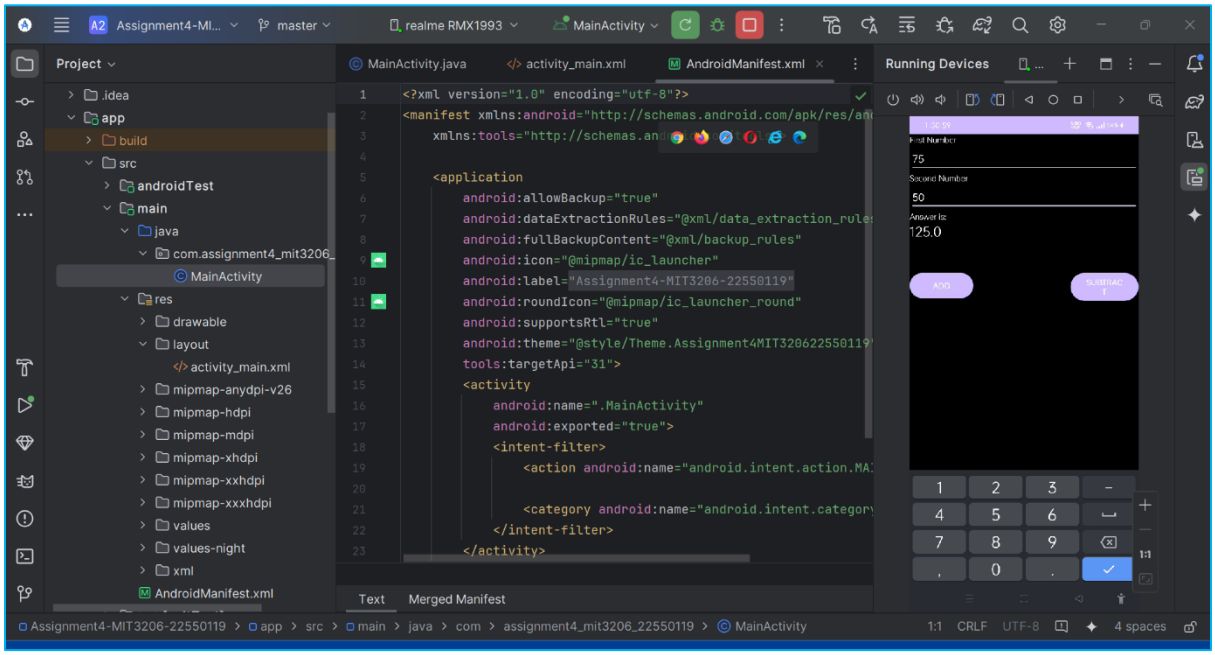
- **Course Module : MIT3206 – Mobile Computing**
- **Course Lecturer : Senior Lecturer Gihan P. Seneviratne Sir**

- **Assignment 4 : Calculator with Addition and Subtraction Functionality**
- **Used Android Studio : Android Studio Koala | 2024.1.1**
- **GitHub Private Repository Link : https://github.com/NavinduMadusanka/Assignment4-MIT3206-22550119.git**

- **Student Name : Kumarage Navindu Madusanka Dias (K.N.M. Dias)**
- **Student Index No : 22550119**
- **Student Registration No : 2022/MIT/011**
- **Email Address : navindu09@gmail.com**
- **Contact No : +94702678624**



Assignment4-MIT3206-22550119

Assignment 4 : Calculator with Addition and Subtraction Functionality

Below is a summary of what I have learned and focused on in this assignment.

1. **Android Components**

   - **API level**

Android Version	API Level	Version Name
Android 7.0	24	Nougat

   - **Classes**

The AppCompatActivity class is used in this assignment.

This is a compatibility library that back ports some features of recent versions of Android to older devices.

   - **Methods**

setOnClickListener

One of the most usable methods in android is setOnClickListener method which helps us to link a listener with certain attributes.

setOnClickListener is a method in Android basically used with buttons, image buttons etc.

We can initiate this method easily like,

public void setOnClickListener(View.OnClickListner).

   - **Permissions**

No special permissions have been granted for this project.

- **Newly learned key points in this assignment**
  - ✓ **Get Value from the EditText and Set value to the TextView**

    TextView result;

    EditText number1,number2;

  - ✓ **Defined the button action**

    Button add,subtract;

  - ✓ **Defined variables & data type**

    float result_num;

    int num1,num2;

  - ✓ **Defined calculations (operation)**

    result_num=num1+num2;

  - ✓ **Getting the result**

    To display stored in sum we have to use setText() as follows :

    result.setText(String.valueOf(result_num));

2. Functionality of the mobile application

2.1 The following interface will appear when the mobile application is activated.

2.2 There is an opportunity to enter two numbers (first number & second number).

2.3 If you want to add the two entered numbers, you have to click on the Add button.

2.4 The first number & second number contain in "number1 & number2" input, assigned two variables num1 and num2.

2.5 These two variables ("number1 and number2") assign "Int" data type.

2.6 These values are added and transferred to a variable called "result_num".

2.7 "result_num" data type is "float".

2.8 The value of this variable (result_num) is shown in the TextView called "result".

2.9 Subtract function is occur in this same way.

3. Layout & Attributes

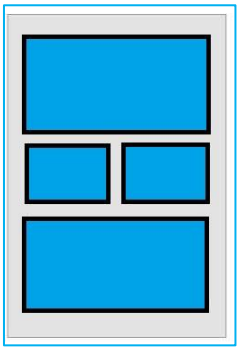
- **RelativeLayout**

It gives us the flexibility to position our component / view based on the relative or sibling component's position.

For the same reason Relative layout is the most used layout after the Linear Layout in Android.

It allows its child view to position relative to each other or relative to the container or another container.

In this assignment I used both Linear and Relative Layouts.



Liner & Relative Layouts

- **Layout Attributes & Other Attributes**

In this assignment I will mention only the special items that I have just learned.

Following are special attributes and applied to the layout in this assignment.

No	Attribute	Description
1	android:ems="10"	android:ems was added in API Level 1. The em is simply the font size. In an element with a 2in font, 1em thus means 2in. Expressing sizes, such as margins and paddings, in em means they are related to the font size, and if the user has a big font or a small font the sizes will be in proportion.
2	android:inputType="number"	Setting the InputType of an EditText tells the system keyboard what kind of input you are expecting. If your edit text is for inputting a phone number, then you want the keyboard to show numbers. It would be annoying for the user to have to manually switch to a numeric keyboard.
3	<Button	Button in linear layout.

4. Running the Application on my android mobile device

I was running the android app for testing in my android mobile device.

My android mobile device is Realme X2 RMX1993.

Below is a photo of my android mobile device (Realme X2 RMX1993) while the app was running.

My android mobile device display setting is set as dark mode option.

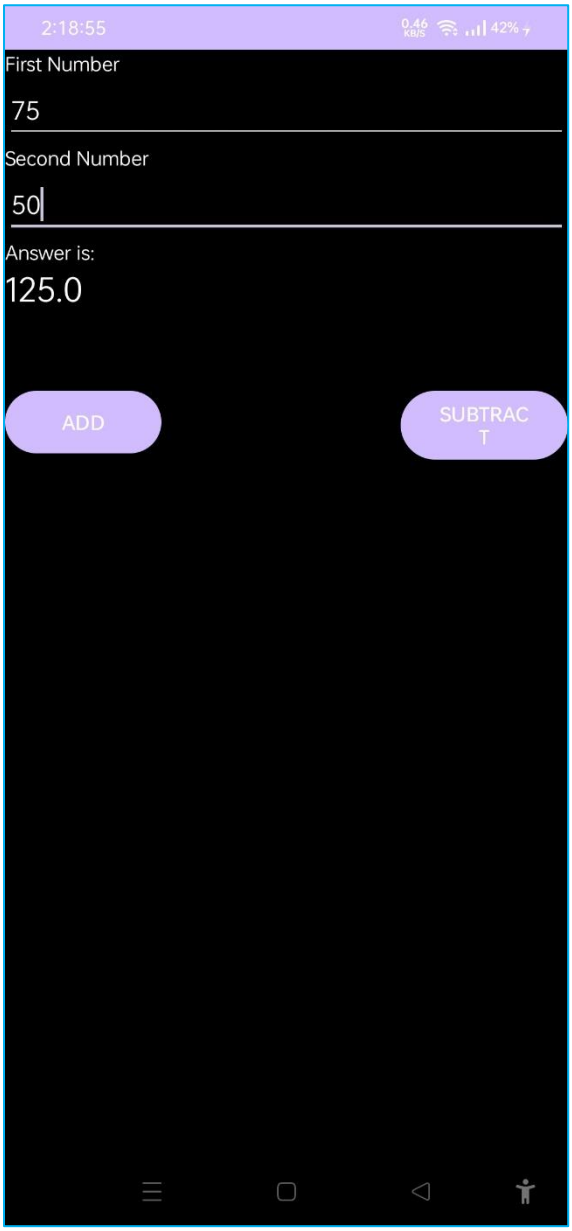


Photo Addition Function : Assignment4-MIT3206-22550119 in Realme X2 RMX1993

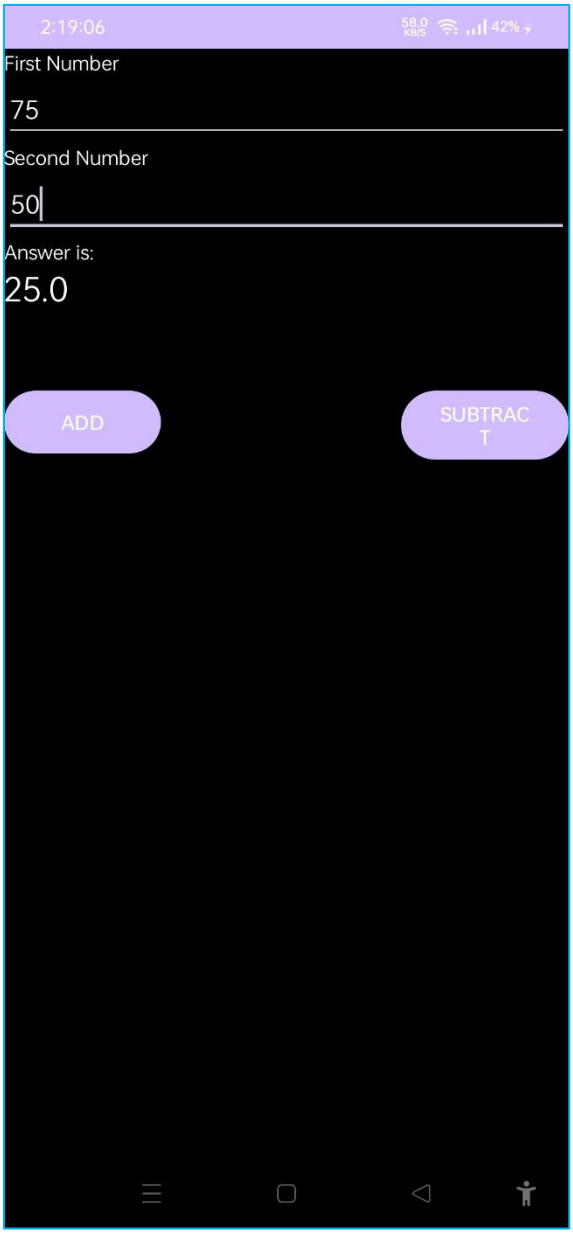


Photo Subtract Function : Assignment4-MIT3206-22550119 in Realme X2 RMX1993

5. Main Coding files

- **MainActivity.java**

```
package com.assignment4_mit3206_22550119;

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;

public class MainActivity extends AppCompatActivity {
    TextView result;
    EditText number1,number2;
    Button add,subtract;

    float result_num;
    int num1,num2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        result=(TextView) findViewById(R.id.result);
        number1=(EditText) findViewById(R.id.number1);
        number2=(EditText) findViewById(R.id.number2);
        add=(Button) findViewById(R.id.add);
        subtract=(Button) findViewById(R.id.subtract);

        add.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v){
                num1=Integer.parseInt(number1.getText().toString());
                num2=Integer.parseInt(number2.getText().toString());
                result_num=num1+num2;
                result.setText(String.valueOf(result_num));
            }
        });

        subtract.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v){
                num1=Integer.parseInt(number1.getText().toString());
                num2=Integer.parseInt(number2.getText().toString());
                result_num=num1-num2;
                result.setText(String.valueOf(result_num));
            }
        });
    }
}
```

- **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:background="@android:color/background_dark"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/n1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="First Number"
        android:textColor="@android:color/background_light" />

    <EditText
        android:id="@+id/number1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number"
        android:textColor="@android:color/background_light" />

    <TextView
        android:id="@+id/n2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Second Number"
        android:textColor="@android:color/background_light" />

    <EditText
        android:id="@+id/number2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number"
        android:textColor="@android:color/background_light" />

    <TextView
        android:id="@+id/Answeris"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Answer is:"
        android:textColor="@android:color/background_light" />

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="76dp"
        android:textColor="@android:color/background_light"
        android:textSize="24sp" />

    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
```

```
        <Button
            android:id="@+id/add"
            android:layout_width="100dp"
            android:layout_height="wrap_content"
            android:layout_alignParentLeft="true"
            android:text="ADD"
            android:textColor="@android:color/background_light" />

        <Button
            android:id="@+id/subtract"
            android:layout_width="107dp"
            android:layout_height="wrap_content"
            android:layout_alignParentRight="true"
            android:text="SUBTRACT"
            android:textColor="@android:color/background_light" />

    </RelativeLayout>

</LinearLayout>
```