

EXP 9

CALCULATOR APP

AIM

To develop a **simple and cute Calculator App** in Android Studio using Kotlin, allowing the user to perform basic arithmetic operations (Addition, Subtraction, Multiplication, Division) with a mobile-friendly interface and clear input/output validation.

ALGORITHM

1. Start the app.
 2. Display two input fields for numbers.
 3. Show buttons: **+** **-** **×** **÷** and (clear).
 4. User enters two numbers and taps a button.
 5. App checks if both inputs are valid numbers:
 - If not: show a toast message “Enter valid numbers”.
 - If valid:
 - Perform the selected operation.
 - Show the result on the screen.
 6. Clear button resets everything.
 7. End.
-

CODE

MainActivity.kt

```
kotlin CopyEdit
package com.example.calci
import
android.os.Bundle import
android.widget.*
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {
    private lateinit var num1: EditText
    private lateinit var num2: EditText
    private lateinit var result: TextView
    private lateinit var addBtn: Button
    private lateinit var subBtn: Button
    private lateinit var mulBtn: Button
    private lateinit var divBtn: Button
    private lateinit var clearBtn: Button
    override fun onCreate(savedInstanceState:
Bundle?) {
```

```

super.onCreate(savedInstanceState)
setContentView(R.layout.activity_main)
    num1 = findViewById(R.id.number1)
num2 = findViewById(R.id.number2)
result = findViewById(R.id.result)
addBtn = findViewById(R.id.add)
subBtn = findViewById(R.id.subtract)
mulBtn = findViewById(R.id.multiply)
divBtn = findViewById(R.id.divide)
clearBtn = findViewById(R.id.clear)
    addBtn.setOnClickListener { calculate("+")
}
    subBtn.setOnClickListener { calculate("-")
}
    mulBtn.setOnClickListener { calculate("*")
}
    divBtn.setOnClickListener { calculate("/")
}
    clearBtn.setOnClickListener {
num1.text.clear()
num2.text.clear()
result.text = ""
    }
    private fun calculate(op:
String) {
        val n1Text =
num1.text.toString()
        val n2Text =
num2.text.toString()
        if (n1Text.isEmpty() || n2Text.isEmpty())
{
            Toast.makeText(this, "Enter valid numbers",
Toast.LENGTH_SHORT).show()
return
        }
        val n1 =
n1Text.toDouble()
        val n2 =
n2Text.toDouble()
        val res
= when (op) {
            "+" -> n1 + n2
            "-" -> n1 - n2
            "*" -> n1 * n2
            "/" -> {
                if (n2 == 0.0) {
                    Toast.makeText(this, "Cannot divide by zero",
Toast.LENGTH_SHORT).show()
                    return
                }
                n1 / n2
            }
            else -> 0.0
        }
        result.text = "Result: $res"
    }
}

```

activity_main.xml (Cute Styling UI)

```

xml CopyEdit
<?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:background="#FFF1F8"      android:padding="24dp"
android:gravity="center">
```

```
    <TextView          android:text="Cute
Calculator 🍷"          android:textSize="28sp"
    android:textColor="#E91E63"
    android:layout_marginBottom="16dp"
    android:textStyle="bold"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
```

```
    <EditText
        android:id="@+id/number1"
    android:hint="Enter Number 1"
    android:inputType="numberDecimal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E91E63"
    android:padding="10dp"
        android:layout_marginBottom="12dp"/>
```

```
    <EditText
        android:id="@+id/number2"
    android:hint="Enter Number 2"
    android:inputType="numberDecimal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E91E63"
    android:padding="10dp"
        android:layout_marginBottom="24dp"/>
```

```
    <LinearLayout
        android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:layout_marginBottom="16dp">
```

```
        <Button
            android:id="@+id/add"
    android:text="+ "
    android:layout_width="0dp"
    android:layout_weight="1"
    android:layout_height="wrap_content"
    android:backgroundTint="#F8BBD0" />
```

```
        <Button
            android:id="@+id/subtract"
    android:text="- "
    android:layout_width="0dp"
    android:layout_weight="1"
    android:layout_height="wrap_content"
    android:backgroundTint="#F8BBD0"
    android:layout_marginStart="8dp"/>
```

```
        <Button
```

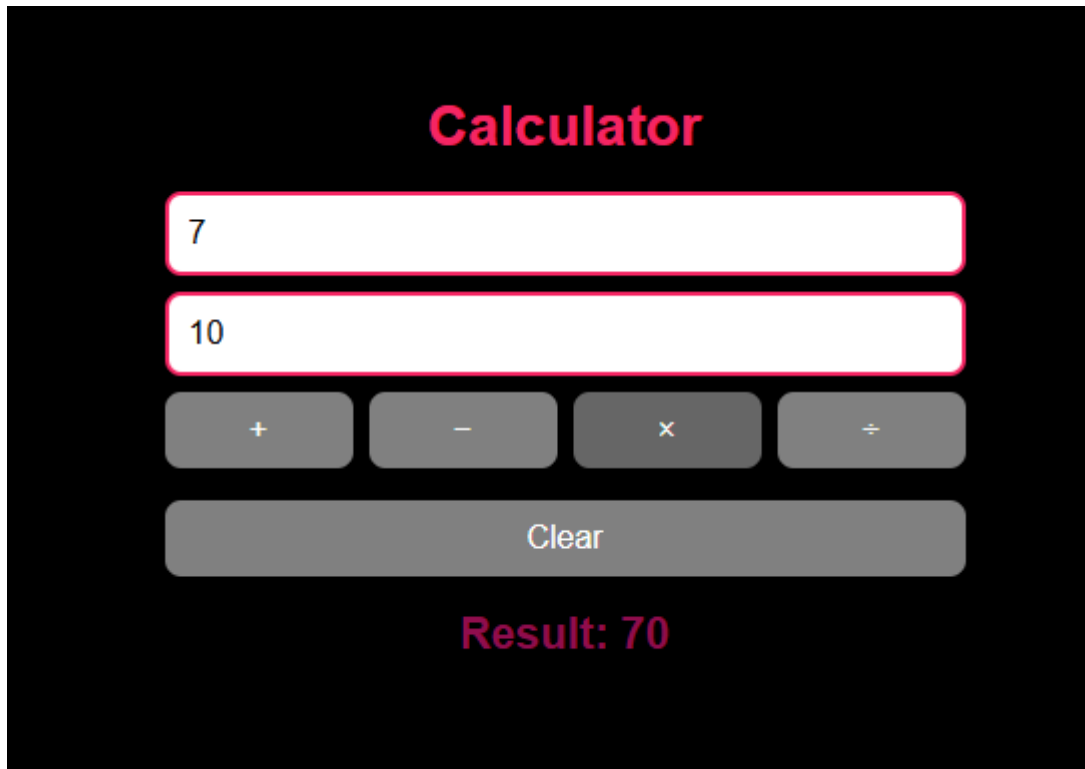
```
        android:id="@+id/multiply"
        android:text="✖"
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:backgroundTint="#F8BBD0"
        android:layout_marginStart="8dp"/>

        <Button
            android:id="@+id/divide"
            android:text="÷"
            android:layout_width="0dp"
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:backgroundTint="#F8BBD0"
            android:layout_marginStart="8dp"/>
    </LinearLayout>

    <Button
        android:id="@+id/clear"
        android:text=" Clear"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:backgroundTint="#CE93D8"
        android:layout_marginBottom="16dp" />

    <TextView
        android:id="@+id/result"
        android:textSize="22sp"
        android:textStyle="bold"
        android:textColor="#880E4F"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

OUTPUT:



RESULT

Once you run the app:

- You can enter two numbers.
- Tap any operation: **+** **-** **×** **÷** **□** Result appears below in bold.
- Clear button resets the input.
- If input is missing or invalid, you'll see a toast message.