1. a) Write a program to implement addition of **two numbers**

XML Code:

```
<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/number1"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:hint="Enter first number"
      android:inputType="number" />

   <EditText
      android:id="@+id/number2"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:hint="Enter second number"
      android:inputType="number" />

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

   <TextView
      android:id="@+id/resultText"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="Result will be displayed here"
      android:textSize="18sp"
      android:paddingTop="16dp" />
</LinearLayout>
```

Kotlin Code:

```
package com.example.myapplication

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

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

      val number1 = findViewById<EditText>(R.id.number1)
      val number2 = findViewById<EditText>(R.id.number2)
      val addButton = findViewById<Button>(R.id.addButton)
      val resultText = findViewById<TextView>(R.id.resultText)

      addButton.setOnClickListener {
         val num1 = number1.text.toString().toIntOrNull()
         val num2 = number2.text.toString().toIntOrNull()

         if (num1 != null && num2 != null) {
            val result = num1 + num2
            resultText.text = "Result: $result"
         } else {
            resultText.text = "Please enter valid numbers"
         }
      }
   }
}
```

---

1 b) Develop an activity to generate **random numbers**

XML Code:

```
<?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"
    android:gravity="center">

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Press the button to generate a number"
        android:textSize="18sp"
        android:layout_marginBottom="16dp" />

    <Button
        android:id="@+id/generateButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Generate Random Number" />
</LinearLayout>
```

Kotlin Code:

```
package com.example.numbergenerator

import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import kotlin.random.Random

class MainActivity : AppCompatActivity() {

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

        val resultTextView: TextView = findViewById(R.id.resultTextView)
        val generateButton: Button = findViewById(R.id.generateButton)

        // Set up click listener for the button
        generateButton.setOnClickListener {
            val randomNumber = Random.nextInt(1, 1000)
            resultTextView.text = "Random Number: $randomNumber"
        }
    }
}
```

---

2. a) Develop a simple list view to select an item from list

XML Code:

```
<?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">

    <!-- ListView to display the list of items -->
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</LinearLayout>
```

Kotlin Code:

```
package com.example.listviewexample

import android.os.Bundle
import android.widget.ArrayAdapter
import android.widget.ListView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import com.example.frag.R

class MainActivity : AppCompatActivity() {

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

```
        setContentView(R.layout.activity_main)

        // Find the ListView from the layout
        val listView: ListView = findViewById(R.id.listView)

        // Data to be displayed in the list
        val data = arrayOf(
            "Item 1", "Item 2", "Item 3",
            "Item 4", "Item 5", "Item 6",
            "Item 7", "Item 8", "Item 9"
        )

        // Create an ArrayAdapter to handle the data
        val adapter = ArrayAdapter(this, android.R.layout.simple_list_item_1, data)

        // Set the adapter to the ListView
        listView.adapter = adapter

        // Handle item clicks
        listView.setOnItemClickListener { parent, view, position, id ->
            val selectedItem = data[position]
            // Display a Toast with the selected item
            Toast.makeText(this, "You selected: $selectedItem", Toast.LENGTH_SHORT).show()
        }
    }
}
```

2b) Develop an activity to edit and save text to display

XML

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

    <EditText
        android:id="@+id/editText"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="100dp"
        android:hint="Enter some text"
        android:inputType="text"
        app:layout_constraintBottom_toTopOf="@+id/buttonSave"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/buttonSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Save Text"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

    <TextView
        android:id="@+id/textViewDisplay"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Entered text will appear here"
        app:layout_constraintTop_toBottomOf="@+id/buttonSave"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        android:layout_marginTop="20dp" />

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

Kotlin

```
package com.example.edit

import android.os.Bundle
import android.widget.Button
```

```
import android.widget.EditText
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    private lateinit var editText: EditText
    private lateinit var buttonSave: Button
    private lateinit var textViewDisplay: TextView

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

        // Initialize views
        editText = findViewById(R.id.editText)
        buttonSave = findViewById(R.id.buttonSave)
        textViewDisplay = findViewById(R.id.textViewDisplay)

        // Set a click listener for the button
        buttonSave.setOnClickListener {
            // Get the text from the EditText
            val enteredText = editText.text.toString()

            // Set the text in the TextView
            textViewDisplay.text = enteredText
        }
    }
}
```

---

3. a) Develop an activity using radio button to display selected option

XML Code:

```
<?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"
    android:gravity="center">

    <!-- RadioGroup containing multiple RadioButtons -->
    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <!-- RadioButton 1 -->
        <RadioButton
            android:id="@+id/radioButton1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Option 1" />

        <!-- RadioButton 2 -->
        <RadioButton
            android:id="@+id/radioButton2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Option 2" />

        <!-- RadioButton 3 -->
        <RadioButton
            android:id="@+id/radioButton3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Option 3" />
    </RadioGroup>

    <!-- Button to trigger action after selection -->
    <Button
        android:id="@+id/showSelectionButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Selected Option"
        android:layout_marginTop="20dp" />
```

```
</LinearLayout>
```

Kotlin Code:

```
package com.example.frag
import android.annotation.SuppressLint
import android.os.Bundle
import android.widget.Button
import android.widget.RadioButton
import android.widget.RadioGroup
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    @SuppressLint("MissingInflatedId")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val radioGroup = findViewById<RadioGroup>(R.id.radioGroup)
        val showSelectionButton = findViewById<Button>(R.id.showSelectionButton)
        showSelectionButton.setOnClickListener {
            val selectedId = radioGroup.checkedRadioButtonId
            if (selectedId != -1) {
                val selectedRadioButton = findViewById<RadioButton>(selectedId)
                Toast.makeText(this, "You selected: ${selectedRadioButton.text}", Toast.LENGTH_SHORT).show()
            } else {
                Toast.makeText(this, "Please select an option", Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

---

3. b) Develop an activity using Toggle switch to display on/off status

XML Code:

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

    <!-- Toggle Switch -->
    <Switch
        android:id="@+id/toggleSwitch"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="OFF"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="200dp"/>

    <!-- TextView to show current state -->
    <TextView
        android:id="@+id/statusText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Status: OFF"
        android:textSize="20sp"
        app:layout_constraintTop_toBottomOf="@id/toggleSwitch"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="40dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Kotlin Code:

```
package com.example.edit

import android.os.Bundle
import android.widget.Switch
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import com.example.toggleswitch.R

class MainActivity : AppCompatActivity() {
    private lateinit var toggleSwitch: Switch
```

```
    private lateinit var statusText: TextView

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

        // Initialize the views
        toggleSwitch = findViewById(R.id.toggleSwitch)
        statusText = findViewById(R.id.statusText)

        // Set an OnCheckedChangeListener to the Switch
        toggleSwitch.setOnCheckedChangeListener { _, isChecked ->
            // Update the TextView based on the switch position
            if (isChecked) {
                statusText.text = "Status: ON"
            } else {
                statusText.text = "Status: OFF"
            }
        }
    }
}
```

4. a) Develop an activity to display dialog window

XML Layout Code:

```
<?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"
    android:gravity="center">

    <!-- Button to show the dialog -->
    <Button
        android:id="@+id/showDialogButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Dialog" />
</LinearLayout>
```

Kotlin Code :

```
package com.example.frag

import android.os.Bundle
import android.widget.Button
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import com.example.frag.R

class MainActivity : AppCompatActivity() {

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

        val showDialogButton = findViewById<Button>(R.id.showDialogButton)

        // Button click listener to show an AlertDialog
        showDialogButton.setOnClickListener {
            showAlertDialog()
        }
    }

    private fun showAlertDialog() {
        // Create an AlertDialog
        val builder = AlertDialog.Builder(this)
        builder.setTitle("Confirmation")
        builder.setMessage("Are you sure you want to proceed?")

        // Positive button
        builder.setPositiveButton("Yes") { dialog, which ->
            Toast.makeText(this, "You clicked Yes!", Toast.LENGTH_SHORT).show()
        }

        // Negative button
        builder.setNegativeButton("No") { dialog, which ->
            Toast.makeText(this, "You clicked No!", Toast.LENGTH_SHORT).show()
        }

        // Create and show the dialog
```

```
        val dialog = builder.create()
        dialog.show()
    }
}
```

---

4. b) Write a program to check network connection of the device

XML Layout Code:

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

    <TextView
        android:id="@+id/connectionStatusTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Checking network..."
        android:textSize="20sp"
        android:layout_marginTop="200dp"
        android:layout_centerHorizontal="true"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Kotlin Code

```
package com.example.counter

import android.net.ConnectivityManager
import android.net.NetworkCapabilities
import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import android.content.Context

class MainActivity : AppCompatActivity() {

    private lateinit var connectionStatusTextView: TextView

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

        connectionStatusTextView = findViewById(R.id.connectionStatusTextView)

        // Check network connectivity
        if (isNetworkAvailable()) {
            connectionStatusTextView.text = "Network is available"
        } else {
            connectionStatusTextView.text = "No network connection"
        }
    }

    private fun isNetworkAvailable(): Boolean {
        val connectivityManager = getSystemService(Context.CONNECTIVITY_SERVICE) as ConnectivityManager
        val network = connectivityManager.activeNetwork
        val networkCapabilities = connectivityManager.getNetworkCapabilities(network)

        // Check if the device is connected to a network (Wi-Fi or Mobile)
        return networkCapabilities != null &&
networkCapabilities.hasCapability(NetworkCapabilities.NET_CAPABILITY_INTERNET)
    }
}
```

---

5. Develop an activity using timepicker to select and display time

XML Layout Code:

```
xml
Copy
<?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"
    android:gravity="center">

    <TimePicker
```

```
        android:id="@+id/timePicker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:timePickerMode="spinner"
        android:layout_marginBottom="20dp" />

    <Button
        android:id="@+id/showTimeButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Selected Time" />

    <TextView
        android:id="@+id/timeDisplay"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Selected Time: "
        android:textSize="18sp"
        android:layout_marginTop="20dp"/>
</LinearLayout>
```

Kotlin Code :

```
package com.example.frag

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

class MainActivity : AppCompatActivity() {

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

        val timePicker = findViewById<TimePicker>(R.id.timePicker)
        val showTimeButton = findViewById<Button>(R.id.showTimeButton)
        val timeDisplay = findViewById<TextView>(R.id.timeDisplay)

        // Set a listener for the button click
        showTimeButton.setOnClickListener {
            // Get the selected hour and minute from the TimePicker
            val hour = timePicker.hour
            val minute = timePicker.minute

            // Format the time to display
            val formattedTime = String.format("%02d:%02d", hour, minute)

            // Display the selected time in the TextView
            timeDisplay.text = "Selected Time: $formattedTime"
        }
    }
}
```

---

6. Develop an activity to display current time in digital clock format

XML Layout Code:

```
<?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:gravity="center"
    android:background="#000000"
    android:padding="16dp">

    <TextView
        android:id="@+id/clockTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00:00"
        android:textSize="48sp"
        android:textColor="#FFFFFF"
        android:textStyle="bold" />
</LinearLayout>
```

Kotlin Code (MainActivity):

```
package com.example.numbergenerator

import android.os.Bundle
```

```
import android.os.Handler
import android.os.Looper
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import com.example.numbergenerator.R
import java.text.SimpleDateFormat
import java.util.*

class MainActivity : AppCompatActivity() {

    private lateinit var clockTextView: TextView
    private val handler = Handler(Looper.getMainLooper())

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

        // Find the TextView by ID
        clockTextView = findViewById(R.id.clockTextView)

        // Start updating the clock
        updateClock()
    }

    private fun updateClock() {
        // Use handler to repeatedly update the clock every second
        handler.post(object : Runnable {
            override fun run() {
                // Get the current time
                val currentTime = Calendar.getInstance().time
                val formatter = SimpleDateFormat("hh:mm:ss a", Locale.getDefault())
                val formattedTime = formatter.format(currentTime)

                // Update the TextView with the formatted time
                clockTextView.text = formattedTime

                // Schedule the next update after 1 second (1000 ms)
                handler.postDelayed(this, 1000)
            }
        })
    }

    override fun onDestroy() {
        super.onDestroy()
        // Stop the updates when the activity is destroyed
        handler.removeCallbacksAndMessages(null)
    }
}
```

---

7. Develop an activity to validate user using username and password

XML Layout Code:

```
<?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"
    android:gravity="center">

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Username"
        android:inputType="textPersonName" />

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Password"
        android:inputType="textPassword" />

    <Button
        android:id="@+id/loginButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_marginTop="16dp" />

    <TextView
        android:id="@+id/resultText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingTop="16dp"
        android:text="" 
```

```
        android:textSize="18sp" />
</LinearLayout>
```

Kotlin Code (MainActivity):

```
package com.example.loginvalidation

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

class MainActivity : AppCompatActivity() {

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

        val usernameField = findViewById<EditText>(R.id.username)
        val passwordField = findViewById<EditText>(R.id.password)
        val loginButton = findViewById<Button>(R.id.loginButton)
        val resultText = findViewById<TextView>(R.id.resultText)

        val validUsername = "admin"
        val validPassword = "1234"

        loginButton.setOnClickListener {
            val username = usernameField.text.toString()
            val password = passwordField.text.toString()

            if (username == validUsername && password == validPassword) {
                resultText.text = "Login Successful"
            } else {
                resultText.text = "Invalid Username or Password"
            }
        }
    }
}
```

---

8. Develop an activity to display counter using increment, decrement and reset button

XML Layout Code:

```
<?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:gravity="center"
    android:padding="16dp">

    <TextView
        android:id="@+id/counterTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="24dp"
        android:text="0"
        android:textSize="32sp"
        android:textStyle="bold" />

    <Button
        android:id="@+id/incrementButton"
        android:layout_width="155dp"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="Increment" />

    <Button
        android:id="@+id/decrementButton"
        android:layout_width="155dp"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="Decrement" />

    <Button
        android:id="@+id/resetButton"
        android:layout_width="155dp"
        android:layout_height="wrap_content"
        android:text="Reset" />

</LinearLayout>
```

Kotlin

```
package com.example.numbergenerator

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

class MainActivity : AppCompatActivity() {
    private var counter = 0

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

        // Find views by ID
        val counterTextView: TextView = findViewById(R.id.counterTextView)
        val incrementButton: Button = findViewById(R.id.incrementButton)
        val decrementButton: Button = findViewById(R.id.decrementButton)
        val resetButton: Button = findViewById(R.id.resetButton)

        // Update the counter display
        fun updateCounter() {
            counterTextView.text = counter.toString()
        }

        // Increment button logic
        incrementButton.setOnClickListener {
            counter++
            updateCounter()
        }

        // Decrement button logic
        decrementButton.setOnClickListener {
            if (counter > 0) counter--  // Prevent negative values
            updateCounter()
        }

        // Reset button logic
        resetButton.setOnClickListener {
            counter = 0
            updateCounter()
        }

        // Initialize counter display
        updateCounter()
    }
}
```

---

9. Develop an application to implement simple calculator

XML Code:

xml

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

    <EditText
        android:id="@+id/number1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="numberDecimal" />

    <EditText
        android:id="@+id/number2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter second number"
        android:inputType="numberDecimal" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="center">

        <Button
            android:id="@+id/addButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
            android:text="Add" />

        <Button
            android:id="@+id/subtractButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Subtract" />

        <Button
            android:id="@+id/multiplyButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Multiply" />

        <Button
            android:id="@+id/divideButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Divide" />
    </LinearLayout>

    <TextView
        android:id="@+id/resultText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Result will be displayed here"
        android:textSize="18sp"
        android:paddingTop="16dp" />
</LinearLayout>
```

KOTLIN

```
package com.example.calcu

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

class MainActivity : AppCompatActivity() {

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

        val number1 = findViewById<EditText>(R.id.number1)
        val number2 = findViewById<EditText>(R.id.number2)
        val resultText = findViewById<TextView>(R.id.resultText)

        val addButton = findViewById<Button>(R.id.addButton)
        val subtractButton = findViewById<Button>(R.id.subtractButton)
        val multiplyButton = findViewById<Button>(R.id.multiplyButton)
        val divideButton = findViewById<Button>(R.id.divideButton)

        addButton.setOnClickListener {
            calculate(number1, number2, resultText) { a, b -> a + b }
        }

        subtractButton.setOnClickListener {
            calculate(number1, number2, resultText) { a, b -> a - b }
        }

        multiplyButton.setOnClickListener {
            calculate(number1, number2, resultText) { a, b -> a * b }
        }

        divideButton.setOnClickListener {
            calculate(number1, number2, resultText) { a, b ->
                if (b != 0.0) a / b else null
            }
        }
    }

    private fun calculate(
        num1Field: EditText,
        num2Field: EditText,
        resultText: TextView,
        operation: (Double, Double) -> Double?
    ) {
        val num1 = num1Field.text.toString().toDoubleOrNull()
```

```
        val num2 = num2Field.text.toString().toDoubleOrNull()

        if (num1 != null && num2 != null) {
            val result = operation(num1, num2)
            resultText.text = if (result != null) "Result: $result" else "Error: Division by zero"
        } else {
            resultText.text = "Please enter valid numbers"
        }
    }
}
```

---

10. Develop a phone dialer activity with call and save options.

XML

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

    <!-- EditText to enter the phone number -->
    <EditText
        android:id="@+id/phoneNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter phone number"
        android:inputType="phone"
        android:textSize="20sp" />

    <!-- Call Button -->
    <Button
        android:id="@+id/callButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Call"
        android:textSize="18sp"
        android:layout_marginTop="16dp" />

    <!-- Save Button -->
    <Button
        android:id="@+id/saveButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Save"
        android:textSize="18sp"
        android:layout_marginTop="16dp" />
</LinearLayout>
```

Kotlin

```
package com.example.dialer

import android.content.Intent
import android.net.Uri
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    private lateinit var phoneNumberEditText: EditText
    private lateinit var callButton: Button
    private lateinit var saveButton: Button

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

        // Initialize views
        phoneNumberEditText = findViewById(R.id.phoneNumberEditText)
        callButton = findViewById(R.id.callButton)
        saveButton = findViewById(R.id.saveButton)
```

```
        // Call button functionality
        callButton.setOnClickListener {
            val phoneNumber = phoneNumberEditText.text.toString().trim()
            if (phoneNumber.isNotEmpty()) {
                // Use an intent to initiate a phone call
                val dialIntent = Intent(Intent.ACTION_DIAL, Uri.parse("tel:$phoneNumber"))
                startActivity(dialIntent)
            } else {
                Toast.makeText(this, "Please enter a phone number", Toast.LENGTH_SHORT).show()
            }
        }

        // Save button functionality
        saveButton.setOnClickListener {
            val phoneNumber = phoneNumberEditText.text.toString().trim()
            if (phoneNumber.isNotEmpty()) {
                // Save the phone number, for simplicity we'll just display it in a Toast
                Toast.makeText(this, "Phone number saved: $phoneNumber", Toast.LENGTH_SHORT).show()

                // Clear the input field
                phoneNumberEditText.text.clear()
            } else {
                Toast.makeText(this, "Please enter a phone number", Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```