

res/values/dimens.xml

<resources>  
 <dimen name="margin\_base">5dp</dimen>  
 <dimen name="margin\_basex2">10dp</dimen>  
 <dimen name="text\_medium">16sp</dimen>  
</resources>

res/values/strings.xml

<resources>  
 <string name="app\_name">SmsReceiverApp</string>  
 <string name="you\_have\_a\_new\_message">Hey! You have a new message</string>  
</resources>

AndroiMainfest.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">  
 <uses-permission android:name="android.permission.RECEIVE\_SMS"/>  
 <uses-permission android:name="android.permission.READ\_SMS"/>  
  
 <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.Baitaptuan5">  
 <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>  
 </application>  
  
</manifest>

res/layout/activity\_main.xml

<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\_content"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Chua co tin nhan"  
 android:textSize="@dimen/text\_medium"/>  
  
</LinearLayout>

MainActivity.java

package com.example.baitaptuan5;  
  
import android.Manifest;  
import android.content.BroadcastReceiver;  
import android.content.Context;  
import android.content.Intent;  
import android.content.IntentFilter;  
import android.content.pm.PackageManager;  
import android.os.Build;  
import android.os.Bundle;  
import android.telephony.SmsMessage;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final int *SMS\_PERMISSION\_CODE* = 100;  
 private BroadcastReceiver broadcastReceiver;  
 private IntentFilter filter;  
 private TextView tvContent;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 tvContent = findViewById(R.id.*tv\_content*);  
  
 // Xin quyền SMS runtime  
 if (ContextCompat.*checkSelfPermission*(this, Manifest.permission.*RECEIVE\_SMS*)  
 != PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this,  
 new String[]{Manifest.permission.*RECEIVE\_SMS*, Manifest.permission.*READ\_SMS*},  
 *SMS\_PERMISSION\_CODE*);  
 }  
  
 initBroadcastReceiver();  
 }  
  
 private void initBroadcastReceiver() {  
 // Khởi tạo filter để lắng nghe tin nhắn đến  
 filter = new IntentFilter("android.provider.Telephony.SMS\_RECEIVED");  
  
 // Tạo broadcastReceiver  
 broadcastReceiver = new BroadcastReceiver() {  
 @Override  
 public void onReceive(Context context, Intent intent) {  
 processReceive(context, intent);  
 }  
 };  
 }  
  
 // Xử lý tin nhắn đến  
 public void processReceive(Context context, Intent intent) {  
 Toast.*makeText*(context, getString(R.string.*you\_have\_a\_new\_message*),  
 Toast.*LENGTH\_SHORT*).show();  
  
 Bundle bundle = intent.getExtras();  
 if (bundle != null) {  
 Object[] pdus = (Object[]) bundle.get("pdus");  
 String format = bundle.getString("format"); // format chỉ có từ API 23  
 if (pdus != null) {  
 StringBuilder smsBuilder = new StringBuilder();  
 for (Object pdu : pdus) {  
 SmsMessage sms;  
 if (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*M*) {  
 sms = SmsMessage.*createFromPdu*((byte[]) pdu, format);  
 } else {  
 sms = SmsMessage.*createFromPdu*((byte[]) pdu);  
 }  
  
 String body = sms.getMessageBody();  
 String address = sms.getDisplayOriginatingAddress();  
  
 smsBuilder.append("From: ")  
 .append(address)  
 .append("\nMessage: ")  
 .append(body)  
 .append("\n\n");  
 }  
 tvContent.setText(smsBuilder.toString());  
 }  
 }  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 if (broadcastReceiver == null) {  
 initBroadcastReceiver();  
 }  
 registerReceiver(broadcastReceiver, filter);  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 unregisterReceiver(broadcastReceiver);  
 }  
  
 // Nhận kết quả xin quyền  
 @Override  
 public void onRequestPermissionsResult(int requestCode,  
 @NonNull String[] permissions,  
 @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == *SMS\_PERMISSION\_CODE*) {  
 if (grantResults.length > 0  
 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
 Toast.*makeText*(this, "Permission granted!", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Permission denied!", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
}