**Lab Exercise 17- Navigation in Jetpack Compose**

**Objective:**

The goal is to create a simple app with two screens and implement **Jetpack Compose Navigation** between them.

**Step-by-Step Exercise:**

**1. Setup Your Project**

Make sure your **build.gradle** has the necessary dependencies for **Jetpack Compose** and **Navigation**.

**Add the required dependencies:**

dependencies {

implementation "androidx.navigation:navigation-compose:2.7.2" // Use the latest version of Navigation

implementation "androidx.compose.ui:ui:1.5.0" // Jetpack Compose UI

implementation "androidx.compose.material3:material3:1.0.1" // Material3 support

}

**2. Create the Navigation Component**

You need to set up a **NavHost** and a **NavController** to manage navigation between two composable screens.

**MainActivity.kt**

import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.activity.enableEdgeToEdge  
import androidx.compose.foundation.layout.Arrangement  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.height  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material3.Button  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
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  
import androidx.navigation.NavController  
import androidx.navigation.compose.NavHost  
import androidx.navigation.compose.composable  
import androidx.navigation.compose.rememberNavController  
import com.example.demonavigation.ui.theme.DemoNavigationTheme  
  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *enableEdgeToEdge*()  
 *setContent* **{** NavigationExampleApp()  
 **}** }  
}  
  
@Composable  
fun NavigationExampleApp() {  
 val navController = rememberNavController()  
  
 *// Setup NavHost with the NavController* NavHost(navController = navController, startDestination = "home") **{** *// Composable for Home Screen  
 composable*("home") **{** HomeScreen(navController) **}** *// Composable for Details Screen  
 composable*("details") **{** DetailsScreen(navController) **}  
 }**}  
  
@Composable  
fun HomeScreen(navController: NavController) {  
 *// Layout for Home Screen* Column(  
 modifier = Modifier.*fillMaxSize*(),  
 verticalArrangement = Arrangement.Center,  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) **{** Text(text = "Welcome to the Home Screen")  
 Spacer(modifier = Modifier.*height*(20.*dp*))  
 Button(onClick = **{** navController.navigate("details") **}**) **{** Text(text = "Go to Details")  
 **}  
 }**}  
  
@Composable  
fun DetailsScreen(navController: NavController) {  
 *// Layout for Details Screen* Column(  
 modifier = Modifier.*fillMaxSize*(),  
 verticalArrangement = Arrangement.Center,  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) **{** Text(text = "This is the Details Screen")  
 Spacer(modifier = Modifier.*height*(20.*dp*))  
 Button(onClick = **{** navController.navigate("home") **}**) **{** Text(text = "Go Back to Home")  
 **}  
 }**}  
  
@Preview(showBackground = true)  
@Composable  
fun DefaultPreview() {  
 NavigationExampleApp()  
}

**Explanation:**

1. **NavHost**: The NavHost composable is where all your navigation takes place. It manages the navigation graph and dictates which screen is displayed based on the current state of the NavController.
2. **NavController**: The NavController is responsible for navigating between screens. You pass the NavController to each screen so that each composable can trigger navigation actions.
3. **Screens**:
   * **HomeScreen**: This screen contains a button that navigates to the DetailsScreen.
   * **DetailsScreen**: This screen contains a button that navigates back to the HomeScreen.
4. **Navigation**:
   * When the "Go to Details" button is pressed, the app navigates from the Home screen to the Details screen using navController.navigate("details").
   * From the Details screen, pressing "Go Back to Home" takes you back to the Home screen using navController.navigate("home").