EX NO:2 **REGISTER NO:210701509**

DATE:

SIMPLE CALCULATOR

AIM :

Develop a simple calculator to perform arithmetic and mathematical functions using Math class.

PROCEDURE:

- Open Android Studio and import the package
- In activity_main.xml drag and drop the buttons
- The button need to perform actions to change the colour, font size and background colour
- Click android virtual device that should control the toolbar
- Design the graphical layout with the textview and buttons
- Run the application
- The version of android and name is displayed
- The theme of the file is also mentioned in a file
- Run the file using the version which is displayed to the users.

PROGRAM CODE:

AndroidManifest.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.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

```
<?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>
```

MainActivity.kt

```
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
    }
}
```

OUTPUT:

RESULT: