**Kanishka Ghosh**

**20BPS1125**

**Assignment Week 3**

**Android Application Development in Kotlin**

**Code:**

package com.example.youtubevideos\_20bps1125  
  
import android.annotation.SuppressLint  
import android.os.Build  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.annotation.RequiresApi  
import androidx.compose.foundation.Image  
import androidx.compose.foundation.layout.Arrangement  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.ExperimentalLayoutApi  
import androidx.compose.foundation.layout.PaddingValues  
import androidx.compose.foundation.layout.Row  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.foundation.layout.height  
import androidx.compose.foundation.layout.padding  
import androidx.compose.foundation.layout.size  
import androidx.compose.foundation.rememberScrollState  
import androidx.compose.foundation.verticalScroll  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.AccountCircle  
import androidx.compose.material.icons.filled.Favorite  
import androidx.compose.material.icons.filled.Home  
import androidx.compose.material.icons.filled.List  
import androidx.compose.material.icons.filled.Lock  
import androidx.compose.material.icons.filled.Menu  
import androidx.compose.material.icons.filled.Search  
import androidx.compose.material.icons.filled.Settings  
import androidx.compose.material.icons.filled.Share  
import androidx.compose.material.icons.filled.Star  
import androidx.compose.material.icons.filled.ThumbUp  
import androidx.compose.material3.BottomAppBar  
import androidx.compose.material3.DrawerValue  
import androidx.compose.material3.ExperimentalMaterial3Api  
import androidx.compose.material3.Icon  
import androidx.compose.material3.IconButton  
import androidx.compose.material3.ModalDrawerSheet  
import androidx.compose.material3.ModalNavigationDrawer  
import androidx.compose.material3.NavigationDrawerItem  
import androidx.compose.material3.NavigationDrawerItemDefaults  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
import androidx.compose.material3.TopAppBar  
import androidx.compose.material3.rememberDrawerState  
import androidx.compose.runtime.Composable  
import androidx.compose.runtime.mutableStateOf  
import androidx.compose.runtime.remember  
import androidx.compose.runtime.rememberCoroutineScope  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.draw.drawBehind  
import androidx.compose.ui.geometry.Offset  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.graphics.vector.ImageVector  
import androidx.compose.ui.layout.ContentScale  
import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.text.font.FontStyle  
import androidx.compose.ui.tooling.preview.Preview  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
import androidx.compose.ui.viewinterop.AndroidView  
import com.example.youtubevideos\_20bps1125.ui.theme.YouTubeVideos\_20BPS1125Theme  
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.YouTubePlayer  
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.listeners.AbstractYouTubePlayerListener  
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.views.YouTubePlayerView  
import kotlinx.coroutines.launch  
  
  
class MainActivity : ComponentActivity() {  
 @RequiresApi(Build.VERSION\_CODES.O)  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContent {  
 YouTubeVideos\_20BPS1125Theme {  
 HomeScreen()  
 }  
 }  
 }  
}  
  
sealed class Screen(val route: String, val title: String, val icon: ImageVector) {  
 object Home : Screen("home", "Home", Icons.Default.Home)  
 object Subscribed : Screen("Subscribed", "Subscribed Channels", Icons.Default.List)  
 object Recent : Screen("Recent", "Recent Videos", Icons.Default.Favorite)  
 object Trending : Screen("Trending", "Trending Videos", Icons.Default.ThumbUp)  
 object Shorts : Screen("Shorts", "Shorts", Icons.Default.Star)  
 object LogOut : Screen("Logout", "Logout Account", Icons.Default.Lock)  
}  
  
@OptIn(ExperimentalMaterial3Api::class)  
@RequiresApi(Build.VERSION\_CODES.O)  
@Composable  
fun HomeScreen() {  
 val screens = listOf(Screen.Home, Screen.Subscribed, Screen.Recent,Screen.Trending,Screen.Shorts,Screen.LogOut)  
 val drawerState =  
 rememberDrawerState(DrawerValue.Closed)  
 val scope = rememberCoroutineScope()  
 val selectedItem = remember { mutableStateOf(screens[0]) }  
 return (  
 ModalNavigationDrawer(  
 drawerState = drawerState,  
 drawerContent = {  
 ModalDrawerSheet {  
 Spacer(Modifier.height(12.dp))  
 screens.forEach { item ->  
 NavigationDrawerItem(  
 icon = { Icon(item.icon, contentDescription = null) },  
 label = { Text(item.title) },  
 selected = item == selectedItem.value,  
 onClick = {  
 scope.launch { drawerState.close() }  
 selectedItem.value = item  
 },  
 modifier = Modifier.padding(NavigationDrawerItemDefaults.ItemPadding)  
 )  
 }  
 }  
 },  
 content = {  
 Scaffold(  
 topBar = {  
 TopAppBar(  
 title = {  
 Row(  
 verticalAlignment = Alignment.CenterVertically,  
 modifier = Modifier  
 .fillMaxWidth()  
 ) {  
 IconButton(onClick = { scope.launch { drawerState.open() } }) {  
 Icon(  
 Icons.Filled.Menu,  
 contentDescription = "",  
 modifier = Modifier  
 .size(30.dp)  
 )  
 }  
 Image(  
 painter = painterResource(id = R.drawable.yt\_logo),  
 contentDescription = null,  
 contentScale = ContentScale.Fit,  
 modifier = Modifier  
 .height(50.dp),  
 )  
  
 Text(text = "YouTube",  
 modifier = Modifier.padding(end = 30.dp),  
 fontSize = 30.sp  
 )  
  
 Icon(  
 Icons.Filled.Search, contentDescription = "Search", modifier = Modifier  
 .size(50.dp)  
 .padding(end = 10.dp)  
 )  
 Icon(  
 Icons.Filled.AccountCircle, contentDescription = "Account Details", modifier = Modifier  
 .size(40.dp)  
 )  
 }  
 },  
 modifier = Modifier.drawBehind {  
 drawLine(  
 Color.Red,  
 Offset(0f, size.height),  
 Offset(size.width, size.height),  
 5f  
 )  
 }  
 )  
 },  
 bottomBar = { CustomBottomBar() },  
 content = { pad -> MainContent(pad) },  
 )  
 }  
 )  
 )  
}  
  
@OptIn(ExperimentalMaterial3Api::class)  
@RequiresApi(Build.VERSION\_CODES.O)  
@Composable  
fun CustomScaffold() {  
 Scaffold(  
 bottomBar = { CustomBottomBar() },  
 content = { pad -> MainContent(pad) },  
 )  
}  
  
  
@Composable  
fun CustomBottomBar() {  
 remember { mutableStateOf(0) }  
 BottomAppBar(  
 modifier = Modifier  
 .drawBehind {  
 drawLine(  
 Color.White,  
 Offset(0f, 0f),  
 Offset(size.width, 0f),  
 4f  
 )  
 }  
 .height(60.dp),  
 containerColor = Color(184, 0, 0, 255),  
 ) {  
 Row(  
 modifier = Modifier.fillMaxWidth(),  
 horizontalArrangement = Arrangement.SpaceEvenly,  
 verticalAlignment =  
 Alignment.CenterVertically  
 ) {  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) {  
 Icon(imageVector = Icons.Default.Home, "", modifier = Modifier.size(30.dp),tint = Color.White)  
 Text(text = "Home", color = Color.White)  
 }  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) {  
 Icon(imageVector = Icons.Default.Star, "", modifier = Modifier.size(30.dp),tint = Color.White)  
 Text(text = "Recent", color = Color.White)  
 }  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) {  
 Icon(imageVector = Icons.Default.Share, "", modifier = Modifier.size(30.dp),tint = Color.White)  
 Text(text = "Share", color = Color.White)  
 }  
 Column(  
 horizontalAlignment = Alignment.CenterHorizontally  
 ) {  
 Icon(imageVector = Icons.Default.Settings, "", modifier = Modifier.size(30.dp),tint = Color.White)  
 Text(text = "Settings", color = Color.White)  
 }  
 }  
 }  
  
}  
  
@Composable  
fun YoutubeScreen(  
 videoId: String  
) {  
 AndroidView(factory = {  
 var view = YouTubePlayerView(it)  
 val fragment = view.addYouTubePlayerListener(  
 object : AbstractYouTubePlayerListener() {  
 override fun onReady(youTubePlayer: YouTubePlayer) {  
 super.onReady(youTubePlayer)  
 youTubePlayer.cueVideo(videoId, 0f)  
 }  
 }  
 )  
 view  
 })  
}  
  
@SuppressLint("RememberReturnType")  
@RequiresApi(Build.VERSION\_CODES.O)  
@OptIn(ExperimentalLayoutApi::class)  
@Composable  
fun MainContent(padding: PaddingValues) {  
 Column(  
 modifier = Modifier.verticalScroll(rememberScrollState())  
 ) {  
 Spacer(modifier = Modifier.height(60.dp))  
 Column() {  
 YoutubeScreen("S8SG1pX5eLA")  
 Spacer(modifier = Modifier.height(10.dp))  
 Row() {  
 Icon(  
 Icons.Filled.AccountCircle, contentDescription = "Account Details", modifier = Modifier  
 .size(40.dp)  
 )  
 Text(text = "Peter Svidler plays the Pirc against Vidit Gujrathi | Blitz Game 1",  
 fontSize = 20.sp  
 )  
 }  
 Row() {  
 Text(text = "26M views . 1 year ago",  
 fontSize = 15.sp,  
 fontStyle = FontStyle.Italic,  
 modifier = Modifier.padding(start = 50.dp)  
 )  
 }  
 }  
  
  
 Spacer(modifier = Modifier.height(20.dp))  
 Column() {  
 YoutubeScreen("3K-vJlVMi5A")  
 Spacer(modifier = Modifier.height(10.dp))  
 Row() {  
 Icon(  
 Icons.Filled.AccountCircle, contentDescription = "Account Details", modifier = Modifier  
 .size(40.dp)  
 )  
 Text(text = "Data Analytics Using AI | Data Analysis Using Artificial Intelligence 2023 | Simplilearn",  
 fontSize = 20.sp  
 )  
 }  
 Row() {  
 Text(text = "89K views . 3 months ago",  
 fontSize = 15.sp,  
 fontStyle = FontStyle.Italic,  
 modifier = Modifier.padding(start = 50.dp)  
 )  
 }  
 }  
  
 Spacer(modifier = Modifier.height(20.dp))  
 Column() {  
 YoutubeScreen("qOwtIj7FM5Q")  
 Spacer(modifier = Modifier.height(10.dp))  
 Row() {  
 Icon(  
 Icons.Filled.AccountCircle, contentDescription = "Account Details", modifier = Modifier  
 .size(40.dp)  
 )  
 Text(text = "What is the snowball effect? How it will help India's economy to grow exponentially | UPSC GS 3",  
 fontSize = 20.sp  
 )  
 }  
 Row() {  
 Text(text = "1.1K views . 2 hours ago",  
 fontSize = 15.sp,  
 fontStyle = FontStyle.Italic,  
 modifier = Modifier.padding(start = 50.dp)  
 )  
 }  
 }  
 Spacer(modifier = Modifier.height(20.dp))  
 Column() {  
 YoutubeScreen("rWv0zzEYB2w")  
 Spacer(modifier = Modifier.height(10.dp))  
 Row() {  
 Icon(  
 Icons.Filled.AccountCircle, contentDescription = "Account Details", modifier = Modifier  
 .size(40.dp)  
 )  
 Text(text = "Using Appwrite service with custom context in React Native",  
 fontSize = 20.sp  
 )  
 }  
 Row() {  
 Text(text = "605 views . 6 hours ago",  
 fontSize = 15.sp,  
 fontStyle = FontStyle.Italic,  
 modifier = Modifier.padding(start = 50.dp)  
 )  
 }  
 }  
 Spacer(modifier = Modifier.height(70.dp))  
 }  
}  
  
  
@RequiresApi(Build.VERSION\_CODES.O)  
@Preview(showBackground = true)  
@Composable  
fun DefaultPreview() {  
 HomeScreen()  
}