**Lab Exercise 1- Simple Project in Jetpack Compose**

**Objective:**

Create a simple Jetpack Compose app that displays a "Hello World" message using the Text composable.

**Steps:**

**Step 1: Set up a New Jetpack Compose Project**

1. Open **Android Studio**.
2. Create a new project:
   * Select **New Project** -> **Empty Compose Activity**.
   * Set the project name to HelloWorldCompose and finish the setup.

Android Studio will generate a basic project with the necessary Jetpack Compose setup.

1. Ensure the build.gradle file has the required Compose dependencies:

dependencies {

implementation "androidx.activity:activity-compose:1.7.0"

implementation "androidx.compose.ui:ui:1.4.0"

implementation "androidx.compose.material:material:1.4.0"

implementation "androidx.compose.ui:ui-tooling-preview:1.4.0"

debugImplementation "androidx.compose.ui:ui-tooling:1.4.0"

}

1. Sync your project to download all dependencies.

**Step 2: Write "Hello World" in Jetpack Compose**

1. Open the MainActivity.kt file.
2. Replace the existing code with the following:

import android.os.Bundle

import androidx.activity.ComponentActivity

import androidx.activity.compose.setContent

import androidx.compose.foundation.layout.\*

import androidx.compose.material.\*

import androidx.compose.runtime.Composable

import androidx.compose.ui.Alignment

import androidx.compose.ui.Modifier

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

import androidx.compose.ui.unit.dp

class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContent {

MyApp()

}

}

}

@Composable

fun MyApp() {

// Main content displaying "Hello World"

Box(

modifier = Modifier.fillMaxSize(),

contentAlignment = Alignment.Center

) {

Text(text = "Hello World!")

}

}

@Preview(showBackground = true)

@Composable

fun DefaultPreview() {

MyApp()

}

**Step 3: Run the Application**

1. Select an emulator or physical device.
2. Run the project.
3. You should see a "Hello World!" message displayed in the center of the screen.

**Explanation:**

1. **setContent**: This function is used to set the Compose UI as the content for the activity.
2. **Text**: The Text composable is used to display simple text. In this case, it shows "Hello World!".
3. **Box**: A Box composable is used to create a container. It wraps the Text composable and aligns it in the center of the screen using contentAlignment = Alignment.Center.
4. **Modifier.fillMaxSize()**: Ensures that the Box takes up the entire screen size.
5. **@Preview**: The @Preview annotation allows you to see a preview of the UI in Android Studio without running the app on a device.

**Conclusion:**

This simple lab exercise introduces you to Jetpack Compose by displaying a "Hello World" message. By completing this, you’ve learned the basics of composables like Text, layout containers like Box, and how to use Modifier for positioning content in Jetpack Compose.