

# GDSC App Development Induction Task Report

## Calculator Application

This report provides a comprehensive breakdown of the **Calculator Application** built using **Jetpack Compose** for the **GDSC App Development Induction Task**. Every component and its functionality is documented in detail.



by Devanshu Dubey

240001024

cse240001024@gmail.com

## Project Overview

The calculator application provides two levels of functionality:

- Level 1: Basic Calculator** – Handles basic arithmetic operations.
- Level 2: Scientific Calculator** – Supports advanced calculations including trigonometry, exponents, and factorials.

The user interface is interactive and intuitive, allowing seamless navigation between basic and scientific modes.

# Tech Stack

Component	Technology Used
Language	Kotlin
Framework	Jetpack Compose
Environment	Android Studio

## Features Overview

### Level 1: Basic Calculator

- Addition (+)
- Subtraction (-)
- Multiplication (\*)
- Division (/)

### Level 2: Scientific Calculator

- Trigonometric Functions:  $\sin()$ ,  $\cos()$ ,  $\tan()$  (Supports Radians/Degrees)
- Logarithmic Functions:  $\log()$ ,  $\ln()$
- Power and Root Calculations:  $x^y$ ,  $\sqrt{x}$
- Factorial Calculation:  $n!$

# Application Structure

## 1. Main Activity

The app starts with MainActivity, which sets the content to the CalculatorUI() function. This function structures the entire user interface.

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContent {
            CalculatorTheme {
                Scaffold(
                    modifier = Modifier.fillMaxSize()
                ) { innerPadding ->
                    CalculatorUI(modifier = Modifier.padding(innerPadding))
                }
            }
        }
    }
}
```

## 2. User Interface (UI)

- **CalculatorUI():** Organizes the calculator layout with:
  - **Display Box:** Displays the current input and calculated output.
  - **Button Grid:** Contains all calculator buttons.

## 3. Calculator Buttons

- **CalculatorButton():** A reusable Composable function for creating buttons.

Parameters:

- **label:** Text displayed on the button.
- **onClick:** Action performed when the button is pressed.

```
@Composable
fun CalculatorButton(label: String, onClick: () -> Unit) {
    ...
}
```

# Expression Handling & Calculation

## 1. Input Management

- `newInput()`: Ensures valid user inputs by:
  - Preventing multiple decimals.
  - Auto-inserting multiplication for constants ( $\pi$ ,  $e$ ).
  - Managing parentheses balance.

## 2. Expression Evaluation

- `calculateResult()`: Initiates expression parsing and returns the evaluated output.
- `evaluateExpression()`: Core logic for expression parsing and computation.

## 3. Function Handling

- `handleFunctions()`: Evaluates trigonometric and logarithmic expressions.
- `handleFactorial()`: Computes factorial for positive integers.
- `handlePower()`: Manages exponentiation operations.
- `handleSquareRoot()`: Computes the square root of a number.

## 4. Error Handling

Errors are gracefully handled to prevent app crashes. Invalid operations return an Error message.

# Utility Functions

Function	Purpose
<code>findClosingParenthesis()</code>	Ensures balanced parentheses handling.
<code>formatResult()</code>	Formats output for better readability.
<code>degreesToRadians()</code>	Converts degrees to radians.

# Demo Video

[Watch Demo Video]([https://youtu.be/7QmbgnLvMpk?si=IB2\\_godHfjyl9rng](https://youtu.be/7QmbgnLvMpk?si=IB2_godHfjyl9rng))

# Installation Instructions

Download and install the latest release from GitHub:

[https://github.com/DevanshuDubey/GDSC\\_APPDEV\\_INDUCTION\\_TASK/releases/latest](https://github.com/DevanshuDubey/GDSC_APPDEV_INDUCTION_TASK/releases/latest)

Or click below to download latest apk:

[Click here](#)

**Thank you for evaluating My GDSC App  
Development Induction Task!  
Hope You Liked My Submission.**