MAD

EX 13

ANDROID MANIFEST

```
<?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.INTERNET"/>
    <uses-permission android:name="android.permission.RECORD_AUDIO"/>

    <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.Ex13"
            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 version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
                tools:context=".MainActivity">

    <EditText
            android:id="@+id/editText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter Text"
            android:layout_marginTop="16dp" android:layout_marginStart="16dp"
android:layout_marginEnd="16dp"
            android:layout_marginBottom="16dp"/>
    <Button
            android:id="@+id/button"
            android:layout_width="match_parent"
            android:layout_height="47dp"
            android:text="Speak"
            android:layout_below="@id/editText"
            android:layout_marginTop="57dp"
            android:layout_marginBottom="16dp"
    />

</RelativeLayout>
```

MAIN ACTIVITY

```
package com.example.ex13

import android.os.Bundle
import android.speech.tts.TextToSpeech
import android.widget.Button
import android.widget.EditText
import androidx.appcompat.app.AppCompatActivity
import java.util.*

class MainActivity : AppCompatActivity(), TextToSpeech.OnInitListener {
    private lateinit var textToSpeech: TextToSpeech

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

        textToSpeech = TextToSpeech(this, this)

        val editText = findViewById<EditText>(R.id.editText)
        val button = findViewById<Button>(R.id.button)

        button.setOnClickListener {
            val text = editText.text.toString()
            if (text.isNotEmpty()) {
                textToSpeech.speak(text, TextToSpeech.QUEUE_FLUSH, null, "")
            }
        }
    }

    override fun onInit(status: Int) {
        if (status == TextToSpeech.SUCCESS) {
            val result = textToSpeech.setLanguage(Locale.US)
            if (result == TextToSpeech.LANG_MISSING_DATA || result ==
TextToSpeech.LANG_NOT_SUPPORTED) {
                // Language data is missing or the language is not supported.
                // Handle error accordingly.
            }
        } else {
            // Initialization failed.
            // Handle error accordingly.
        }
    }

    override fun onDestroy() {
        if (textToSpeech.isSpeaking) {
            textToSpeech.stop()
        }
        textToSpeech.shutdown()
        super.onDestroy()
    }
}
```

EX 14

ANDROID MANIFEST

```
<?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.RECORD_AUDIO" />
    <uses-permission android:name="android.permission.INTERNET" />

    <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.Ex14"
            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 version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
                tools:context=".MainActivity">

    <ImageButton
            android:id="@+id/micButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:src="@android:drawable/ic_btn_speak_now"/>

    <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/micButton"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"
            android:text="Output appears here..."
            android:textSize="20sp" />

</RelativeLayout>
```

MAIN ACTIVITY

```
package com.example.ex14

import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.speech.RecognitionListener
import android.speech.RecognizerIntent
import android.speech.SpeechRecognizer
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.ImageButton
import android.widget.TextView
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import java.util.*

class MainActivity : AppCompatActivity() {

    private lateinit var micButton: ImageButton
    private lateinit var textView: TextView
    private lateinit var speechRecognizer: SpeechRecognizer

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

        micButton = findViewById(R.id.micButton)
        textView = findViewById(R.id.textView)

        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.RECORD_AUDIO)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.RECORD_AUDIO),
                1
            )
        }

        micButton.setOnClickListener {
            startListening()
        }

        speechRecognizer = SpeechRecognizer.createSpeechRecognizer(this)
        speechRecognizer.setRecognitionListener(object : RecognitionListener {
            override fun onReadyForSpeech(params: Bundle?) {}
            override fun onBeginningOfSpeech() {}
            override fun onRmsChanged(rmsdB: Float) {}
            override fun onBufferReceived(buffer: ByteArray?) {}
            override fun onEndOfSpeech() {}
            override fun onError(error: Int) {
                textView.text = "Error: $error"
            }
            override fun onResults(results: Bundle?) {
                val result =
results?.getStringArrayList(SpeechRecognizer.RESULTS_RECOGNITION)
                textView.text = result?.get(0) ?: ""
            }
            override fun onPartialResults(partialResults: Bundle?) {}
            override fun onEvent(eventType: Int, params: Bundle?) {}
        })
    }
    private fun startListening() {
        val intent = Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH)
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
            RecognizerIntent.LANGUAGE_MODEL_FREE_FORM)
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault())
        speechRecognizer.startListening(intent)
    }

    override fun onDestroy() {
        super.onDestroy()
        speechRecognizer.destroy()
    }
}
```

EX 15

ANDROID MANIFEST

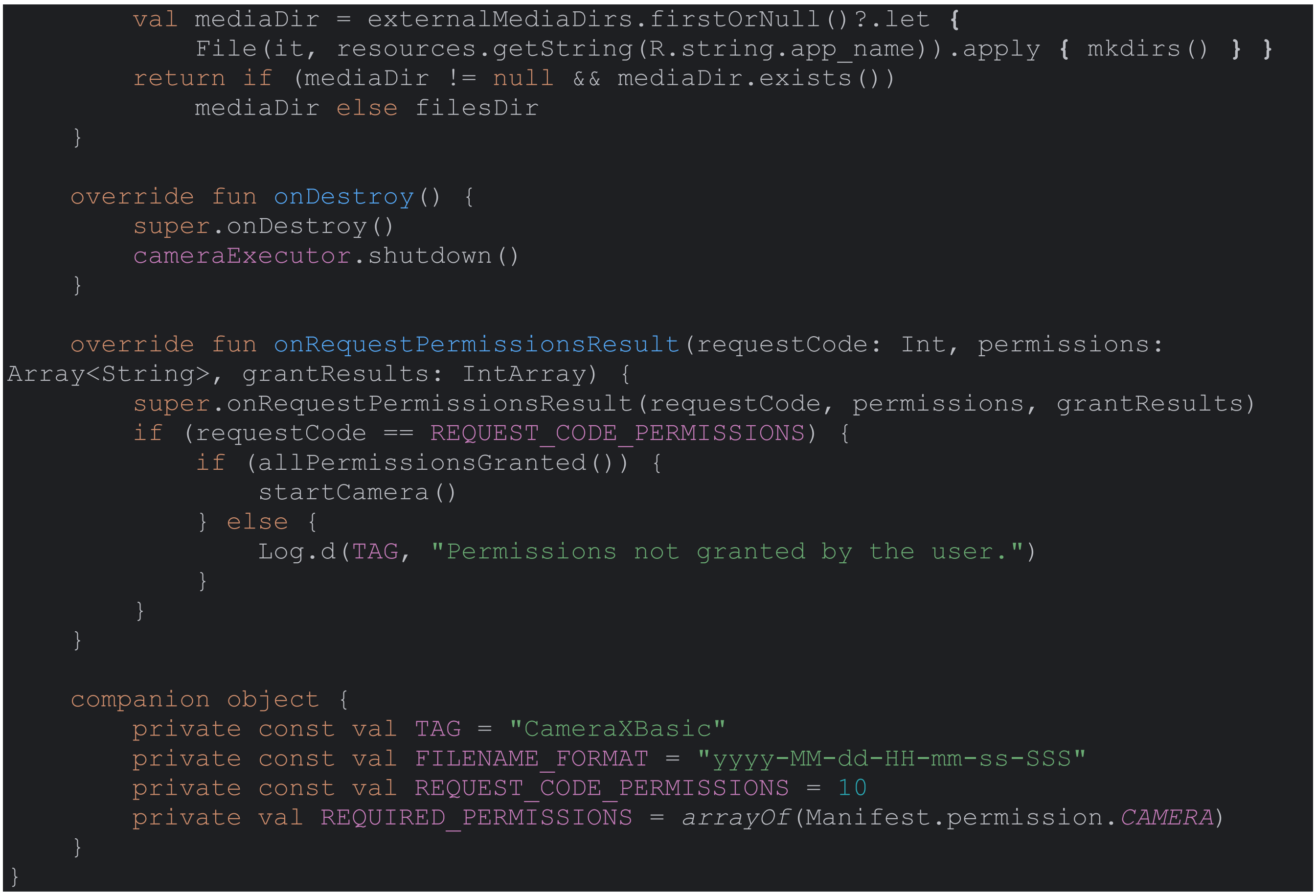
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-feature android:name="android.hardware.camera"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>


    <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.Ex15"
            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 version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
                tools:context=".MainActivity">

    <ImageView
            android:id="@+id/imageView"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

            android:layout_marginTop="16dp"
            android:adjustViewBounds="true"
            android:scaleType="fitCenter"
            android:src="@android:drawable/ic_menu_camera"
    />

    <Button
            android:id="@+id/btnTakePicture"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Take Picture"
            android:layout_below="@id/imageView"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"/>
    <androidx.camera.view.PreviewView
            android:id="@+id/previewView"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:visibility="invisible"/>

</RelativeLayout>
```

MAIN ACTIVITY

```
package com.example.ex15

import android.Manifest
import android.content.pm.PackageManager
import android.os.Bundle
import android.util.Log
import android.widget.Button
import android.widget.ImageView
import androidx.activity.result.contract.ActivityResultContracts
import androidx.appcompat.app.AppCompatActivity
import androidx.camera.core.*
import androidx.camera.lifecycle.ProcessCameraProvider
import androidx.camera.view.PreviewView
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import androidx.core.net.toUri
import java.io.File
import java.text.SimpleDateFormat
import java.util.*
import java.util.concurrent.ExecutorService
import java.util.concurrent.Executors

class MainActivity : AppCompatActivity() {

    private lateinit var btnTakePicture: Button
    private lateinit var imageView: ImageView
    private lateinit var outputDirectory: File
    private lateinit var cameraExecutor: ExecutorService
    private lateinit var previewView: PreviewView

    private var imageCapture: ImageCapture? = null

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

        btnTakePicture = findViewById(R.id.btnTakePicture)
        imageView = findViewById(R.id.imageView)
        previewView=findViewById(R.id.previewView)

        if (allPermissionsGranted()) {
            startCamera()
        } else {
            ActivityCompat.requestPermissions(
                this, REQUIRED_PERMISSIONS, REQUEST_CODE_PERMISSIONS)
        }

        btnTakePicture.setOnClickListener { takePhoto() }

        outputDirectory = getOutputDirectory()
        cameraExecutor = Executors.newSingleThreadExecutor()
    }

    private fun takePhoto() {
        val imageCapture = imageCapture ?: return

        val photoFile = File(
            outputDirectory,
            SimpleDateFormat(FILENAME_FORMAT, Locale.US)
                .format(System.currentTimeMillis()) + ".jpg")

        val outputOptions =
ImageCapture.OutputFileOptions.Builder(photoFile).build()

        imageCapture.takePicture(
            outputOptions, ContextCompat.getMainExecutor(this), object :
ImageCapture.OnImageSavedCallback {
                override fun onError(exception: ImageCaptureException) {
                    Log.e(TAG, "Photo capture failed: ${exception.message}",
exception)
                }

                override fun onImageSaved(outputFileResults:
ImageCapture.OutputFileResults) {
                    val savedUri = outputFileResults.savedUri ?: photoFile.toUri()
                    Log.d(TAG, "Photo capture succeeded: $savedUri")
                    imageView.setImageURI(savedUri)
                }
            })
    }

    private fun startCamera() {
        val cameraProviderFuture = ProcessCameraProvider.getInstance(this)

        cameraProviderFuture.addListener({
            val cameraProvider: ProcessCameraProvider = cameraProviderFuture.get()

            val preview = Preview.Builder()
                .build()
                .also {
                    it.setSurfaceProvider(previewView.surfaceProvider)
                }

            imageCapture = ImageCapture.Builder()
                .build()

            val cameraSelector = CameraSelector.DEFAULT_BACK_CAMERA

            try {
                cameraProvider.unbindAll()
                cameraProvider.bindToLifecycle(
                    this, cameraSelector, preview, imageCapture)
            } catch (exc: Exception) {
                Log.e(TAG, "Use case binding failed", exc)
            }

        }, ContextCompat.getMainExecutor(this))
    }

    private fun allPermissionsGranted() = REQUIRED_PERMISSIONS.all {
        ContextCompat.checkSelfPermission(
            baseContext, it) == PackageManager.PERMISSION_GRANTED
    }

    private fun getOutputDirectory(): File {
```



```
        val mediaDir = externalMediaDirs.firstOrNull()?.let {
            File(it, resources.getString(R.string.app_name)).apply { mkdirs() } }
        return if (mediaDir != null && mediaDir.exists())
            mediaDir else filesDir
    }

    override fun onDestroy() {
        super.onDestroy()
        cameraExecutor.shutdown()
    }

    override fun onRequestPermissionsResult(requestCode: Int, permissions:
Array<String>, grantResults: IntArray) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults)
        if (requestCode == REQUEST_CODE_PERMISSIONS) {
            if (allPermissionsGranted()) {
                startCamera()
            } else {
                Log.d(TAG, "Permissions not granted by the user.")
            }
        }
    }

    companion object {
        private const val TAG = "CameraXBasic"
        private const val FILENAME_FORMAT = "yyyy-MM-dd-HH-mm-ss-SSS"
        private const val REQUEST_CODE_PERMISSIONS = 10
        private val REQUIRED_PERMISSIONS = arrayOf(Manifest.permission.CAMERA)
    }
}
```

EX 12

ANDROID MANIFEST

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <queries>
        <package android:name="com.google.android.gm" />
    </queries>


    <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.Ex12"
            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>
```

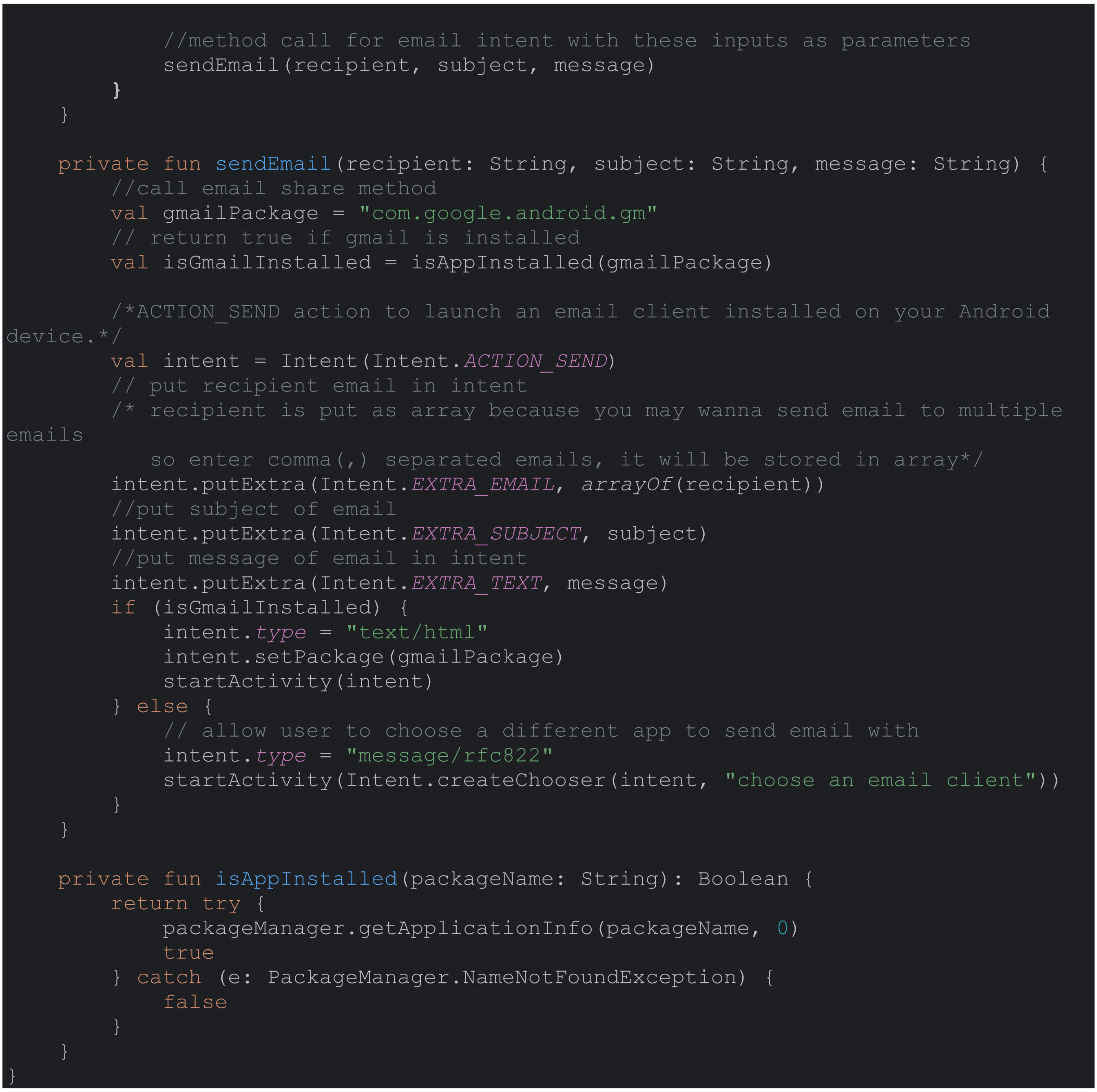
ACTIVITY MAIN

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
              android:orientation="vertical"
              android:padding="10dp"
              tools:context=".MainActivity">

    <!--EditText: Input the recipient-->
    <EditText
            android:id="@+id/recipientEt"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Recipient email(s)"
            android:inputType="textEmailAddress"
            android:padding="10dp"
             />

    <!--EditText: Input the subject of email-->
    <EditText
            android:id="@+id/subjectEt"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Subject"
```

```
        android:layout_marginTop="2dp"
        android:layout_marginBottom="2dp"
        android:inputType="text|textCapSentences"
        android:padding="10dp"
         />

    <!--EditText: Input the message-->
    <EditText
        android:id="@+id/messageEt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="start"
        android:hint="Enter message here..."
        android:inputType="text|textCapSentences"
        android:minHeight="150dp"
        android:padding="10dp"
         />

    <!--Button:   Launch existing email clients to send email-->
    <Button
        android:id="@+id/sendEmailBtn"
        style="@style/Base.Widget.AppCompat.Button.Colored"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="end"
        android:text="send Email" />

</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.ex12

import android.content.Intent
import android.content.pm.PackageManager
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    //declare views
    private lateinit var mRecipientEt: EditText
    private lateinit var mSubjectEt: EditText
    private lateinit var mMessageEt: EditText
    private lateinit var mSendEmailBtn: Button

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

        //initializing views with activity_main.xml
        mRecipientEt = findViewById(R.id.recipientEt)
        mSubjectEt = findViewById(R.id.subjectEt)
        mMessageEt = findViewById(R.id.messageEt)
        mSendEmailBtn = findViewById(R.id.sendEmailBtn)

        //button click to get input and call sendEmail method
        mSendEmailBtn.setOnClickListener {
            //get input from EditTexts and save in variables
            val recipient = mRecipientEt.text.toString().trim()
            val subject = mSubjectEt.text.toString().trim()
            val message = mMessageEt.text.toString().trim()
```



```
            //method call for email intent with these inputs as parameters
            sendEmail(recipient, subject, message)
        }
    }

    private fun sendEmail(recipient: String, subject: String, message: String) {
        //call email share method
        val gmailPackage = "com.google.android.gm"
        // return true if gmail is installed
        val isGmailInstalled = isAppInstalled(gmailPackage)

        /*ACTION_SEND action to launch an email client installed on your Android
device.*/
        val intent = Intent(Intent.ACTION_SEND)
        // put recipient email in intent
        /* recipient is put as array because you may wanna send email to multiple
emails
           so enter comma(,) separated emails, it will be stored in array*/
        intent.putExtra(Intent.EXTRA_EMAIL, arrayOf(recipient))
        //put subject of email
        intent.putExtra(Intent.EXTRA_SUBJECT, subject)
        //put message of email in intent
        intent.putExtra(Intent.EXTRA_TEXT, message)
        if (isGmailInstalled) {
            intent.type = "text/html"
            intent.setPackage(gmailPackage)
            startActivity(intent)
        } else {
            // allow user to choose a different app to send email with
            intent.type = "message/rfc822"
            startActivity(Intent.createChooser(intent, "choose an email client"))
        }
    }

    private fun isAppInstalled(packageName: String): Boolean {
        return try {
            packageManager.getApplicationInfo(packageName, 0)
            true
        } catch (e: PackageManager.NameNotFoundException) {
            false
        }
    }
}
```

EX 11

ANDROID MANIFEST

```
<?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.SEND_SMS" />
    <uses-feature android:name="android.hardware.telephony"
android:required="false" />

    <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.Ex11"
            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 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="vertical"
              android:padding="16dp">

    <EditText
            android:id="@+id/editTextPhoneNumber"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Phone Number" />

    <EditText
            android:id="@+id/editTextMessage"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Message" />

    <Button
            android:id="@+id/buttonSend"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Send" />

</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.ex11

import android.Manifest
import android.content.pm.PackageManager
import android.os.Bundle
import android.telephony.SmsManager
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat

class MainActivity : AppCompatActivity() {
    private lateinit var phoneNumberEditText: EditText
    private lateinit var messageEditText: EditText
    private lateinit var sendButton: Button

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

        phoneNumberEditText = findViewById(R.id.editTextPhoneNumber)
        messageEditText = findViewById(R.id.editTextMessage)
        sendButton = findViewById(R.id.buttonSend)

        sendButton.setOnClickListener {
            val phoneNumber = phoneNumberEditText.text.toString()
            val message = messageEditText.text.toString()

            if (ContextCompat.checkSelfPermission(this,
Manifest.permission.SEND_SMS)
                != PackageManager.PERMISSION_GRANTED
            ) {
                ActivityCompat.requestPermissions(
                    this,
                    arrayOf(Manifest.permission.SEND_SMS),
                    PERMISSION_REQUEST_CODE
                )
            } else {
                sendMessage(phoneNumber, message)
            }
        }
    }

    private fun sendMessage(phoneNumber: String, message: String) {
        try {
            val smsManager = SmsManager.getDefault()
            smsManager.sendTextMessage(phoneNumber, null, message, null, null)
            Toast.makeText(this, "Message sent successfully",
Toast.LENGTH_SHORT).show()
        } catch (e: Exception) {
            Toast.makeText(this, "Failed to send message",
Toast.LENGTH_SHORT).show()
            e.printStackTrace()
        }
    }

    override fun onRequestPermissionsResult(
        requestCode: Int,
        permissions: Array<out String>,
        grantResults: IntArray
    ) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults)
        if (requestCode == PERMISSION_REQUEST_CODE) {
            if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                val phoneNumber = phoneNumberEditText.text.toString()
                val message = messageEditText.text.toString()
                sendMessage(phoneNumber, message)
            } else {
                Toast.makeText(
                    this,
                    "Permission denied, message cannot be sent",
                    Toast.LENGTH_SHORT
                ).show()
            }
        }
    }

    companion object {
        const val PERMISSION_REQUEST_CODE = 100
    }
}
```

EX 10

ANDROID MANIFEST

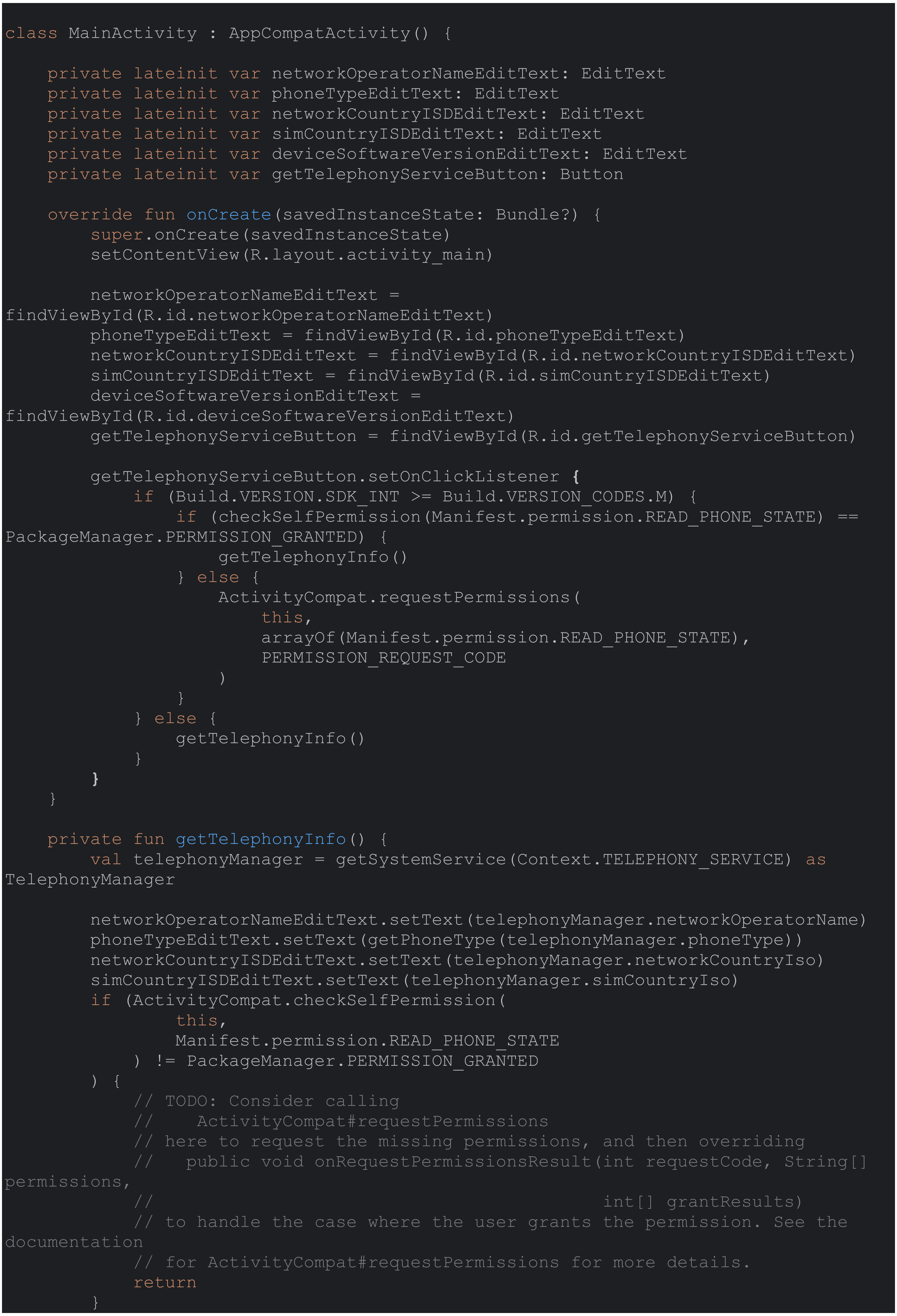
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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:supportsRtl="true"
            android:theme="@style/Theme.Ex10"
            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 version="1.0" encoding="utf-8"?>
<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="16dp"
              tools:context=".MainActivity">

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Network Operator Name:" />

    <EditText
            android:id="@+id/networkOperatorNameEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:enabled="false" />

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Phone Type:" />

    <EditText
            android:id="@+id/phoneTypeEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:enabled="false" />

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Network Country ISD:" />

    <EditText
            android:id="@+id/networkCountryISDEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:enabled="false" />

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="SIM Country ISD:" />

    <EditText
            android:id="@+id/simCountryISDEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:enabled="false" />

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Device Software Version:" />

    <EditText
            android:id="@+id/deviceSoftwareVersionEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:enabled="false" />

    <Button
            android:id="@+id/getTelephonyServiceButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:text="Get Telephony Service" />

</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.ex10

import android.Manifest
import android.content.Context
import android.content.pm.PackageManager
import android.os.Build
import android.os.Bundle
import android.telephony.TelephonyManager
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
```



```
class MainActivity : AppCompatActivity() {

    private lateinit var networkOperatorNameEditText: EditText
    private lateinit var phoneTypeEditText: EditText
    private lateinit var networkCountryISDEditText: EditText
    private lateinit var simCountryISDEditText: EditText
    private lateinit var deviceSoftwareVersionEditText: EditText
    private lateinit var getTelephonyServiceButton: Button

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

        networkOperatorNameEditText =
findViewById(R.id.networkOperatorNameEditText)
        phoneTypeEditText = findViewById(R.id.phoneTypeEditText)
        networkCountryISDEditText = findViewById(R.id.networkCountryISDEditText)
        simCountryISDEditText = findViewById(R.id.simCountryISDEditText)
        deviceSoftwareVersionEditText =
findViewById(R.id.deviceSoftwareVersionEditText)
        getTelephonyServiceButton = findViewById(R.id.getTelephonyServiceButton)

        getTelephonyServiceButton.setOnClickListener {
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
                if (checkSelfPermission(Manifest.permission.READ_PHONE_STATE) ==
PackageManager.PERMISSION_GRANTED) {
                    getTelephonyInfo()
                } else {
                    ActivityCompat.requestPermissions(
                        this,
                        arrayOf(Manifest.permission.READ_PHONE_STATE),
                        PERMISSION_REQUEST_CODE
                    )
                }
            } else {
                getTelephonyInfo()
            }
        }
    }

    private fun getTelephonyInfo() {
        val telephonyManager = getSystemService(Context.TELEPHONY_SERVICE) as
TelephonyManager

        networkOperatorNameEditText.setText(telephonyManager.networkOperatorName)
        phoneTypeEditText.setText(getPhoneType(telephonyManager.phoneType))
        networkCountryISDEditText.setText(telephonyManager.networkCountryIso)
        simCountryISDEditText.setText(telephonyManager.simCountryIso)
        if (ActivityCompat.checkSelfPermission(
                this,
                Manifest.permission.READ_PHONE_STATE
            ) != PackageManager.PERMISSION_GRANTED
        ) {
            // TODO: Consider calling
            //    ActivityCompat#requestPermissions
            // here to request the missing permissions, and then overriding
            //   public void onRequestPermissionsResult(int requestCode, String[]
permissions,
            //                                          int[] grantResults)
            // to handle the case where the user grants the permission. See the
documentation
            // for ActivityCompat#requestPermissions for more details.
            return
        }
deviceSoftwareVersionEditText.setText(telephonyManager.deviceSoftwareVersion)
    }

    private fun getPhoneType(phoneType: Int): String {
        return when (phoneType) {
            TelephonyManager.PHONE_TYPE_CDMA -> "CDMA"
            TelephonyManager.PHONE_TYPE_GSM -> "GSM"
            TelephonyManager.PHONE_TYPE_NONE -> "None"
            else -> "Unknown"
        }
    }

    override fun onRequestPermissionsResult(requestCode: Int, permissions:
Array<out String>, grantResults: IntArray) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults)
        if (requestCode == PERMISSION_REQUEST_CODE) {
            if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                getTelephonyInfo()
            } else {
                Toast.makeText(this, "Permission denied",
Toast.LENGTH_SHORT).show()
            }
        }
    }

    companion object {
        private const val PERMISSION_REQUEST_CODE = 100
    }
}
```

EX 9 (not working)

ANDROID MANIFEST

```
<?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.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS" />

    <uses-permission android:name="android.permission.VIBRATE"/>
    <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.Ex9"
            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>
        <receiver android:name=".AlarmReceiver"/>

    </application>

</manifest>
```

ACTIVITY MAIN

```
<?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="vertical">

    <!--Added Time picker just to pick the alarm time-->
    <!--gravity is aligned to center-->
    <TimePicker
            android:id="@+id/timePicker"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center" />

    <!--Added Toggle Button to set the alarm on or off-->
    <!--ByDefault toggleButton is set to false-->
    <ToggleButton
            android:id="@+id/toggleButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_margin="20dp"
            android:checked="false"
            android:onClick="OnToggleClicked" />
    <!--"OnToggleClicked" method will be implemented in MainActivity.java -->

</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.ex9

import android.app.AlarmManager
import android.app.PendingIntent
import android.content.Intent
import android.os.Bundle
import android.view.View
import android.widget.TimePicker
import android.widget.Toast
import android.widget.ToggleButton
import androidx.appcompat.app.AppCompatActivity
import java.util.*

class MainActivity : AppCompatActivity() {
    private lateinit var alarmTimePicker: TimePicker
    private lateinit var pendingIntent: PendingIntent
    private lateinit var alarmManager: AlarmManager

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

        alarmTimePicker = findViewById(R.id.timePicker)
        alarmManager = getSystemService(ALARM_SERVICE) as AlarmManager
    }

    // onToggleClicked() method is implemented for the time functionality
    fun onToggleClicked(view: View) {
        var time: Long
        if ((view as ToggleButton).isChecked) {
            Toast.makeText(this@MainActivity, "ALARM ON",
Toast.LENGTH_SHORT).show()
            val calendar = Calendar.getInstance()

            // calendar is called to get the current time in hour and minute
            calendar[Calendar.HOUR_OF_DAY] = alarmTimePicker.currentHour
            calendar[Calendar.MINUTE] = alarmTimePicker.currentMinute

            // using intent I have class AlarmReceiver class which inherits
            // BroadcastReceiver
            val intent = Intent(this, AlarmReceiver::class.java)

            // we call broadcast using pendingIntent
            pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0)

            time = calendar.timeInMillis - calendar.timeInMillis % 60000
            if (System.currentTimeMillis() > time) {
                // setting time as AM and PM
                if (calendar[Calendar.AM_PM] == 0) time += 1000 * 60 * 60 *
12.toLong()
                else time += 1000 * 60 * 60 * 24.toLong()
            }
            // Alarm rings continuously until the toggle button is turned off
            alarmManager.setRepeating(AlarmManager.RTC_WAKEUP, time, 10000,
pendingIntent)
            // alarmManager.set(AlarmManager.RTC_WAKEUP, System.currentTimeMillis()
+ (time * 1000), pendingIntent);
        } else {
            alarmManager.cancel(pendingIntent)
            Toast.makeText(this@MainActivity, "ALARM OFF",
Toast.LENGTH_SHORT).show()
        }
    }
}
```

ALARM RECEIVER.KT

```
package com.example.ex9
import android.content.BroadcastReceiver
import android.content.Context
import android.content.Intent
import android.media.Ringtone
import android.media.RingtoneManager
import android.net.Uri
import android.os.Build
import android.os.Vibrator
import android.widget.Toast

class AlarmReceiver : BroadcastReceiver() {
    override fun onReceive(context: Context, intent: Intent) {
        // We will use vibrator first
        val vibrator = context.getSystemService(Context.VIBRATOR_SERVICE) as
Vibrator
        vibrator.vibrate(4000)

        Toast.makeText(context, "Alarm! Wake up! Wake up!",
Toast.LENGTH_LONG).show()
        var alarmUri: Uri? =
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM)
        if (alarmUri == null) {
            alarmUri =
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION)
        }

        // Setting default ringtone
        val ringtone: Ringtone = RingtoneManager.getRingtone(context, alarmUri)

        // Play ringtone
        ringtone.play()
    }
}
```

EX 8

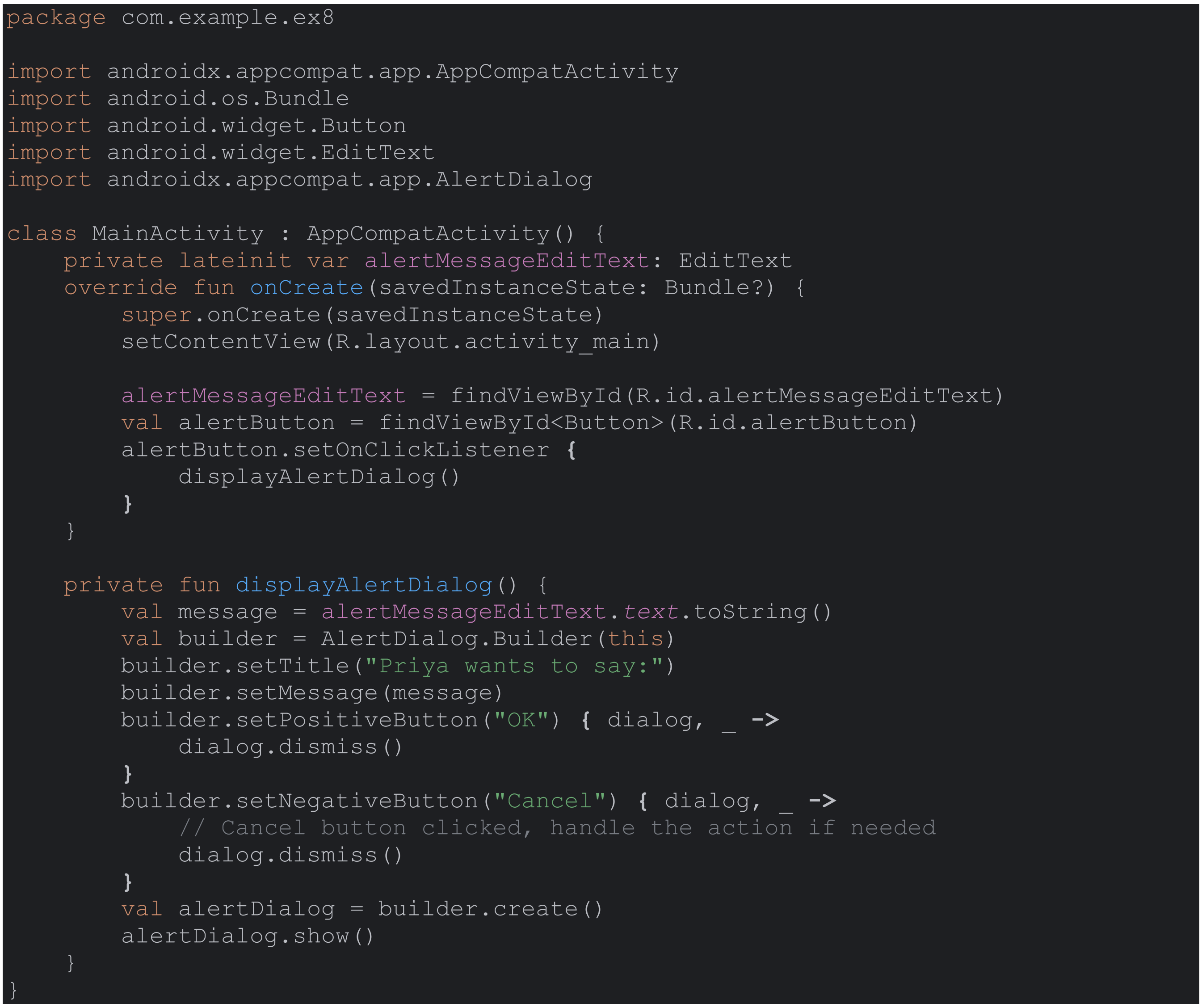
ANDROID MANIFEST

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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.Ex8"
            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 version="1.0" encoding="utf-8"?>
<RelativeLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        tools:ignore="MissingConstraints"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        tools:context=".MainActivity">

    <EditText
            android:id="@+id/alertMessageEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter your message"
            android:layout_margin="16dp"
            android:inputType="text" tools:layout_editor_absoluteY="40dp"
tools:layout_editor_absoluteX="16dp"/>
    <Button
            android:id="@+id/alertButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Display"
            android:layout_below="@id/alertMessageEditText"
            android:layout_centerHorizontal="true"
            tools:layout_editor_absoluteY="121dp"
tools:layout_editor_absoluteX="158dp"
    />
</RelativeLayout>
```

MAIN ACTIVITY



```kotlin
package com.example.ex8

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import androidx.appcompat.app.AlertDialog

class MainActivity : AppCompatActivity() {
    private lateinit var alertMessageEditText: EditText
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        alertMessageEditText = findViewById(R.id.alertMessageEditText)
        val alertButton = findViewById<Button>(R.id.alertButton)
        alertButton.setOnClickListener {
            displayAlertDialog()
        }
    }

    private fun displayAlertDialog() {
        val message = alertMessageEditText.text.toString()
        val builder = AlertDialog.Builder(this)
        builder.setTitle("Priya wants to say:")
        builder.setMessage(message)
        builder.setPositiveButton("OK") { dialog, _ ->
            dialog.dismiss()
        }
        builder.setNegativeButton("Cancel") { dialog, _ ->
            // Cancel button clicked, handle the action if needed
            dialog.dismiss()
        }
        val alertDialog = builder.create()
        alertDialog.show()
    }
}
```

EX 7

ANDROID MANIFEST

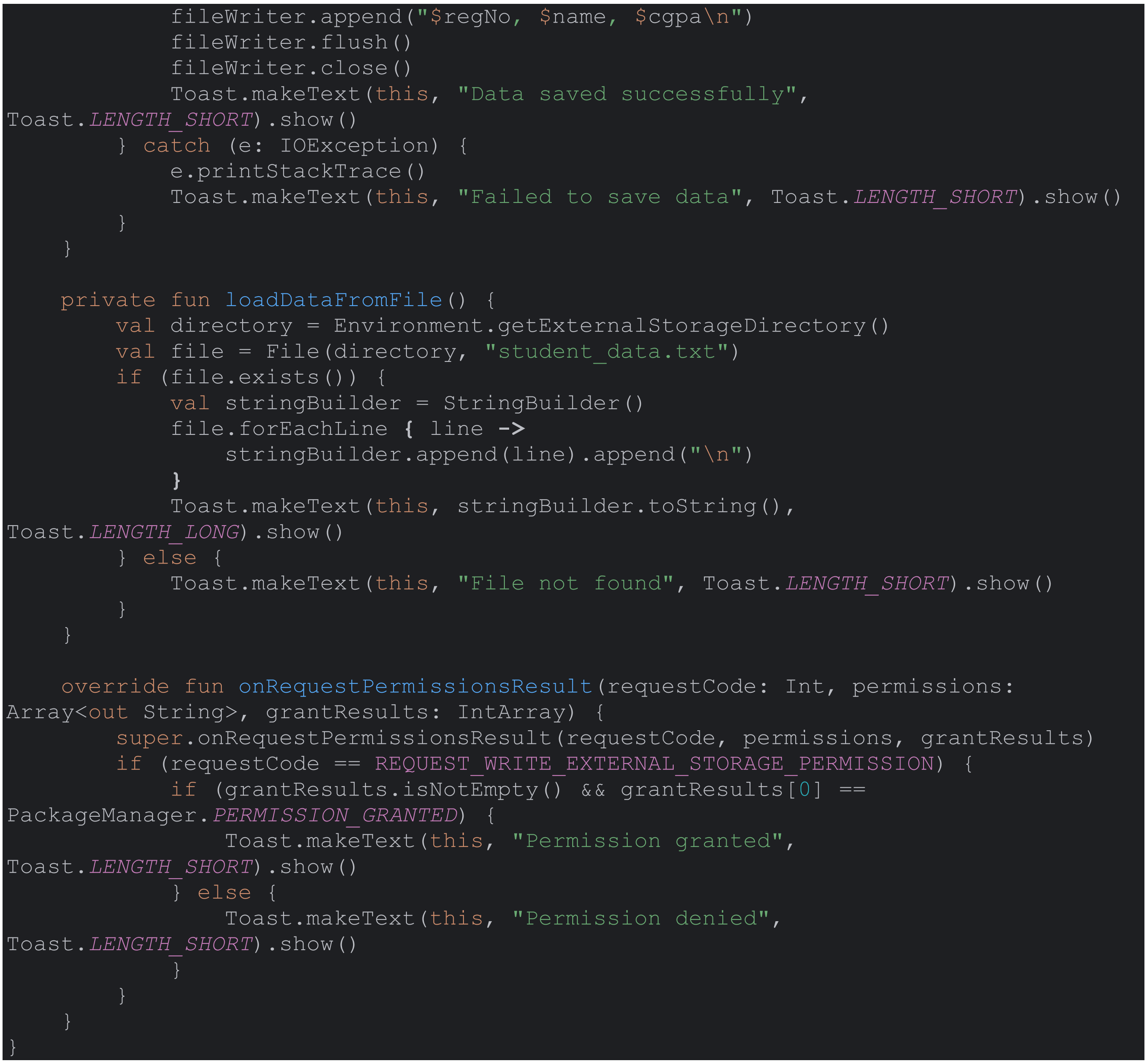
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

    <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.Ex7"
            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 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">

    <EditText
            android:id="@+id/editTextRegNo"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Register Number"/>

    <EditText
            android:id="@+id/editTextName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/editTextRegNo"
            android:hint="Name"/>

    <EditText
            android:id="@+id/editTextCGPA"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/editTextName"
            android:hint="CGPA"/>

    <Button
            android:id="@+id/buttonSave"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/editTextCGPA"
            android:text="Save"/>

    <Button
            android:id="@+id/buttonLoad"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/buttonSave"
            android:text="Load"/>

</RelativeLayout>
```

MAIN ACTIVITY

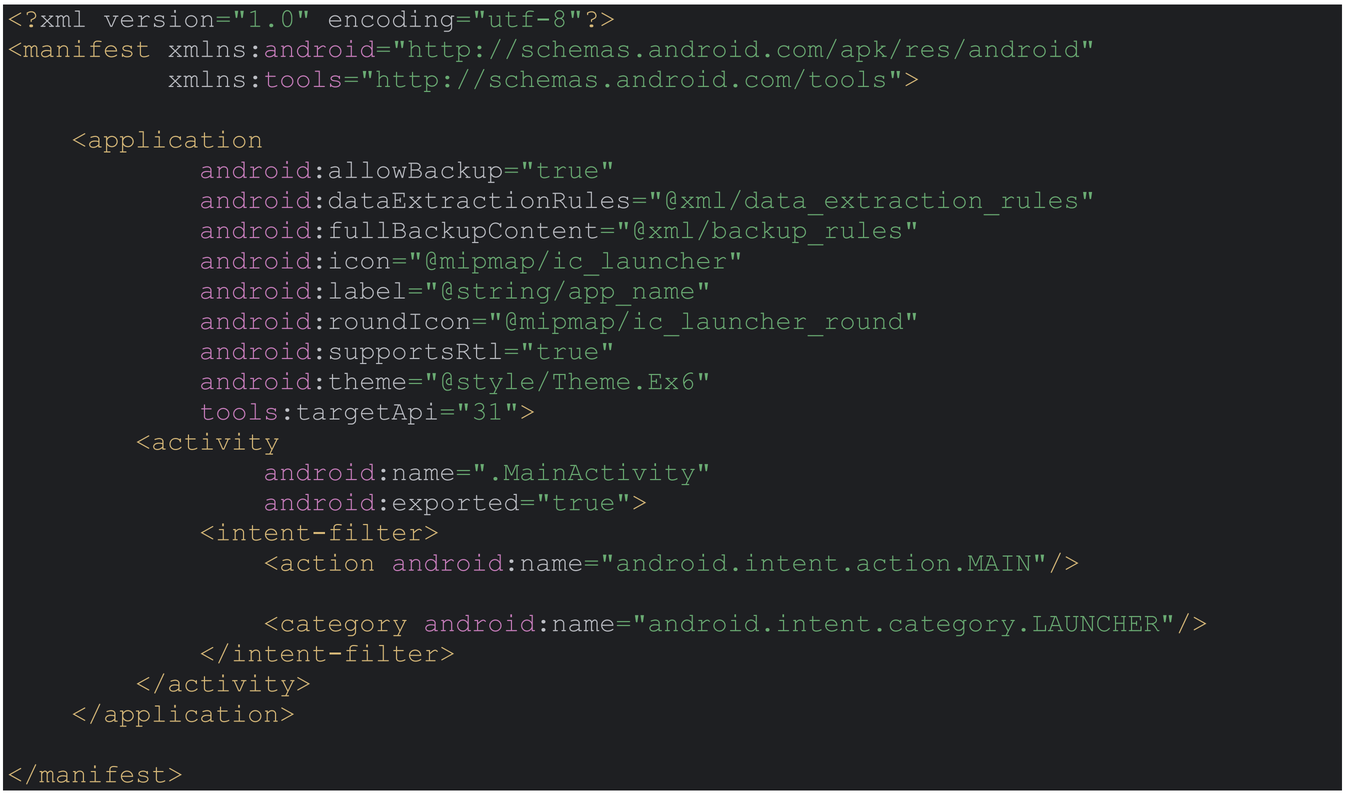
```
package com.example.ex7

import android.Manifest
import android.content.pm.PackageManager
import android.os.Bundle
import android.os.Environment
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import java.io.File
import java.io.FileWriter
import java.io.IOException

class MainActivity : AppCompatActivity() {

    private val REQUEST_WRITE_EXTERNAL_STORAGE_PERMISSION = 101

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

        val editTextRegNo = findViewById<EditText>(R.id.editTextRegNo)
        val editTextName = findViewById<EditText>(R.id.editTextName)
        val editTextCGPA = findViewById<EditText>(R.id.editTextCGPA)
        val buttonSave = findViewById<Button>(R.id.buttonSave)
        val buttonLoad = findViewById<Button>(R.id.buttonLoad)

        buttonSave.setOnClickListener {
            if (checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE) !=
PackageManager.PERMISSION_GRANTED) {

requestPermissions(arrayOf(Manifest.permission.WRITE_EXTERNAL_STORAGE),
REQUEST_WRITE_EXTERNAL_STORAGE_PERMISSION)
            } else {
                saveDataToFile(editTextRegNo.text.toString(),
editTextName.text.toString(), editTextCGPA.text.toString())
            }
        }

        buttonLoad.setOnClickListener {
            loadDataFromFile()
        }
    }

    private fun saveDataToFile(regNo: String, name: String, cgpa: String) {
        val directory = Environment.getExternalStorageDirectory()
        val file = File(directory, "student_data.txt")
        try {
            val fileWriter = FileWriter(file, true)
```



```
            fileWriter.append("$regNo, $name, $cgpa\n")
            fileWriter.flush()
            fileWriter.close()
            Toast.makeText(this, "Data saved successfully",
Toast.LENGTH_SHORT).show()
        } catch (e: IOException) {
            e.printStackTrace()
            Toast.makeText(this, "Failed to save data", Toast.LENGTH_SHORT).show()
        }
    }

    private fun loadDataFromFile() {
        val directory = Environment.getExternalStorageDirectory()
        val file = File(directory, "student_data.txt")
        if (file.exists()) {
            val stringBuilder = StringBuilder()
            file.forEachLine { line ->
                stringBuilder.append(line).append("\n")
            }
            Toast.makeText(this, stringBuilder.toString(),
Toast.LENGTH_LONG).show()
        } else {
            Toast.makeText(this, "File not found", Toast.LENGTH_SHORT).show()
        }
    }

    override fun onRequestPermissionsResult(requestCode: Int, permissions:
Array<out String>, grantResults: IntArray) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults)
        if (requestCode == REQUEST_WRITE_EXTERNAL_STORAGE_PERMISSION) {
            if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                Toast.makeText(this, "Permission granted",
Toast.LENGTH_SHORT).show()
            } else {
                Toast.makeText(this, "Permission denied",
Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

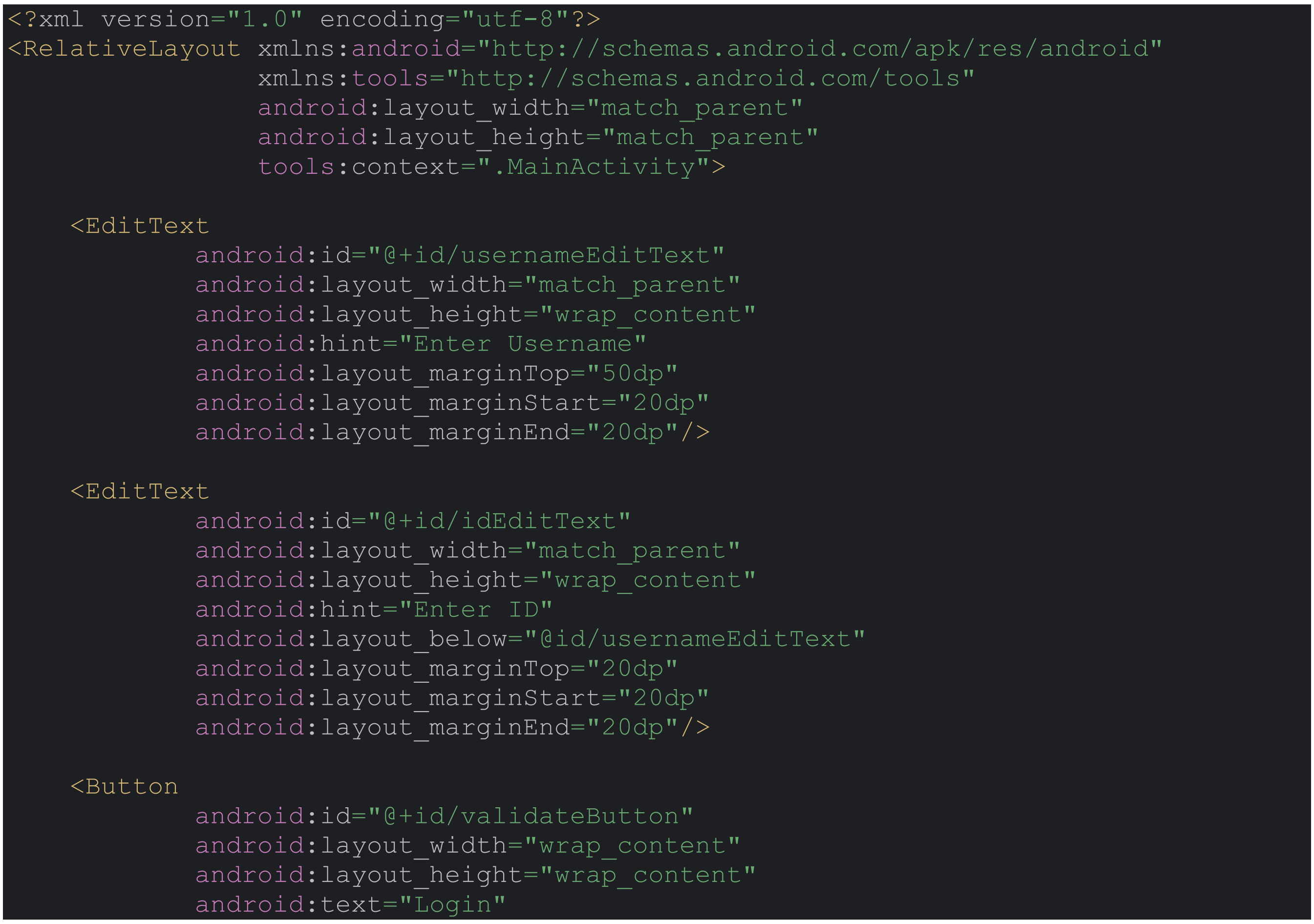
EX 6

ANDROID MANIFEST



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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.Ex6"
            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 MAIN1



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
                tools:context=".MainActivity">

    <EditText
            android:id="@+id/usernameEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter Username"
            android:layout_marginTop="50dp"
            android:layout_marginStart="20dp"
            android:layout_marginEnd="20dp"/>

    <EditText
            android:id="@+id/idEditText"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter ID"
            android:layout_below="@id/usernameEditText"
            android:layout_marginTop="20dp"
            android:layout_marginStart="20dp"
            android:layout_marginEnd="20dp"/>

    <Button
            android:id="@+id/validateButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Login"
            android:layout_below="@id/idEditText"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"/>

</RelativeLayout>
```

ACTIVITY MAIN 2

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
                tools:context=".MainActivity2">

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Login Successful!"
            android:textSize="24sp"
            android:layout_centerInParent="true"/>

</RelativeLayout>
```

MAIN ACTIVITY 1

```
package com.example.ex6
import android.content.Intent
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

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

        val validateButton: Button = findViewById(R.id.validateButton)
        val usernameEditText: EditText = findViewById(R.id.usernameEditText)
        val idEditText: EditText = findViewById(R.id.idEditText)

        validateButton.setOnClickListener {
            val username = usernameEditText.text.toString().trim()
            val id = idEditText.text.toString().trim()

            if (username.isEmpty() || id.isEmpty()) {
                // Check if any field is empty
                // Show error message as toast
                Toast.makeText(this, "Username and ID cannot be empty",
Toast.LENGTH_SHORT).show()
                return@setOnClickListener
            }

            if (!username.matches(Regex("[a-zA-Z]+"))) {
                // Check if username contains only alphabets
                // Show error message as toast
                Toast.makeText(this, "Username should contain only alphabets",
Toast.LENGTH_SHORT).show()
                return@setOnClickListener
            }
            if (!id.matches(Regex("\\d{4}"))) {
                // Check if ID contains exactly 4 digits
                // Show error message as toast
                Toast.makeText(this, "ID should contain exactly 4 digits",
Toast.LENGTH_SHORT).show()
                return@setOnClickListener
            }

            // If all validations pass, start MainActivity2
            val intent = Intent(this, MainActivity2::class.java)
            intent.putExtra("username", username)
            intent.putExtra("id", id)
            startActivity(intent)
        }
    }
}
```

MAIN ACTIVITY 2

```
package com.example.ex6

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity

class MainActivity2 : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main2)
    }
}
```

EX 3

ANDROID MANIFEST

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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.Shapes"
            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 version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
        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"
        tools:ignore="MissingConstraints"

        tools:context=".MainActivity">

    <com.example.shapes.CanvasView
            android:id="@+id/canvasView"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
tools:layout_editor_absoluteY="0dp"
            tools:layout_editor_absoluteX="-16dp"
            android:background="@color/lavendar"/>

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

MAIN ACTIVITY

```
package com.example.shapes

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

CANVASVIEW.KT

```
package com.example.shapes

import android.content.Context
import android.graphics.Canvas
import android.graphics.Color
import android.graphics.Paint
import android.util.AttributeSet
import android.view.View

class CanvasView @JvmOverloads constructor(
    context: Context, attrs: AttributeSet? = null, defStyleAttr: Int = 0
) : View(context, attrs, defStyleAttr) {

    override fun onDraw(canvas: Canvas) {
            super.onDraw(canvas)

        val paint:Paint=Paint()

        //line
        paint.setColor(Color.GREEN)
        paint.strokeWidth=8f
        canvas.drawLine(750f,800f,750f,1200f,paint)
        drawText(canvas, "Line", 690f, 750f)

        //circle
        paint.style=Paint.Style.FILL
        paint.setColor(Color.YELLOW)
        canvas.drawCircle(290f,350f,150f,paint)
        drawText(canvas, "Circle", 220f, 150f)

        //rectangle
        paint.style=Paint.Style.FILL
        paint.setColor(Color.RED)
        canvas.drawRect(850f,650f,600f,200f,paint)
        drawText(canvas, "Rectangle", 620f, 150f)


        //square
        paint.style=Paint.Style.FILL
        paint.setColor(Color.BLUE)
        canvas.drawRect(200f,1150f,500f,850f,paint)
        drawText(canvas, "Square", 250f, 750f)

    }

    private fun drawText(canvas: Canvas, s: String,x: Float, y: Float) {
        val textPaint = Paint().apply {
            color = Color.BLACK
            textSize = 50f
        }
        canvas.drawText(s, x, y, textPaint)
    }
}
```

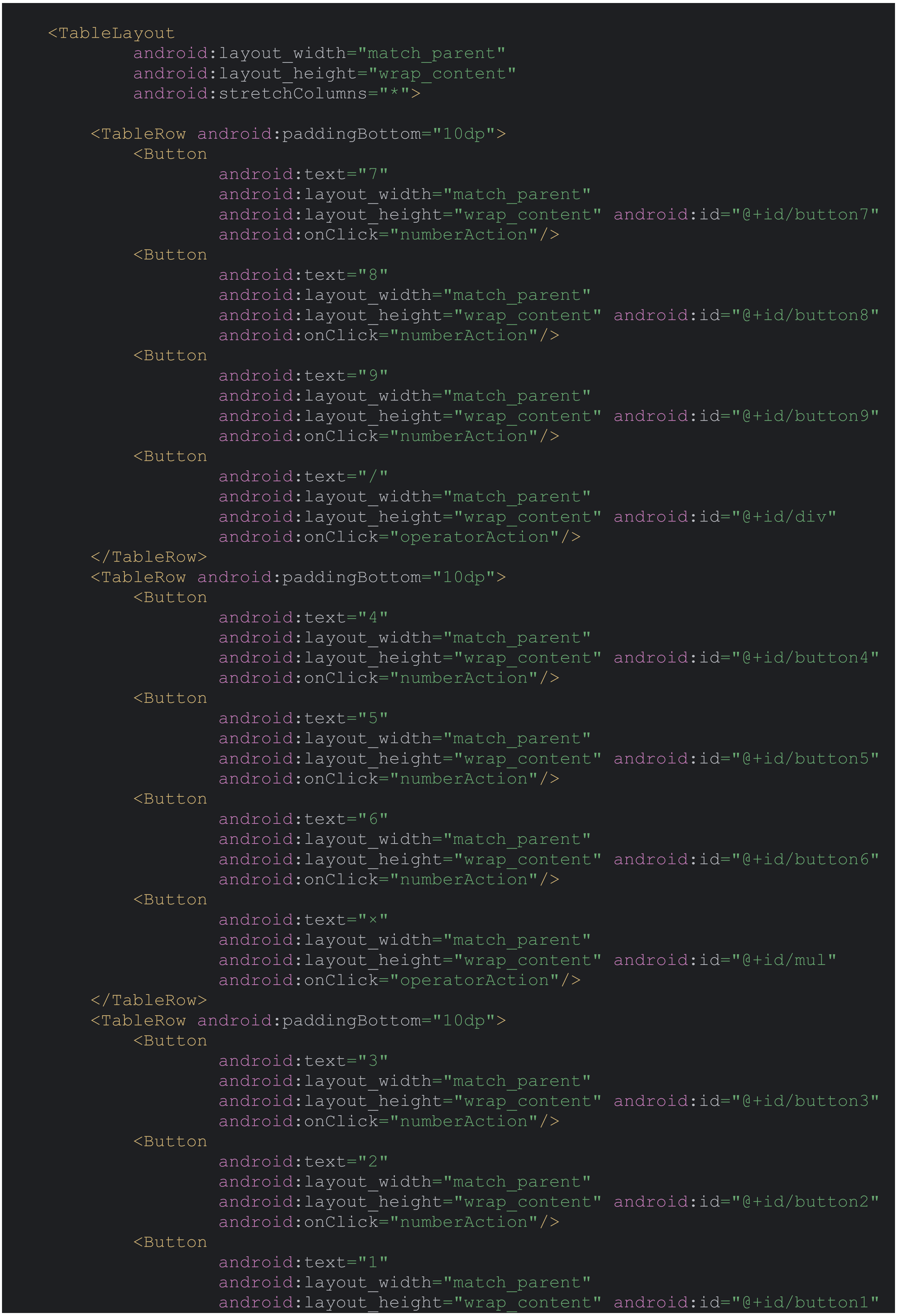
EX 2

ANDROID MANIFEST

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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.Calci"
            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 version="1.0" encoding="utf-8"?>
<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"
        tools:context=".MainActivity"
        tools:ignore="HardcodedText">

    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="20dp">

        <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textSize="25sp"
                android:id="@+id/input"
                android:text=""/>


        <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textSize="40sp"
                android:id="@+id/output"
                android:text=""/>
    </LinearLayout>
```

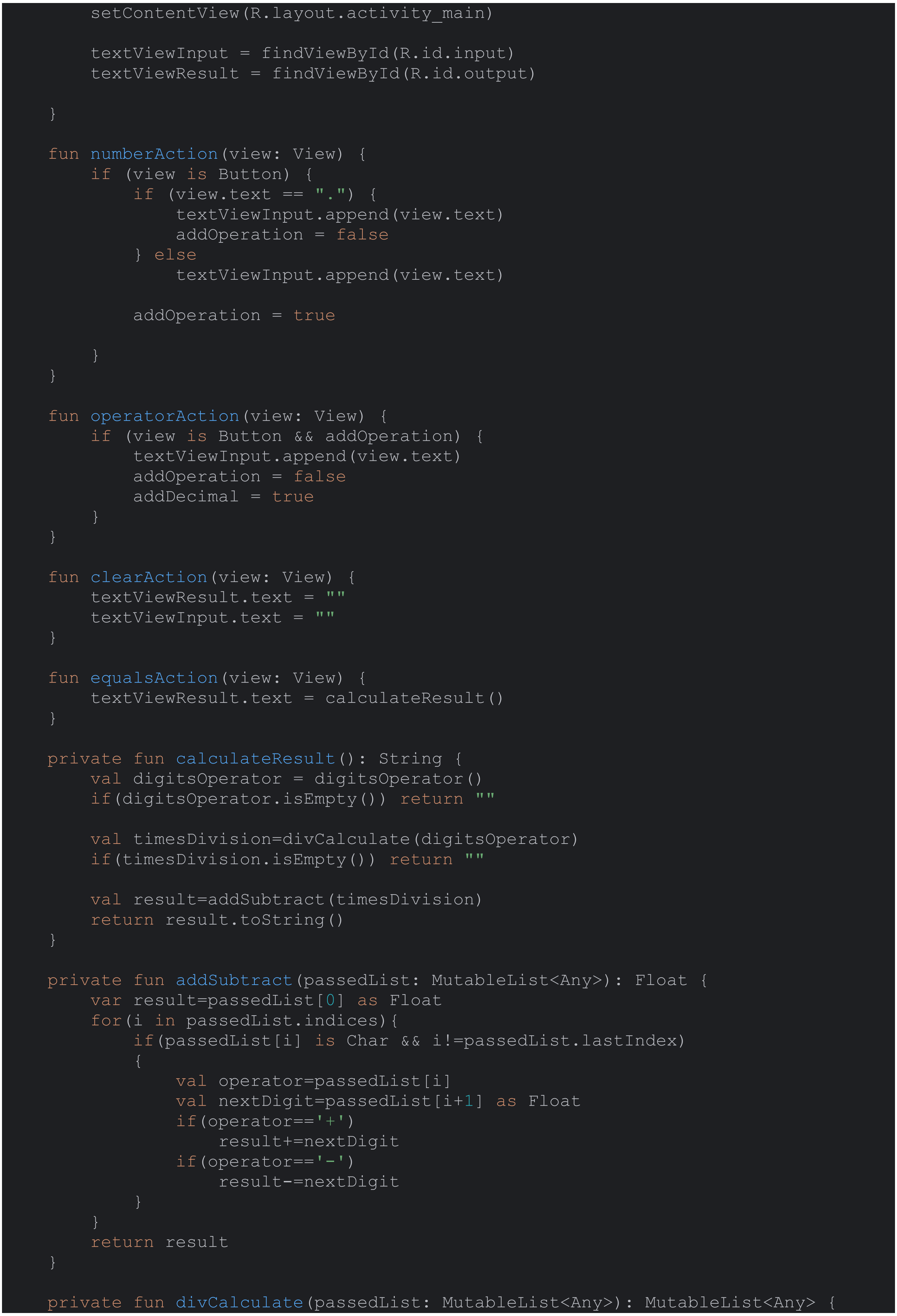


```
<TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:stretchColumns="*">

    <TableRow android:paddingBottom="10dp">
        <Button
                android:text="7"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button7"
                android:onClick="numberAction"/>
        <Button
                android:text="8"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button8"
                android:onClick="numberAction"/>
        <Button
                android:text="9"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button9"
                android:onClick="numberAction"/>
        <Button
                android:text="/"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/div"
                android:onClick="operatorAction"/>
    </TableRow>
    <TableRow android:paddingBottom="10dp">
        <Button
                android:text="4"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button4"
                android:onClick="numberAction"/>
        <Button
                android:text="5"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button5"
                android:onClick="numberAction"/>
        <Button
                android:text="6"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button6"
                android:onClick="numberAction"/>
        <Button
                android:text="×"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/mul"
                android:onClick="operatorAction"/>
    </TableRow>
    <TableRow android:paddingBottom="10dp">
        <Button
                android:text="3"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button3"
                android:onClick="numberAction"/>
        <Button
                android:text="2"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button2"
                android:onClick="numberAction"/>
        <Button
                android:text="1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" android:id="@+id/button1"
```

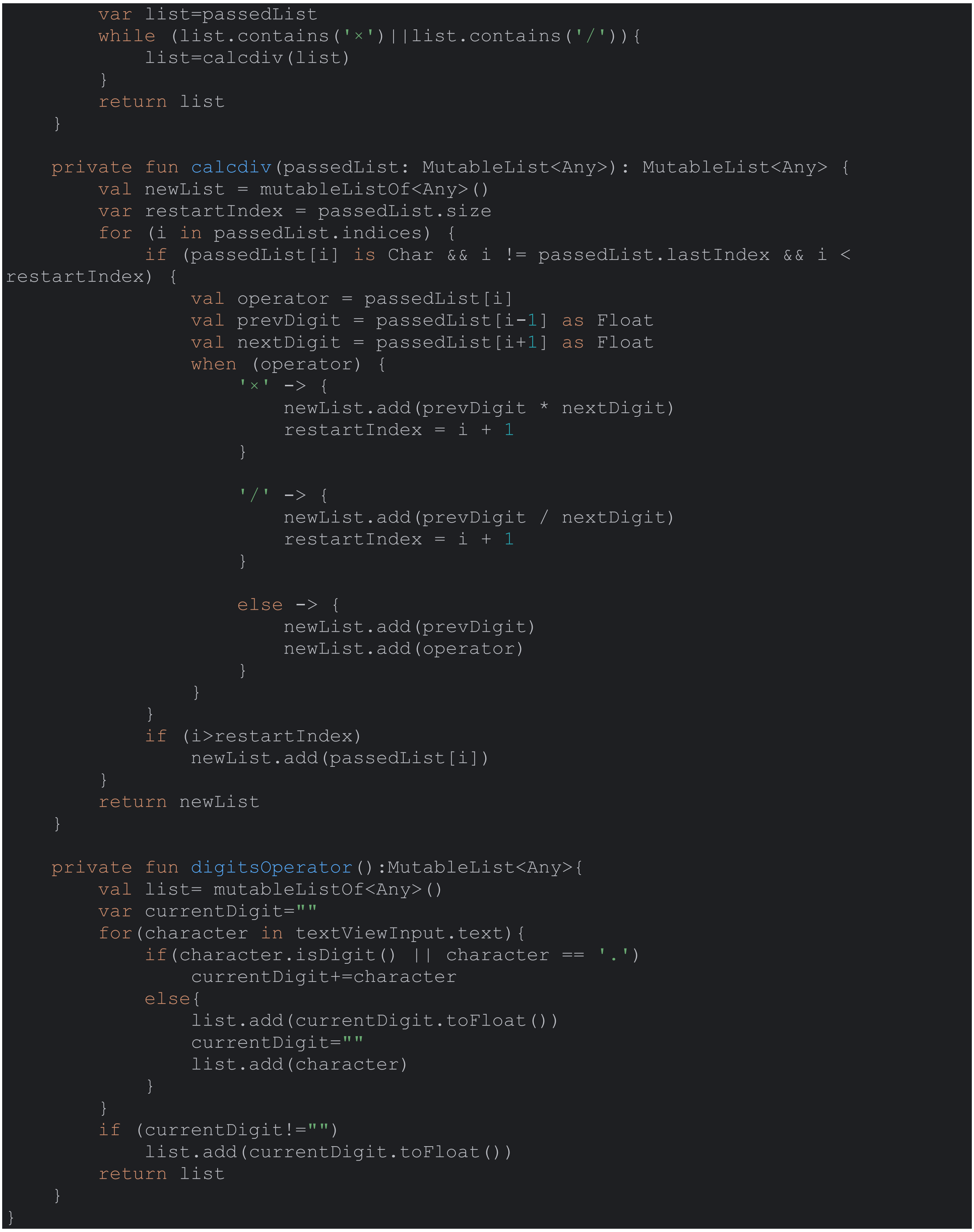
```
                    android:onClick="numberAction"/>
            <Button
                    android:text="-"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content" android:id="@+id/sub"
                    android:onClick="operatorAction"/>
        </TableRow>

        <TableRow android:paddingBottom="10dp">
            <Button
                    android:text="."
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content" android:id="@+id/dot"
                    android:onClick="numberAction"/>
            <Button
                    android:text="0"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content" android:id="@+id/button0"
                    android:onClick="numberAction"/>
            <Button
                    android:text="="
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content" android:id="@+id/equals"
                    android:onClick="equalsAction"/>
            <Button
                    android:text="+"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content" android:id="@+id/add"
                    android:onClick="operatorAction"/>
        </TableRow>
        <TableRow android:paddingBottom="10dp">
            <Button
                    android:text="Clear"
                    android:layout_width="match_parent"
                    android:layout_weight="3"
                    android:layout_height="wrap_content" android:id="@+id/clear"
                    android:onClick="clearAction"/>

        </TableRow>
    </TableLayout>


</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.calci

import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity


class MainActivity : AppCompatActivity() {
    private var addOperation = false
    private var addDecimal = true
    private lateinit var textViewInput: TextView
    private lateinit var textViewResult: TextView


    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
```



```
        setContentView(R.layout.activity_main)

        textViewInput = findViewById(R.id.input)
        textViewResult = findViewById(R.id.output)

    }

    fun numberAction(view: View) {
        if (view is Button) {
            if (view.text == ".") {
                textViewInput.append(view.text)
                addOperation = false
            } else
                textViewInput.append(view.text)

            addOperation = true

        }
    }

    fun operatorAction(view: View) {
        if (view is Button && addOperation) {
            textViewInput.append(view.text)
            addOperation = false
            addDecimal = true
        }
    }

    fun clearAction(view: View) {
        textViewResult.text = ""
        textViewInput.text = ""
    }

    fun equalsAction(view: View) {
        textViewResult.text = calculateResult()
    }

    private fun calculateResult(): String {
        val digitsOperator = digitsOperator()
        if(digitsOperator.isEmpty()) return ""

        val timesDivision=divCalculate(digitsOperator)
        if(timesDivision.isEmpty()) return ""

        val result=addSubtract(timesDivision)
        return result.toString()
    }

    private fun addSubtract(passedList: MutableList<Any>): Float {
        var result=passedList[0] as Float
        for(i in passedList.indices){
            if(passedList[i] is Char && i!=passedList.lastIndex)
            {
                val operator=passedList[i]
                val nextDigit=passedList[i+1] as Float
                if(operator=='+')
                    result+=nextDigit
                if(operator=='-')
                    result-=nextDigit
            }
        }
        return result
    }

    private fun divCalculate(passedList: MutableList<Any>): MutableList<Any> {
```



```
        var list=passedList
        while (list.contains('×')||list.contains('/')){
            list=calcdiv(list)
        }
        return list
    }

    private fun calcdiv(passedList: MutableList<Any>): MutableList<Any> {
        val newList = mutableListOf<Any>()
        var restartIndex = passedList.size
        for (i in passedList.indices) {
            if (passedList[i] is Char && i != passedList.lastIndex && i <
restartIndex) {
                val operator = passedList[i]
                val prevDigit = passedList[i-1] as Float
                val nextDigit = passedList[i+1] as Float
                when (operator) {
                    '×' -> {
                        newList.add(prevDigit * nextDigit)
                        restartIndex = i + 1
                    }

                    '/' -> {
                        newList.add(prevDigit / nextDigit)
                        restartIndex = i + 1
                    }

                    else -> {
                        newList.add(prevDigit)
                        newList.add(operator)
                    }
                }
            }
            if (i>restartIndex)
                newList.add(passedList[i])
        }
        return newList
    }

    private fun digitsOperator():MutableList<Any>{
        val list= mutableListOf<Any>()
        var currentDigit=""
        for(character in textViewInput.text){
            if(character.isDigit() || character == '.')
                currentDigit+=character
            else{
                list.add(currentDigit.toFloat())
                currentDigit=""
                list.add(character)
            }
        }
        if (currentDigit!="")
            list.add(currentDigit.toFloat())
        return list
    }
}
```

EX 1

ANDROID MANIFEST

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
          xmlns:tools="http://schemas.android.com/tools">

    <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.Demo"
            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 version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/rootView"
        tools:context=".MainActivity">

    <TextView
            android:id="@+id/text"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Rajalakshmi Engineering College"
            android:textAlignment="center"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.068"/>
    <Button
            android:id="@+id/fontsize"
            android:text="Change font size"
            android:layout_width="485dp"
            android:layout_height="47dp"
            app:layout_constraintTop_toBottomOf="@id/text"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="32dp"
            app:layout_constraintHorizontal_bias="0.513"/>
    <Button
            android:id="@+id/fontcolor"
            android:text="Change font color"
            android:layout_width="444dp"
            android:layout_height="47dp"
            app:layout_constraintTop_toBottomOf="@id/text"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="188dp"/>
    <Button
            android:id="@+id/bgcolor"
            android:text="Change background color"
            android:layout_width="446dp"
            android:layout_height="47dp"
            app:layout_constraintTop_toBottomOf="@id/text"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
android:layout_marginTop="112dp"
            app:layout_constraintHorizontal_bias="0.497"/>

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

MAIN ACTIVITY

```
package com.example.demo

import android.graphics.Color
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import com.example.demo.R.*

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(layout.activity_main)
        val rootView=
findViewById<androidx.constraintlayout.widget.ConstraintLayout>(id.rootView)
        val textView: TextView = findViewById(id.text)
        val increaseButton: Button = findViewById(id.fontsize)
        val changetextcolor:Button=findViewById(id.fontcolor)
        val changebgcolor:Button=findViewById(id.bgcolor)

        increaseButton.setOnClickListener {
            // Increase font size by a certain factor
            val currentSize = textView.textSize
            val newSize = currentSize + 5 // You can adjust this value
            textView.textSize = newSize
        }
        changetextcolor.setOnClickListener{
            textView.setTextColor(Color.GREEN)
        }
        changebgcolor.setOnClickListener {
            rootView.setBackgroundColor(Color.BLUE)
        }
    }
}
```

MAD EX-4

Activity_main.xml

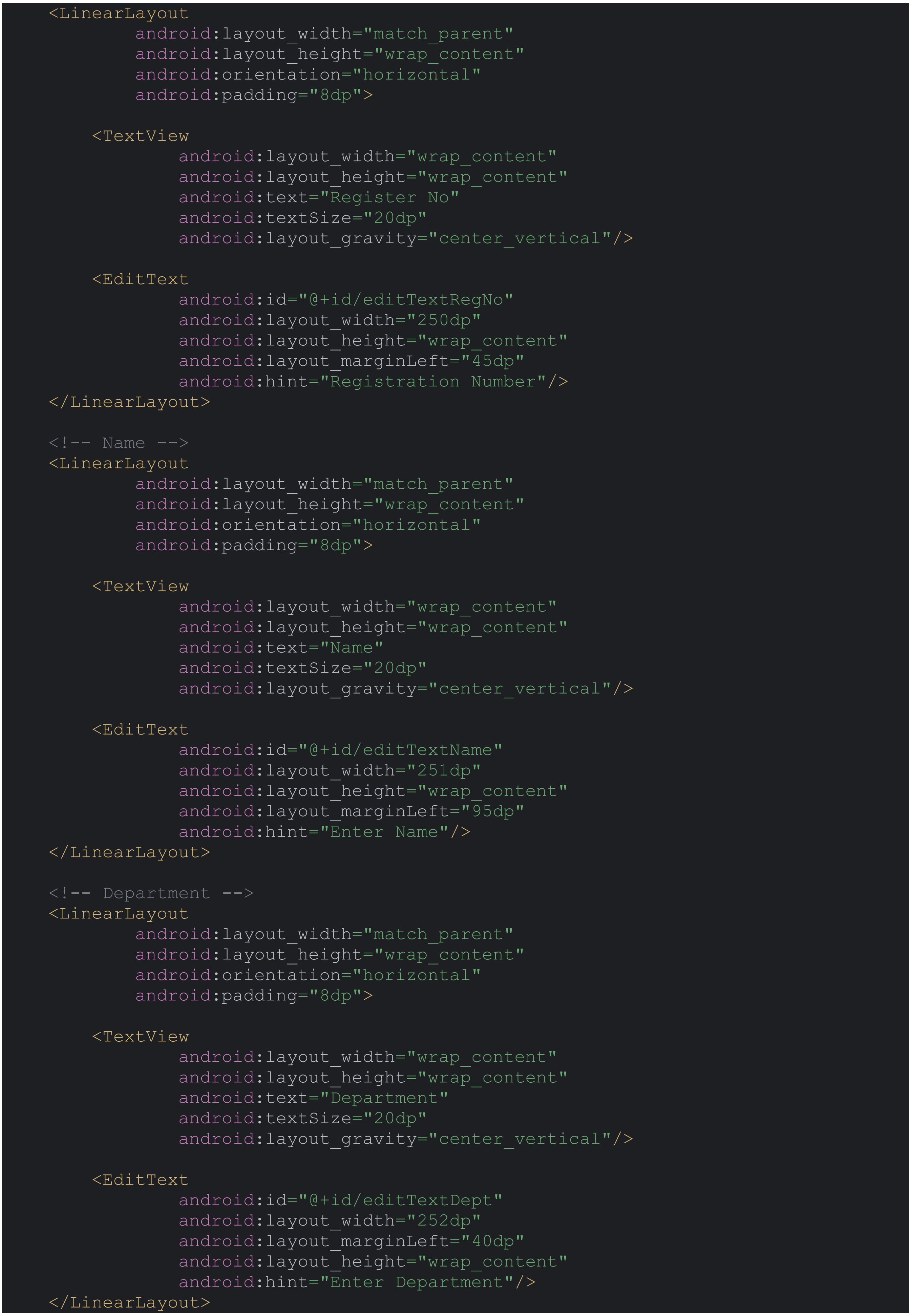
```
package com.example.ex4

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup

class StudentMarkDetailsFragment : Fragment() {
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        return inflater.inflate(R.layout.fragment_student_mark_details, container,
false)
    }
}
```

MainActivity.kt

```
package com.example.ex4

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.appcompat.widget.Toolbar

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

        supportFragmentManager.beginTransaction()
            .replace(R.id.container_basic_details, StudentBasicDetailsFragment())
            .replace(R.id.container_mark_details, StudentMarkDetailsFragment())
            .commit()

        val toolbar: Toolbar = findViewById(R.id.toolbar)
        setSupportActionBar(toolbar)
    }
}
```

fragment_student_basic_details.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
              android:layout_width="match_parent"
              android:layout_height="match_parent"
              android:orientation="vertical">

    <!-- Registration Number -->

    <TextView
            android:text="Student Basic Details"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textAlignment="center"
            android:textSize="30dp"
            android:id="@+id/textView2"/>
```



```
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Register No"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextRegNo"
                android:layout_width="250dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="45dp"
                android:hint="Registration Number"/>
    </LinearLayout>

    <!-- Name -->
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Name"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextName"
                android:layout_width="251dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="95dp"
                android:hint="Enter Name"/>
    </LinearLayout>

    <!-- Department -->
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Department"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextDept"
                android:layout_width="252dp"
                android:layout_marginLeft="40dp"
                android:layout_height="wrap_content"
                android:hint="Enter Department"/>
    </LinearLayout>
```

```
</LinearLayout>
```

Fragment_student_mark_details.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
              android:layout_width="match_parent"
              android:layout_height="match_parent"
              android:orientation="vertical">

    <!-- SSLC Marks -->
    <TextView
            android:text="Student Mark Details"
            android:textAlignment="center"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="30dp"
            android:id="@+id/textView"/>
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="SSLC Marks"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextSSLCMarks"
                android:layout_width="251dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="40dp"
                android:hint="Enter SSLC Marks"/>
    </LinearLayout>

    <!-- HSC Marks -->
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="HSC Marks"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextHSCMarks"
                android:layout_width="251dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="45dp"
                android:hint="Enter HSC Marks"/>
    </LinearLayout>

    <!-- UG Marks -->
    <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="8dp">

        <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="UG CGPA"
                android:textSize="20dp"
                android:layout_gravity="center_vertical"/>

        <EditText
                android:id="@+id/editTextUGMarks"
                android:layout_width="251dp"
                android:layout_height="wrap_content"
                android:layout_marginLeft="60dp"
                android:hint="Enter UG CGPA"/>
    </LinearLayout>

</LinearLayout>
```

Studentbasicdetails.kt

```
package com.example.ex4
import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
class StudentBasicDetailsFragment : Fragment() {
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        return inflater.inflate(R.layout.fragment_student_basic_details, container,
false)
    }
}
```

StudentMarkDetailskt

```
package com.example.ex4

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup

class StudentMarkDetailsFragment : Fragment() {
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        return inflater.inflate(R.layout.fragment_student_mark_details, container,
false)
    }
}
```

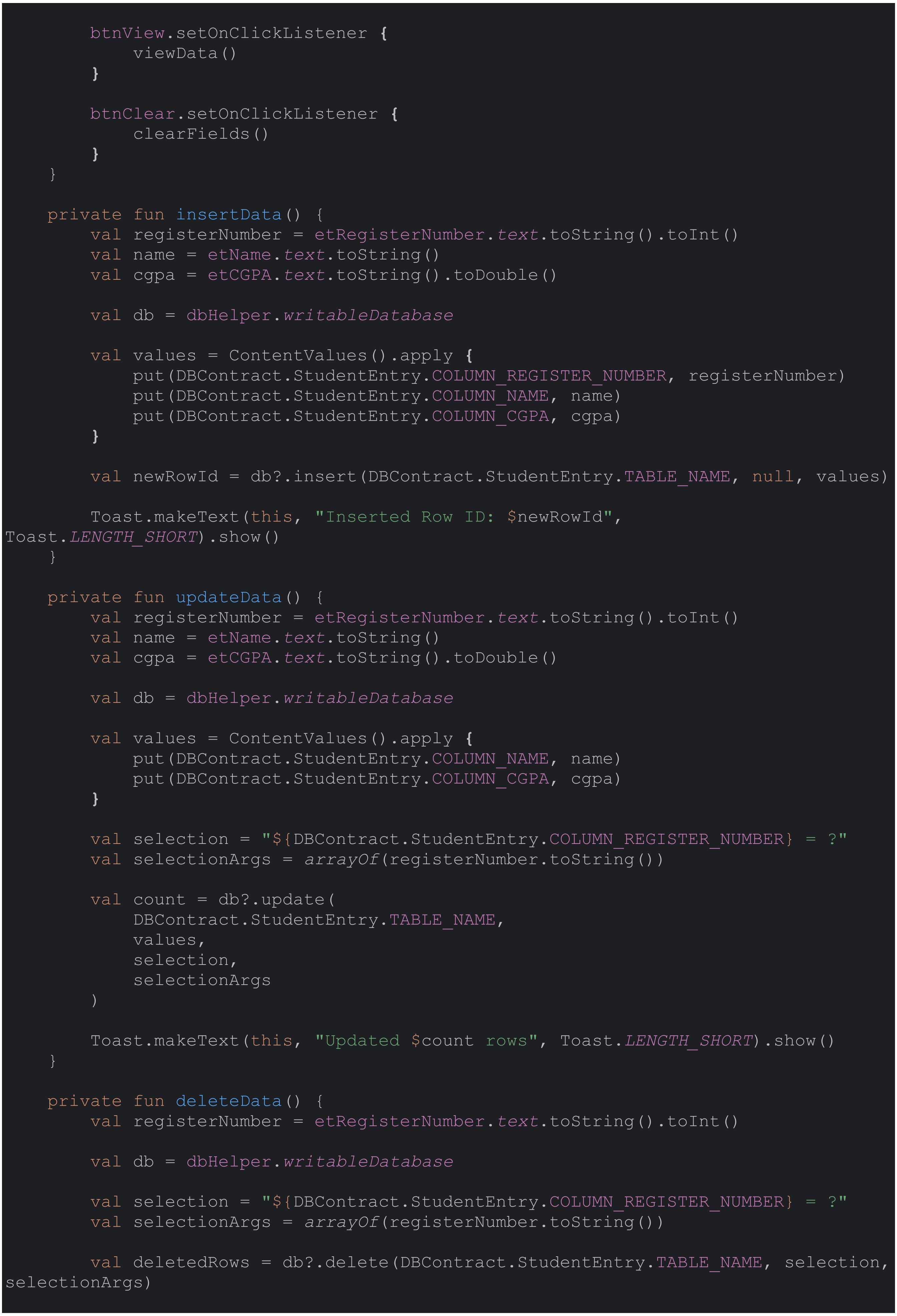
MAD EX-5

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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="16dp"
              tools:context=".MainActivity">

    <EditText
            android:id="@+id/etRegisterNumber"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Register Number"
            android:inputType="number" />

    <EditText
            android:id="@+id/etName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Name" />

    <EditText
            android:id="@+id/etCGPA"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="CGPA"
            android:inputType="numberDecimal" />

    <Button
            android:id="@+id/btnAdd"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Add" />

    <Button
            android:id="@+id/btnModify"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Modify" />

    <Button
            android:id="@+id/btnDelete"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Delete" />

    <Button
            android:id="@+id/btnView"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="View" />

    <Button
            android:id="@+id/btnClear"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Clear" />
    <TextView
            android:text=""
            android:textSize="15dp"
            android:layout_width="376dp"
            android:layout_height="308dp" android:id="@+id/tvDetails"/>

</LinearLayout>
```

MainActivity.kt

```
package com.example.ex5

import UsersDBHelper
import android.content.ContentValues
import android.content.Context
import android.database.Cursor
import android.database.sqlite.SQLiteDatabase
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity



class MainActivity : AppCompatActivity() {

    private lateinit var etRegisterNumber: EditText
    private lateinit var etName: EditText
    private lateinit var etCGPA: EditText
    private lateinit var btnAdd: Button
    private lateinit var btnModify: Button
    private lateinit var btnDelete: Button
    private lateinit var btnView: Button
    private lateinit var btnClear: Button
    private lateinit var tvDetails: TextView

    private lateinit var dbHelper: UsersDBHelper

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

        dbHelper = UsersDBHelper(this)

        etRegisterNumber = findViewById(R.id.etRegisterNumber)
        etName = findViewById(R.id.etName)
        etCGPA = findViewById(R.id.etCGPA)
        btnAdd = findViewById(R.id.btnAdd)
        btnModify = findViewById(R.id.btnModify)
        btnDelete = findViewById(R.id.btnDelete)
        btnView = findViewById(R.id.btnView)
        btnClear = findViewById(R.id.btnClear)
        tvDetails = findViewById(R.id.tvDetails)

        btnAdd.setOnClickListener {
            insertData()
        }

        btnModify.setOnClickListener {
            updateData()
        }

        btnDelete.setOnClickListener {
            deleteData()
        }
```



```
        btnView.setOnClickListener {
            viewData()
        }

        btnClear.setOnClickListener {
            clearFields()
        }
    }

    private fun insertData() {
        val registerNumber = etRegisterNumber.text.toString().toInt()
        val name = etName.text.toString()
        val cgpa = etCGPA.text.toString().toDouble()

        val db = dbHelper.writableDatabase

        val values = ContentValues().apply {
            put(DBContract.StudentEntry.COLUMN_REGISTER_NUMBER, registerNumber)
            put(DBContract.StudentEntry.COLUMN_NAME, name)
            put(DBContract.StudentEntry.COLUMN_CGPA, cgpa)
        }

        val newRowId = db?.insert(DBContract.StudentEntry.TABLE_NAME, null, values)

        Toast.makeText(this, "Inserted Row ID: $newRowId",
Toast.LENGTH_SHORT).show()
    }

    private fun updateData() {
        val registerNumber = etRegisterNumber.text.toString().toInt()
        val name = etName.text.toString()
        val cgpa = etCGPA.text.toString().toDouble()

        val db = dbHelper.writableDatabase

        val values = ContentValues().apply {
            put(DBContract.StudentEntry.COLUMN_NAME, name)
            put(DBContract.StudentEntry.COLUMN_CGPA, cgpa)
        }

        val selection = "${DBContract.StudentEntry.COLUMN_REGISTER_NUMBER} = ?"
        val selectionArgs = arrayOf(registerNumber.toString())

        val count = db?.update(
            DBContract.StudentEntry.TABLE_NAME,
            values,
            selection,
            selectionArgs
        )

        Toast.makeText(this, "Updated $count rows", Toast.LENGTH_SHORT).show()
    }

    private fun deleteData() {
        val registerNumber = etRegisterNumber.text.toString().toInt()

        val db = dbHelper.writableDatabase

        val selection = "${DBContract.StudentEntry.COLUMN_REGISTER_NUMBER} = ?"
        val selectionArgs = arrayOf(registerNumber.toString())

        val deletedRows = db?.delete(DBContract.StudentEntry.TABLE_NAME, selection,
selectionArgs)
```

```
        Toast.makeText(this, "Deleted $deletedRows rows",
Toast.LENGTH_SHORT).show()
    }

    private fun viewData() {
        val db = dbHelper.readableDatabase

        val cursor = db?.query(
            DBContract.StudentEntry.TABLE_NAME,
            null,
            null,
            null,
            null,
            null,
            null
        )

        tvDetails.text = ""
        cursor?.moveToFirst()
        while (cursor?.moveToNext() == true) {
            val registerNumber =
cursor.getInt(cursor.getColumnIndexOrThrow(DBContract.StudentEntry.COLUMN_REGISTER_
NUMBER))
            val name =
cursor.getString(cursor.getColumnIndexOrThrow(DBContract.StudentEntry.COLUMN_NAME))
            val cgpa =
cursor.getDouble(cursor.getColumnIndexOrThrow(DBContract.StudentEntry.COLUMN_CGPA))

            tvDetails.append("Register Number: $registerNumber, Name: $name, CGPA:
$cgpa\n")
        }

        cursor?.close()
    }

    private fun clearFields() {
        etRegisterNumber.text.clear()
        etName.text.clear()
        etCGPA.text.clear()
    }

    override fun onDestroy() {
        dbHelper.close()
        super.onDestroy()
    }
}
```

DBContract.kt

```
package com.example.ex5

import android.provider.BaseColumns

class DBContract {
    class StudentEntry : BaseColumns {
        companion object {
            const val TABLE_NAME = "Student"
            const val COLUMN_REGISTER_NUMBER = "RegisterNumber"
            const val COLUMN_NAME = "Name"
            const val COLUMN_CGPA = "CGPA"
        }
    }
}
```

UserModel.kt

```
package com.example.ex5

data class UserModel(val registerNumber: Int, val name: String, val cgpa: Double)
```

UsersDBHelper.kt

```
import  android.content.Context
import android.database.sqlite.SQLiteDatabase
import android.database.sqlite.SQLiteOpenHelper
import com.example.ex5.DBContract

class UsersDBHelper(context: Context) : SQLiteOpenHelper(context, DATABASE_NAME,
null, DATABASE_VERSION) {

    override fun onCreate(db: SQLiteDatabase) {
        val SQL_CREATE_ENTRIES =
            "CREATE TABLE ${DBContract.StudentEntry.TABLE_NAME} (" +
                    "${DBContract.StudentEntry.COLUMN_REGISTER_NUMBER} INTEGER
PRIMARY KEY," +
                    "${DBContract.StudentEntry.COLUMN_NAME} TEXT," +
                    "${DBContract.StudentEntry.COLUMN_CGPA} REAL)"

        db.execSQL(SQL_CREATE_ENTRIES)
    }

    override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {
        db.execSQL("DROP TABLE IF EXISTS ${DBContract.StudentEntry.TABLE_NAME}")
        onCreate(db)
    }

    companion object {
        const val DATABASE_VERSION = 1
        const val DATABASE_NAME = "Users.db"
    }
}
```