

Practical: 9

AIM- What is Difference between ListView and RecyclerView? Create SMS Android Application that can receive SMS by BroadcastReceiver and Interface, Send SMS and read & display all SMS of Inbox by using RecyclerView according to below instructions.

Submitted By: Dhruv.S.Lakhani
Enrollment number: 20012011055



**Ganpat
University**
॥ विद्यया समाजोत्कर्षः ॥

**U.V. Patel
College of
Engineering**

**Department of Computer
Engineering/Information Technology**

Practical-9

AIM:- What is Difference between ListView and RecyclerView? Create SMS Android Application that can receive SMS by BroadcastReceiver and Interface, Send SMS and read & display all SMS of Inbox by using RecyclerView according to below instructions.

Activity main.xml:-

```
<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:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<com.google.android.material.card.MaterialCardView
    android:id="@+id/card"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toTopOf="@id/l1"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:cardCornerRadius="7dp"
    app:cardElevation="7dp"
    android:layout_marginTop="10dp"
    android:layout_marginHorizontal="20dp">
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="500dp"
        android:backgroundTint="@color/cardview_shadow_end_color"/>
</com.google.android.material.card.MaterialCardView>
<LinearLayout
    android:id="@+id/l1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:weightSum="100"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent">
```

Practical: 9

```
app:layout_constraintTop_toBottomOf="@id/card"
app:layout_constraintBottom_toBottomOf="parent"
android:orientation="horizontal"
android:layout_margin="20dp">
```

```
<LinearLayout
    android:layout_weight="90"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/phoneNoInput"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Mobile No"
        app:endIconMode="custom"
        app:endIconDrawable="@drawable/ic_baseline_phone_24"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/phoneNo"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>

    </com.google.android.material.textfield.TextInputLayout>
    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/msgInput"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Message"
        app:endIconMode="custom"
        app:endIconDrawable="@drawable/ic_baseline_message_24"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/msg"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>

    </com.google.android.material.textfield.TextInputLayout>
</LinearLayout>
<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/sendButton"
    app:fabSize="normal"
    android:layout_marginTop="35dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginHorizontal="10dp"
    android:src="@drawable/ic_baseline_send_24"/>
```

```
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Sms item view.xml:-

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:layout_marginHorizontal="10dp">

    <ImageView
        android:id="@+id/icon"
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:padding="5dp"
        android:background="@drawable/shape"
        android:src="@drawable/ic_baseline_person_24"/>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <TextView
            android:id="@+id/textViewPhoneNo"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Medium Text"
            android:textStyle="bold"
            android:textAppearance="?android:attr/textAppearanceMedium"
            android:layout_marginLeft="10dp"
            android:layout_marginTop="5dp"
            android:padding="2dp"
            android:textColor="#4d4d4d" />

        <TextView
            android:id="@+id/textViewMessage"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="TextView"
            android:layout_marginLeft="10dp"/>

    </LinearLayout>
</LinearLayout>
```

MainActivity.kt:-

```
package com.example.madpractical9_20012011055
```

```
import android.Manifest
import android.content.IntentFilter
import android.content.pm.PackageManager
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.Telephony
import android.telephony.SmsManager
import android.widget.ListView
import androidx.appcompat.app.AlertDialog
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import com.example.madpractical9_20012011055.databinding.ActivityMainBinding
```

```
class MainActivity : AppCompatActivity() {
    private lateinit var binding : ActivityMainBinding
    private val SMS_PERMISSION_CODE = 100
    private lateinit var lv: ListView
    private lateinit var al: ArrayList<SMSView>
    private lateinit var smsReceiver:SMSBroadcastReceiver
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)
        lv = binding.listView
        al = ArrayList()

        if(checkRequestPermission()){
            loadSMSInbox()
        }
        smsReceiver = SMSBroadcastReceiver()
        registerReceiver(smsReceiver, IntentFilter(Telephony.Sms.Intents.SMS_RECEIVED_ACTION))
        smsReceiver.setListner(ListenerImplement())
        binding.sendButton.setOnClickListener{
            val phone = binding.phoneNo.text.toString()
            val msg = binding.msg.text.toString()
            sendSms(phone,msg)
        }
    }
    private fun sendSms(sPhoneNo: String?,sMsg: String?){
        if(!checkRequestPermission()){
            return
        }
        val smsmanager = SmsManager.getDefault()
    }
}
```

Practical: 9

```
if(smsmanager != null){
    smsmanager.sendTextMessage(sPhoneNo,null,sMsg,null,null)
}
}
inner class ListenerImplement:SMSBroadcastReceiver.Listener{
    override fun onTextReceived(sPhoneNo: String?, sMsg: String?) {
        val builder : AlertDialog.Builder = AlertDialog.Builder(this@MainActivity)
        builder.setTitle("New SMS Received")
        builder.setMessage("$sPhoneNo\n$sMsg")
        builder.setCancelable(true)
        builder.show()
        loadSMSInbox()
    }
}
private val isSMSReadPermission: Boolean
    get() = ContextCompat.checkSelfPermission(this, Manifest.permission.READ_SMS) ==
PackageManager.PERMISSION_GRANTED
private val isSMSWritePermission: Boolean
    get() = ContextCompat.checkSelfPermission(this, Manifest.permission.SEND_SMS) ==
PackageManager.PERMISSION_GRANTED

private fun requestSMSPermission() {
    if (ActivityCompat.shouldShowRequestPermissionRationale(this, Manifest.permission.READ_SMS)) {
        // You may display a non-blocking explanation here, read more in the documentation:
        // https://developer.android.com/training/permissions/requesting.html
    }
    ActivityCompat.requestPermissions(this, arrayOf(Manifest.permission.READ_SMS,
        Manifest.permission.SEND_SMS,
        Manifest.permission.RECEIVE_SMS),
        SMS_PERMISSION_CODE)
}
private fun checkRequestPermission(): Boolean {
    return if (!isSMSReadPermission || !isSMSWritePermission) {
        requestSMSPermission()
        false
    } else true
}
private fun loadSMSInbox() {
    if (!checkRequestPermission()) return
    val uriSMS = Uri.parse("content://sms/inbox")
    val c = contentResolver.query(uriSMS, null, null, null, null)
    al.clear()
    while (c!!.moveToNext()) {
        al.add(SMSView(c.getString(2),c.getString(12)))
    }
    lv.adapter = SMSViewAdapter(this,al)
}
override fun onDestroy() {
```

Practical: 9

```
    unregisterReceiver(smsReceiver)
    super.onDestroy()
}
}
```

BroadcastReceiver.kt:-

```
package com.example.madpractical9_20012011055
```

```
import android.content.BroadcastReceiver
import android.content.Context
import android.content.Intent
import android.os.Build
import android.provider.Telephony
```

```
class SMSBroadcastReceiver : BroadcastReceiver() {
    interface Listner {
        fun onTextReceived(sPhoneNo: String?, sMsg: String?)
    }
    private var listner: Listner? = null
    fun setListner(lis: Listner) {
        listner = lis
    }
    override fun onReceive(context: Context, intent: Intent) {
        if (intent.action == Telephony.Sms.Intents.SMS_RECEIVED_ACTION) {
            var sPhoneNo = ""
            var sSMSBody = ""
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
                for (smsMessage in Telephony.Sms.Intents.getMessagesFromIntent(intent)) {
                    sPhoneNo = smsMessage.displayOriginatingAddress
                    sSMSBody += smsMessage.messageBody
                }
                if (listner != null) {
                    listner?.onTextReceived(sPhoneNo, sSMSBody)
                }
            }
        }
    }
}
```

SMSView.kt:-

```
package com.example.madpractical9_20012011055
```

```
class SMSView(val phoneNo:String,val msg:String) {
}
```

SMSViewAdapter.kt:-

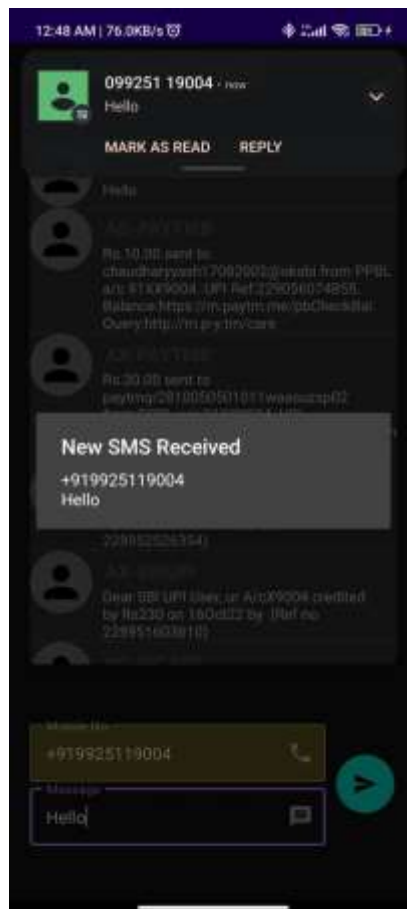
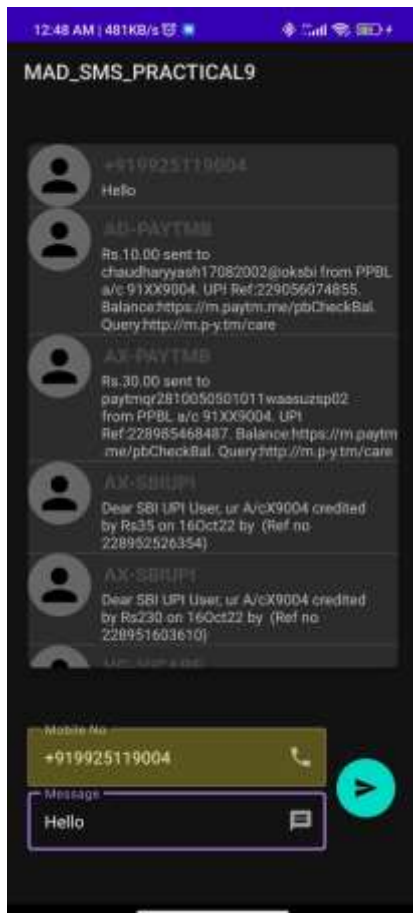
```
package com.example.madpractical9_20012011055
```

Practical: 9

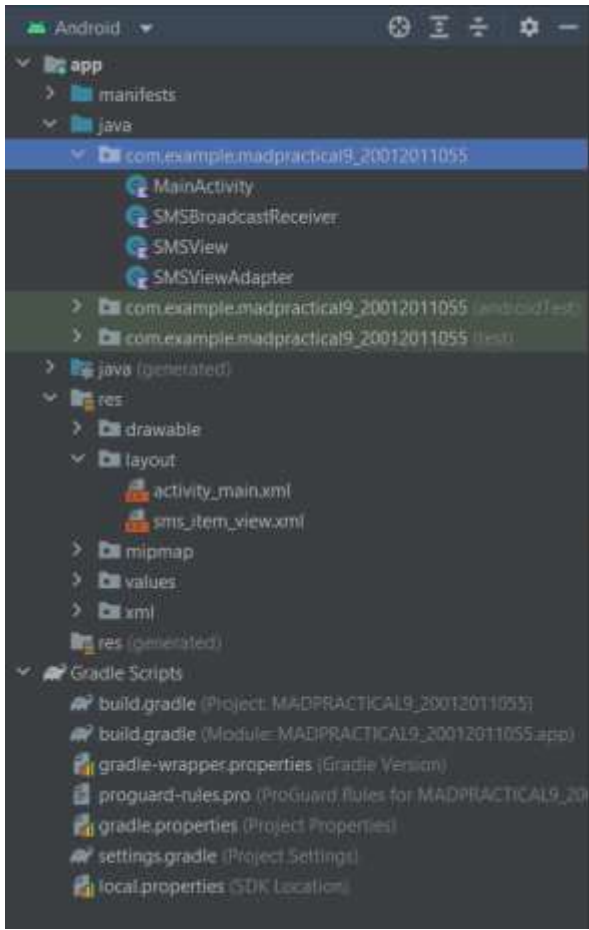
```
import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.AdapterView
import com.example.madpractical9_20012011055.databinding.SmsItemViewBinding

class SMSViewAdapter(context: Context, private val array: ArrayList<SMSView>):
    ArrayAdapter<SMSView>(context,array.size,array) {
    override fun getView(position: Int, convertView: View?, parent: ViewGroup): View {
        //val currentItemView:View = LayoutInflater.from(context).inflate(R.layout.sms_item_view, parent,
false)
        // get the position of the view from the ArrayAdapter
        val currentSms: SMSView? = getItem(position)
        val binding = SmsItemViewBinding.inflate(LayoutInflater.from(context))
        binding.textViewPhoneNo.text = currentSms!!.phoneNo
        binding.textViewMessage.text = currentSms.msg
        return binding.root
        //return super.getView(position, convertView, parent)
    }
}
```

Output:-



Practical: 9



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"
    package="com.example.madpractical9_20012011055">

    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    <uses-permission android:name="android.permission.READ_SMS" />
    <uses-permission android:name="android.permission.SEND_SMS" />

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="MAD_SMS_PRACTICAL9"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.MADPRACTICAL9_20012011055"
        tools:targetApi="31">
```

Practical: 9

```
<receiver
  android:name=".SMSBroadcastReceiver"
  android:enabled="true"
  android:exported="true">
</receiver>

<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>
```