

Code:

```
package com.example.vitv20bci0095

import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.compose.foundation.clickable
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.fillMaxWidth
import androidx.compose.foundation.lazy.LazyColumn
import androidx.compose.foundation.lazy.items
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.runtime.Composable
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.remember
import androidx.compose.ui.Modifier
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.viewinterop.AndroidView
import com.example.vitv20bci0095.ui.theme.VITV20BCI0095Theme
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.YouTubePlayer
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.listeners.AbstractYouTubePlayerListener
import com.pierfrancescosoffritti.androidyoutubeplayer.core.player.views.YouTubePlayerView

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            VITV20BCI0095Theme {
                // A surface container using the 'background' color from
the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    val videoIds = listOf("lKAbgGDarlw", "lGtnla6Py20",
"eJAAGbm27pY", "DoTkzj3wiHE", "sMmZpNz1zXI")
                    val selectedVideoId = remember { mutableStateOf("") }
                    YoutubeScreen(videoIds, selectedVideoId.value) {
                        selectedVideoId.value = videoId
                    }
                }
            }
        }
    }
}

@Composable
fun YoutubeScreen(
    videoIds: List<String>,
    selectedVideoId: String,
    onVideoItemClick: (String) -> Unit
) {
```

```

LazyColumn {
    items(videoIds) { videoId ->
        VideoItem(
            videoId = videoId,
            onVideoItemClick = onVideoItemClick,
            isSelected = videoId == selectedVideoId
        )
    }
}

@Composable
fun VideoItem(
    videoId: String,
    onVideoItemClick: (String) -> Unit,
    isSelected: Boolean
) {
    Surface(
        modifier = Modifier
            .fillMaxWidth()
            .clickable { onVideoItemClick(videoId) },
        color = if (isSelected) MaterialTheme.colorScheme.primary else
MaterialTheme.colorScheme.background
    ) {
        AndroidView(factory = { context ->
            val view = YouTubePlayerView(context)
            val fragment = view.addYouTubePlayerListener(
                object : AbstractYouTubePlayerListener() {
                    override fun onReady(youtubePlayer: YouTubePlayer) {
                        super.onReady(youtubePlayer)
                        youtubePlayer.cueVideo(videoId, 0f)
                    }
                }
            )
            view
        })
    }
}

```

Gradle:

```

plugins {
    id 'com.android.application'
    id 'org.jetbrains.kotlin.android'
}

android {
    namespace 'com.example.vitv20bci0095'
    compileSdk 33

    defaultConfig {
        applicationId "com.example.vitv20bci0095"
        minSdk 24
        targetSdk 33
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
        vectorDrawables {
            useSupportLibrary true
        }
    }
}

```

```

        buildTypes {
            release {
                minifyEnabled false
                proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
            }
        }
        compileOptions {
            sourceCompatibility JavaVersion.VERSION_1_8
            targetCompatibility JavaVersion.VERSION_1_8
        }
        kotlinOptions {
            jvmTarget = '1.8'
        }
        buildFeatures {
            compose true
        }
        composeOptions {
            kotlinCompilerExtensionVersion '1.3.2'
        }
        packagingOptions {
            resources {
                excludes += '/META-INF/{AL2.0,LGPL2.1}'
            }
        }
    }
}

dependencies {
    implementation 'androidx.core:core-ktx:1.8.0'
    implementation platform('org.jetbrains.kotlin:kotlin-bom:1.8.0')
    implementation 'androidx.lifecycle:lifecycle-runtime-ktx:2.3.1'
    implementation 'androidx.activity:activity-compose:1.5.1'
    implementation platform('androidx.compose:compose-bom:2022.10.00')
    implementation 'androidx.compose.ui:ui'
    implementation 'androidx.compose.ui:ui-graphics'
    implementation 'androidx.compose.ui:ui-tooling-preview'
    implementation 'androidx.compose.material3:material3'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.5'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'
    androidTestImplementation platform('androidx.compose:compose-bom:2022.10.00')
    androidTestImplementation 'androidx.compose.ui:ui-test-junit4'
    debugImplementation 'androidx.compose.ui:ui-tooling'
    debugImplementation 'androidx.compose.ui:ui-test-manifest'
    implementation ("com.pierfrancescosoffritti.androidyoutubeplayer:core:11.1.0")
}

```

AndroidManifest.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"

```

```
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.VITV20BCI0095"
    tools:targetApi="31">
    <activity
        android:name=".MainActivity"
        android:exported="true"
        android:label="@string/app_name"
        android:theme="@style/Theme.VITV20BCI0095">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER"
/>
        </intent-filter>
    </activity>
</application>

</manifest>
```