

# **Design and Develop a Scientific Calculator Mobile App**

---

*Project Report submitted in fulfilment of the requirement for  
the degree of*

***Bachelor of Technology***

***By***

***UTKARSH VERMA (10300116021)***

***SOURAV KUMAR RAJAK (10300116035)***

***SAURAV KUMAR (10300116045)***

***SATYAM RAJ (10300116046)***



Department of Computer Science and Engineering

Haldia Institute of Technology

# Certificate

---

It is certified that the work contained in the project report titled "*Design and developed a Scientific Calculator Mobile App*" by Utkarsh Verma, Sourav Kumar Rajak, Saurav Kumar, Satyam Raj has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor: .....

Name: Dr. Subhankar Joardar

Department: Computer Science & Engineering

Haldia Institute of Technology

Month: July

Year: 2020

# Declaration

---

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included. I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not miss represented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the institute and can also evoke panel action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.



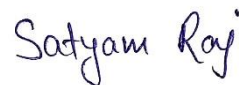
Utkarsh Verma  
(16/CSE/021)



Sourav kumar Rajak  
(16/CSE/035)



Saurav Kumar  
(16/CSE/045)



Satyam Raj  
(16/CSE/046)

Date: 05.07.2020

# Approval Sheet

---

This project report entitled Design and Develop a Scientific Calculator Mobile App is approved for the degree of Bachelor of Technology in Computer Science and Engineering

Examiners

.....  
.....  
.....

Date: .....

Place: .....

# Acknowledgement

---

We, the project team members hereby acknowledge various sources and individuals who have been a major part in the completion of the project. First of all, we are thankful to our project mentor Dr. Subhankar Joardar for helping us in providing support of computer science professionals of our college.

# Abstract

---

The application will be android based replica of specific scientific calculators available in the market. The ability of android based Scientific calculator will be similar to that of calculators.

The android application, will be developed using XML for front end and Java for backend. The application screen will contain the numerical options from 0-9, along with arithmetic operators.

The application will take input from the user consisting the operation, and print the output on the same screen. When the user presses the equals to symbol.

In the upcoming android application will have the capability to perform complex trigonometric and logarithmic computation.

# Table of Content

---

<u>CONTENT</u>	<u>Page no</u>
Introduction	<u>8</u>
Aim & Objective	<u>10</u>
Hardware & Software Requirement	<u>11</u>
Review & Design	<u>12</u>
Basic Operations	<u>13</u>
Algorithm	<u>15</u>
Source Code	<u>16</u>
Screen Shots	<u>43</u>
Advantages	<u>50</u>
Disadvantages	<u>51</u>
Analysis	<u>52</u>
Limitation & Future Scope	<u>53</u>
Conclusion	<u>54</u>
References	<u>55</u>

# Introduction

---

A scientific calculator app is a replica of electronic calculator, usually but not always handheld, designed to calculate problems in science (especially physics), engineering, and mathematics. They have almost completely replaced slide rules in almost all traditional applications, and are widely used in both education and professional settings.

In certain contexts such as higher education, scientific calculators have been superseded by graphing calculators, which offer a superset of scientific calculator functionality along with the ability to graph input data and write and store programs for the device. There is also some overlap with the financial calculator market.

## Functions

Modern scientific calculator app generally has many more features than a standard four or five-function calculator, and the feature set differs developer to developer; however, the defining features of a scientific calculator include:

- scientific notation
- floating point arithmetic
- logarithmic functions, using both base 10 and base e
- trigonometric functions (some including hyperbolic trigonometry)
- exponential functions and roots beyond the square root
- In addition, high-end scientific calculator app will include
- Permutation and combination operations are possible
- Factorials
- fractions
- programmability — see Programmable calculator
- Radian and degree options available

## Uses

Scientific calculator app are used widely in any situation where quick access to certain mathematical functions is needed, especially those such as trigonometric functions that were once traditionally looked up in tables; they are also used in situations requiring back-of-the-envelope calculations of very large numbers, as in some aspects of astronomy, physics, and chemistry.

They are very often required for math classes from the junior high school level through college, and are generally either permitted or required on many standardized tests covering math and science subjects.

## History



The first scientific calculator that included all of the basic features above was the programmable Hewlett-Packard HP-9100A, released in 1968, though the Wang LOCI-2 and the Mathatronics Mathatron had some features later identified with scientific calculator designs. The HP-9100 series was built entirely from discrete transistor logic with no integrated circuits, and was one of the first uses of the CORDIC algorithm for trigonometric computation in a personal computing device, as well as the first calculator based on reverse Polish notation entry. HP became closely identified with RPN calculators from then on, and even today some of their high-end calculators (particularly the long-lived HP-12C financial calculator and the HP-48 series of graphing calculators) still offer RPN as their default input mode due to having garnered a very large following.

The HP-35, introduced on February 1, 1972, was Hewlett-Packard's first pocket calculator and the world's first handheld scientific calculator. Like some of HP's desktop calculators it used reverse Polish notation. Introduced at US\$395, the HP-35 was available from 1972 to 1975. HP continues to develop and market high-end scientific calculators, like the HP-35s and HP-49 series, which have been favoured by scientists and engineers, in labs, offices, as well as in the field.

Texas Instruments, after the introduction of several units with scientific notation, came out with a handheld scientific calculator on January 15, 1974 in the form of the SR-50. TI continues to be a major player in the calculator market, with their long-running TI-30 series being one of the most widely used scientific calculators in classrooms.

# Aim and Objectives

---

A scientific calculator app is a handy and more acceptable replacement for old-fashioned electronic calculator available in the market. With the increase in the bars among the number of smartphones multiplying in the market, the popularity of scientific calculator is increasing exponentially.

The old-fashioned electronic scientific calculator, although being portable limited to specification. The modification if needed cannot be implanted to any of the traditional calculators via any update, to the actually existing circuitry.

Whereas, the scientific calculator app comes in parallel to the currently existing application running over the same platform. And in further addition to this, it also holds the scope for upgradation if needed. Following are the some of the functionality provided by the scientific calculator app:

- scientific notation
- floating-point arithmetic
- logarithmic functions, using both base 10 and base e
- trigonometric functions (some including hyperbolic trigonometry)
- exponential functions and roots beyond the square root
- quick access to constants such as pi and e

## *Aim of the Study*

The basic aim of this research study is to design an effective and accurate scientific calculator application for android platform by implementing advance mathematical approach using java language. In order to print accurate output on the application screen.

## *Objective of the Study*

The following are the objective of this study.

1. To develop or improve existing algorithm to make the calculation accurate and faster.
2. Performance Evaluation of the designed app by comparing the performance with the existing scientific calculator application.
3. Design attractive and user-friendly front

# Hardware and Software Requirements

---

## Hardware

### Microsoft Windows Specific:

- A PC or laptop or tablet with x86 32-or 64-bit processor.
- Microsoft Windows 7/8/10, 32-bit or 64-bit

### Android Specific:

- A tablet or smartphone with ARM-based processor. At least 1.5 GHz clock rate recommended for convenient performance.
- The device should have at least 256 MB of RAM.
- The device should have at least 512 MB of ROM.
- Android 4.1 or newer.

## Software Specification

- Front End: XML
- Back End: Java

# Review of Design

---

Then we began with the design phase of the system. System design is a solution, a “HOW TO” approach to the creation of a new system. It translates system requirements into ways by which they can be made operational. It is a translational from a user-oriented document to a document-oriented programmer. For that, it provides the understanding and procedural details necessary for the implementation. Here we use Flowchart to supplement the working of the new system. The system thus made should be reliable, durable and above all should have least possible maintenance costs. It should overcome all the drawbacks of the Old existing system and most important of all meet the user requirements.

## *Problem*

There are some problems which are faced by Scientific calculator App:

- Does not calculate complex roots of nth degree polynomial
- Few libraries used are outdated and does not support new GUIs.
- Presently it can only be used by Engineering students only.

# Basic Operations

---

## *Addition*

The addition (sum function) is used by clicking on the "+" button . The function results in  $a+b$ .

## *Subtraction*

The subtraction (minus function) is used by clicking on the "-" button. The function results in  $a-b$ .

## *Multiplication*

The multiplication (times function) is used by clicking on the "x" button . The function results in  $a*b$ .

## *Division*

The division (divide function) is used by clicking on the "/" button or using the keyboard. The function results in  $a/b$ .

## *Sign*

The sign key (negative key) is used by clicking on the "(-)" button. The function results in  $1*x$ .

## *Square*

The square function is used by clicking on the " $x^2$ " button. The function results in  $x*x$ .

## *Square Root*

The square root function is used by clicking on the " $\sqrt{\phantom{x}}$ " button.

## *Raise to power*

The raise to the power (y raised to the x function) is used by clicking on the " $y^x$ " button.

## *Natural Exponential*

The natural exponential (e raised to the x) is used by clicking on the " $e^x$ " button. The result is e (2.71828...) raised to x.

## *Logarithm*

The logarithm (LOG) is used by clicking on the "LOG" button.

### *Natural Logarithm*

The Natural logarithm (LN) is used by clicking on the "LN" button.

### *Inverse*

Multiplicative inverse (reciprocal function) is used by pressing the "1/x" button. This function is the same as  $x^{-1}$  or dividing 1 by the number.

### *Exponent*

Numbers with exponents of 10 are displayed with an "e", for example 4.5e+100 or 4.5e-100. This function represents  $10^x$ . Numbers are automatically displayed in the format when the number is too large or too small for the display.

### *Factorial*

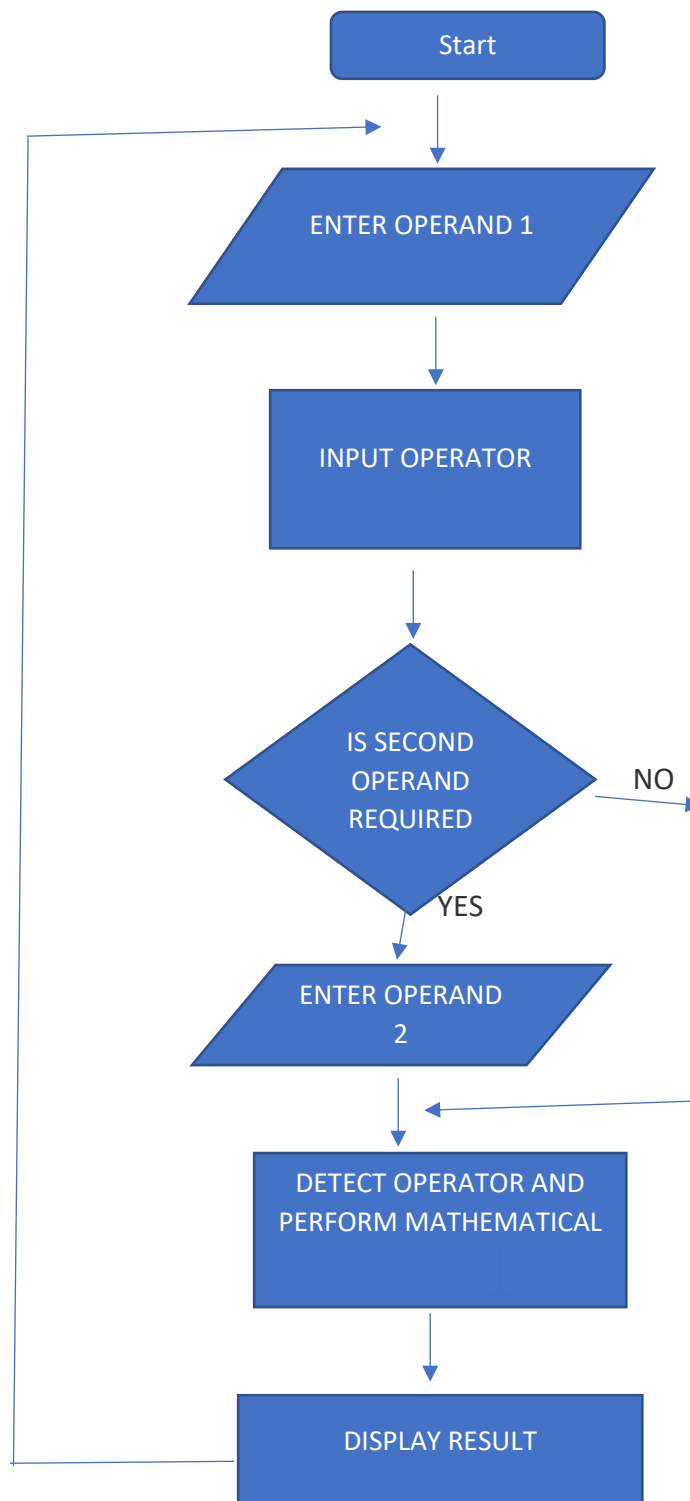
The Factorial function is used by clicking the "!" button or type "!".

### *PI*

PI is a mathematical constant of the ratio of a circle's circumference to its diameter.

# Algorithms

---



# Source Code

---

- *MainActivity.java*

```
package com.finalyearproject.calc;

import android.content.Context;
import android.content.SharedPreferences;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.text.Html;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener{
    private TextView outputResult;
    private TextView shiftDisplay;
    private TextView degreeRad;
    private boolean isDegree = false;
    private boolean isInverse = false;
    private String lastResultObtain = "";
    private String resultObject;
    private String currentDisplayedInput = "";
    private String inputToBeParsed = "";
    private Calculator mCalculator;
    private static String PREFS_NAME = "memory";
    private Button button0, button1,
    button2, button3, button4, button5, button6, button7,
        button8, button9, buttonClear, buttonDivide, buttonMultiply, buttonSubtract,
        buttonAdd, buttonPercentage, buttonEqual, buttonDecimal,
    closeParenthesis, openParenthesis, buttonAnswer,
        buttonSingleDelete, buttonExp;
    private TextView labelFactorial, labelCombination, labelPermutation, labelPi,
    labelE, labelComma, labelCubeRoot, labelCube,
        labelInverseX, labelInverseSin, labelInverseCos, labelInverseTan,
    labelExponential, labelTenPowerX, labelRCL,
        labelSTO, labelMMinus, labelFloat, labelDeg;
    private Button buttonSin, buttonLn, buttonCos, buttonLog, buttonTan,
    buttonSquareRoot, buttonXSquare, buttonYPowerX,
        buttonRnd;
    private Button buttonShift, buttonRad, buttonAbs, buttonMr, buttonMs,
    buttonMPlus;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        requestWindowFeature(Window.FEATURE_NO_TITLE);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
        WindowManager.LayoutParams.FLAG_FULLSCREEN);
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        mCalculator = new Calculator();
        outputResult = (TextView) findViewById(R.id.display);
        outputResult.setText("");
        shiftDisplay = (TextView) findViewById(R.id.shift_display);
        degreeRad = (TextView) findViewById(R.id.degree);
        button0 = (Button) findViewById(R.id.zero_button);
        button1 = (Button) findViewById(R.id.one_button);
        button2 = (Button) findViewById(R.id.two_button);
```



```

button3 = (Button) findViewById(R.id.three_button);
button4 = (Button) findViewById(R.id.four_button);
button5 = (Button) findViewById(R.id.five_button);
button6 = (Button) findViewById(R.id.six_button);
button7 = (Button) findViewById(R.id.seven_button);
button8 = (Button) findViewById(R.id.eight_button);
button9 = (Button) findViewById(R.id.nine_button);
buttonDivide = (Button) findViewById(R.id.division);
buttonMultiply = (Button) findViewById(R.id.multiplication);
buttonSubtract = (Button) findViewById(R.id.subtraction);
buttonAdd = (Button) findViewById(R.id.addition);
buttonPercentage = (Button) findViewById(R.id.percent);
buttonDecimal = (Button) findViewById(R.id.dot);
closeParenthesis = (Button) findViewById(R.id.close_bracket);
openParenthesis = (Button) findViewById(R.id.open_bracket);
buttonExp = (Button) findViewById(R.id.exp);
buttonSquareRoot = (Button) findViewById(R.id.square_root);
buttonXSquare = (Button) findViewById(R.id.x_square);
buttonYPowerX = (Button) findViewById(R.id.x_power_y);
buttonSin = (Button) findViewById(R.id.sin_sign);
buttonCos = (Button) findViewById(R.id.cos_sign);
buttonTan = (Button) findViewById(R.id.tan_sign);
buttonLn = (Button) findViewById(R.id.natural_log);
buttonLog = (Button) findViewById(R.id.log);
buttonRnd = (Button) findViewById(R.id.hys);
buttonDivide.setText(Html.fromHtml(Helpers.division));
buttonSquareRoot.setText(Html.fromHtml(Helpers.squareRoot));
buttonXSquare.setText(Html.fromHtml(Helpers.xSquare));
buttonYPowerX.setText(Html.fromHtml(Helpers.yPowerX));
buttonShift = (Button) findViewById(R.id.shift);
buttonRad = (Button) findViewById(R.id.rad);
buttonAbs = (Button) findViewById(R.id.abs);
buttonMr = (Button) findViewById(R.id.mr);
buttonMs = (Button) findViewById(R.id.ms);
buttonMPlus = (Button) findViewById(R.id.m_plus);
buttonClear = (Button) findViewById(R.id.clear);
buttonSingleDelete = (Button) findViewById(R.id.single_delete);
buttonEqual = (Button) findViewById(R.id.equal_sign);
buttonAnswer = (Button) findViewById(R.id.ans);
labelFactorial = (TextView) findViewById(R.id.factorial);
labelCombination = (TextView) findViewById(R.id.combination);
labelPermutation = (TextView) findViewById(R.id.permutation);
labelPi = (TextView) findViewById(R.id.pi);
labelE = (TextView) findViewById(R.id.e);
labelComma = (TextView) findViewById(R.id.comma);
labelCubeRoot = (TextView) findViewById(R.id.cube_root);
labelCube = (TextView) findViewById(R.id.cube);
labelInverseX = (TextView) findViewById(R.id.one_over_x);
labelInverseSin = (TextView) findViewById(R.id.inverse_sin);
labelInverseCos = (TextView) findViewById(R.id.inverse_cos);
labelInverseTan = (TextView) findViewById(R.id.inverse_tan);
labelExponential = (TextView) findViewById(R.id.expo);
labelTenPowerX = (TextView) findViewById(R.id.ten_power_x);
labelRCL = (TextView) findViewById(R.id.rcl);
labelSTO = (TextView) findViewById(R.id.sto);
labelMMinus = (TextView) findViewById(R.id.m_minus);
labelFloat = (TextView) findViewById(R.id.float_number);
labelDeg = (TextView) findViewById(R.id.degree);
labelInverseSin.setText(Html.fromHtml(Helpers.inverseSin));
labelInverseCos.setText(Html.fromHtml(Helpers.inverseCos));
labelInverseTan.setText(Html.fromHtml(Helpers.inverseTan));
labelExponential.setText(Html.fromHtml(Helpers.exponential));
labelTenPowerX.setText(Html.fromHtml(Helpers.tenPowerX));
labelCubeRoot.setText(Html.fromHtml(Helpers.cubeSquare));
labelCube.setText(Html.fromHtml(Helpers.cubeRoot));
labelPi.setText(Html.fromHtml(Helpers.pi));
button0.setOnClickListener(this);
button1.setOnClickListener(this);

```

```

        button2.setOnClickListener(this);
        button3.setOnClickListener(this);
        button4.setOnClickListener(this);
        button5.setOnClickListener(this);
        button6.setOnClickListener(this);
        button7.setOnClickListener(this);
        button8.setOnClickListener(this);
        button9.setOnClickListener(this);
        buttonClear.setOnClickListener(this);
        buttonDivide.setOnClickListener(this);
        buttonMultiply.setOnClickListener(this);
        buttonSubtract.setOnClickListener(this);
        buttonAdd.setOnClickListener(this);
        buttonPercentage.setOnClickListener(this);
        buttonEqual.setOnClickListener(this);
        buttonAnswer.setOnClickListener(this);
        buttonDecimal.setOnClickListener(this);
        closeParenthesis.setOnClickListener(this);
        openParenthesis.setOnClickListener(this);
        buttonSingleDelete.setOnClickListener(this);
        buttonExp.setOnClickListener(this);
        buttonSquareRoot.setOnClickListener(this);
        buttonXSquare.setOnClickListener(this);
        buttonYPowerX.setOnClickListener(this);
        buttonSin.setOnClickListener(this);
        buttonCos.setOnClickListener(this);
        buttonTan.setOnClickListener(this);
        buttonLn.setOnClickListener(this);
        buttonLog.setOnClickListener(this);
        buttonRnd.setOnClickListener(this);
        buttonShift.setOnClickListener(this);
        buttonRad.setOnClickListener(this);
        buttonAbs.setOnClickListener(this);
        buttonMr.setOnClickListener(this);
        buttonMs.setOnClickListener(this);
        buttonMPlus.setOnClickListener(this);
    }

    private void obtainInputValues(String input){
        switch (input){
            case "0":
                currentDisplayedInput += "0";
                inputToBeParsed += "0";
                break;
            case "1":
                if(isInverse){
                    currentDisplayedInput += "π";
                    inputToBeParsed += "pi";
                }else{
                    currentDisplayedInput += "1";
                    inputToBeParsed += "1";
                }
                toggleInverse();
                toggleShiftButton();
                break;
            case "2":
                if(isInverse){
                    currentDisplayedInput += "e";
                    inputToBeParsed += "e";
                }else{
                    currentDisplayedInput += "2";
                    inputToBeParsed += "2";
                }
                toggleInverse();
                toggleShiftButton();
                break;
            case "3":
                if(isInverse){
                    currentDisplayedInput += ",";

```

```

inputToBeParsed += ",";
} else {
currentDisplayedInput += "3";
inputToBeParsed += "3";
}
toggleInverse();
toggleShiftButton();
break;
case "4":
if(isInverse) {
currentDisplayedInput += "!(";
inputToBeParsed += "!(";
} else {
currentDisplayedInput += "4";
inputToBeParsed += "4";
}
toggleInverse();
toggleShiftButton();
break;
case "5":
if(isInverse) {
currentDisplayedInput += "comb(";
inputToBeParsed += "comb(";
} else {
currentDisplayedInput += "5";
inputToBeParsed += "5";
}
toggleInverse();
toggleShiftButton();
break;
case "6":
if(isInverse) {
currentDisplayedInput += "permu(";
inputToBeParsed += "permu(";
} else {
currentDisplayedInput += "6";
inputToBeParsed += "6";
}
toggleInverse();
toggleShiftButton();
break;
case "7":
currentDisplayedInput += "7";
inputToBeParsed += "7";
break;
case "8":
currentDisplayedInput += "8";
inputToBeParsed += "8";
break;
case "9":
currentDisplayedInput += "9";
inputToBeParsed += "9";
break;
case ".":
currentDisplayedInput += ".";
inputToBeParsed += ".";
break;
case "+":
currentDisplayedInput += "+";
inputToBeParsed += "+";
break;
case "-":
currentDisplayedInput += "-";
inputToBeParsed += "-";
break;
case "÷":
currentDisplayedInput += "÷";
inputToBeParsed += "/";

```

```

break;
case "x":
currentDisplayedInput += "*";
inputToBeParsed += "*";
break;
case "(":
currentDisplayedInput += "(";
inputToBeParsed += "(";
break;
case ")":
currentDisplayedInput += ")";
inputToBeParsed += ")";
break;
case "%":
if(isInverse){
currentDisplayedInput += "1÷";
inputToBeParsed += "1÷";
}else{
currentDisplayedInput += "%";
inputToBeParsed += "%";
}
toggleInverse();
toggleShiftButton();
break;
case "ln":
if(isInverse){
currentDisplayedInput += "e^";
inputToBeParsed += "e^";
}else{
currentDisplayedInput += "ln(";
inputToBeParsed += "ln(";
}
toggleInverse();
toggleShiftButton();
break;
case "log":
if(isInverse){
currentDisplayedInput += "10^";
inputToBeParsed += "10^";
}else{
currentDisplayedInput += "log(";
inputToBeParsed += "log(";
}
toggleInverse();
toggleShiftButton();
break;
case "√":
if(isInverse){
currentDisplayedInput += "3√(";
inputToBeParsed += "crt(";
}else{
currentDisplayedInput += "√(";
inputToBeParsed += "sqrt(";
}
toggleInverse();
toggleShiftButton();
break;
case "Yx":
currentDisplayedInput += "^";
inputToBeParsed += "^";
break;
case "sin":
if(isInverse){
currentDisplayedInput += "asin(";
inputToBeParsed += "asin(";
}else{
currentDisplayedInput += "sin(";
inputToBeParsed += "sin(";
}

```

```

}
toggleInverse();
toggleShiftButton();
break;
case "cos":
if(isInverse){
currentDisplayedInput += "acos(";
inputToBeParsed += "acos(";
}else{
currentDisplayedInput += "cos(";
inputToBeParsed += "cos(";
}
toggleInverse();
toggleShiftButton();
break;
case "tan":
if(isInverse){
currentDisplayedInput += "atan(";
inputToBeParsed += "atan(";
}else{
currentDisplayedInput += "tan(";
inputToBeParsed += "tan(";
}
toggleInverse();
toggleShiftButton();
break;
case "exp":
currentDisplayedInput += "E";
inputToBeParsed += "E0";
break;
case "x2":
if(isInverse){
currentDisplayedInput += "^3";
inputToBeParsed += "^3";
}else{
currentDisplayedInput += "^2";
inputToBeParsed += "^2";
}
toggleInverse();
toggleShiftButton();
break;
case "rnd":
double ran = Math.random();
currentDisplayedInput += String.valueOf(ran);
inputToBeParsed += String.valueOf(ran);
break;
case "ABS":
currentDisplayedInput += "abs(";
inputToBeParsed += "abs(";
break;
case "MR":
String mValue = getStoredPreferenceValue(MainActivity.this);
String result = removeTrailingZero(mValue);
if(!result.equals("0")){
currentDisplayedInput += result;
inputToBeParsed += result;
}
break;
case "MS":
clearMemoryStorage(MainActivity.this);
break;
case "M+":
if (isInverse){
double inputValueMinus = isANumber(outputResult.getText().toString());
if(!Double.isNaN(inputValueMinus)){
subtractMemoryStorage(MainActivity.this, inputValueMinus);
}
}else{

```

```

        double inputValue = isANumber(outputResult.getText().toString());
        if(!Double.isNaN(inputValue)){
            addToMemoryStorage(MainActivity.this, inputValue);
        }
        toggleInverse();
        toggleShiftButton();
        break;
    }
    outputResult.setText(currentDisplayedInput);
}
@Override
public void onClick(View view) {
    Button button = (Button) view;
    String data = button.getText().toString();
    //Toast.makeText(this, "Click " + data, Toast.LENGTH_LONG).show();
    if(data.equals("AC")){
        outputResult.setText("");
        currentDisplayedInput = "";
        inputToBeParsed = "";
    }
    else if(data.equals("Del")){
        String enteredInput = outputResult.getText().toString();
        if(enteredInput.length() > 0){
            enteredInput = enteredInput.substring(0, enteredInput.length() -
1);

            currentDisplayedInput = enteredInput;
            inputToBeParsed = enteredInput;
            outputResult.setText(currentDisplayedInput);
        }
    }else if(data.equals("=")){
        String enteredInput = outputResult.getText().toString();
        // call a function that will return the result of the calculate.
        resultObject = mCalculator.getResult(currentDisplayedInput,
inputToBeParsed);
        outputResult.setText(removeTrailingZero(resultObject));
    }else if(data.equals("Ans")){
        if (resultObject != null) {
            String enteredInput = outputResult.getText().toString();
            enteredInput += resultObject;
            //currentDisplayedInput = enteredInput;
            inputToBeParsed = enteredInput;
            outputResult.setText(removeTrailingZero(enteredInput));
        }else {
            Toast.makeText(MainActivity.this , "No Answer found" ,
Toast.LENGTH_SHORT).show();
        }

    }else if(data.equals("SHIFT")){
        if(!isInverse){
            isInverse = true;
        }else{
            isInverse = false;
        }
        toggleShiftButton();
    }else if(data.equals("RAD")){
        buttonRad.setText("DEG");
        degreeRad.setText("RAD");
    }
    else if(data.equals("DEG")){
        buttonRad.setText("RAD");
        degreeRad.setText("DEG");
    }else{
        obtainInputValues(data);
    }
}

```

```

private String removeTrailingZero(String formattingInput){
    if(!formattingInput.contains(".")){
        return formattingInput;
    }
    int dotPosition = formattingInput.indexOf(".");
    String newValue = formattingInput.substring(dotPosition,
formattingInput.length());
    if(newValue.equals(".0")){
        return formattingInput.substring(0, dotPosition);
    }
    return formattingInput;
}
private void toggleInverse(){
    if(isInverse){
        isInverse = false;
    }
}
private void toggleShiftButton(){
    if(isInverse){
        shiftDisplay.setText("SHIFT");
    }else{
        shiftDisplay.setText("");
    }
}

private double isANumber(String numberInput){
    double result = Double.NaN;
    try{
        result = Double.parseDouble(numberInput);
    }catch(NumberFormatException nfe){
    }

    return result;
}
private void addToMemoryStorage(Context context, double inputToStore){
    float returnPrefValue = getPreference(context);
    float newValue = returnPrefValue + (float)inputToStore;
    setPreference(context, newValue);
}
private void subtractMemoryStorage(Context context, double inputToStore){
    float returnPrefValue = getPreference(context);
    float newValue = returnPrefValue - (float)inputToStore;
    setPreference(context, newValue);
}
private void clearMemoryStorage(Context context){
    setPreference(context, 0);
}
private String getStoredPreferenceValue(Context context){
    float returnedValue = getPreference(context);
    return String.valueOf(returnedValue);
}
static public boolean setPreference(Context c, float value) {
    SharedPreferences settings = c.getSharedPreferences(PREFS_NAME,
0);

    settings = c.getSharedPreferences(PREFS_NAME, 0);
    SharedPreferences.Editor editor = settings.edit();
    editor.putFloat("key", value);
    return editor.commit();
}
static public float getPreference(Context c) {
    SharedPreferences settings = c.getSharedPreferences(PREFS_NAME,
0);

    settings = c.getSharedPreferences(PREFS_NAME, 0);
    float value = settings.getFloat("key", 0);
    return value;
}
}

```

- *Calculator.java*

```
package com.finalyearproject.calc;

import com.fathzer.soft.javaluator.DoubleEvaluator;
import com.fathzer.soft.javaluator.Function;
import com.fathzer.soft.javaluator.Parameters;

import java.util.ArrayList;
import java.util.Iterator;

public class Calculator {
    final Function sqrt = new Function("sqrt", 1);
    final Function factorial = new Function("!", 1);
    final Function cuberoot = new Function("crt", 1);
    final Function combination = new Function("comb", 2);
    final Function permutation = new Function("permu", 2);
    Parameters params;
    DoubleEvaluator evaluator;
    private double previousSum = 0;
    private double currentSum = 0;
    private String currentDisplay = "";
    //private String expressionUsedForParsing = "";
    private boolean isRadians = false;
    public Calculator() {
        addFunctions();
        //Adds the functions to the evaluator
        evaluator = new DoubleEvaluator(params) {
            @Override
            protected Double evaluate(Function function, Iterator arguments, Object
evaluationContext) {
                if (function == sqrt)
                    return Math.sqrt((Double) arguments.next());
                else if (function == cuberoot) {
                    return Math.cbrt((Double) arguments.next());
                }
                else if (function == combination) {
                    double numberInputs = 0;
                    ArrayList<Double> saveValue = new ArrayList<Double>();
                    while (arguments.hasNext()) {
                        numberInputs = (Double) arguments.next();
                        saveValue.add(numberInputs);
                    }
                    double firstArgument = saveValue.get(0);
                    double secondArgument = saveValue.get(1);
                    double denominator = getFactorial((int) firstArgument);
                    double nominator = getFactorial((int) secondArgument) *
(getFactorial((int) (firstArgument - secondArgument)));
                    return denominator / nominator;
                }
                else if (function == permutation) {
                    double numberInputs = 0;
                    ArrayList<Double> saveValue = new ArrayList<Double>();
                    while (arguments.hasNext()) {
                        numberInputs = (Double) arguments.next();
                        saveValue.add(numberInputs);
                    }
                    double firstArgument = saveValue.get(0);
                    double secondArgument = saveValue.get(1);
                    double denominator = getFactorial((int) firstArgument);
                    double nominator = (getFactorial((int) (firstArgument -
secondArgument)));
                    return denominator / nominator;
                }
                else if (function == factorial) {
                    double result = 1;

```



```

        double num = (Double) arguments.next();
        for (int i = 2; i <= num; i++) {
            result = result * i;
        }
        return result;
    } else
        return super.evaluate(function, arguments, evaluationContext);
    }
};
}

private int getFactorial(int n) {
    int result;
    if (n==0 || n==1)
        return 1;
    result = getFactorial(n-1) * n;
    return result;
}

public void addFunctions() {
    params = DoubleEvaluator.getDefaultParameters();
    params.add(sqrt);
    params.add(factorial);
    params.add(cuberoot);
    params.add(combination);
    params.add(permutation);
}

public String getResult(String currentDisplay, String expressionUsedForParsing)
{
    //Tries to parse the information as it is entered, if the parser can't
handle it, the word error is shown on screen
    try {
        System.out.println("Displayed Output " + expressionUsedForParsing);
        currentSum =
evaluator.evaluate(fixExpression(expressionUsedForParsing));
        currentSum = convertToRadians(currentSum);
        currentDisplay = String.valueOf(currentSum);
        //previousSum = currentSum;
    } catch (Exception e) {
        currentDisplay = "Error";
    }
    return currentDisplay;
}

public double convertToRadians(double sum) {
    double newSum = sum;
    if (isRadians == true)
        newSum = Math.toRadians(sum);
    return newSum;
}

//Used to show display to user
public String getCurrentDisplay() {
    return currentDisplay;
}

//Handles fixing the expression before parsing. Adding parens, making sure
parens can multiply with each other,
public String fixExpression(String exp) {
    int openParens = 0;
    int closeParens = 0;
    char openP = '(';
    char closeP = ')';
    String expr = exp;
    for (int i = 0; i < exp.length(); i++) {
        if (exp.charAt(i) == openP)
            openParens++;
        else if (exp.charAt(i) == closeP)
            closeParens++;
    }
    while (openParens > 0) {
        expr += closeP;
        openParens--;
    }
}

```

```

    }
    while (closeParens > 0) {
        expr = openP + expr;
        closeParens--;
    }
    expr = multiplicationForParens(expr);
    return expr;
}
//Used to fix multiplication between parentheses
public String multiplicationForParens(String s) {
    String fixed = "";
    for (int position = 0; position < s.length(); position++) {
        fixed += s.charAt(position);
        if (position == s.length() - 1)
            continue;
        if (s.charAt(position) == ')' && s.charAt(position + 1) == '(')
            fixed += '*';
        if (s.charAt(position) == '(' && s.charAt(position + 1) == ')')
            fixed += '1';
    }
    return fixed;
}
}

```

- *Helpers.java*

```

package com.finalyearproject.calc;

import android.content.Context;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.Toast;

public class Helpers {
    public static String division = "&divide;";
    public static String inverseSin = "sin<sup>-1</sup>";
    public static String inverseCos = "cos<sup>-1</sup>";
    public static String inverseTan = "tan<sup>-1</sup>";
    public static String exponential = "e<sup>x</sup>";
    public static String tenPowerX = "10<sup>x</sup>";
    public static String cubeSquare = "3&radic;";
    public static String cubeRoot = "x<sup>3</sup>";
    public static String yPowerX = "Y<sup>x</sup>";
    public static String squareRoot = "&radic;";
    public static String xSquare = "x<sup>2</sup>";
    public static String pi = "&pi;";
    public static void displayErrorMessage(Context context){
        Toast.makeText(context, "Input field must not be zero",
        Toast.LENGTH_LONG).show();
    }
    public static boolean isZero(EditText input){
        if(Double.parseDouble(input.getText().toString()) == 0){
            return true;
        }
        return false;
    }
    public static int getTopicId(Bundle bundle, String inputValue){
        int id = 0;
        if(bundle != null){
            id = bundle.getInt(inputValue);
        }
        return id;
    }
}

```

- *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:id="@+id/linearLay0"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@color/md_blue_grey_400"
    tools:context="com.finalyearproject.calc.MainActivity">

    <LinearLayout
        android:id="@+id/top_label"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:textAlignment="center">

        <TextView
            android:id="@+id/shift_display"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp"
            android:textColor="@color/divider" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp"
            android:textColor="@color/divider" />

        <TextView
            android:id="@+id/float_number"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp"
            android:text="@string/no_text"
            android:textColor="@color/divider" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp" />

        <TextView
            android:id="@+id/degree"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:padding="8dp"
            android:text="DEG"
            android:textAlignment="center"
            android:textColor="@color/md_white_1000" />
    </LinearLayout>
</LinearLayout>
```

```

        android:id="@+id/display_screen"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/btn_bg">
        <TextView
            android:id="@+id/display"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="right"
            android:layout_gravity="right"
            android:cursorVisible="true"
            android:textColorHint="@color/icons"
            android:paddingLeft="10dp"
            android:paddingRight="10dp"
            android:singleLine="true"
            android:text="ln2"
            android:textColor="@color/primary_text"
            android:textSize="60sp" />
    </LinearLayout>
    <!--First Row-->
    <!--Second Row Button-->
    <LinearLayout
        android:id="@+id/first_row"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:textAlignment="center">

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="center"
            android:padding="4dp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="center"
            android:padding="4dp" />

        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="center"
            android:padding="4dp"
            android:textColor="@color/divider" />

        <TextView
            android:id="@+id/rc1"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="center"
            android:padding="4dp"
            android:text="@string/no_text"
            android:textColor="@color/divider" />

        <TextView
            android:id="@+id/sto"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:gravity="center"
            android:padding="4dp"
            android:text="@string/no_text"

```

```

        android:textColor="@color/divider" />

<TextView
    android:id="@+id/m_minus"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/m_minus"
    android:textColor="@color/divider"/>
</LinearLayout>

<LinearLayout
    android:id="@+id/second_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/shift"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/shift"
        android:textColor="@color/calculator_color"
        android:textSize="14sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/rad"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/rad"
        android:textColor="@color/calculator_color"
        android:textSize="14sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/abs"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/abs"
        android:textColor="@color/calculator_color"
        android:textSize="14sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/mr"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"

```

```

        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/mr"
        android:textColor="@color/calculator_color"
        android:textSize="14sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

<Button
    android:id="@+id/ms"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/ms"
    android:textColor="@color/calculator_color"
    android:textSize="14sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/m_plus"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/m_plus"
    android:textColor="@color/calculator_color"
    android:textSize="14sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />
</LinearLayout>
<!--Third Row Text-->
<LinearLayout
    android:id="@+id/third_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:textAlignment="center"
    android:weightSum="6">

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:textColor="@color/divider" />

    <TextView
        android:id="@+id/inverse_sin"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/inverse_sin"
        android:textColor="@color/divider"
        android:textSize="13dp" />

    <TextView
        android:id="@+id/inverse_cos"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"

```

```

        android:gravity="center"
        android:padding="4dp"
        android:text="@string/inverse_cos"
        android:textColor="@color/divider"
        android:textSize="13dp" />

<TextView
    android:id="@+id/inverse_tan"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/inverse_tan"
    android:textColor="@color/divider"
    android:textSize="13dp" />

<TextView
    android:id="@+id/expo"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/expo"
    android:textColor="@color/divider"
    android:textSize="13dp" />

<TextView
    android:id="@+id/ten_power_x"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/ten_power"
    android:textColor="@color/divider"
    android:textSize="13dp" />
</LinearLayout>
<!--Fourth Row Button-->
<LinearLayout
    android:id="@+id/fourth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/hys"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/rnd"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/sin_sign"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"

```

```

        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/sin_sign"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

<Button
    android:id="@+id/cos_sign"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/cos_sign"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/tan_sign"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/tan_sign"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/natural_log"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/natural_log"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/log"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/log"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />
</LinearLayout>

```



```

        <!--Fifth Row Text-->
<LinearLayout
    android:id="@+id/fifth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:textAlignment="center"
    android:weightSum="6">

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:textColor="@color/divider" />

    <TextView
        android:id="@+id/cube_root"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/cube_root"
        android:textColor="@color/divider"
        android:textSize="13dp" />

    <TextView
        android:id="@+id/cube"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/cube"
        android:textColor="@color/divider"
        android:textSize="13dp" />

    <TextView
        android:id="@+id/one_over_x"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/one_over_x"
        android:textColor="@color/divider"
        android:textSize="13dp" />

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text=""
        android:textColor="@color/divider" />

    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text=""
        android:textColor="@color/divider" />
</LinearLayout>

```

```

        <!--Sixth Row Button-->
<LinearLayout
    android:id="@+id/sixth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/x_power_y"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/x_power_y"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/square_root"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/square_root"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/x_square"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/x_square"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/percent"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/percent"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="16sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

```

```

<Button
    android:id="@+id/open_bracket"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/open_bracket"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/close_bracket"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/close_bracket"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="16sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />
</LinearLayout>
    <!--Seventh Row 5 Buttons-->
<LinearLayout
    android:id="@+id/seventh_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:layout_weight="1"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/seven_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/seven_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/eight_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/eight_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

```

```

<Button
    android:id="@+id/nine_button"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/nine_button"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/single_delete"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/del"
    android:padding="8dp"
    android:text="Del"
    android:textAllCaps="false"
    android:textColor="#dfd4d4"
    android:textSize="20sp"
    android:textStyle="bold"/>

<Button
    android:id="@+id/clear"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/del"
    android:padding="8dp"
    android:text="@string/clear"
    android:textAllCaps="false"
    android:textColor="#dfd4d4"
    android:textSize="20sp"
    android:textStyle="bold"/>
</LinearLayout>
<!--Eighth Row 5 Text-->
<LinearLayout
    android:id="@+id/eighth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:textAlignment="center">

    <TextView
        android:id="@+id/factorial"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/factorial"
        android:textColor="@color/divider"
        android:textSize="13dp" />

    <TextView
        android:id="@+id/combination"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"

```

```

        android:text="@string/combination"
        android:textColor="@color/divider"
        android:textSize="13dp" />

<TextView
    android:id="@+id/permutation"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/permutation"
    android:textColor="@color/divider"
    android:textSize="13dp" />

<TextView
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text=""
    android:textColor="@color/divider" />

<TextView
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text=""
    android:textColor="@color/divider" />
</LinearLayout>
<!--Nine Row 5 Buttons-->
<LinearLayout
    android:id="@+id/ninth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:textAlignment="center"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/four_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/four_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/five_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/five_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"

```

```

        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

<Button
    android:id="@+id/six_button"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/six_button"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/multiplication"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/multiplication"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/division"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/division"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />
</LinearLayout>
<!--Ten Row 5 Text-->
<LinearLayout
    android:id="@+id/tenth_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:textAlignment="center">

    <TextView
        android:id="@+id/pi"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/pi"
        android:textColor="@color/divider" />

    <TextView

```

```

        android:id="@+id/e"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:padding="4dp"
        android:text="@string/e"
        android:textColor="@color/divider" />

<TextView
    android:id="@+id/comma"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text="@string/comma"
    android:textColor="@color/divider" />

<TextView
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text=""
    android:textColor="@color/divider" />

<TextView
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:gravity="center"
    android:padding="4dp"
    android:text=""
    android:textColor="@color/divider" />
</LinearLayout>
<!--Eleven Row 5 Buttons-->
<LinearLayout
    android:id="@+id/Eleven_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:textAlignment="center"
    android:theme="@style/MyButton">

    <Button
        android:id="@+id/one_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/one_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

    <Button
        android:id="@+id/two_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"
        android:background="@drawable/cicle_bg"

```



```

        android:padding="8dp"
        android:text="@string/two_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

<Button
    android:id="@+id/three_button"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/three_button"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/addition"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/addition"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/subtraction"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/subtraction"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />
</LinearLayout>
<!--Twelve Row 5 Buttons-->
<LinearLayout
    android:id="@+id/twelve_row"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:layout_weight="1"
    android:textAlignment="center">

    <Button
        android:id="@+id/zero_button"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_marginRight="1dp"
        android:layout_weight="1"

```



```

        android:background="@drawable/cicle_bg"
        android:padding="8dp"
        android:text="@string/zero_button"
        android:textAllCaps="false"
        android:textColor="@color/calculator_color"
        android:textSize="20sp"
        android:textStyle="bold"
        android:theme="@style/MyButton" />

<Button
    android:id="@+id/dot"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/dot"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/exp"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/exp"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/ans"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/ans"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

<Button
    android:id="@+id/equal_sign"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_marginRight="1dp"
    android:layout_weight="1"
    android:background="@drawable/cicle_bg"
    android:padding="8dp"
    android:text="@string/equal_sign"
    android:textAllCaps="false"
    android:textColor="@color/calculator_color"
    android:textSize="20sp"
    android:textStyle="bold"
    android:theme="@style/MyButton" />

```

```
</LinearLayout>  
</LinearLayout>
```

# Screen shots

---

## Addition



## Subtraction



## Multiplication



## Division



***Sin(radian)***



***Tan(radian)***



## Logarithmic



## Factorial





### Combination



### Permutation



## Natural Logarithm



## Sign





### ***Raise to the power***



### ***Square Root***



# Advantages

---

- *Faster computation of complex problems*

Scientific calculator can compute large values in a second within a flash. The android application supports the functionality to operate on multiple operator at the same time, using the mathematical definition provided.

- *Accurate computation*

Scientific calculator produces accurate results for complex calculation, operating on multiple operator at the same time. Using the calculator app ensure error free results that cannot be guaranteed by counting manually

- *Easier accessibility*

The scientific calculator mobile application comes to be very handy when it comes to accessibility, as it is readily available along with multiple application in parallel

- *Unit conversions*

Sometime unit conversion can be very painful. As the defined measured value is needed to be recalled. But when it comes to Scientific calculator this turns out to be very handy. As the formulas and defined are values are hard coded while manufacturing the application. And thus, making the conversion easier and simpler.

- *Reduces time and Effort*

Calculators were developed with the prime motives to reduce efforts. And the presence of Scientific calculators is an addition to the comfort and effort. And even the scientific calculator app provides the exact functionality provided by the specific calculators.

- *Minimizes mathematical error*

The chances of error are found to be maximum in case of manual computation. And when it comes to complex calculations the probability of error increases exponentially. Scientific calculator takes the mathematical definition into consideration while computation and minimizes the chances of error.

# Disadvantages

---

- *Dependency increases*

With the maximization in the need of the calculator the dependency increases for the urge of calculator. Even for the simple calculations the user approaches for the calculator without giving human capabilities a single chance

- *Approach for the result becomes irrelevant*

The motive to get the result become the prime objective. The approach to the problem becomes out of question. As a result, basic skill is hampered.

- *Limits the Knowledge of the user*

The user becomes result centric. The method of using calculators never encourages the user to know the roots of the problem. And the generalized and accurate approach to conclude an answer.

- *Basic learning skills are hampered*

Mathematics is second derivate of practice. And the use of calculator eliminates the scope of practice mathematics mentally or manually. In long run, even for basic calculation the user needs a calculator, in order to get the output.

# Analysis

---

Android based Scientific calculator is probably the best alternative for the specific scientific calculators available in the market. The scientific calculator comes in parallel with the other android applications and cannot be coined as specific for a device. With the functionality provided by scientific calculators are almost similar to specific scientific calculators. And that can also be called the reason for its popularity in the recent days.

The two significant aspects of popularity of Android based scientific calculator application are:

1. Fast and accurate calculation
2. Portability

## *Fast and Accurate Calculation*

The scientific calculator can perform all the functionality offered by the specific calculator, designed only to perform calculations. The scientific calculator app, takes the input from the user and compute it according to the mathematical definition, producing the accurate result. The other reason for its popularity is its speed. With the high processing processor android based scientific calculator produces result within a flash.

## *Portability*

The specific calculator design to perform mathematical calculation are hard core circuitry, which is design only to perform numerical operations. Whereas, the Android based scientific calculator is the in parallel application, i.e., scientific calculator is one the application among various applications available in android device. The parallel presence multiplies its popularity.

# Limitation and Future scope

---

## *Limitations*

- History of the calculation performed is not available
- Unable to perform operations related to calculus
- Unable to compute equations directly.

## *Future Scope*

- Will have the ability to perform more complex operations.
- Will be made available on play store, for educational purposes.
- Will always be free.

# Conclusion

---

Android based Scientific calculator is the fastest growing replacement for the specific calculators available in the market. The scientific calculator app, with its capabilities to perform its functionality in presence of the other apps over the same device, is the main reason for its over growing popularity.

Android based Scientific calculator has the capability to compute the complex calculations as per mathematical definition. And producing the output accurate and fast. It shares the functionality performed by the specific calculators. And hold the capability of upgradation over the same circuitry via online/offline updates.

With the ease to access to complex calculation, it also reduces the capability of the user over problem solving abilities. And increases its dependency on calculator for performing basic computation.

Neither it provides the scope for learning. The user becomes result centric without expanding its ideas about the flow to solve the problems.

# References

---

- “Scientific calculator”  
[https://en.wikipedia.org/wiki/Scientific\\_calculator](https://en.wikipedia.org/wiki/Scientific_calculator)
- “Algorithms for trigonometric implementation”  
<https://www.scribd.com/document/34526338>
- “Advantages and Disadvantages of scientific calculator”  
<https://www.scribd.com/document/345263938/Advantages-and-Disadvantages-of-Using-Calculator>
- “Objective for scientific calculators”  
<https://www.tomsguide.com/us/pictures-story/645-best-calculator-apps.html>
- “Algorithm and hint for Scientific Calculator”  
<https://stackoverflow.com/questions/11260746/algorithm-and-hint-for-scientific-calculator>
- “What algorithm do scientific calculators use to calculate logarithms”  
<https://math.stackexchange.com/questions/2873604/what-algorithm-do-scientific-calculators-use-to-calculate-logarithms>
- “Java mini calculators”  
[https://www.academia.edu/8313886/Java\\_Mini\\_project\\_-\\_Calculator](https://www.academia.edu/8313886/Java_Mini_project_-_Calculator)