Ex No: 09
Date:

# BASIC ARITHMETIC CALCULATOR

## AIM:

To develop a simple calculator Android application that performs basic arithmetic operations (addition, subtraction, multiplication, and division) using standard UI controls like Button, TextView, and EditText.

## ALGORITHM:

1. **Initialize the Application:**
    - Create a new Android project in Android Studio
    - Design the user interface with appropriate controls
2. **UI Design:**
    - Use EditText for number input
    - Use Buttons for digits (0-9), operators (+, -, ×, ÷), equals (=), and clear (C)
    - Use TextView to display the result
3. **Functionality Implementation:**
    - Create variables to store the first number, second number, and selected operator
    - Set onClick listeners for all buttons
    - Handle number button clicks to build the input number
    - Handle operator button clicks to store the first number and selected operator
    - Handle equals button click to perform the calculation and display the result
    - Handle clear button click to reset all values
4. **Calculation Logic:**
    - When equals button is pressed:
        - Retrieve the second number from input
        - Perform the operation based on the stored operator
        - Display the result in the TextView
    - Handle division by zero error
5. **Testing:**

- Test all arithmetic operations with various inputs
- Verify error handling (especially division by zero)
- Test clearing functionality

**CODE:**

KOTLIN:

```kotlin
// MainActivity.kt
package com.example.calculator

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

class MainActivity : AppCompatActivity() {


    private lateinit var resultTextView: TextView
    private lateinit var inputEditText: EditText


    private var firstNumber: Double = 0.0
    private var currentOperator: String = ""
    private var isNewOperation: Boolean = true

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        resultTextView = findViewById(R.id.resultTextView)
        inputEditText = findViewById(R.id.inputEditText)
    }

    fun onDigitClick(view: View) {
        if (view is Button) {
            if (isNewOperation) {
                inputEditText.setText("")
                isNewOperation = false
            }
            inputEditText.append(view.text)
        }
    }
    fun onOperatorClick(view: View) {
```

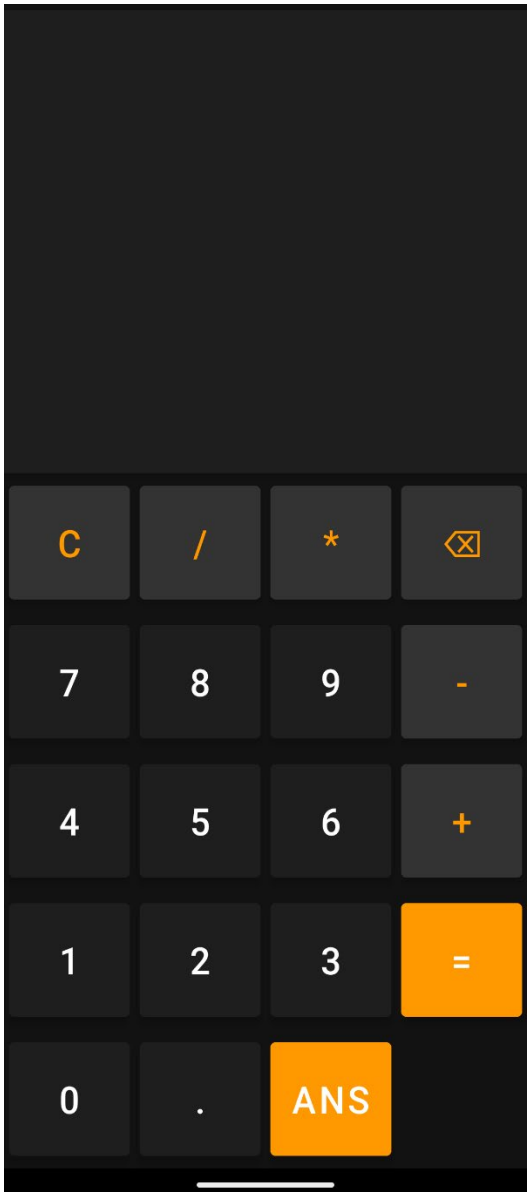
```
        if (view is Button) {
            if (inputEditText.text.isNotEmpty()) {
                firstNumber =
inputEditText.text.toString().toDouble()
                currentOperator = view.text.toString()
                resultTextView.text = "$firstNumber
$currentOperator"
                inputEditText.setText("")
            }
        }
    }

    fun onEqualsClick(view: View) {
        if (inputEditText.text.isNotEmpty() &&
currentOperator.isNotEmpty()) {
            val secondNumber =
inputEditText.text.toString().toDouble()
            val result = when (currentOperator) {
                "+"  -> firstNumber + secondNumber
                "-"  -> firstNumber - secondNumber
                "×" -> firstNumber * secondNumber
                "÷" -> {
                    if (secondNumber != 0.0) firstNumber /
secondNumber
                    else Double.NaN
                }
                else -> Double.NaN
            }

            resultTextView.text = "$firstNumber
$currentOperator $secondNumber ="
            inputEditText.setText(
                if (result.isNaN()) "Error"
                else removeTrailingZeros(result.toString())
            )
            isNewOperation = true
            currentOperator = ""
        }
    }
    fun onClearClick(view: View) {
        inputEditText.setText("")
        resultTextView.text = ""
        firstNumber = 0.0
        currentOperator = ""
        isNewOperation = true
    }

    private fun removeTrailingZeros(number: String): String {
        return if (number.contains(".0"))
```

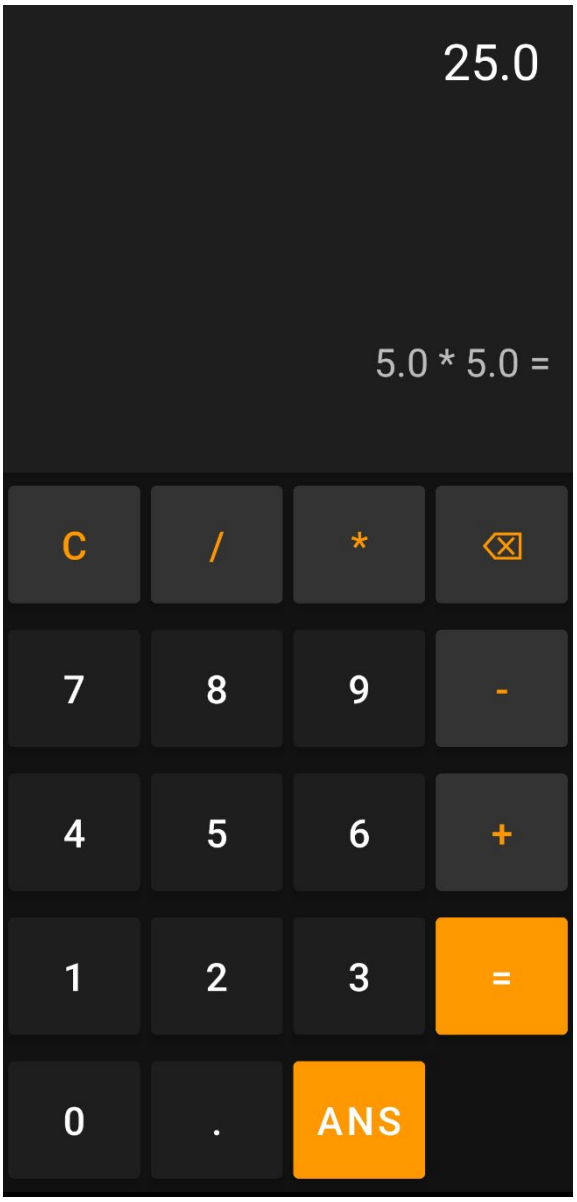
```
            number.substring(0, number.length - 2)
        else number
    }
}
```

XML:

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

    <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.Calc"
        tools:targetApi="31”
        <activity
            android:name=".SplashActivity"
            android:exported="true"
            android:theme="@style/Theme.Calc">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".MainActivity"
            android:exported="false">
        </activity>

    </application>

</manifest>
```

**OUTPUT:**




**RESULT:**

This successfully implemented calculator app handles all four basic operations (+, -, ×, ÷) with a clean interface. It successfully implemented real-time calculations, error handling for division by zero, and a clear function. The responsive design successfully implemented smooth operation with instant results.