

03 - Simple Calculator

Ex. No. : 3

Roll No. :

Date :

Reg. No. :

Aim

Develop a native Calculator Application.

Objective

In this application we are going to start a new project Calculator.

First of all, to start with we are going to create a simple calculator to do simple arithmetic operations (i.e., one operator and two operands).

Procedure

Start Android Studio.

Create a new Project.

Application Name: 'Calculator'.

First of all insert a Linear Layout (Vertical) inside the Relative Layout.

Delete the Hello World.

Next Insert a Text Field inside the Linear Layout (Vertical).

Go to Design Tab and insert Buttons for operators.

Place the components and change the id's for all components according to this table:

Component	Id	Text
TextView	textView	First No.
EditText	etFNo	
TextView	textView2	Second No.
EditText	etSNo	
TextView	textView2	Result
EditText	etRes	

Insert a Linear Layout (Horizontal) inside the Linear Layout (Vertical).

Place the components and change the id's for all components according to this table:

Component	Id	Text
Button	btnAdd	+
Button	btnSub	-
Button	btnMul	*
Button	btnDiv	/

Open MainActivity.java file.

First we need to define some variables for each item in the UI.

```
EditText etFNo, etSNo, etRes;  
Button btnAdd, btnSub, btnMul, btnDiv;
```

Assign the UI elements to these variables using findViewById() inside onCreate().

```
etFNo = (EditText) findViewById(R.id.etFNo);  
etSNo = (EditText) findViewById(R.id.etSNo);  
etRes = (EditText) findViewById(R.id.etRes);  
btnAdd = (Button) findViewById(R.id.btnAdd);  
btnSub = (Button) findViewById(R.id.btnSub);  
btnMul = (Button) findViewById(R.id.btnMul);  
btnDiv = (Button) findViewById(R.id.btnDiv);
```

Register the buttons for onClickListener event.

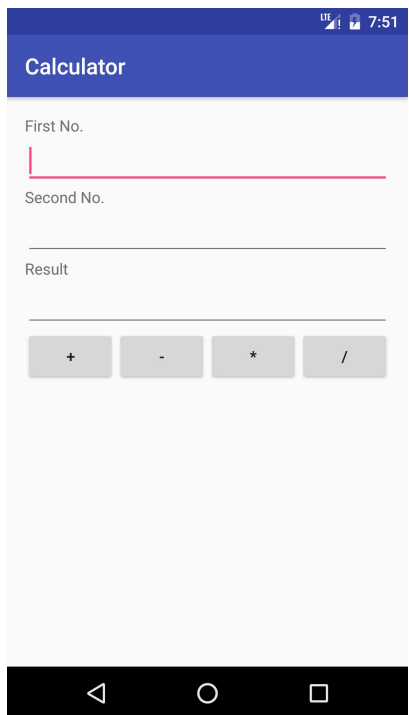
```
btnAdd.setOnClickListener(this);  
btnSub.setOnClickListener(this);  
btnMul.setOnClickListener(this);  
btnDiv.setOnClickListener(this);
```

Override the onClick() method and add the following code:

```
public void onClick(View v) {  
    float fNo = 0, sNo = 0, res = 0;  
    String op = "";  
    fNo = Float.parseFloat(etFNo.getText().toString());  
    sNo = Float.parseFloat(etSNo.getText().toString());  
    switch (v.getId()) {  
        case R.id.btnAdd:  
            res = fNo + sNo;  
            break;  
        case R.id.btnSub:  
            res = fNo - sNo;  
            break;  
        case R.id.btnMul:  
            res = fNo * sNo;
```

```
        break;
    case R.id.btnDiv:
        res = fNo / sNo;
        break;
    }
    etRes.setText("" + res);
}
```

Output



Calculator

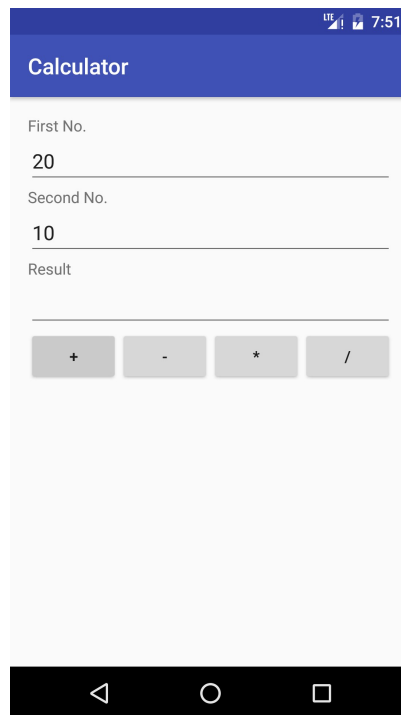
First No.

Second No.

Result

+ - * /

This screenshot shows the initial state of a mobile calculator application. The title bar is blue with the word "Calculator" in white. The status bar at the top shows "LTE", signal strength, and the time "7:51". The main area has three input fields: "First No.", "Second No.", and "Result", each followed by a horizontal line. Below these fields are four buttons: "+", "-", "*", and "/". The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.



Calculator

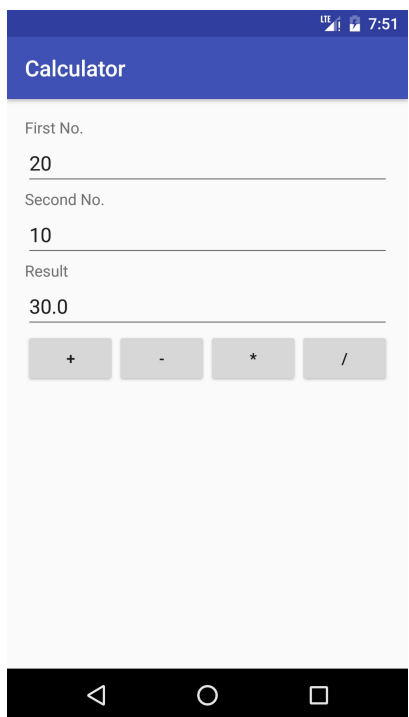
First No.
20

Second No.
10

Result

+ - * /

This screenshot shows the calculator app with the numbers "20" and "10" entered into the "First No." and "Second No." fields respectively. The "Result" field is still empty. The layout and UI elements are consistent with the previous screenshot.



Calculator

First No.
20

Second No.
10

Result
30.0

+ - * /

This screenshot shows the calculator app after a calculation. The "Result" field now displays "30.0". The "First No." field still contains "20" and the "Second No." field still contains "10". The layout and UI elements are consistent with the previous screenshots.