

**Ex. No. : 07**

**Date: 24/04/2025**

**Register No.: 221701023**

**Name: JANAKIRAMAN K**

---

## **Telephony services**

### **Aim**

Implement an application to get Telephony services.

### ***Procedure:***

***Step 1 : File -> NewProject***

*Provide the application name and Click “Next”*

***Step 2 : Select the target android devices***

*Select the minimum SDK to run the application. Click “Next”.*

***Step 3 : Choose the activity for the application (By default choose “Blank Activity”).***

*Click “Next”.*

***Step 4 : Enter activity name and click.***

***Step 5 : Edit the program.***

***Step 6 : Run the application, 2-ways to run the application.***

*1. Running through emulator*

*2. Running through mobile device*

### ***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">
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />

    <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:supportRtl="true"
        android:theme="@style/Theme._7exp23"
        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>

    </application>

</manifest>
```

### ***Activity\_main.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Fetching Telephony Info..."
        android:textSize="18sp"
        android:textColor="#000000"/>
</RelativeLayout>
```

### ***MainActivity.kt***

```
package com.example.a7exp23

import android.Manifest
import android.content.pm.PackageManager
import android.os.Bundle
import android.telephony.TelephonyManager
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat

class MainActivity : AppCompatActivity() {

    private val PERMISSION_REQUEST_CODE = 101

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        if (ContextCompat.checkSelfPermission(this, Manifest.permission.READ_PHONE_STATE)
            != PackageManager.PERMISSION_GRANTED
        ) {
            ActivityCompat.requestPermissions(
                this,
```

```

        arrayOf(Manifest.permission.READ_PHONE_STATE),
        PERMISSION_REQUEST_CODE
    )
} else {
    displayTelephonyInfo()
}
}

override fun onRequestPermissionsResult(
    requestCode: Int, permissions: Array<out String>, grantResults: IntArray
) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)

    if (requestCode == PERMISSION_REQUEST_CODE && grantResults.isNotEmpty()
        && grantResults[0] == PackageManager.PERMISSION_GRANTED
    ) {
        displayTelephonyInfo()
    } else {
        findViewById<TextView>(R.id.textView).text = "Permission Denied"
    }
}

private fun displayTelephonyInfo() {
    val telephonyManager = getSystemService(TELEPHONY_SERVICE) as
TelephonyManager

    val info = StringBuilder()
    info.append("Network Operator: ${telephonyManager.networkOperatorName}\n")
    info.append("SIM Country: ${telephonyManager.simCountryIso.uppercase()}\n")
    info.append("Phone Type: ${getPhoneTypeString(telephonyManager.phoneType)}\n")


    // Instead of IMEI, use ANDROID_ID which works on all versions
    val androidId = android.provider.Settings.Secure.getString(
        contentResolver,
        android.provider.Settings.Secure.ANDROID_ID
    )
    info.append("Android ID: $androidId\n")

    findViewById<TextView>(R.id.textView).text = info.toString()
}

```

```
private fun getPhoneTypeString(type: Int): String {  
    return when (type) {  
        TelephonyManager.PHONE_TYPE_GSM -> "GSM"  
        TelephonyManager.PHONE_TYPE_CDMA -> "CDMA"  
        TelephonyManager.PHONE_TYPE_SIP -> "SIP"  
        TelephonyManager.PHONE_TYPE_NONE -> "None"  
        else -> "Unknown"  
    }  
}  
  
}
```

## ***Output***



Network Operator: T-Mobile  
SIM Country: US  
Phone Type: GSM  
Android ID: 97c271e90b663bb6

## **Result:**

**The Telephony Services experiment has been successfully completed**