; import androidx.core.graphics.Insets; import androidx.core.view.ViewCompat; import androidx.core.view.WindowInsetsCompat;  public class SecondActivity extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  EdgeToEdge.*enable*(this);  setContentView(R.layout.*activity\_second*);   String message = getIntent().getStringExtra("EXTRA\_MESSAGE");  TextView secTextView = findViewById(R.id.*textViewSecond*);  secTextView.setText(message);   TextView greetText = findViewById(R.id.*greetTextView*);   Button buttonBack = findViewById(R.id.*buttonBack*);  buttonBack.setOnClickListener(v -> finish());   } } |

**Output**





**TASK 2**

MainActivity.kt

|  |
| --- |
| package com.example.musicplayerapp  import android.content.Intent import android.os.Bundle import android.widget.Button import androidx.appcompat.app.AppCompatActivity  class MainActivity : AppCompatActivity() {  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  setContentView(R.layout.*activity\_main*)   val startMusicButton: Button = findViewById(R.id.*startMusicButton*)  val stopMusicButton: Button = findViewById(R.id.*stopMusicButton*)   startMusicButton.setOnClickListener {  val intent = Intent(this, MusicService::class.*java*)  startService(intent) // Start the foreground service  }   stopMusicButton.setOnClickListener {  val intent = Intent(this, MusicService::class.*java*)  stopService(intent) // Stop the foreground service  }  } } |

Activity\_main.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MainActivity">    <LinearLayout  xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:gravity="center"  android:orientation="vertical">   <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:src="@drawable/ic\_music\_note"  android:scaleType="fitCenter"  android:maxWidth="100dp"  android:maxHeight="100dp"  android:adjustViewBounds="true" />  <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Igauan malam (Insomniacks)'" />   <Button  android:id="@+id/startMusicButton"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Start Music" />  <Button  android:id="@+id/stopMusicButton"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Stop Music" />  </LinearLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

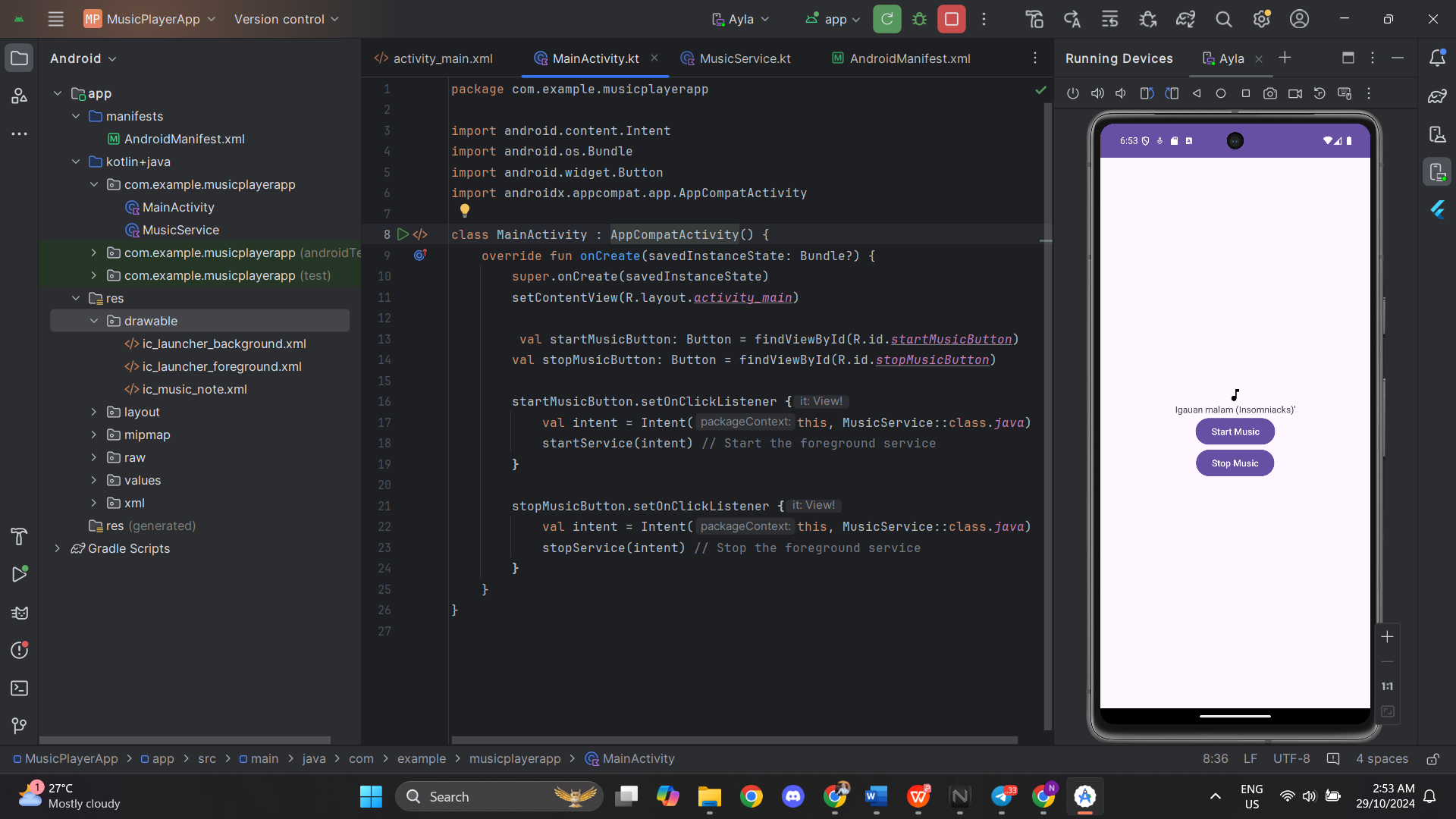
MusicService.kt

|  |
| --- |
| package com.example.musicplayerapp  import android.app.NotificationChannel import android.app.NotificationManager import android.app.PendingIntent import android.app.Service import android.content.Intent import android.media.MediaPlayer import android.os.Build import android.os.IBinder import androidx.core.app.NotificationCompat  class MusicService : Service() {   private lateinit var mediaPlayer: MediaPlayer   override fun onCreate() {  super.onCreate()  mediaPlayer = MediaPlayer.create(this, R.raw.*song*)  mediaPlayer.*isLooping* = true // Loop playback if required  }   override fun onStartCommand(intent: Intent?, flags: Int, startId: Int): Int {  startForegroundServiceWithNotification()  if (!mediaPlayer.*isPlaying*) {  mediaPlayer.start()  }  return *START\_STICKY* }   private fun startForegroundServiceWithNotification() {  createNotificationChannel()   val notificationIntent = Intent(this, MainActivity::class.*java*)  val pendingIntent = PendingIntent.getActivity(  this,  0,  notificationIntent,  PendingIntent.*FLAG\_UPDATE\_CURRENT* or PendingIntent.*FLAG\_IMMUTABLE* )   val notification = NotificationCompat.Builder(this, "music\_service\_channel")  .setContentTitle("Music Player")  .setContentText("Playing music...")  .setSmallIcon(R.drawable.*ic\_music\_note*) // Ensure this icon exists in your resources  .setContentIntent(pendingIntent)  .build()   startForeground(1, notification)  }   private fun createNotificationChannel() {  if (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*O*) {  val serviceChannel = NotificationChannel(  "music\_service\_channel",  "Music Service Channel",  NotificationManager.*IMPORTANCE\_LOW* )  val manager = getSystemService(NotificationManager::class.*java*)  manager?.createNotificationChannel(serviceChannel)  }  }   override fun onDestroy() {  mediaPlayer.stop()  mediaPlayer.release()  super.onDestroy()  }   override fun onBind(intent: Intent?): IBinder? {  return null  } } |

AndroidManifest.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <manifest xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools">   <!-- Permission required for foreground services -->  <uses-permission android:name="android.permission.FOREGROUND\_SERVICE" />   <application  android:allowBackup="true"  android:dataExtractionRules="@xml/data\_extraction\_rules"  android:fullBackupContent="@xml/backup\_rules"  android:icon="@mipmap/ic\_launcher"  android:label="@string/app\_name"  android:roundIcon="@mipmap/ic\_launcher\_round"  android:supportsRtl="true"  android:theme="@style/Theme.MusicPlayerApp"  tools:targetApi="31">   <activity  android:name=".MainActivity"  android:exported="true">  <intent-filter>  <action android:name="android.intent.action.MAIN" />  <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  </activity>   <service  android:name=".MusicService"  android:foregroundServiceType="mediaPlayback" />   </application> </manifest> |

**Output:**



**TASK 3**

MainActivity.kt

|  |
| --- |
| import android.content.Intent import android.content.IntentFilter import android.os.Bundle import android.widget.Button import androidx.appcompat.app.AppCompatActivity import androidx.localbroadcastmanager.content.LocalBroadcastManager import com.example.custombroadcastreceiverapp.R  class MainActivity : AppCompatActivity() {  private lateinit var myReceiver: MyCustomBroadcastReceiver  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  setContentView(R.layout.*activity\_main*)  myReceiver = MyCustomBroadcastReceiver()  var sendBroadcastButton : Button = findViewById(R.id.*sendBroadcastButton*)  sendBroadcastButton.setOnClickListener **{** // Create an intent with the custom action  val intent = Intent("com.example.CUSTOM\_ACTION")  // Send the broadcast  LocalBroadcastManager.getInstance(this).sendBroadcast(intent)  **}** }  override fun onStart() {  super.onStart()  // Register the receiver with the intent filter  LocalBroadcastManager.getInstance(this)  .registerReceiver(myReceiver,  IntentFilter("com.example.CUSTOM\_ACTION"))  }  override fun onStop() {  super.onStop()  // Unregister the receiver to prevent memory leaks   LocalBroadcastManager.getInstance(this).unregisterReceiver(myReceiver)  } } |

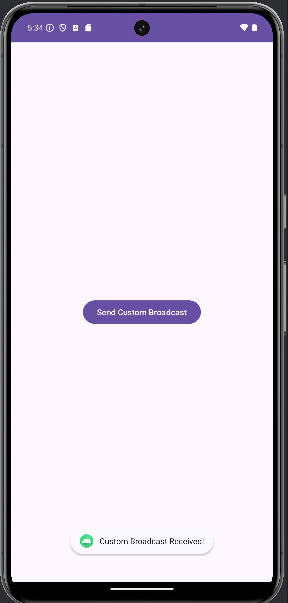
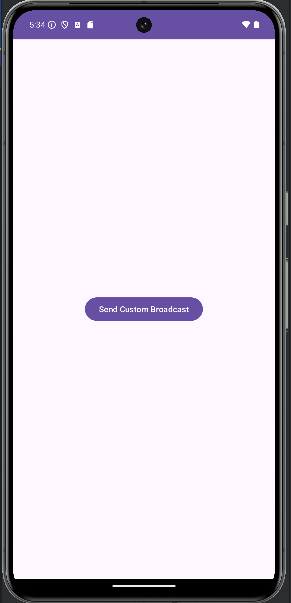
Activity\_main.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context="MainActivity">  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">  <Button  android:id="@+id/sendBroadcastButton"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Send Custom Broadcast"  android:layout\_centerInParent="true"/>  </RelativeLayout>  </androidx.constraintlayout.widget.ConstraintLayout> |

MyCustomBroadcastReceiver.kt

|  |
| --- |
| import android.content.BroadcastReceiver import android.content.Context import android.content.Intent import android.util.Log import android.widget.Toast class MyCustomBroadcastReceiver : BroadcastReceiver() {  private val TAG = "MyBroadcastReceiver"  override fun onReceive(context: Context, intent: Intent) {  // Check if the action matches the expected custom action  if (intent.*action* == "com.example.CUSTOM\_ACTION") {  Log.d(TAG, "Custom Broadcast Received!")  Toast.makeText(context, "Custom Broadcast Received!",  Toast.*LENGTH\_SHORT*).show()  }  } } |

**Output:**



**Answer the following question**

1. **In the provided implementation, why is the custom action string("com.example.CUSTOM\_ACTION") used? What would happen if this string were changed in one place but not another?**

A broadcast event is uniquely identified within the application by the custom action string, which guarantees that only appropriate those receiving react. The receiver won't identify the broadcast and what you want to happen won't occur if this string is altered just once.

1. **What are the advantages of using LocalBroadcastManager over regular broadcasts?**

Security: Local broadcasts are not intercepted by other apps because they remain inside the app.

Performance: Because local broadcasts do not involve inter-process communication, they are quicker.

Memory Efficiency: Because in-app messaging uses fewer resources, they are optimized for it.

1. **Describe how you would modify the code to send a broadcast with additional data (using Intent.putExtra). Provide an example.**

To send data with a broadcast, use Intent.putExtra

Intent intent = new Intent("com.example.CUSTOM\_ACTION");

intent.putExtra("EXTRA\_MESSAGE", "Hello from broadcast!");

LocalBroadcastManager.getInstance(this).sendBroadcast(intent);

1. **If you wanted to receive broadcasts from multiple activities, how would you modify the existing BroadcastReceiver setup?**

Register in Every Activity: To manage the broadcast separately, register the same BroadcastReceiver in every activity.  
  
Centralized BroadcastReceiver: To centralize broadcast handling in one location, create a BroadcastReceiver in a BaseActivity that all other activities extend.