

# TP N° 2

## Prise en main d'Android

2024 - 2025  
Dr. Abdelkader Ouared



# Les étapes



Installer Android Studio

Créer/Configurer un projet Android

Créer et gérer des appareils virtuels

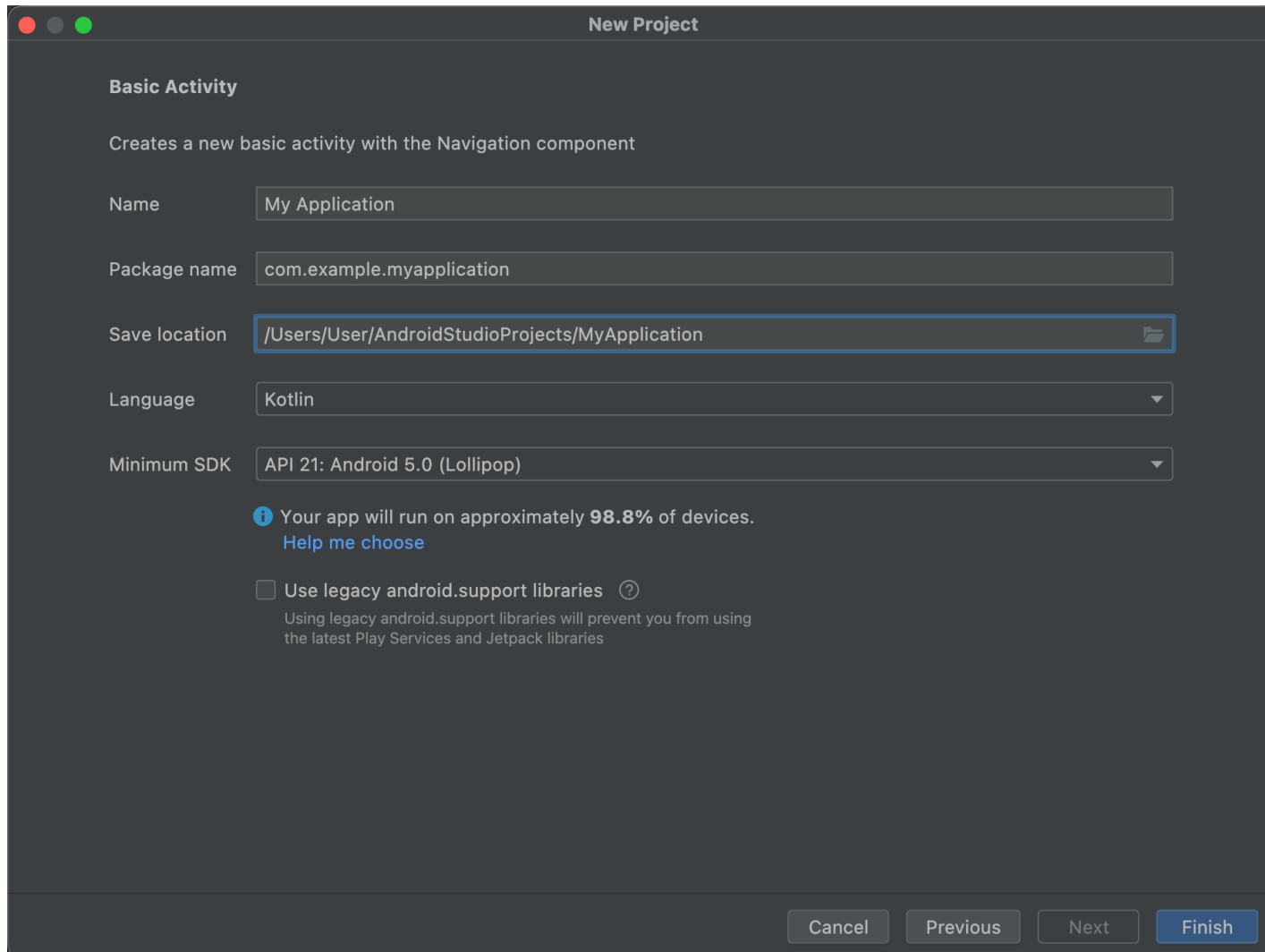
Générer l'APK

Utiliser les Pratiques CI/CD

# Installation Android Studio

<https://developer.android.com/studio>

# Configurer votre projet



**New Project**

**Basic Activity**

Creates a new basic activity with the Navigation component

Name: My Application

Package name: com.example.myapplication

Save location: /Users/User/AndroidStudioProjects/MyApplication

Language: Kotlin

Minimum SDK: API 21: Android 5.0 (Lollipop)

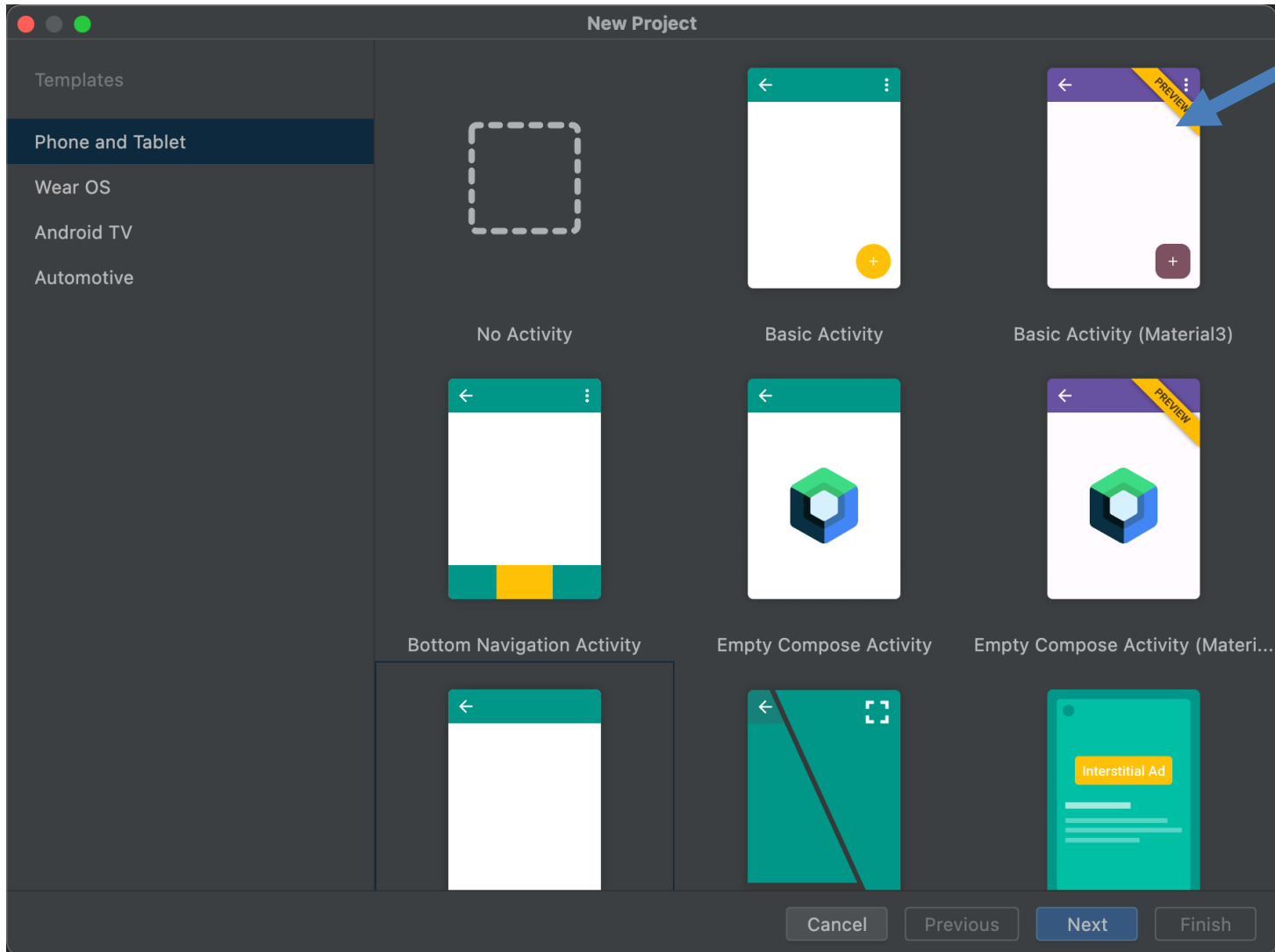
**i** Your app will run on approximately **98.8%** of devices.  
[Help me choose](#)

☐ Use legacy android.support libraries **?**  
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Cancel Previous Next **Finish**

# Créer un projet

**JETPACK  
COMPOSE**



# Mon app



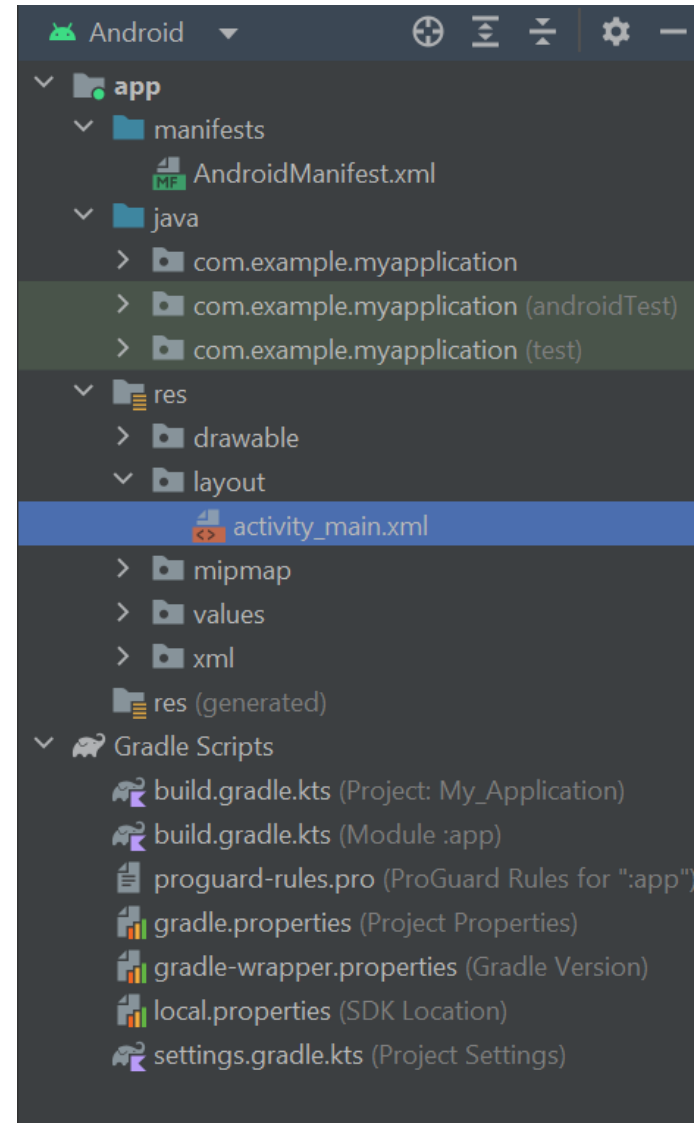
```
graph LR; Manifest[Manifest] --- Java[Java]; Java --- Res[Res];
```

Manifest

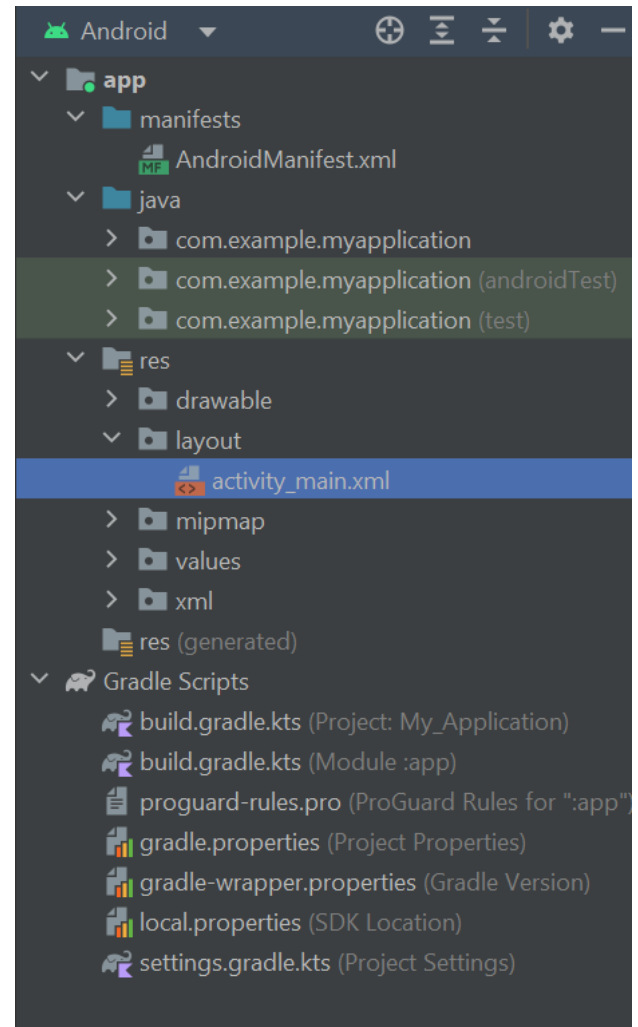
Java

Res

# Structure de projet



# Structure de projet





# Mon app

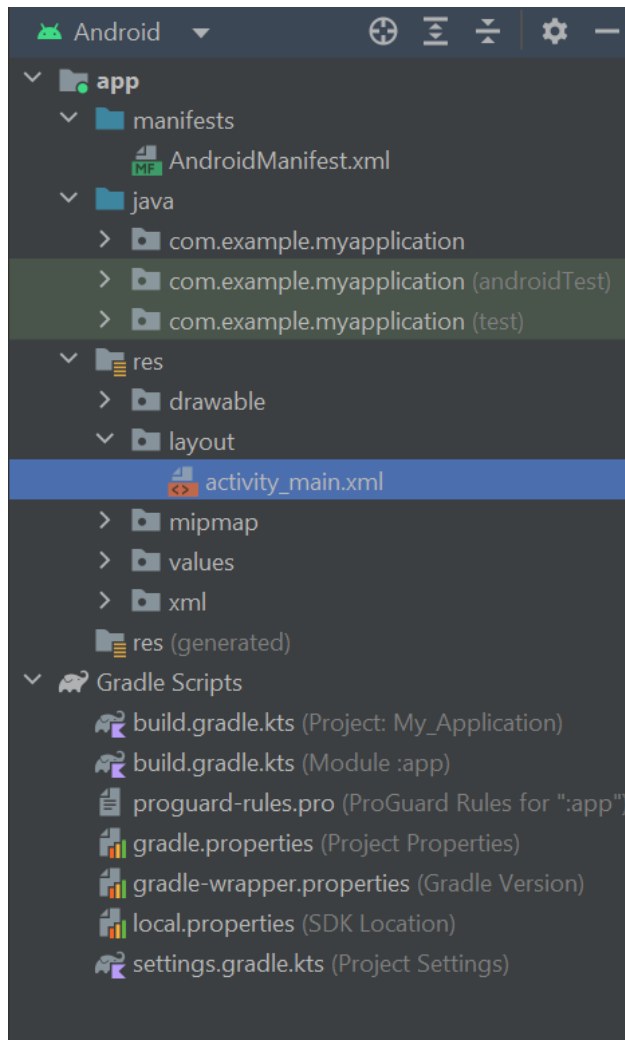


Manifest

The diagram illustrates the components of an Android application. It features three rounded rectangular boxes arranged horizontally. The first box on the left is red and labeled 'Manifest'. The second box in the middle is blue and labeled 'Java'. The third box on the right is also blue and labeled 'Res'. All three boxes have a subtle drop shadow, giving them a 3D appearance. The background is white, and there is a green bar at the bottom left and a dark grey bar at the bottom right.

Java

Res



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <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:supportRtl="true"
        android:theme="@style/Theme.MyApplication"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

# Mon app



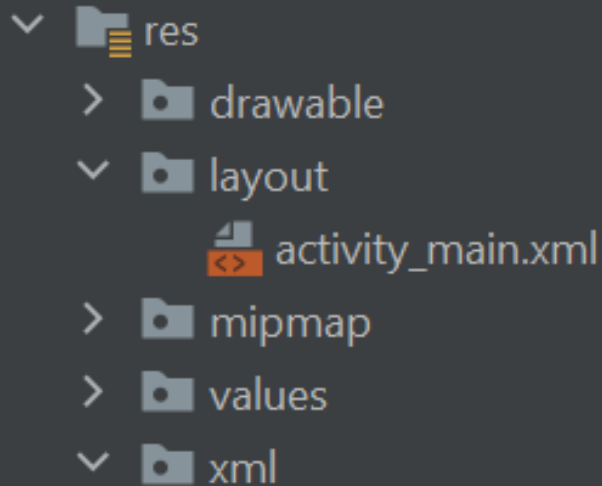
Manifest

The diagram illustrates the components of an Android application. At the top, the text 'Mon app' is centered. Below it, three rounded square boxes are arranged horizontally. The first two boxes, 'Manifest' and 'Java', are blue, while the third box, 'Res', is red. All boxes have a slight drop shadow. A green bar is at the bottom left, and a dark grey bar is at the bottom right.

Java

Res

# Mon app



**Drawable:** images

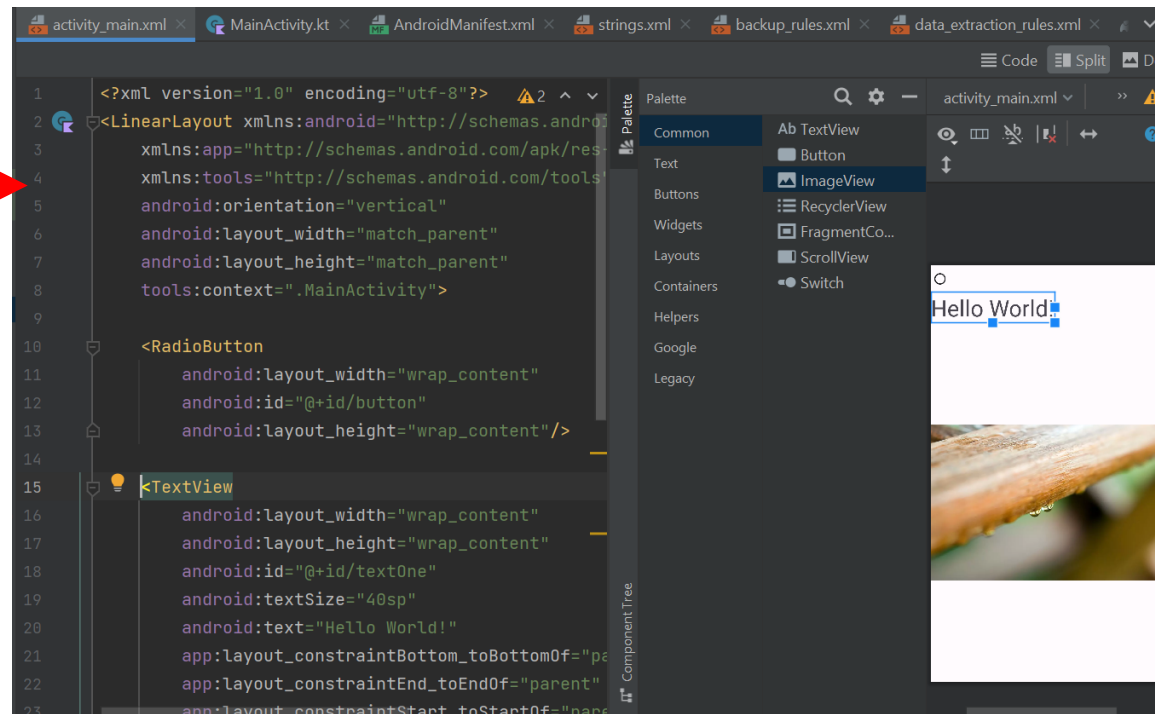
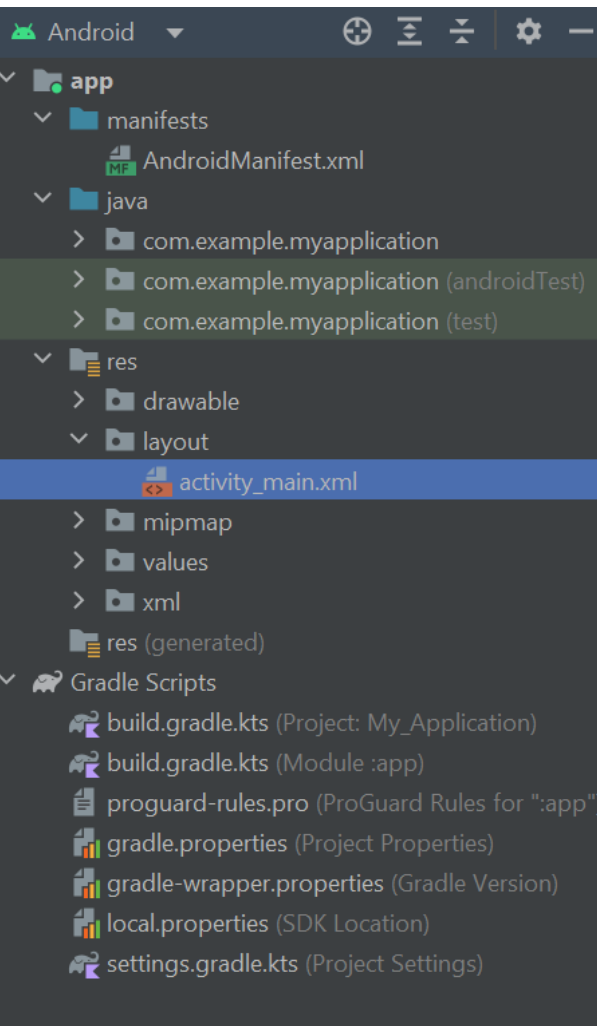
**Layout:** xml file (activity)

**Mipmap:** icone

**Values:** String, Colors, Themes

**Xml:** menu, autre options..

# Mon app



# Mon app



Manifest

The diagram illustrates the components of an Android application. It consists of three rounded square boxes arranged horizontally. The first box on the left is blue and labeled 'Manifest'. The middle box is red and labeled 'Java'. The third box on the right is blue and labeled 'Res'. The red 'Java' box is slightly larger and has a white border, making it stand out as the central component.

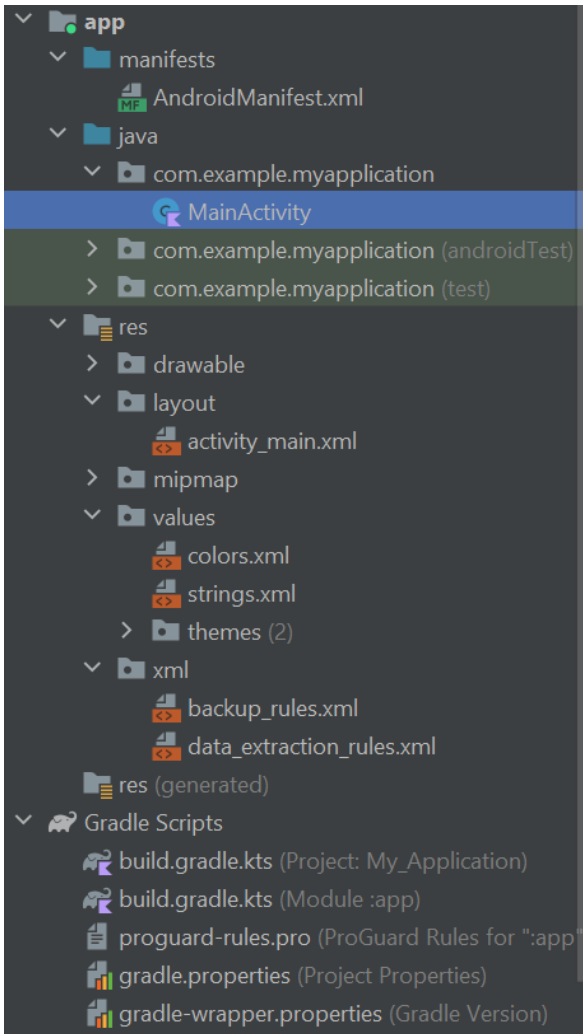
Java

Res

# Mon app



# Mon app



```
package com.example.myapplication

import ...

class MainActivity : AppCompatActivity() {

    @SuppressWarnings("MissingInflatedId")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

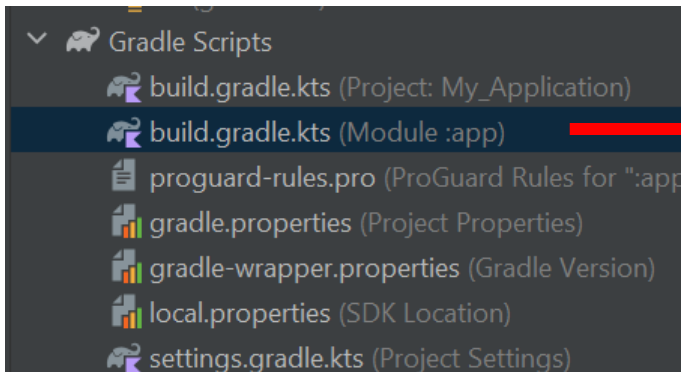
        val text = findViewById<TextView>(R.id.textOne)
        val radio = findViewById<RadioButton>(R.id.button)

        radio.isEnabled = true

        text.setOnClickListener { it: View!
            text.text = "I have click"
        }
    }
}
```

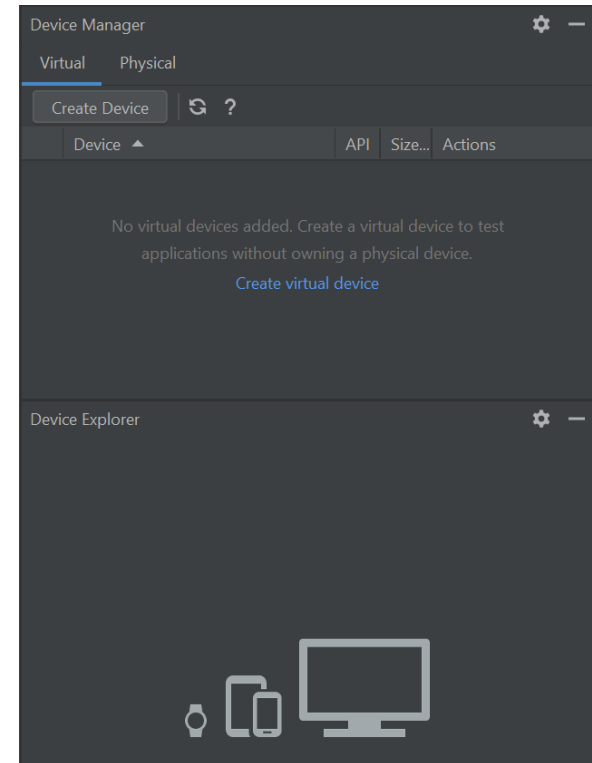


# Gradle Script

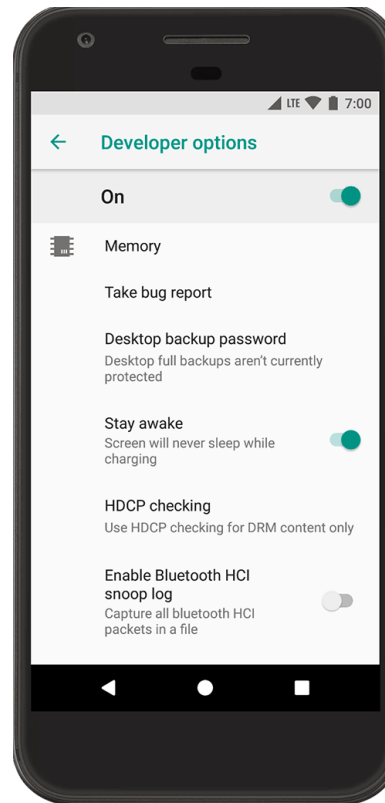


```
plugins {  
    id("com.android.application")  
    id("org.jetbrains.kotlin.android")  
}  
  
android {  
    namespace = "com.example.myapplication"  
    compileSdk = 33  
  
    defaultConfig {  
        applicationId = "com.example.myapplication"  
        minSdk = 24  
        targetSdk = 33  
        versionCode = 1  
        versionName = "1.0"  
  
        testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner"  
    }  
  
    buildTypes {  
        release {  
            isMinifyEnabled = false  
            proguardFiles(getDefaultProguardFile("proguard-android-optimize.txt"))  
        }  
    }  
}
```

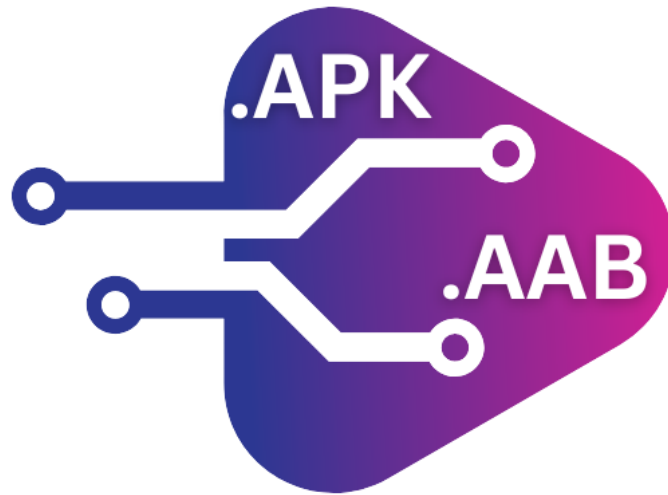
# Créer et gérer des appareils virtuels



# Configurer les options pour les développeurs sur l'appareil



# Génération de l'APK (Android application package)



Android App Bundle (AAB)

# Utiliser Github: Pratiques CI/CD



# Thank you !

Questions ? [abdelkader.ouared@univ-tiaret.dz](mailto:abdelkader.ouared@univ-tiaret.dz)

