# 实验7 Service

## 一、实验目的

* 熟练掌握Service。

## 二、实验要求

* 实现上述的基本功能要求。说明：商品内容可以写在代码中。在此基础上，可以进一步优化界面，也可以进一步添加其他功能。
* 实验报告提交内容：1）结合系统界面（截图）介绍APP功能；2）核心实现代码（Java或Kotlin代码）和布局文件等。

## 三、实验内容

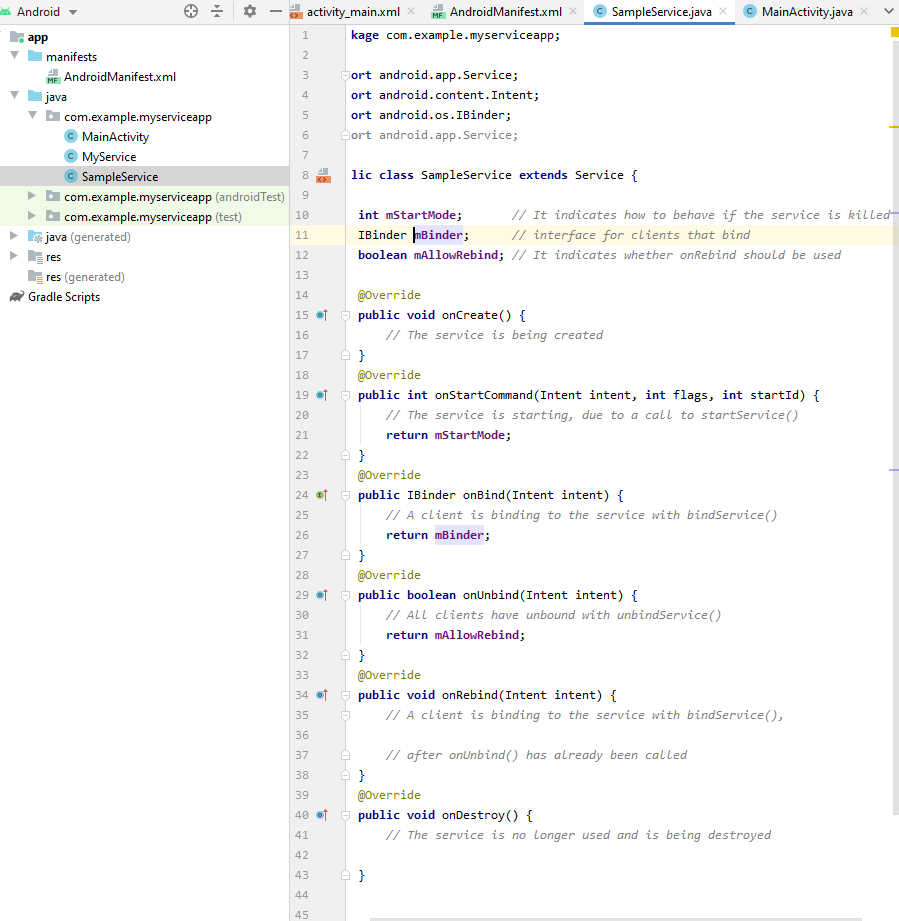
Step1:

Working with the Android Skeleton Service

Following is the skeleton **service** that describes about each of the lifecycle methods.

SampleService.java

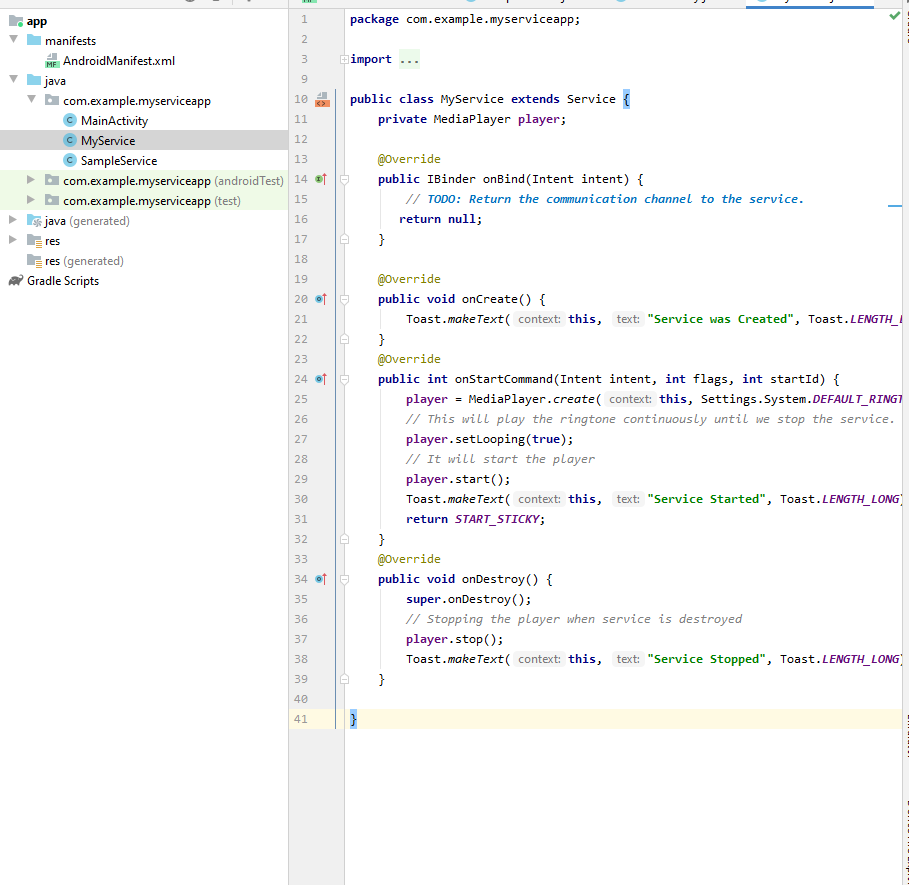
**package** com.example.myserviceapp;  
  
**import** android.app.Service;  
**import** android.content.Intent;  
**import** android.os.IBinder;  
**import** android.app.Service;  
  
**public class** SampleService **extends** Service {  
  
 **int mStartMode**; *// It indicates how to behave if the service is killed* IBinder **mBinder**; *// interface for clients that bind* **boolean mAllowRebind**; *// It indicates whether onRebind should be used* @Override  
 **public void** onCreate() {  
 *// The service is being created* }  
 @Override  
 **public int** onStartCommand(Intent intent, **int** flags, **int** startId) {  
 *// The service is starting, due to a call to startService()* **return mStartMode**;  
 }  
 @Override  
 **public** IBinder onBind(Intent intent) {  
 *// A client is binding to the service with bindService()* **return mBinder**;  
 }  
 @Override  
 **public boolean** onUnbind(Intent intent) {  
 *// All clients have unbound with unbindService()* **return mAllowRebind**;  
 }  
 @Override  
 **public void** onRebind(Intent intent) {  
 *// A client is binding to the service with bindService(),  
  
 // after onUnbind() has already been called* }  
 @Override  
 **public void** onDestroy() {  
 *// The service is no longer used and is being destroyed* }  
  
  
}



Step2:

Working with the MyService.java

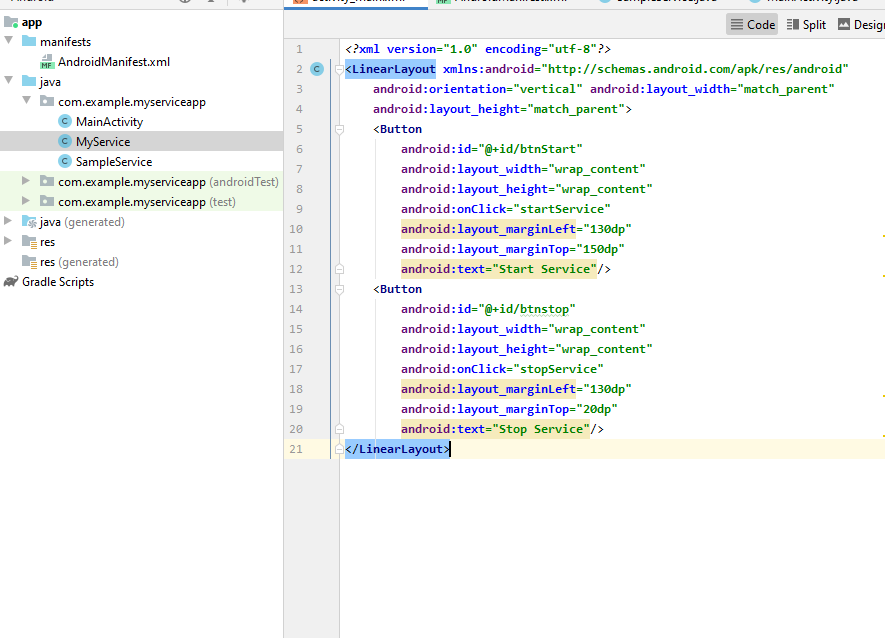
**package** com.example.myserviceapp;  
  
**import** android.app.Service;  
**import** android.content.Intent;  
**import** android.media.MediaPlayer;  
**import** android.os.IBinder;  
**import** android.provider.Settings;  
**import** android.widget.Toast;  
  
**public class** MyService **extends** Service {  
 **private** MediaPlayer **player**;  
  
 @Override  
 **public** IBinder onBind(Intent intent) {  
 *//* ***TODO: Return the communication channel to the service.* return null**;  
 }  
  
 @Override  
 **public void** onCreate() {  
 Toast.*makeText*(**this**, **"Service was Created"**, Toast.***LENGTH\_LONG***).show();  
 }  
 @Override  
 **public int** onStartCommand(Intent intent, **int** flags, **int** startId) {  
 **player** = MediaPlayer.*create*(**this**, Settings.System.***DEFAULT\_RINGTONE\_URI***);  
 *// This will play the ringtone continuously until we stop the service.* **player**.setLooping(**true**);  
 *// It will start the player* **player**.start();  
 Toast.*makeText*(**this**, **"Service Started"**, Toast.***LENGTH\_LONG***).show();  
 **return *START\_STICKY***;  
 }  
 @Override  
 **public void** onDestroy() {  
 **super**.onDestroy();  
 *// Stopping the player when service is destroyed* **player**.stop();  
 Toast.*makeText*(**this**, **"Service Stopped"**, Toast.***LENGTH\_LONG***).show();  
 }  
  
}



Step3:

Working with the Activity\_main.xml

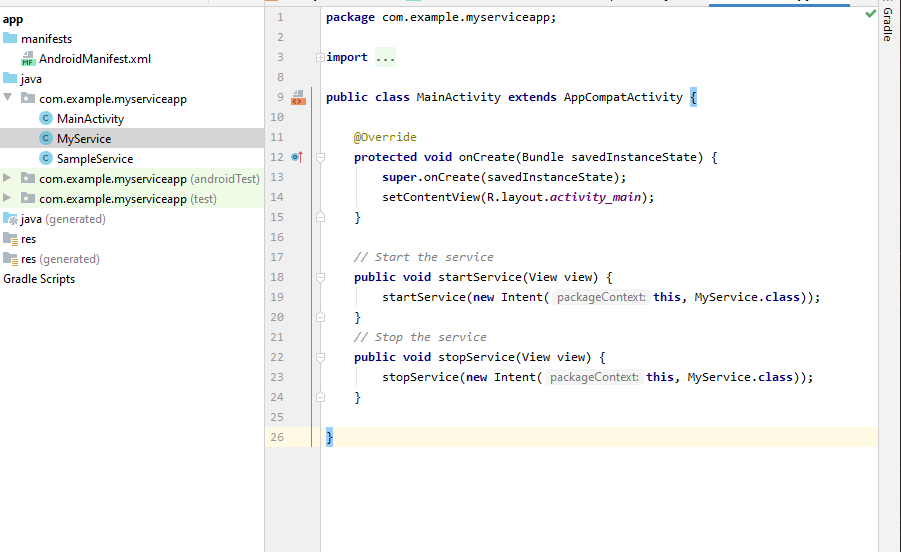
*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**Button  
 android:id="@+id/btnStart"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="startService"  
 android:layout\_marginLeft="130dp"  
 android:layout\_marginTop="150dp"  
 android:text="Start Service"**/>  
 <**Button  
 android:id="@+id/btnstop"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="stopService"  
 android:layout\_marginLeft="130dp"  
 android:layout\_marginTop="20dp"  
 android:text="Stop Service"**/>  
</**LinearLayout**>



Step4:

Working with the MainActivity.java

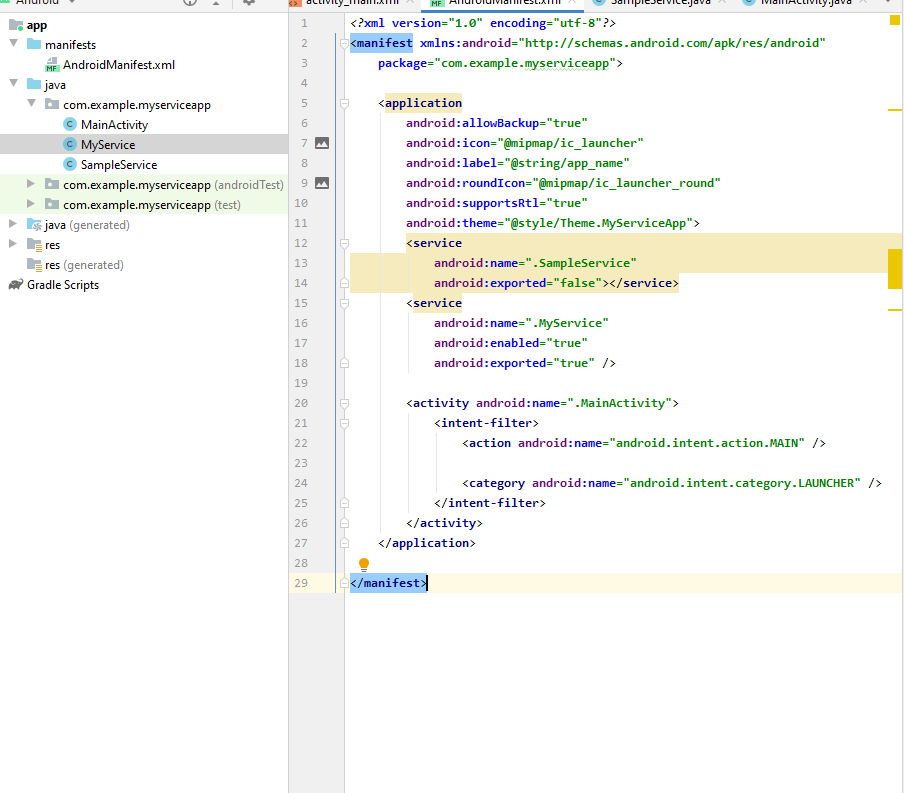
**package** com.example.myserviceapp;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.View;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 }  
  
 *// Start the service* **public void** startService(View view) {  
 startService(**new** Intent(**this**, MyService.**class**));  
 }  
 *// Stop the service* **public void** stopService(View view) {  
 stopService(**new** Intent(**this**, MyService.**class**));  
 }  
  
}



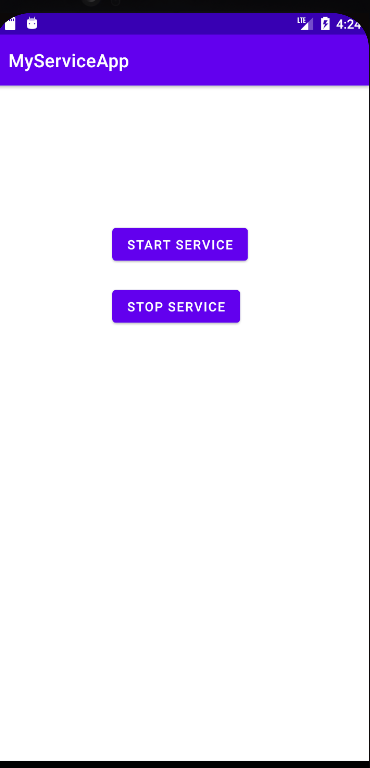
Step5:

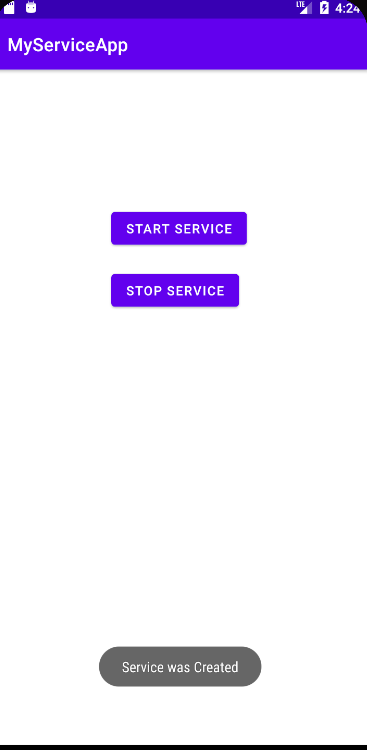
## Working with the AndroidManifest.xml

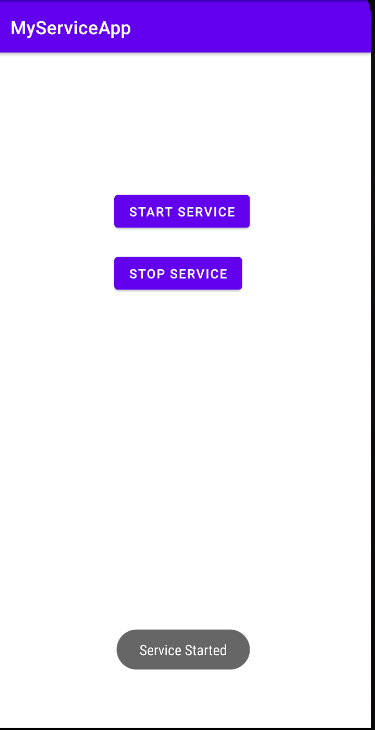
*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.myserviceapp"**>  
  
 <**application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.MyServiceApp"**>  
 <**service  
 android:name=".SampleService"  
 android:exported="false"**></**service**>  
 <**service  
 android:name=".MyService"  
 android:enabled="true"  
 android:exported="true"** />  
  
 <**activity android:name=".MainActivity"**>  
 <**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
  
 <**category android:name="android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
 </**application**>  
  
</**manifest**>

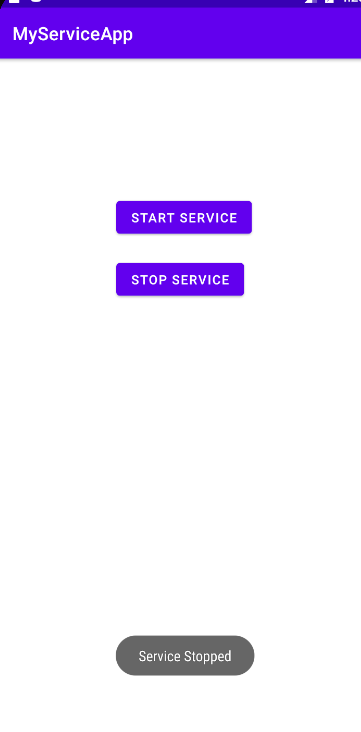


**四，实验结果：**

****

****

****

****