# EXPERIMENT NO. 7

Aim :- Write an Android application to implement Basic Calculator using Android.

## Theory :-

The practical aims to create a user-friendly calculator application using Android Studio, leveraging Java for logic implementation and XML for interface design. The app will feature numeric buttons (0-9), operation buttons (+, -, *, /), a clear button (AC), and an equal button (=) to perform basic arithmetic operations. The primary objectives include designing an intuitive interface, accurate calculations with proper error handling.

## XML Layout Design :

The app's XML layout represents the UI design for a basic calculator Android app using ConstraintLayout in Android Studio. The layout consists of a TextView at top to display results and several rows of buttons organized in LinearLayouts. Each row represents a set of numeric and operator buttons. The form includes 'res/layout/activity_main.xml' which essentially defines the visual structure of the app. It also uses two drawable files for giving styles to the buttons, 'button_bg.xml' for all buttons, and 'yellow_button_bg.xml' for especially a yellow button.

The root layout used in this XML file is 'ConstraintLayout'. The TextView is used to display the result of calculations. Attributes such as 'android:textSize="48sp"' determines the font size, and 'app:layout_constraintTop_toTopOf="parent"' ensures it is constrained to top of parent layout. LinearLayouts are used to organize buttons into rows. AppCompactButtons elements represent buttons for numerics and operators. Buttons also have styles applied to them using 'style="@style/CalculatorButtonStyle"' to maintain a consistent appearance throughout the application.

Java Programming Logic :-

Firstly, the most important file (hub of the program) is 'src/main/java/com.example.basiccalculator/MainActivity.java'. To connect with particulair XML, the 'onCreate' methods are initialized.

Built-in Functions :

1) findViewById():

This method is used to retrieve UI components from layout XML file by their respective IDs. It is a built-in method provided by Android framework.

2> setOnClickListener():

This method is used to set click event listeners on buttons, enabling them to respond to user interactions.

3> Integer. parseInt():

This built-in method converts a string representation of a number to it's integer equivalent. It is used to extract numeric value from button text when a digit button is clicked.

4> String. valueOf():

This method converts various types of values, including integers, to their string representation.

5> append():

This method is part of TextView class and is used to append text to existing text in TextView. It is used to dynamically update the displayed expression as the user inputs digits and operators.

6> setText():

This method is used to set the text content of a TextView. It is used to display error messages, clear the TextView, or display the final result.

7) Logical operators (&&):

These built-in operators are used for the logical AND operators. They are used in conditional statements to evaluate the multiple conditions together.

These built-in functions and methods which are provided by the Android SDK and Java language are essential for implementing the core functionality of the calculator application, including user interaction handling, data manipulation, and error checking.

Conclusion :-

Thus, we developed a basic calculator by understanding and implementing different built-in functions and XML Layout functions and attributes provided by the Android Studio for effective development.