MyCalc 😄 :

- New Project > Empty Views Activity > MyCalc (Name of the App with Kotlin Language) > Finish
- Go to build.gradle.scripts(Module: app) and update the compileSdk with latest like today is 36 is latest.
- Go to libs.versions.toml and update the kotlin latest version like today's the latest version is 2.2.20 and sync it.

Go to app > resource (res) folder > values and then create a new variable resource file with name "style" this will contain the style of our app icon.

In that file "style.xml" add this code :

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <style name = "CalculatorButton" parent ="Widget.MaterialComponents.Button.UnelevatedButton">
        <item name = "android:layout_width">0dp</item>
        <item name = "android:layout_height">0dp</item>
        <item name = "android:layout_margin">4dp</item>
        <item name = "android:textSize">24sp</item>
        <item name = "android:textColor">@android:color/white</item>
        <item name = "backgroundTint">@null</item>
        <item name = "rippleColor">#33FFFFFF</item>
        <item name = "cornerRadius">24dp</item>
    </style>
</resources>
```

→ Create a new file under res > drawable > new Resource File with name (button_background)

In that file the code is this :

```
<?xml version="1.0" encoding="utf-8"?>
<ripple xmlns:android="http://schemas.android.com/apk/res/android"
    android:color="#33FFFFFF">
    <item>
        <shape android:shape="rectangle">
            <corners android:radius="24dp"/>
            <solid android:color="#424242"/>
        </shape>
    </item>
</ripple>
```

→ Create a new file same in the drawable > new Resource File with name (button_operator_background)

When we add this code into this file :

```
xml version="1.0" encoding="utf-8"?>
<ripple xmlns:android="http://schemas.android.com/apk/res/android"
  android:color="#33FFFFFF">
  <item>
    <shape android:shape="rectangle">
      <corners android:radius="24dp"/>
      <solid android:color="@color/operatorcolor"/>
    </shape>
  </item>
</ripple>
```

Then here is an error in the @ color/operatorcolor. Because we need to initialize the color first in values > colors.xml file. So, we provide the color info/code into this file (colors.xml file) :

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <color name="colorPrimary">#2196F3</color>
  <color name = "colorPrimaryDary">#1976D2</color>
  <color name = "colorAccent">#FF4081</color>
  <color name = "operatorColor">#FF9800</color>
  <color name = "numberColor">#424242</color>
  <color name = "equalColor">#4CAF50</color>
  <color name = "clearColor">#F44336</color>
  <color name = "displayBG">#1E1E1E</color>
  <color name = "displayText">#FFFFFF</color>
</resources>
```

Similarly we have button_equal_background, Code :

```
<?xml version="1.0" encoding="utf-8"?>
<ripple xmlns:android="http://schemas.android.com/apk/res/android"
  android:color="#3FFFFF">
  <item>
    <shape android:shape="rectangle">
      <corners android:radius="24dp"/>
      <solid android:color="@color/equalColor"/>
    </shape>
  </item>
</ripple>
```

Also button_clear_background :

```
<?xml version="1.0" encoding="utf-8"?>
<ripple xmlns:android="http://schemas.android.com/apk/res/android"
  android:color="#3FFFFF">
  <item>
```

```
    <shape android:shape="rectangle">
        <corners android:radius="24dp"/>
        <solid android:color="@color/clearColor"/>
    </shape>
  </item>
</ripple>
```

- activity_mail.xml is the main activity where we initialize all the main UI of the MyCalc App:

First of app select the select the background of the app. There is the cardview and under that view we have the two linear layouts of the textviews.

Inside that cardview we have a linear layout.

Code is this in the “activity_main/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:id="@+id/main"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity"
  android:background="#CECECE">

  <!-- Display Section -->
  <androidx.cardview.widget.CardView
    android:id="@+id/displayCard"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="16dp"
    app:cardCornerRadius="24dp"
    app:layout_constraintTop_toTopOf="parent">

    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="vertical"
      android:background="@color/displayBG"
      android:padding="24dp">

      <TextView
        android:id="@+id/previouscalculationtextview"
```

```
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="end"
            android:textColor="#80FFFFFF"
            android:textSize="20sp"
            android:maxLines="1" />

        <EditText
            android:id="@+id/resultTextView"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="48sp"
            android:textColor="@color/displayText"
            android:text="0"
            android:maxLines="1"
            android:gravity="end"
            android:layout_marginTop="8dp"
            android:inputType="none" />
    </LinearLayout>
  </androidx.cardview.widget.CardView>

  <androidx.cardview.widget.CardView
    android:id="@+id/buttonsCard"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_margin="16dp"
    app:cardCornerRadius="24dp"
    app:cardElevation="8dp"
    app:layout_constraintTop_toBottomOf="@id/displayCard"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintBottom_toBottomOf="parent">

    <GridLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:columnCount="4"
        android:rowCount="5">

        <!-- Row 1 -->
        <Button
            android:id="@+id/btnClear"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="C"
```

```
            android:background="@drawable/button_clear_background" />

        <Button
            android:id="@+id/btnBackspace"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="⌫"
            android:background="@drawable/button_operator_background" />

        <com.google.android.material.button.MaterialButton
            android:id="@+id/btnPercent"
            style = "@style/CalculatorButton"
            android:layout_columnWeight="1"
            android:layout_rowWeight="1"
            android:background="@drawable/button_operator_background"
            android:text="%"/>

        <Button
            android:id="@+id/btnDivide"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="/"
            android:background="@drawable/button_operator_background" />

        <!-- Row 2 -->
        <Button
            android:id="@+id/btn7"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="7"
            android:background="@drawable/button_clear_background" />

        <Button
            android:id="@+id/btn8"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="8"
            android:background="@drawable/button_operator_background" />

        <com.google.android.material.button.MaterialButton
            android:id="@+id/btn9"
            style = "@style/CalculatorButton"
```

```
            android:layout_columnWeight="1"
            android:layout_rowWeight="1"
            android:background="@drawable/button_operator_background"
            android:text="9"/>

        <Button
            android:id="@+id/btnMultiply"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="*"
            android:background="@drawable/button_operator_background" />

        <!-- Row 3 -->
        <Button
            android:id="@+id/btn4"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="4"
            android:background="@drawable/button_clear_background" />

        <Button
            android:id="@+id/btn5"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="5"
            android:background="@drawable/button_operator_background" />

        <com.google.android.material.button.MaterialButton
            android:id="@+id/btn6"
            style = "@style/CalculatorButton"
            android:layout_columnWeight="1"
            android:layout_rowWeight="1"
            android:background="@drawable/button_operator_background"
            android:text="6"/>

        <Button
            android:id="@+id/btnSubtract"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="-"
            android:background="@drawable/button_operator_background" />
```

```
        <!-- Row 4 -->
        <Button
            android:id="@+id/btn1"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="1"
            android:background="@drawable/button_clear_background" />

        <Button
            android:id="@+id/btn2"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="2"
            android:background="@drawable/button_operator_background" />

        <com.google.android.material.button.MaterialButton
            android:id="@+id/btn3"
            style = "@style/CalculatorButton"
            android:layout_columnWeight="1"
            android:layout_rowWeight="1"
            android:background="@drawable/button_operator_background"
            android:text="3"/>

        <Button
            android:id="@+id/btnAddtion"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="+"
            android:background="@drawable/button_operator_background" />

        <!-- Row 5 -->

        <Button
            android:id="@+id/btn0"
            style="@style/CalculatorButton"
            android:layout_columnWeight="2"
            android:layout_columnSpan="2"
            android:text="0"
            android:layout_rowWeight="1"
            android:background="@drawable/button_background" />

        <Button
            android:id="@+id/btnDot"
```

```
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="."
            android:background="@drawable/button_background" />

        <Button
            android:id="@+id/btnEqual"
            style="@style/CalculatorButton"
            android:layout_rowWeight="1"
            android:layout_columnWeight="1"
            android:text="="
            android:background="@drawable/button_equal_background" />
    </GridLayout>
  </androidx.cardview.widget.CardView>

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

Now set our backend of the app in the MainActivity.kt file.

First, we initialize of the buttons of the app :

```
// Let's first set all of the buttons from 0 to 9 :
val btn0:Button = findViewById<Button>(R.id.btn0)
val btn1:Button = findViewById<Button>(R.id.btn1)
val btn2:Button = findViewById<Button>(R.id.btn2)
val btn3:Button = findViewById<Button>(R.id.btn3)
val btn4:Button = findViewById<Button>(R.id.btn4)
val btn5:Button = findViewById<Button>(R.id.btn5)
val btn6:Button = findViewById<Button>(R.id.btn6)
val btn7:Button = findViewById<Button>(R.id.btn7)
val btn8:Button = findViewById<Button>(R.id.btn8)
val btn9:Button = findViewById<Button>(R.id.btn9)

// Initalization of the Operators :
val btnAdd:Button = findViewById<Button>(R.id.btnAddtion)
val btnMinus:Button = findViewById<Button>(R.id.btnSubtract)
val btnMultiply:Button = findViewById<Button>(R.id.btnMultiply)
val btnDivide:Button = findViewById<Button>(R.id.btnDivide)
val btnPercent:Button = findViewById<Button>(R.id.btnPercent)

// Initalization of the Others remaining buttons :

val btnClear:Button = findViewById<Button>(R.id.btnClear)
val btnEqual:Button = findViewById<Button>(R.id.btnEqual)
```

```
val btnDot:Button = findViewById<Button>(R.id.btnDot)
val btnBackSpace:Button = findViewById<Button>(R.id.btnBackspace)
```

--

```
private lateinit var resultTextView: TextView
private lateinit var previousCalculationTextView: TextView

private var firsNumber = 0.0
private var operation = ""
private var isNewOperation = true
```

lateint means we initialize the variable without value, this code is before the override fun in MainActivity.kt

```
// Initialazation of the lateinit variables :
resultTextView = findViewById(R.id.resultTextView)
previousCalculationTextView = findViewById(R.id.previouscalculationtextview)
```

Code of the setOnClickListener of the Nos and append function in kotlin :

```
// Let's set what is enter on screen when the user click on the button from 0 to 9 :
    btn0.setOnClickListener {}
    btn1.setOnClickListener {}
    btn2.setOnClickListener {}
    btn3.setOnClickListener {}
    btn4.setOnClickListener {}
    btn5.setOnClickListener {}
    btn6.setOnClickListener {}
    btn7.setOnClickListener {}
    btn8.setOnClickListener {}
    btn9.setOnClickListener {}

    // Here is the append Function :
    private fun appendNumber(number : String) {
        if(isNewOperation) {
            resultTextView.text = number
            isNewOperation=false
        }
        else {
            resultTextView.text = "${resultTextView.text}$number"
        }
    }
```

MainActivity.kt Code :

```
package com.example.mycalc

import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import android.widget.Toast
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MainActivity : AppCompatActivity() {
    private lateinit var resultTextView: TextView
    private lateinit var previousCalculationTextView: TextView

    private var firsNumber = 0.0
    private var operation = ""
    private var isNewOperation = true

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

        // Initialazation of the lateinit variables :
        resultTextView = findViewById(R.id.resultTextView)
        previousCalculationTextView = findViewById(R.id.previouscalculationtextview)


        // Let's first set all of the buttons from 0 to 9 :
        val btn0: Button = findViewById<Button>(R.id.btn0)
        val btn1: Button = findViewById<Button>(R.id.btn1)
        val btn2: Button = findViewById<Button>(R.id.btn2)
        val btn3: Button = findViewById<Button>(R.id.btn3)
        val btn4: Button = findViewById<Button>(R.id.btn4)
        val btn5: Button = findViewById<Button>(R.id.btn5)
        val btn6: Button = findViewById<Button>(R.id.btn6)
        val btn7: Button = findViewById<Button>(R.id.btn7)
        val btn8: Button = findViewById<Button>(R.id.btn8)
        val btn9: Button = findViewById<Button>(R.id.btn9)

        // Initalization of the Operators :
        val btnAdd: Button = findViewById<Button>(R.id.btnAddtion)
        val btnMinus: Button = findViewById<Button>(R.id.btnSubtract)
        val btnMultiply: Button = findViewById<Button>(R.id.btnMultiply)
        val btnDivide: Button = findViewById<Button>(R.id.btnDivide)
```

```
    val btnPercent: Button = findViewById<Button>(R.id.btnPercent)

    // Initalization of the Others remaining buttons :

    val btnClear: Button = findViewById<Button>(R.id.btnClear)
    val btnEqual: Button = findViewById<Button>(R.id.btnEqual)
    val btnDot: Button = findViewById<Button>(R.id.btnDot)
    val btnBackSpace: Button = findViewById<Button>(R.id.btnBackspace)

    // Let's set what is enter on screen when the user click on the button from 0 to 9 :
    btn0.setOnClickListener { appendNumber("0") }
    btn1.setOnClickListener { appendNumber("1") }
    btn2.setOnClickListener { appendNumber("2") }
    btn3.setOnClickListener { appendNumber("3") }
    btn4.setOnClickListener { appendNumber("4") }
    btn5.setOnClickListener { appendNumber("5") }
    btn6.setOnClickListener { appendNumber("6") }
    btn7.setOnClickListener { appendNumber("7") }
    btn8.setOnClickListener { appendNumber("8") }
    btn9.setOnClickListener { appendNumber("9") }
    btnDot.setOnClickListener { appendNumber(".") }

    btnAdd.setOnClickListener { setOperation("+") }
    btnMinus.setOnClickListener { setOperation("-") }
    btnMultiply.setOnClickListener { setOperation("*") }
    btnDivide.setOnClickListener { setOperation("/") }

    btnEqual.setOnClickListener { calculateResult() }
    btnClear.setOnClickListener { clearCalculator() }

    btnBackSpace.setOnClickListener { deleteNum() }
    }

  // setOpeation function :
  fun setOperation(operator: String) {
    firsNumber = resultTextView.text.toString().toDouble()
    operation = operator
    isNewOperation = true
    previousCalculationTextView.text = "$firsNumber $operation"
    resultTextView.text = "0.0"
  }

    // Here is the append Function :
    private fun appendNumber(number: String) {
        if (isNewOperation) {
            resultTextView.text = number
```

```
            isNewOperation = false
        } else {
            resultTextView.text = "${resultTextView.text}$number"
        }
    }

    // a Function of the calculation for the Calculator :
    private fun calculateResult() {
        try {
            val secondNumber :Double = resultTextView.text.toString().toDouble()
            val result: Double = when(operation) {
                "+" -> firsNumber+secondNumber
                "-" -> firsNumber-secondNumber
                "/"-> firsNumber/secondNumber
                "*" -> firsNumber*secondNumber
                else -> secondNumber
            }
            previousCalculationTextView.text = "$firsNumber $operation $secondNumber"
            resultTextView.text = result.toString()
            isNewOperation = true
        } catch (e: Exception) {
            resultTextView.text = "Error"
        }
    }

    // Clear Function :
    private fun clearCalculator() {
        resultTextView.text = "0"
        previousCalculationTextView.text = ""
        operation = ""
        firsNumber = 0.0
        isNewOperation = true;
    }

    // Delete Function :
    private fun deleteNum() {
        if(resultTextView.text.isNotEmpty() && resultTextView.text != "0.0" && resultTextView.text != "Error") {
            resultTextView.text = resultTextView.text.drop(1)
        } else {
            Toast.makeText(this, "Invalid Operator", Toast.LENGTH_SHORT).show()
        }
    }
}
```