

## MainActivity.kt

```
package com.example.simplecal
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 lateinit var inputDisplay: TextView
    private lateinit var outputDisplay: TextView
    private var firstValue: Double = 0.0
    private var secondValue: Double = 0.0
    private var currentOperation: String? = null
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        inputDisplay = findViewById(R.id.input)
        outputDisplay = findViewById(R.id.output)
        val buttons = listOf<Button>(
            findViewById(R.id.btn0),
            findViewById(R.id.btn1),
            findViewById(R.id.btn2),
            findViewById(R.id.btn3),
            findViewById(R.id.btn4),
            findViewById(R.id.btn5),
            findViewById(R.id.btn6),
            findViewById(R.id.btn7),
            findViewById(R.id.btn8),
            findViewById(R.id.btn9),
            findViewById(R.id.add),
            findViewById(R.id.subtract),
            findViewById(R.id.multiply),
            findViewById(R.id.divide),
            findViewById(R.id.equal),
            findViewById(R.id.btnClear)
        )
        for (button in buttons) {
            button.setOnClickListener { onButtonClick(button) }
        }
    }
    private fun onButtonClick(button: Button) {
        when (button.id) {
            R.id.btnClear -> clear()
            R.id.equal -> calculateResult()
            in listOf(R.id.add, R.id.subtract, R.id.multiply, R.id.divide) ->
```

```

        setOperation(button.text.toString())
        else -> appendToInput(button.text.toString())
    }
}

private fun clear() {
    inputDisplay.text = ""
    outputDisplay.text = ""
    firstValue = 0.0
    secondValue = 0.0
    currentOperation = null
}

private fun appendToInput(value: String) {
    inputDisplay.append(value)
}

private fun setOperation(operation: String) {
    if (inputDisplay.text.isNotEmpty()) {
        firstValue = inputDisplay.text.toString().toDouble()
        currentOperation = operation
        inputDisplay.text = ""
    }
}

private fun calculateResult() {
    if (inputDisplay.text.isNotEmpty() && currentOperation != null) {
        secondValue = inputDisplay.text.toString().toDouble()
        val result = when (currentOperation) {
            "+" -> firstValue + secondValue
            "-" -> firstValue - secondValue
            "*" -> firstValue * secondValue
            "/" -> if (secondValue != 0.0) firstValue / secondValue else Double.NaN
            else -> 0.0
        }
        outputDisplay.text = result.toString()
        currentOperation = null
    }
}
}

```

### activity\_main.xml

```

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

```

```

        android:id="@+id/input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="32sp"
        android:padding="16dp"
        android:background="#E0E0E0"
        android:layout_alignParentTop="true"
        android:gravity="end"/>
<TextView
    android:id="@+id/output"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="24sp"
    android:layout_below="@id/input"
    android:gravity="end"/>
<GridLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/output"
    android:layout_marginTop="16dp"
    android:rowCount="5"
    android:columnCount="4">
    <Button android:id="@+id/btn1" android:text="1" />
    <Button android:id="@+id/btn2" android:text="2" />
    <Button android:id="@+id/btn3" android:text="3" />
    <Button android:id="@+id/add" android:text="+" />
    <Button android:id="@+id/btn4" android:text="4" />
    <Button android:id="@+id/btn5" android:text="5" />
    <Button android:id="@+id/btn6" android:text="6" />
    <Button android:id="@+id/subtract" android:text="-" />
    <Button android:id="@+id/btn7" android:text="7" />
    <Button android:id="@+id/btn8" android:text="8" />
    <Button android:id="@+id/btn9" android:text="9" />
    <Button android:id="@+id/multiply" android:text="*" />
    <Button android:id="@+id/btnClear" android:text="C" />
    <Button android:id="@+id/btn0" android:text="0" />
    <Button android:id="@+id/equal" android:text="=" />
    <Button android:id="@+id/divide" android:text="/" />
</GridLayout>
</RelativeLayout>

```

## AndroidManifest.xml

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.simplecalc">

```

```

<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportRtl="true"
    android:theme="@style/Theme.AppCompat.Light.NoActionBar">
    <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>

```

## MainActivity2.kt

```

package com.example.simplecal
import android.os.Bundle
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
class MainActivity2 : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets
        }
    }
}

```