

# Programmation Multimédia

INSTITUT GRASSET  
TECHNIQUES DE L'INFORMATIQUE

Enseignant : Pascal Guy

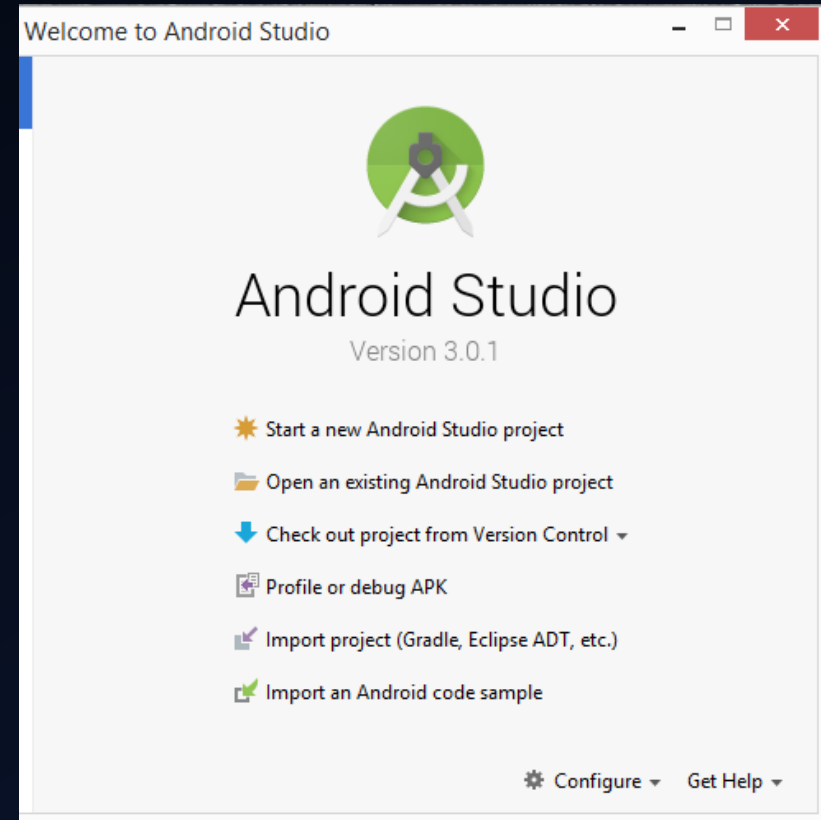
Cours 10 : Lecture de vidéo

# Sommaire

- Création d'une nouvelle application
- Création de l'Activity
- Codage
- XML

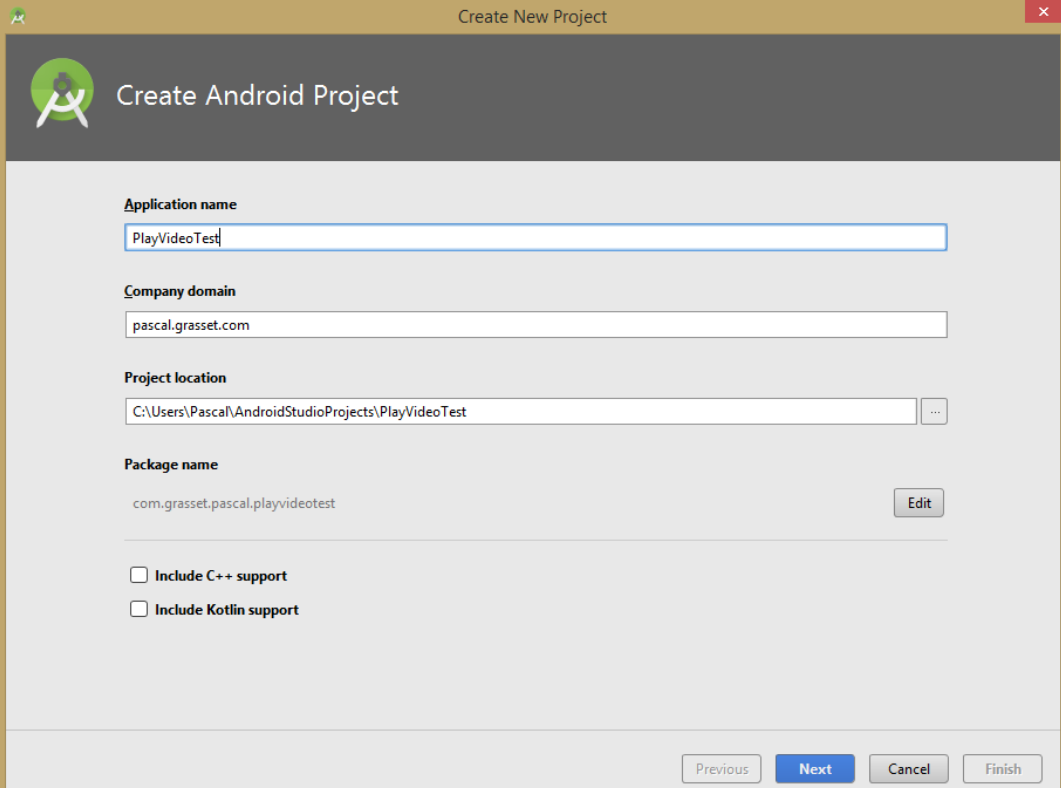
# Ajouter un titre de diapositive - 2

## CRÉER UN NOUVEAU PROJET



# Configurer le projet

- Application Name : PlayVideoTest
- Choisir le dossier ou le projet sera créé.




The screenshot shows the 'Create New Project' dialog box in Android Studio. The dialog has a title bar with the Android Studio logo and the text 'Create New Project'. Below the title bar, there is a header section with the Android Studio logo and the text 'Create Android Project'. The main area of the dialog contains several input fields and checkboxes:

- Application name:** A text field containing 'PlayVideoTest'.
- Company domain:** A text field containing 'pascal.grasset.com'.
- Project location:** A text field containing 'C:\Users\Pascal\AndroidStudioProjects\PlayVideoTest' with a browse button (three dots) to its right.
- Package name:** A text field containing 'com.grasset.pascal.playvideotest' with an 'Edit' button to its right.
- Include C++ support:** A checkbox that is currently unchecked.
- Include Kotlin support:** A checkbox that is currently unchecked.

At the bottom of the dialog, there are four buttons: 'Previous', 'Next' (highlighted in blue), 'Cancel', and 'Finish'.

# Choisir l'API

Create New Project

 Target Android Devices

### Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**

API 21: Android 5.0 (Lollipop)

By targeting **API 21 and later**, your app will run on approximately **71,3%** of devices. [Help me choose](#)

☐ Include Android Instant App support

☐ **Wear**

API 21: Android 5.0 (Lollipop)

☐ **TV**

API 21: Android 5.0 (Lollipop)

☐ **Android Auto**

☐ **Android Things**

API 24: Android 7.0 (Nougat)

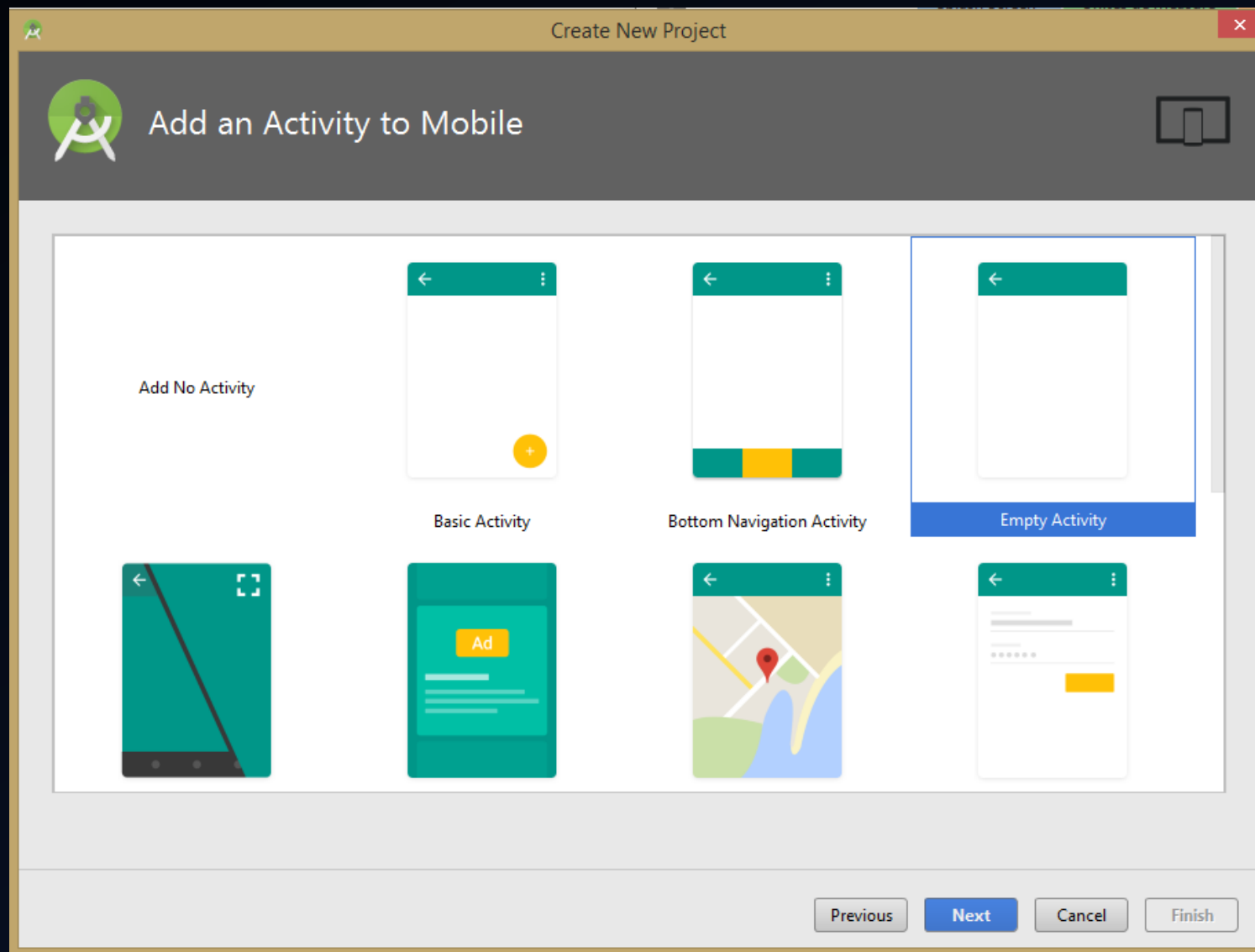
Previous

Next

Cancel

Finish

# Ajoutez une Activity vide

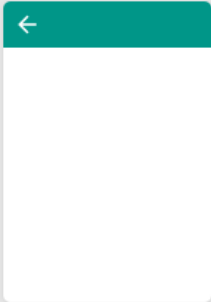


# Configurer l'Activity

Create New Project

## Configure Activity

Creates a new empty activity



**Activity Name**

MainActivity

☒ Generate Layout File

**Layout Name**

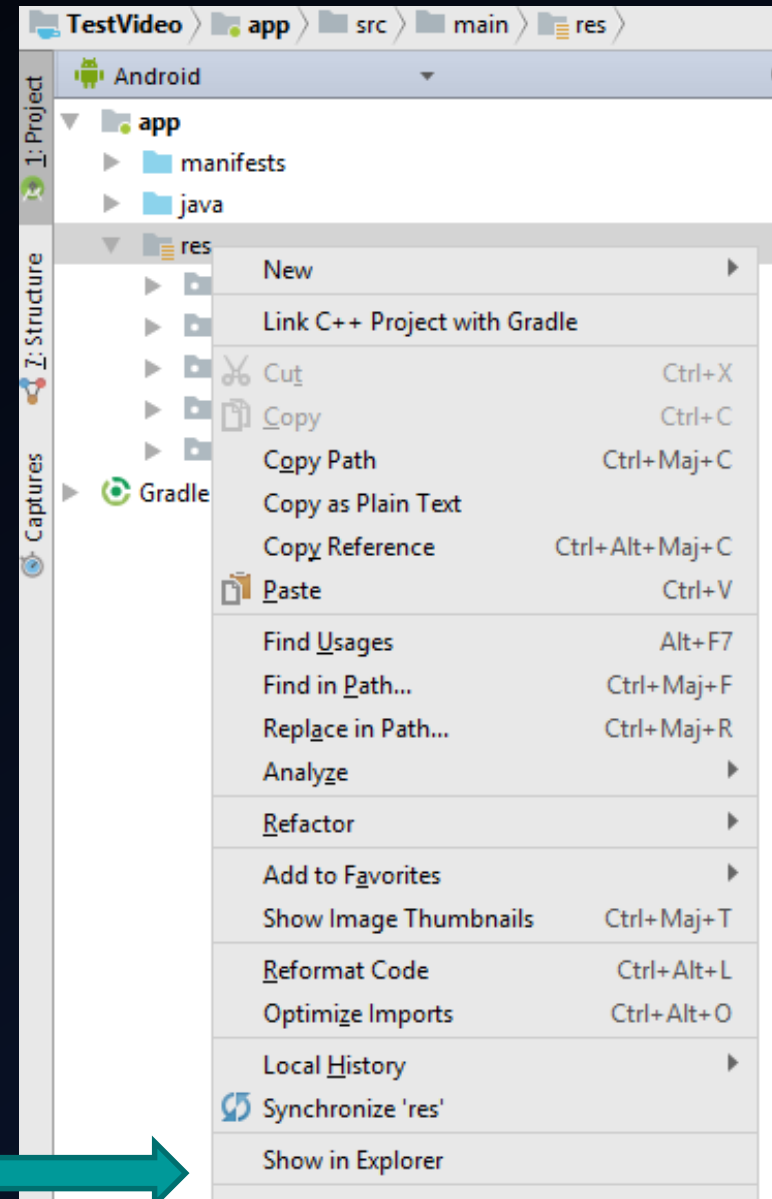
activity\_main

☒ Backwards Compatibility (AppCompat)

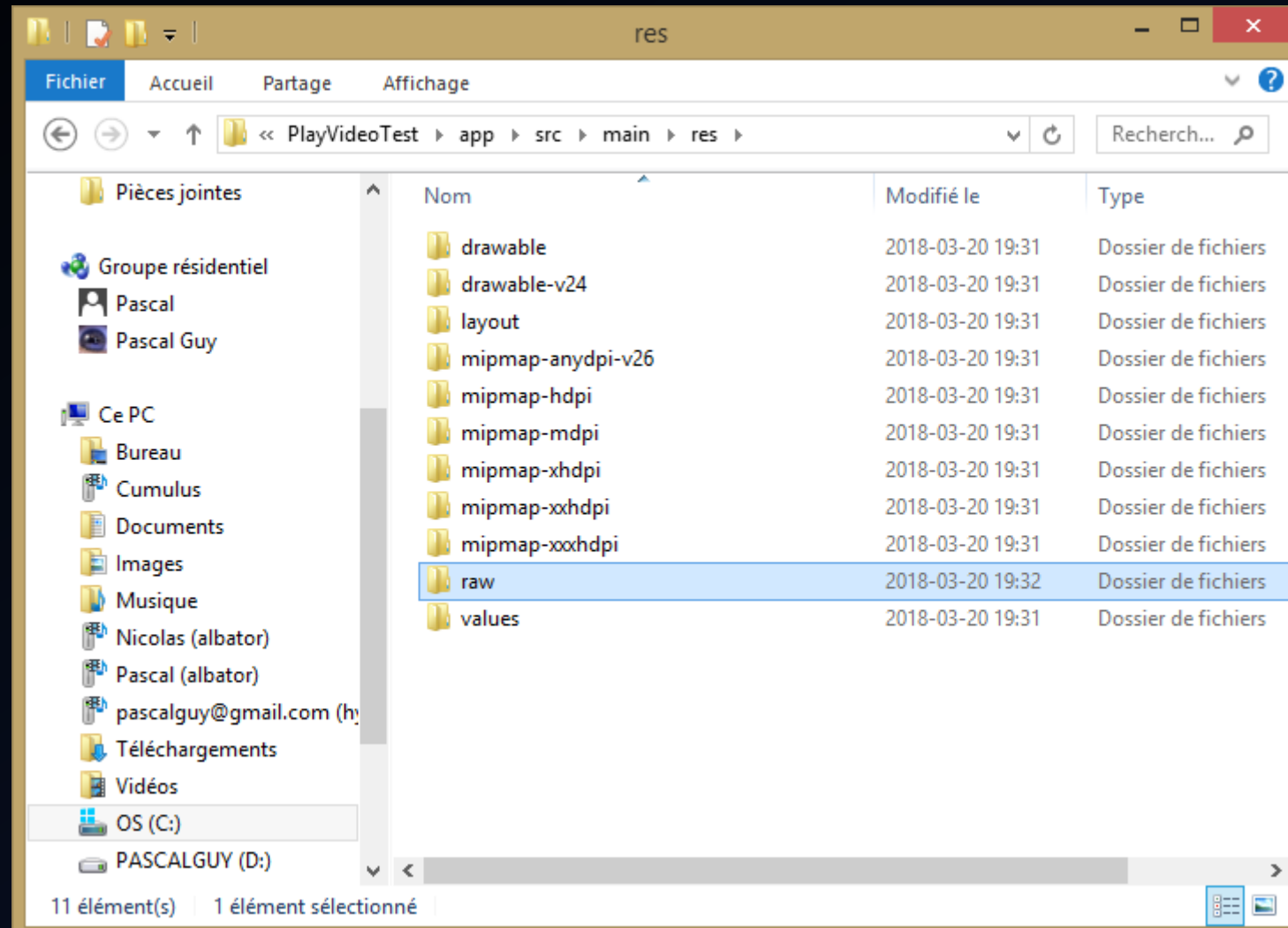
Previous Next Cancel Finish

# Création du dossier raw

- Dans le dossier res du projet, créez un nouveau dossier nommé raw.
- Vous pouvez accéder au dossier res en cliquant droit dessus et en choisissant Show in Explorer.



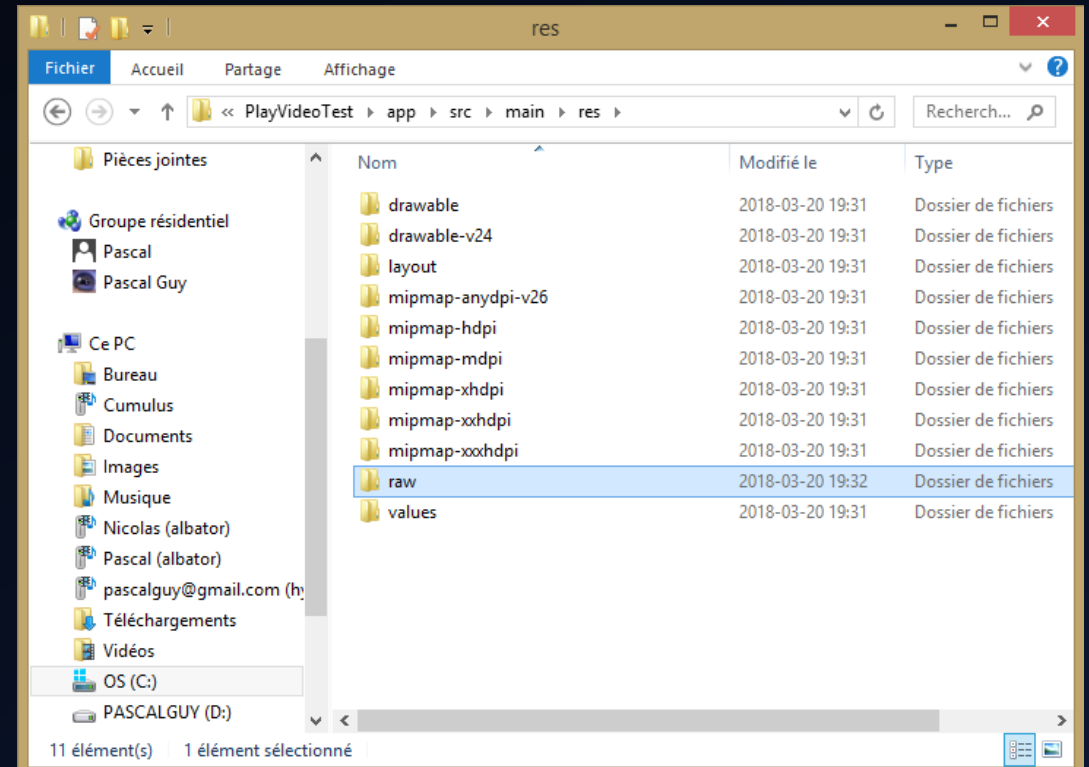


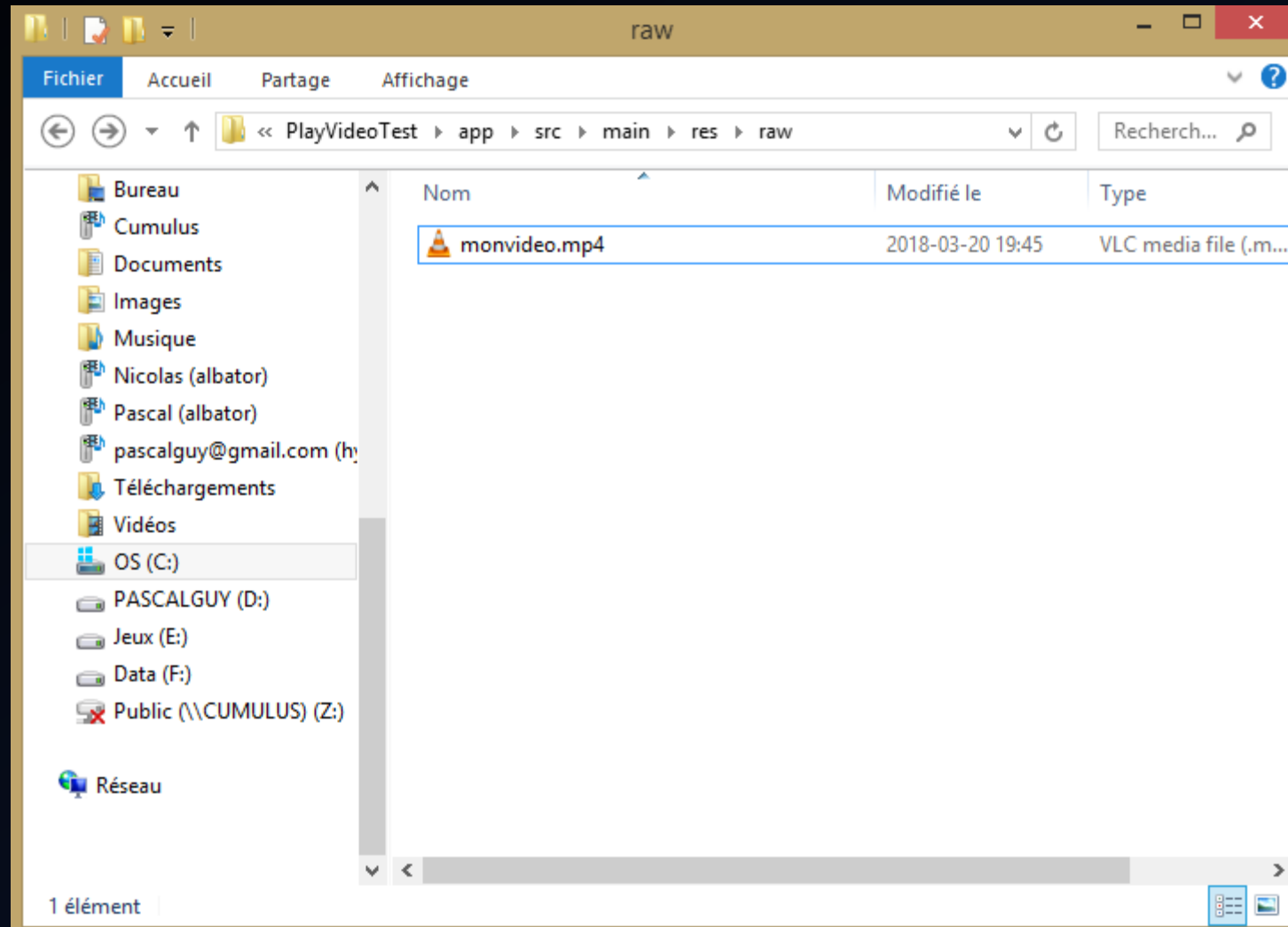


# Insérer la vidéo

- Placez une vidéo dans le dossier raw
- Vous trouverez les standards d'encodage de vidéo supportés à l'adresse suivante :

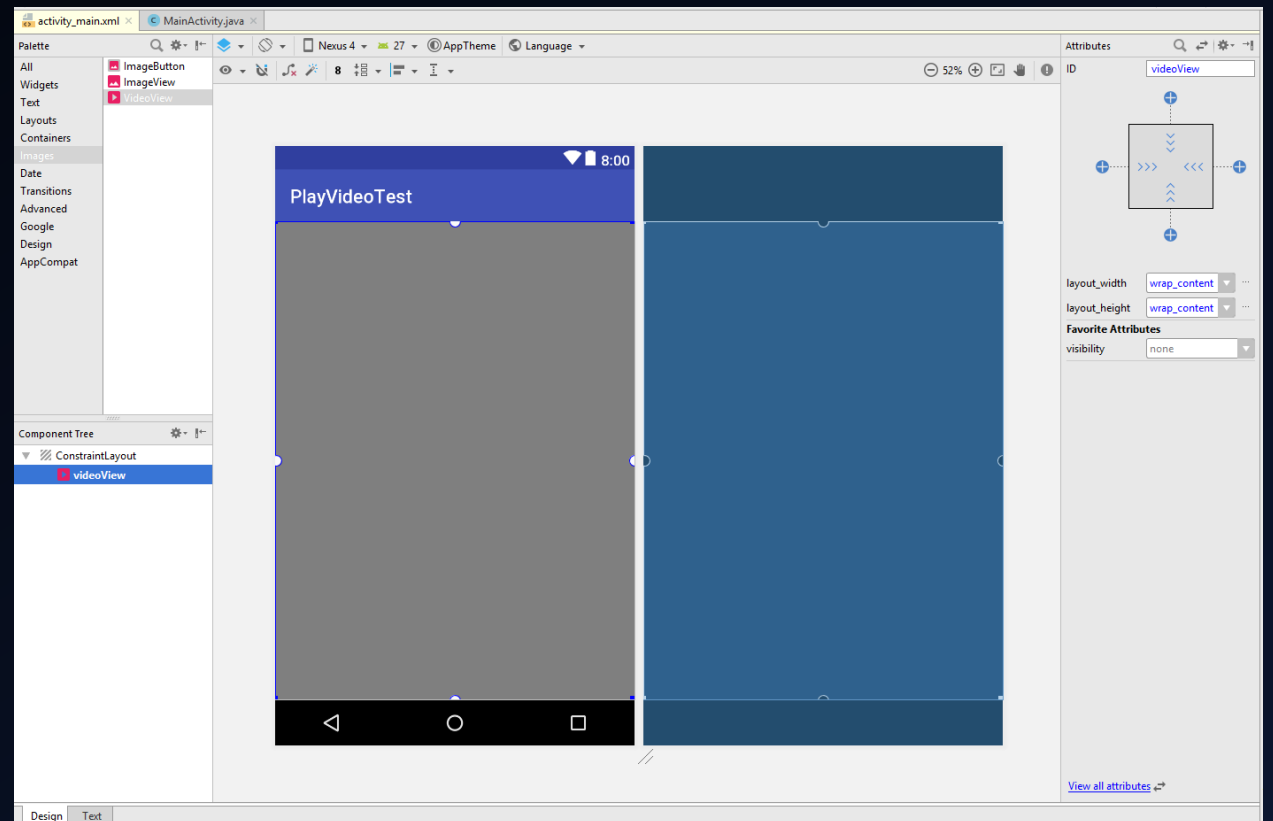
<https://developer.android.com/guide/topics/media/media-formats.html>





# VideoView

- Dans le Layout, placer un objet VideoView



# Aller dans MainActivity.java

A screenshot of an IDE window with two tabs: 'activity\_main.xml' and 'MainActivity.java'. The 'MainActivity.java' tab is active, showing the following Java code:

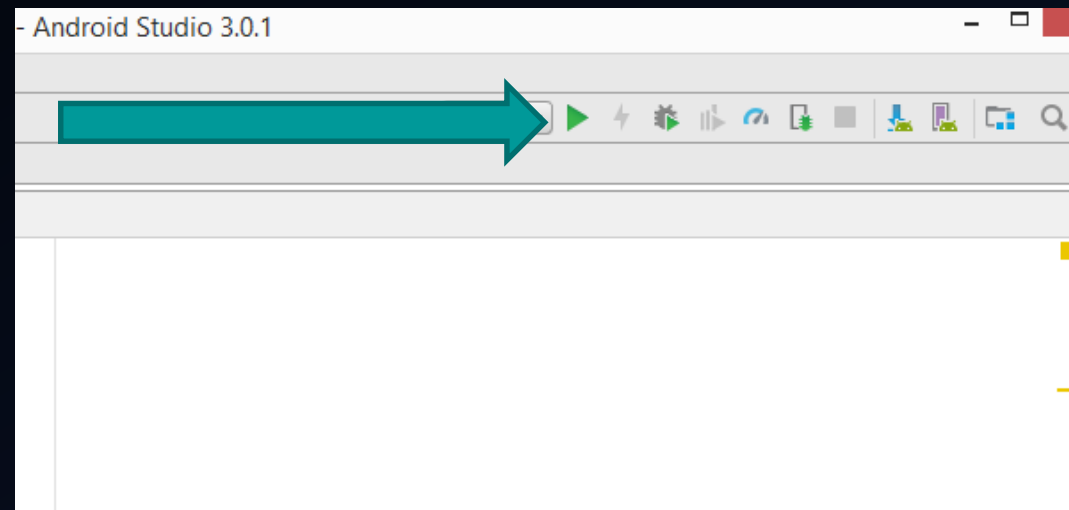
```
1 package com.grasset.pascal.playvideotest;
2
3 import ...
4
5
6 public class MainActivity extends AppCompatActivity {
7
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_main);
12     }
13 }
14
```

The code is displayed with line numbers 1 through 14 on the left. The package name is 'com.grasset.pascal.playvideotest'. The class 'MainActivity' extends 'AppCompatActivity'. The 'onCreate' method is annotated with '@Override' and calls 'super.onCreate' and 'setContentView' with 'R.layout.activity\_main'. The line numbers 14 and 15 are highlighted in yellow.

# Ajoutez le code suivant

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        VideoView mvideoView = (VideoView) findViewById(R.id.mvideoView);  
        String uriPath = "android.resource://com.tiri.pguy.testvideo/" + R.raw.test;  
  
        Uri uri = Uri.parse(uriPath);  
        mvideoView.setVideoURI(uri);  
        mvideoView.requestFocus();  
        mvideoView.start();  
    }  
}
```

# Testez l'application



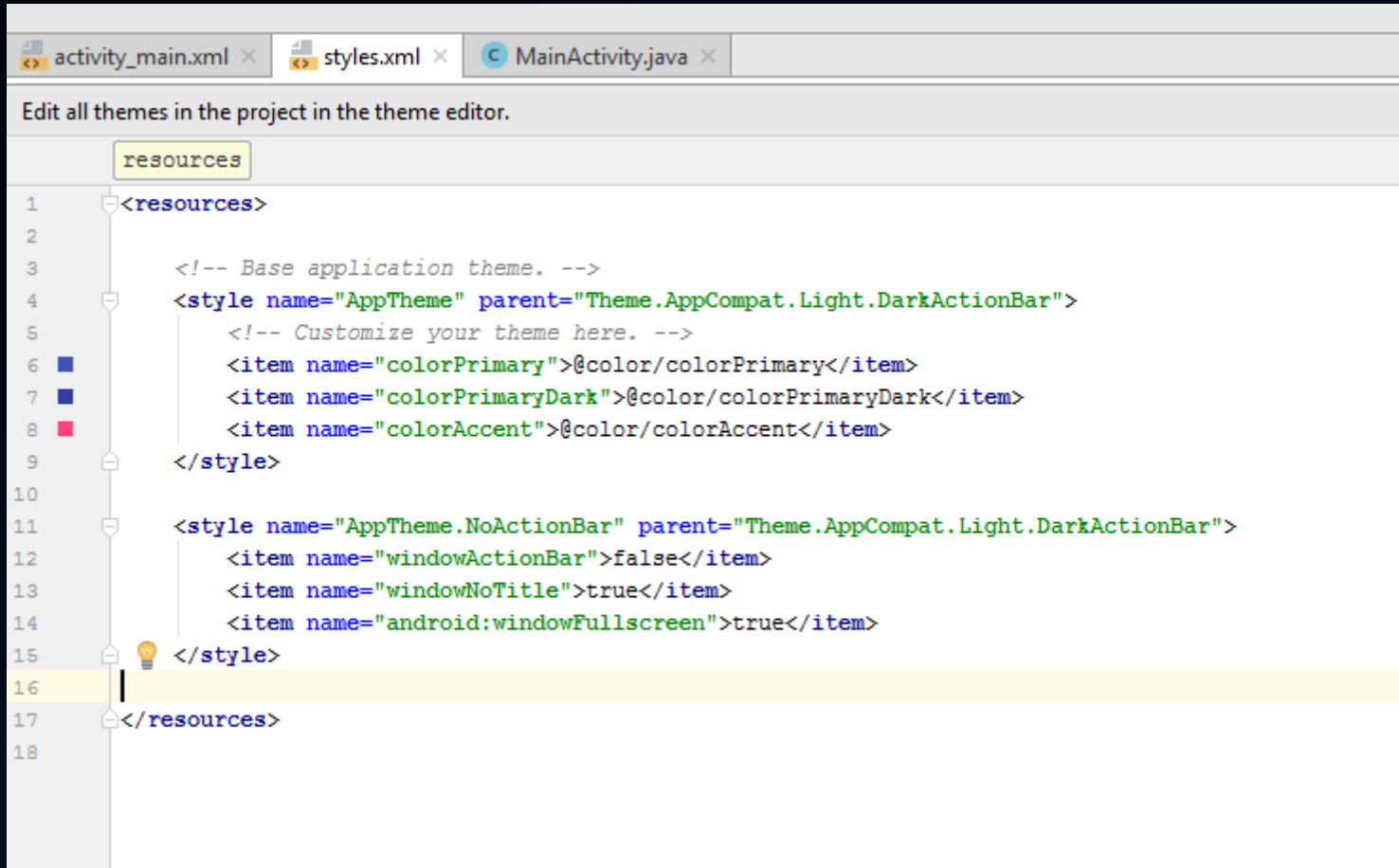
# Pour avoir la vidéo plein écran



```
activity_main.xml x MainActivity.java x
android.support.constraint.ConstraintLayout | VideoView
1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="fill_parent"
6     android:layout_height="fill_parent"
7     tools:context="com.tiri.pguy.testvideo.MainActivity">
8
9     <VideoView
10         android:id="@+id/mvideoView"
11         android:layout_alignParentTop="true"
12         android:layout_alignParentBottom="true"
13         android:layout_alignParentLeft="true"
14         android:layout_alignParentRight="true"
15         android:layout_width="fill_parent"
16         android:layout_height="fill_parent"
17         app:layout_constraintStart_toStartOf="parent"
18         app:layout_constraintTop_toTopOf="parent" />
19
20 </android.support.constraint.ConstraintLayout>
21
```

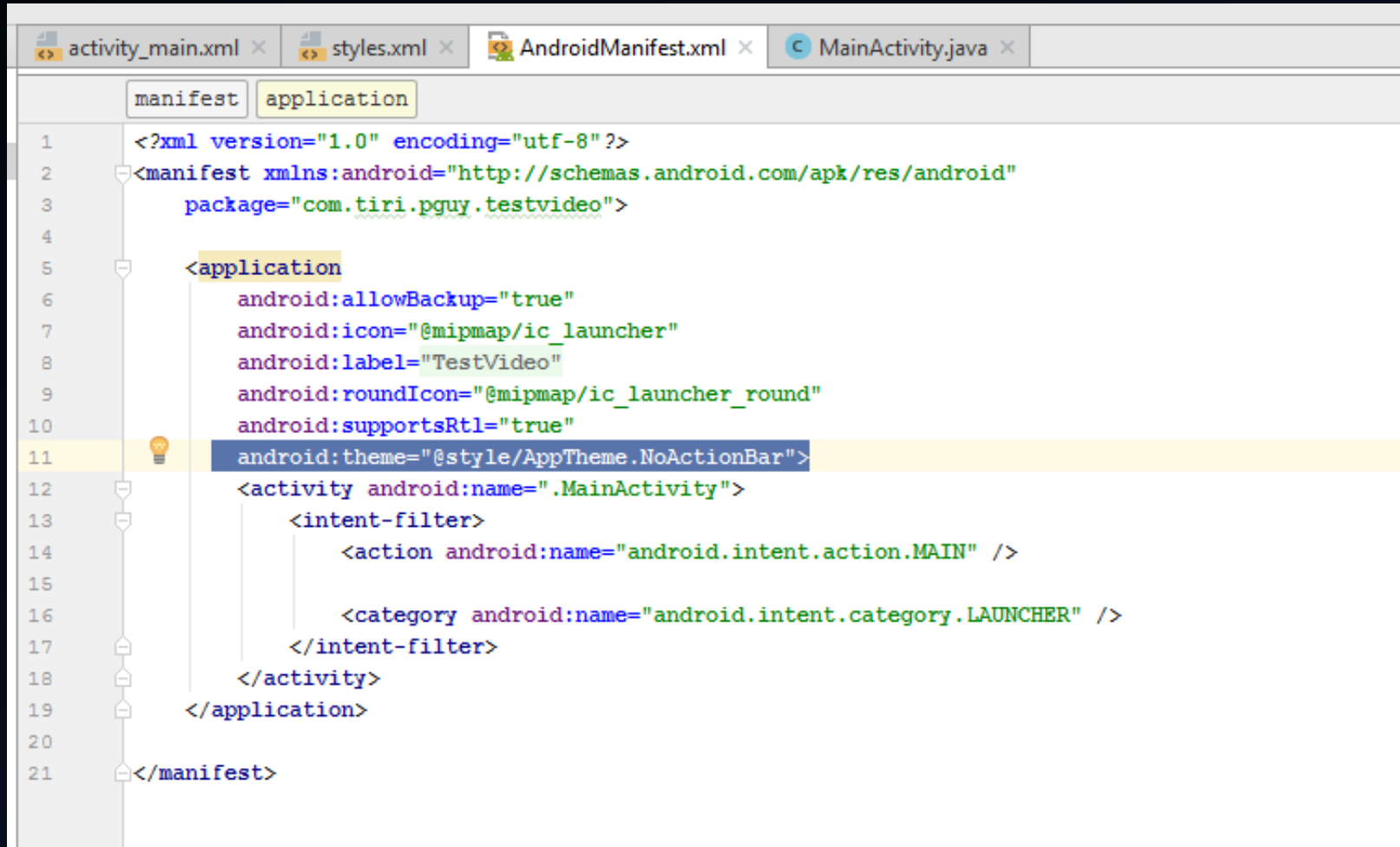


# Pour enlever la barre titre



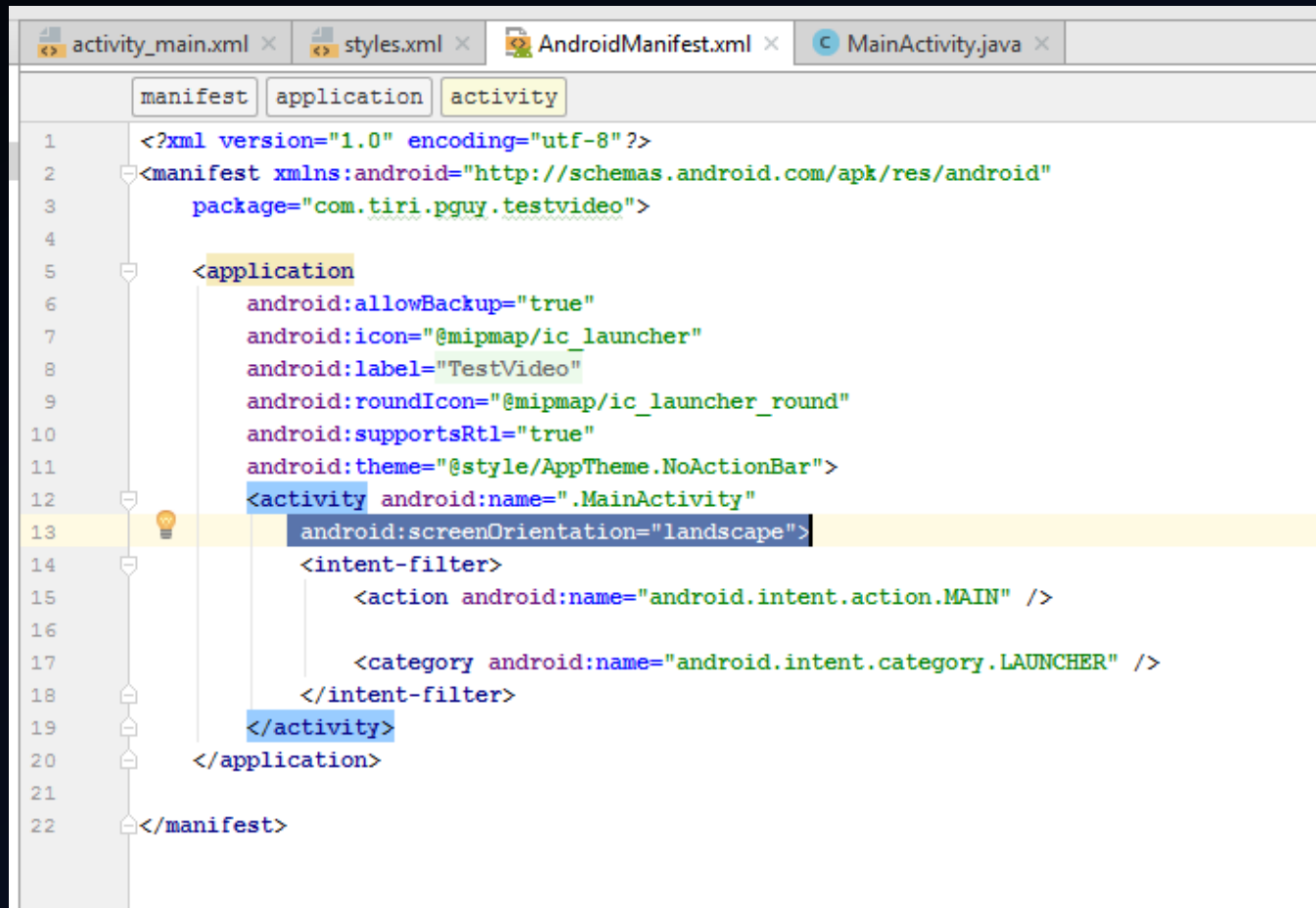
```
1 <resources>
2
3     <!-- Base application theme. -->
4     <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
5         <!-- Customize your theme here. -->
6         <item name="colorPrimary">@color/colorPrimary</item>
7         <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
8         <item name="colorAccent">@color/colorAccent</item>
9     </style>
10
11     <style name="AppTheme.NoActionBar" parent="Theme.AppCompat.Light.DarkActionBar">
12         <item name="windowActionBar">false</item>
13         <item name="windowNoTitle">true</item>
14         <item name="android:windowFullscreen">true</item>
15     </style>
16 </resources>
```

# Puis dans AndroidManifest.xml



```
activity_main.xml x styles.xml x AndroidManifest.xml x MainActivity.java x
manifest application
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.tiri.pguy.testvideo">
4
5     <application
6         android:allowBackup="true"
7         android:icon="@mipmap/ic_launcher"
8         android:label="TestVideo"
9         android:roundIcon="@mipmap/ic_launcher_round"
10        android:supportsRtl="true"
11        android:theme="@style/AppTheme.NoActionBar">
12        <activity android:name=".MainActivity">
13            <intent-filter>
14                <action android:name="android.intent.action.MAIN" />
15
16                <category android:name="android.intent.category.LAUNCHER" />
17            </intent-filter>
18        </activity>
19    </application>
20
21 </manifest>
```

# Pour que le mode Landscape s'affiche automatiquement



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.tiri.pguy.testvideo">
4
5     <application
6         android:allowBackup="true"
7         android:icon="@mipmap/ic_launcher"
8         android:label="TestVideo"
9         android:roundIcon="@mipmap/ic_launcher_round"
10        android:supportsRtl="true"
11        android:theme="@style/AppTheme.NoActionBar">
12        <activity android:name=".MainActivity"
13            android:screenOrientation="landscape">
14            <intent-filter>
15                <action android:name="android.intent.action.MAIN" />
16
17                <category android:name="android.intent.category.LAUNCHER" />
18            </intent-filter>
19        </activity>
20    </application>
21
22 </manifest>
```