**Minor Project 2- Simple Two-Number Calculator in Jetpack Compose**

**Objective:**

**Lab Exercise: Simple Two-Number Calculator in Jetpack Compose**

**Objective:**

Create a basic calculator in Jetpack Compose that performs addition, subtraction, multiplication, and division operations between two numbers. This exercise will help you understand how to manage state and handle user input in Compose.

**Steps:**

**Step 1: Set Up Your Project**

1. **Open Android Studio** and create a new project:
   * Select **New Project** -> **Empty Compose Activity**.
   * Name your project SimpleCalculator.
2. **Ensure Compose dependencies are included** in your build.gradle:

**Step 2: Create the Calculator UI**

1. **Define the Composable Function for the Calculator**:

Open MainActivity.kt and define a composable function that creates a simple calculator.

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.layout.\*

import androidx.compose.material3.\*

import androidx.compose.runtime.\*

import androidx.compose.ui.Modifier

import androidx.compose.ui.tooling.preview.Preview

import androidx.compose.ui.unit.dp

import com.example.simplecalculator.ui.theme.SimpleCalculatorTheme

class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContent {

SimpleCalculatorTheme {

// A surface container using the background color from the theme

Surface(modifier = Modifier.fillMaxSize(), color = MaterialTheme.colorScheme.background) {

CalculatorScreen()

}

}

}

}

}

@Composable

fun CalculatorScreen() {

var num1 by remember { mutableStateOf("") }

var num2 by remember { mutableStateOf("") }

var result by remember { mutableStateOf<String?>(null) }

var operation by remember { mutableStateOf("") }

Column(

modifier = Modifier

.fillMaxSize()

.padding(16.dp),

verticalArrangement = Arrangement.Center,

horizontalAlignment = Alignment.CenterHorizontally

) {

Text("Simple Calculator", style = MaterialTheme.typography.titleLarge)

Spacer(modifier = Modifier.height(16.dp))

OutlinedTextField(

value = num1,

onValueChange = { num1 = it },

label = { Text("Number 1") },

keyboardOptions = KeyboardOptions(keyboardType = KeyboardType.Number)

)

Spacer(modifier = Modifier.height(8.dp))

OutlinedTextField(

value = num2,

onValueChange = { num2 = it },

label = { Text("Number 2") },

keyboardOptions = KeyboardOptions(keyboardType = KeyboardType.Number)

)

Spacer(modifier = Modifier.height(16.dp))

Row {

Button(onClick = { operation = "Add" }) {

Text("Add")

}

Spacer(modifier = Modifier.width(8.dp))

Button(onClick = { operation = "Subtract" }) {

Text("Subtract")

}

Spacer(modifier = Modifier.width(8.dp))

Button(onClick = { operation = "Multiply" }) {

Text("Multiply")

}

Spacer(modifier = Modifier.width(8.dp))

Button(onClick = { operation = "Divide" }) {

Text("Divide")

}

}

Spacer(modifier = Modifier.height(16.dp))

Button(onClick = {

val number1 = num1.toDoubleOrNull()

val number2 = num2.toDoubleOrNull()

if (number1 != null && number2 != null) {

result = when (operation) {

"Add" -> (number1 + number2).toString()

"Subtract" -> (number1 - number2).toString()

"Multiply" -> (number1 \* number2).toString()

"Divide" -> if (number2 != 0.0) (number1 / number2).toString() else "Cannot divide by zero"

else -> "Select an operation"

}

} else {

result = "Invalid input"

}

}) {

Text("Calculate")

}

Spacer(modifier = Modifier.height(16.dp))

Text(

text = result ?: "",

style = MaterialTheme.typography.bodyLarge

)

}

}

@Preview(showBackground = true)

@Composable

fun PreviewCalculatorScreen() {

SimpleCalculatorTheme {

CalculatorScreen()

}

}

* **Explanation**:
  + **var num1 and var num2**: These mutable states hold the input values for the two numbers.
  + **var result**: This mutable state holds the result of the calculation.
  + **var operation**: This mutable state keeps track of the selected operation (Add, Subtract, Multiply, Divide).
  + **OutlinedTextField**: These are used to input the numbers.
  + **Row with Button**: Used for selecting the operation.
  + **Calculate Button**: Performs the selected operation and updates the result.

**Running the Application**

1. **Run the project** on an emulator or physical device.
2. **Interact with the UI**:
   * Enter two numbers in the input fields.
   * Select an operation (Add, Subtract, Multiply, Divide).
   * Click "Calculate" to see the result.