

20BCE1458

AYUSH KAPRI

ASSIGNMENT-3

CODE

```
package com.example.vit_20bce1458_youtube

import android.os.Bundle
import android.view.ViewGroup
import android.webkit.WebChromeClient
import android.webkit.WebSettings
import android.webkit.WebView
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.height
import androidx.compose.foundation.layout.width
import androidx.compose.material.MaterialTheme
import androidx.compose.material.Surface
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.unit.dp
import androidx.compose.ui.viewinterop.AndroidView
import
com.example.vit_20bce1458_youtube.ui.theme.Vit_20bce1458_youtubeTheme

class MainActivity : ComponentActivity() {
    private val VIDEO_ID_1 = "f--fD8Y0RnA"
    private val VIDEO_ID_2 = "LpHnHRI6gLc"
    private val VIDEO_ID_3 = "N3Xu1rO6feY"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            Vit_20bce1458_youtubeTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colors.background
                ) {
                    Column(modifier = Modifier.fillMaxSize()) {
                        YouTubePlayerWebView(VIDEO_ID_1)
                        YouTubePlayerWebView(VIDEO_ID_2)
                        YouTubePlayerWebView(VIDEO_ID_3)
                    }
                }
            }
        }
    }
}
```

```

    }
}

@Composable
private fun YouTubePlayerWebView(videoId: String) {
    AndroidView(
        factory = { context ->
            WebView(context).apply {
                layoutParams = ViewGroup.LayoutParams(
                    ViewGroup.LayoutParams.MATCH_PARENT,
                    ViewGroup.LayoutParams.MATCH_PARENT
                )
                settings.apply {
                    javaScriptEnabled = true
                    mediaPlaybackRequiresUserGesture = false
                    allowFileAccess = false
                    allowContentAccess = false
                    allowFileAccessFromFileURLs = false
                    allowUniversalAccessFromFileURLs = false
                    mixedContentMode =
WebSettings.MIXED_CONTENT_NEVER_ALLOW
                }
                webChromeClient = WebChromeClient()
                loadVideo(videoId)
            }
        },
        update = { webView ->
            webView.loadVideo(videoId)
        },
        modifier = Modifier
            .height(244.dp)
            .width(380.dp)
    )
}

private fun WebView.loadVideo(videoId: String) {
    val videoUrl =
"https://www.youtube.com/embed/$videoId?autoplay=1&fs=0&rel=0&showinfo=0&lo
op=1"
    loadUrl(videoUrl)
}
}

```