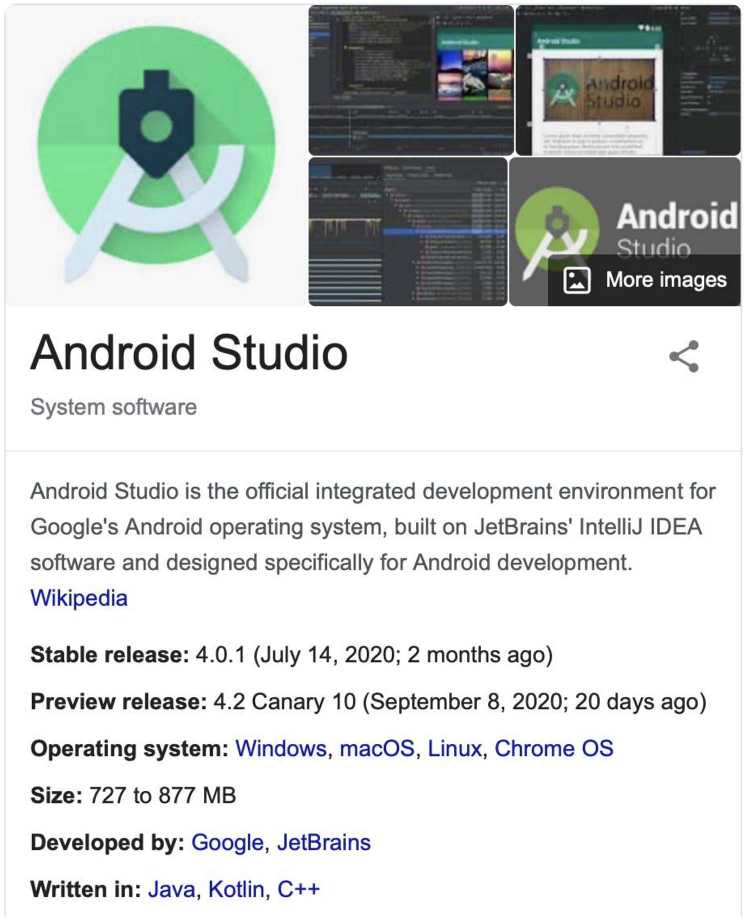
COM4510/6510
Software Development for Mobile Devices

Lab 1: Creating your first app

Dr Po Yang
The University of Sheffield
po.yang@sheffield.ac.uk

Creating your first App

- The default IDE is AndroidStudio
  - https://developer.android.com/studio/index.html
- We can use the lab computers
- You can use your own computer (PC or MAC)
- Open Android Studio NOW!
  - it will take a lot to load



Android Studio

System software

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Wikipedia

Stable release: 4.0.1 (July 14, 2020; 2 months ago)

Preview release: 4.2 Canary 10 (September 8, 2020; 20 days ago)

Operating system: Windows, macOS, Linux, Chrome OS

Size: 727 to 877 MB

Developed by: Google, JetBrains

Written in: Java, Kotlin, C++

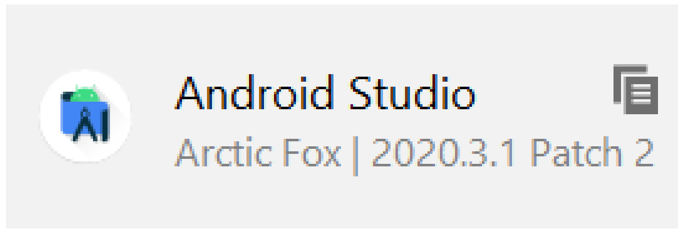
Set Up

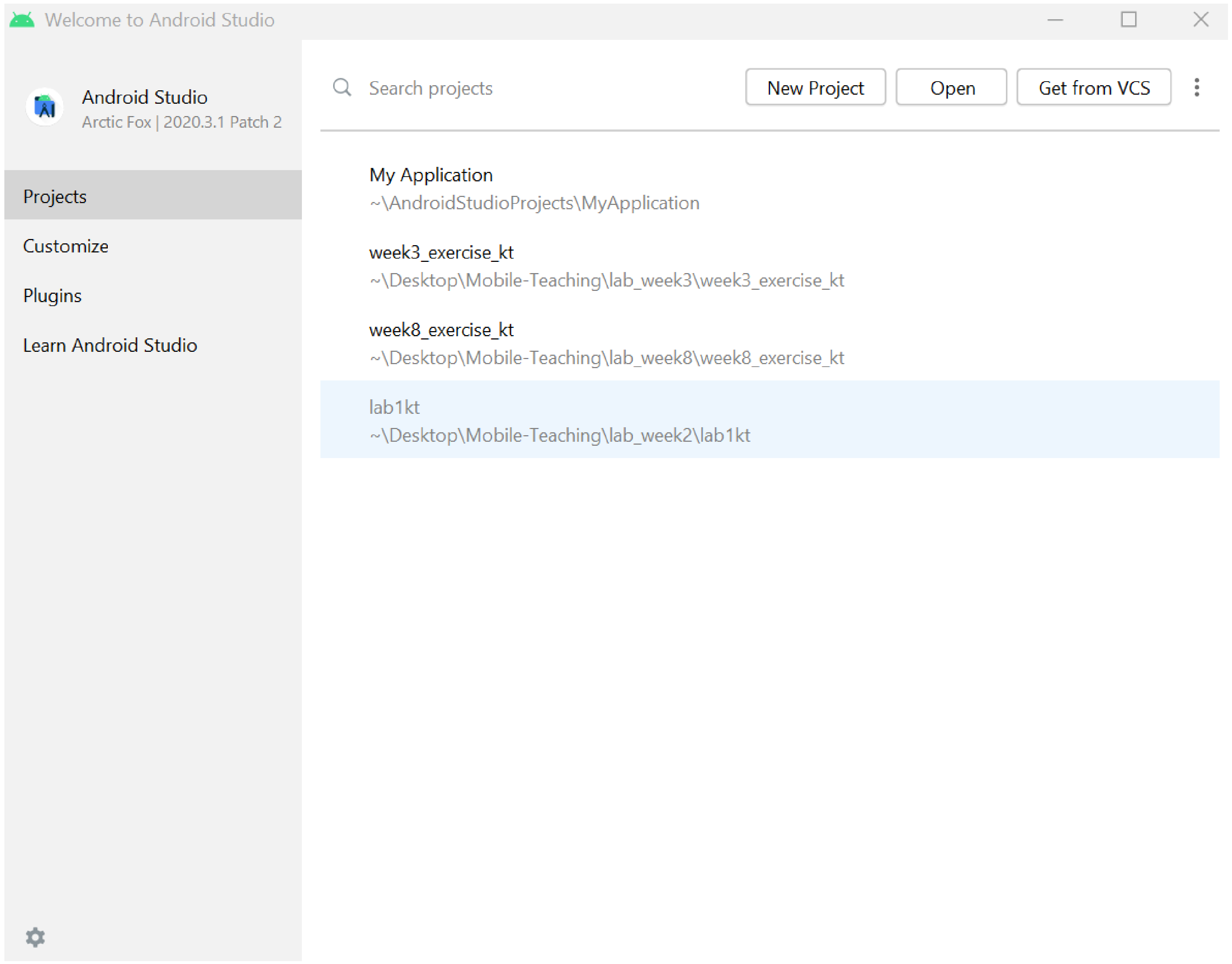
- It will take quite some time to load
- make sure to come well in advance to get the laptop and to open Android Studio
- Accept all standard settings, e.g.
  - *I do not have a version of Android Studio installed*
  - click next next until it start downloading components

The University Of Sheffield.

Click until…

- Arctic Fox : 2020. 3. 1


Android Studio
Arctic Fox | 2020.3.1 Patch 2


Welcome to Android Studio

Android Studio
Arctic Fox | 2020.3.1 Patch 2

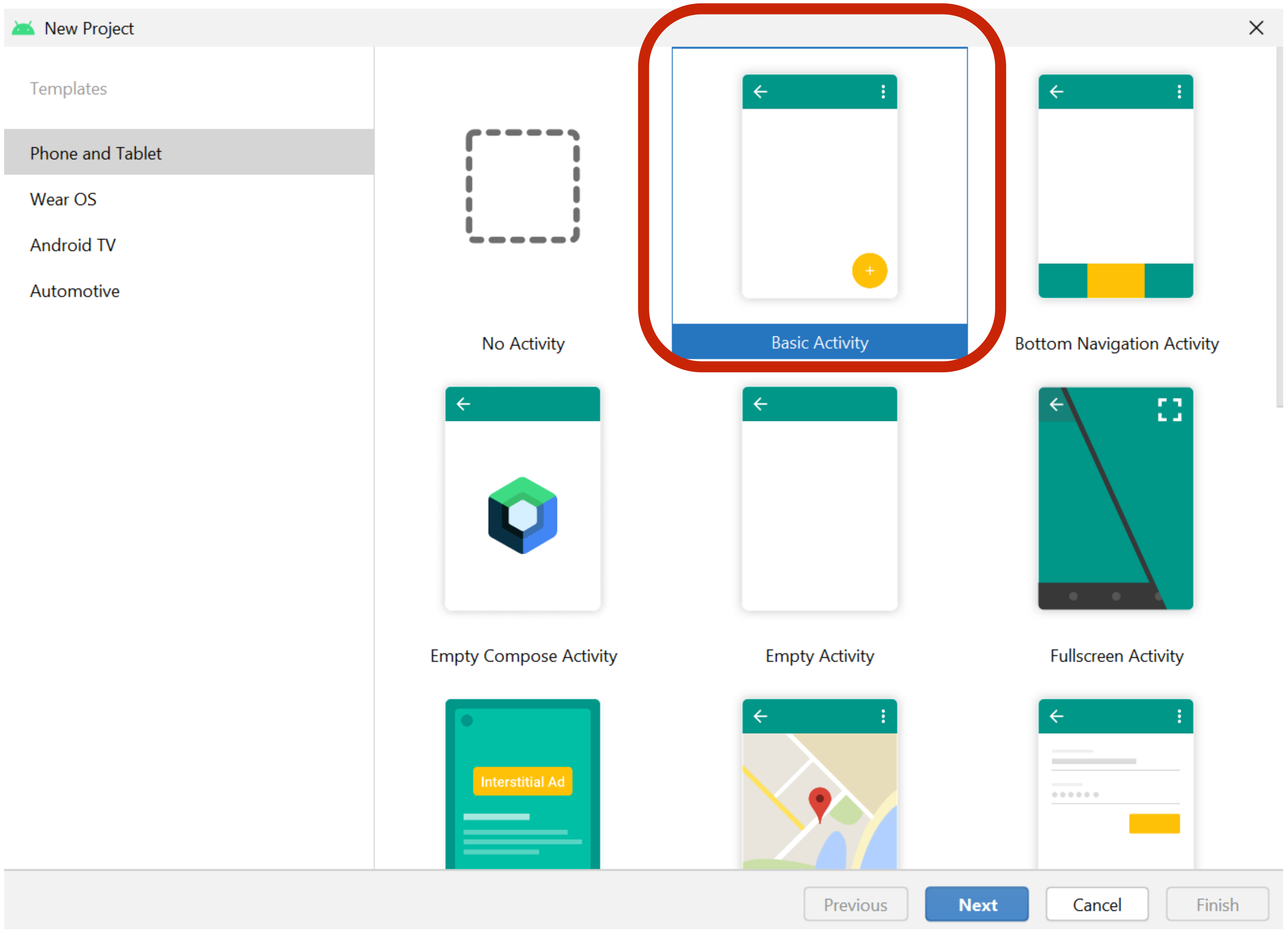
My Application
~\AndroidStudioProjects\MyApplication

week3_exercise_kt
~\Desktop\Mobile-Teaching\lab_week3\week3_exercise_kt

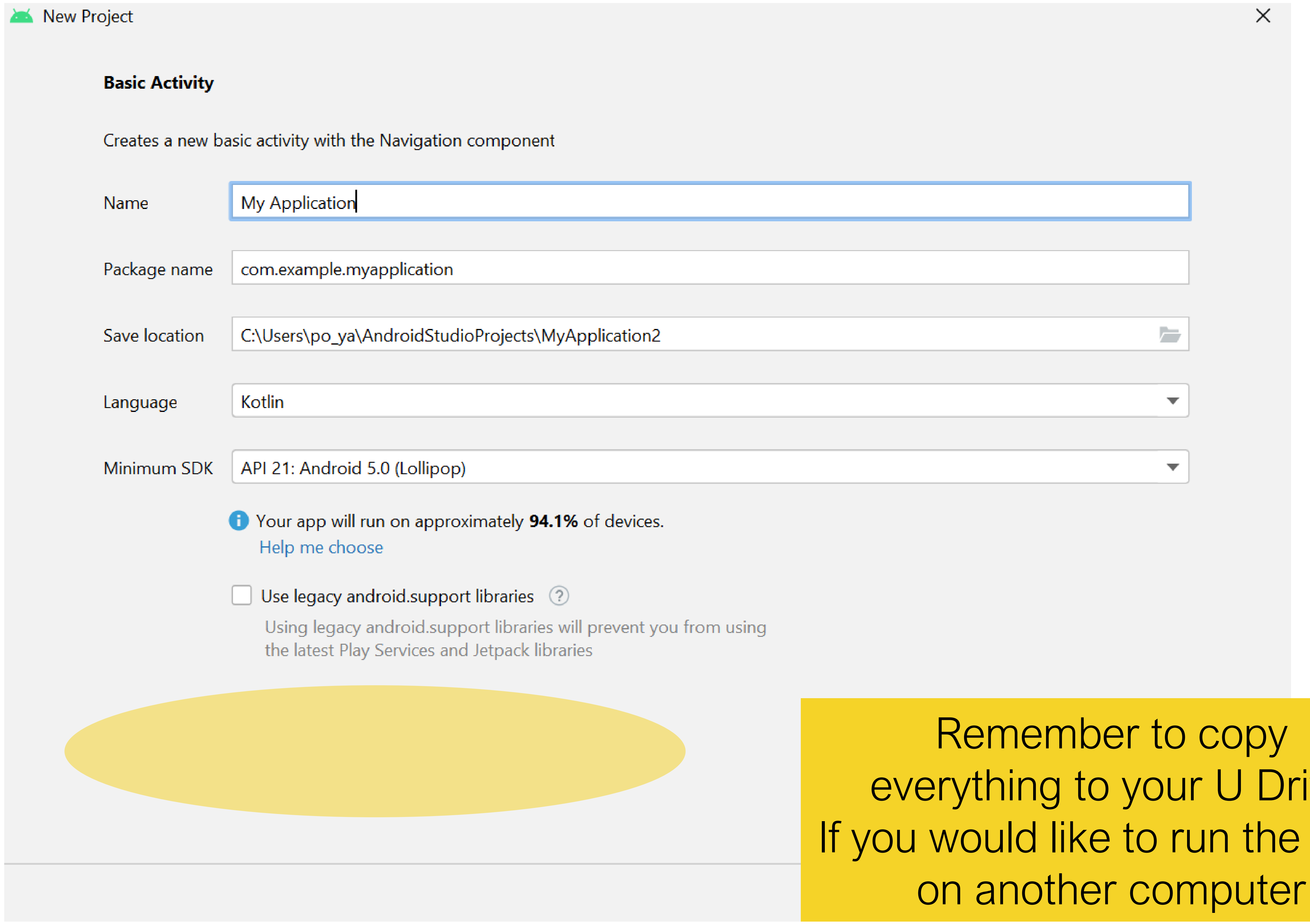
week8_exercise_kt
~\Desktop\Mobile-Teaching\lab_week8\week8_exercise_kt

lab1kt
~\Desktop\Mobile-Teaching\lab_week2\lab1kt

- Continue and select the first app



Configuration



New Project

Basic Activity

Creates a new basic activity with the Navigation component

Name	My Application
Package name	com.example.myapplication
Save location	C:\Users\po_ya\AndroidStudioProjects\MyApplication2
Language	Kotlin
Minimum SDK	API 21: Android 5.0 (Lollipop)

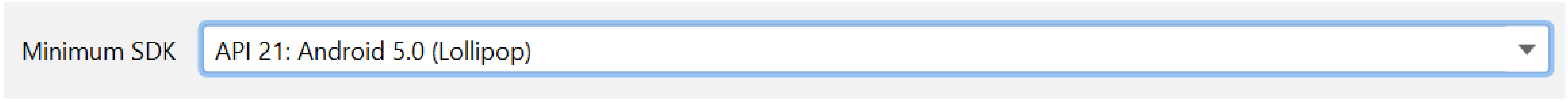
Your app will run on approximately **94.1%** of devices.
Help me choose

☐ Use legacy android.support libraries

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

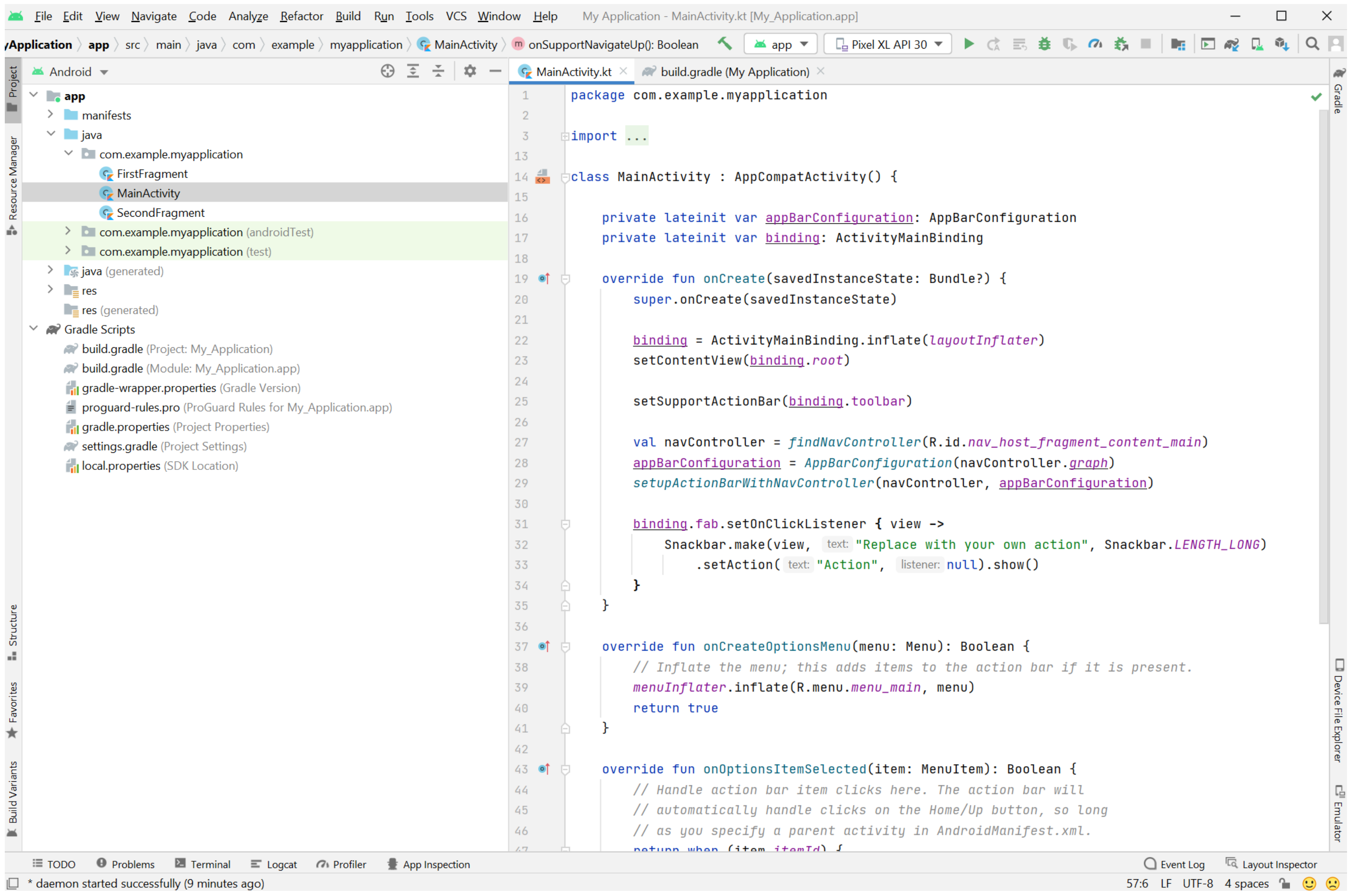
Remember to copy everything to your U Drive
If you would like to run the code on another computer

Configuration


Minimum SDK: API 21: Android 5.0 (Lollipop)

The first time you create an app

- It will take a while to set up
  - Look at the bottom of the screen
    - there will be Gradle and indexing going on
      - let them finish



File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help My Application - MainActivity.kt [My_Application.app]

Application 〉 app 〉 src 〉 main 〉 java 〉 com 〉 example 〉 myapplication 〉 MainActivity 〉 onSupportNavigateUp(): Boolean | app | Pixel XL API 30

Android

```
app
  manifests
  java
    com.example.myapplication
      FirstFragment
      MainActivity
      SecondFragment
    com.example.myapplication (androidTest)
    com.example.myapplication (test)
  java (generated)
  res
  res (generated)
Gradle Scripts
  build.gradle (Project: My_Application)
  build.gradle (Module: My_Application.app)
  gradle-wrapper.properties (Gradle Version)
  proguard-rules.pro (ProGuard Rules for My_Application.app)
  gradle.properties (Project Properties)
  settings.gradle (Project Settings)
  local.properties (SDK Location)
```

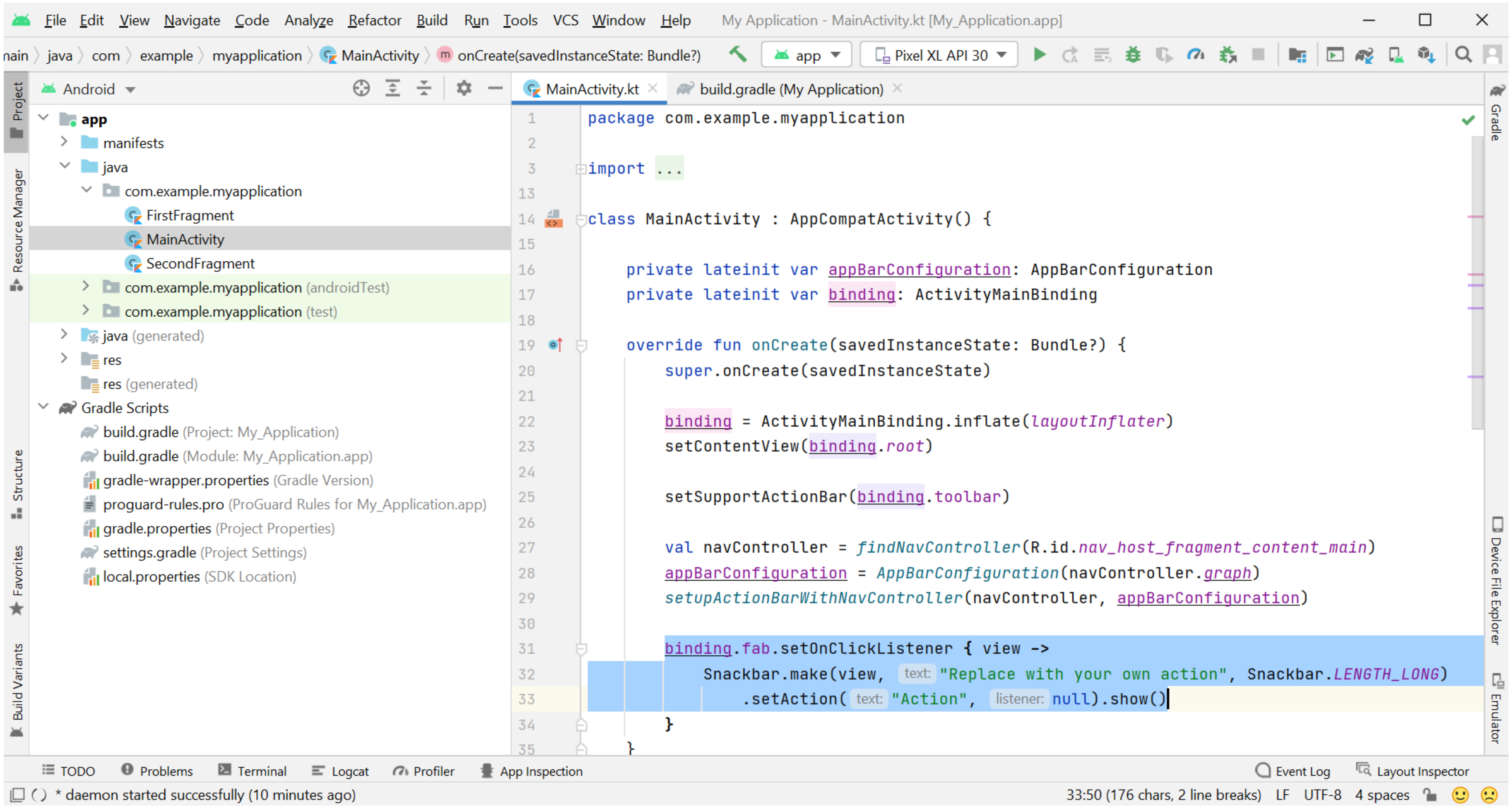
MainActivity.kt | build.gradle (My Application)

```kotlin
package com.example.myapplication

import ...

class MainActivity : AppCompatActivity() {

    private lateinit var appBarConfiguration: AppBarConfiguration
    private lateinit var binding: ActivityMainBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)

        val navController = findNavController(R.id.nav_host_fragment_content_main)
        appBarConfiguration = AppBarConfiguration(navController.graph)
        setupActionBarWithNavController(navController, appBarConfiguration)

        binding.fab.setOnClickListener { view ->
            Snackbar.make(view, text: "Replace with your own action", Snackbar.LENGTH_LONG)
                .setAction( text: "Action",  listener: null).show()
        }
    }

    override fun onCreateOptionsMenu(menu: Menu): Boolean {
        // Inflate the menu; this adds items to the action bar if it is present.
        menuInflater.inflate(R.menu.menu_main, menu)
        return true
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        return when (item.itemId) {
```

TODO | Problems | Terminal | Logcat | Profiler | App Inspection | Event Log | Layout Inspector

* daemon started successfully (9 minutes ago) 57:6 LF UTF-8 4 spaces

Your first app



File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
My Application - MainActivity.kt [My_Application.app]
main › java › com › example › myapplication › MainActivity › onCreate(savedInstanceState: Bundle?)
app
Pixel XL API 30
Android
MainActivity.kt
build.gradle (My Application)

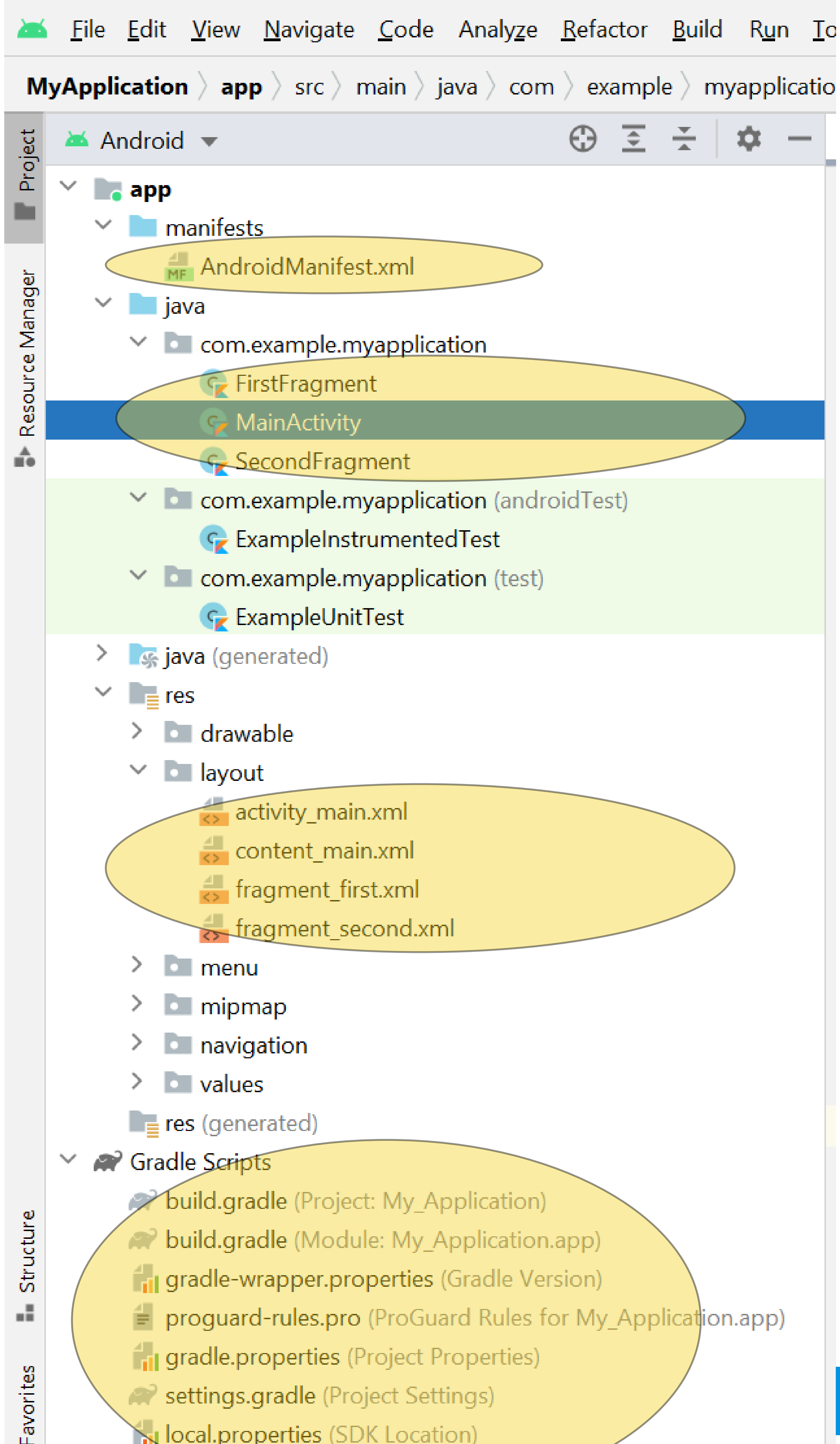
```
app
  manifests
  java
    com.example.myapplication
      FirstFragment
      MainActivity
      SecondFragment
    com.example.myapplication (androidTest)
    com.example.myapplication (test)
  java (generated)
  res
  res (generated)
Gradle Scripts
  build.gradle (Project: My_Application)
  build.gradle (Module: My_Application.app)
  gradle-wrapper.properties (Gradle Version)
  proguard-rules.pro (ProGuard Rules for My_Application.app)
  gradle.properties (Project Properties)
  settings.gradle (Project Settings)
  local.properties (SDK Location)
```

```kotlin
package com.example.myapplication

import ...

class MainActivity : AppCompatActivity() {

    private lateinit var appBarConfiguration: AppBarConfiguration
    private lateinit var binding: ActivityMainBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)

        val navController = findNavController(R.id.nav_host_fragment_content_main)
        appBarConfiguration = AppBarConfiguration(navController.graph)
        setupActionBarWithNavController(navController, appBarConfiguration)

        binding.fab.setOnClickListener { view ->
            Snackbar.make(view, text: "Replace with your own action", Snackbar.LENGTH_LONG)
                .setAction( text: "Action", listener: null).show()
        }
    }
```

```
* daemon started successfully (10 minutes ago)
33:50 (176 chars, 2 line breaks)  LF  UTF-8  4 spaces
```

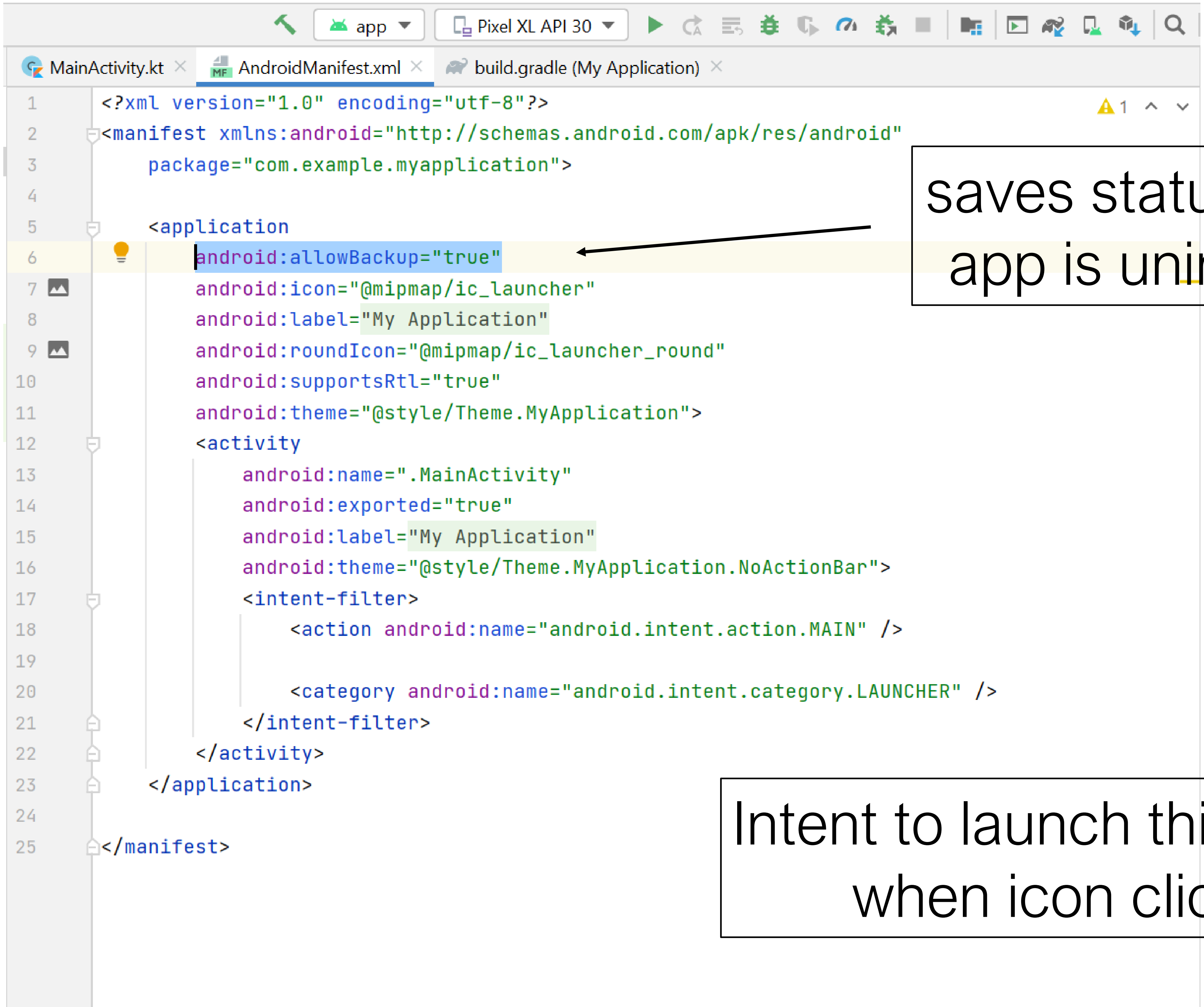
Please note that AndroiStudio presents a virtual view of the file space. For example the folder `java` is not directly under the folder `app`: it will be `app/src/main/layout`. The folder res is `app/src/res/main`. The cradle files are instead under `app/`

You do not need to care about that unless you want to access the files from via the file system



```
File  Edit  View  Navigate  Code  Analyze  Refactor  Build  Run  To
MyApplication 〉 app 〉 src 〉 main 〉 java 〉 com 〉 example 〉 myapplicatio
Android
app
  manifests
    AndroidManifest.xml
  java
    com.example.myapplication
      FirstFragment
      MainActivity
      SecondFragment
    com.example.myapplication (androidTest)
      ExampleInstrumentedTest
    com.example.myapplication (test)
      ExampleUnitTest
  java (generated)
  res
    drawable
    layout
      activity_main.xml
      content_main.xml
      fragment_first.xml
      fragment_second.xml
    menu
    mipmap
    navigation
    values
  res (generated)
Gradle Scripts
  build.gradle (Project: My_Application)
  build.gradle (Module: My_Application.app)
  gradle-wrapper.properties (Gradle Version)
  proguard-rules.pro (ProGuard Rules for My_Application.app)
  gradle.properties (Project Properties)
  settings.gradle (Project Settings)
  local.properties (SDK Location)
```

Manifest

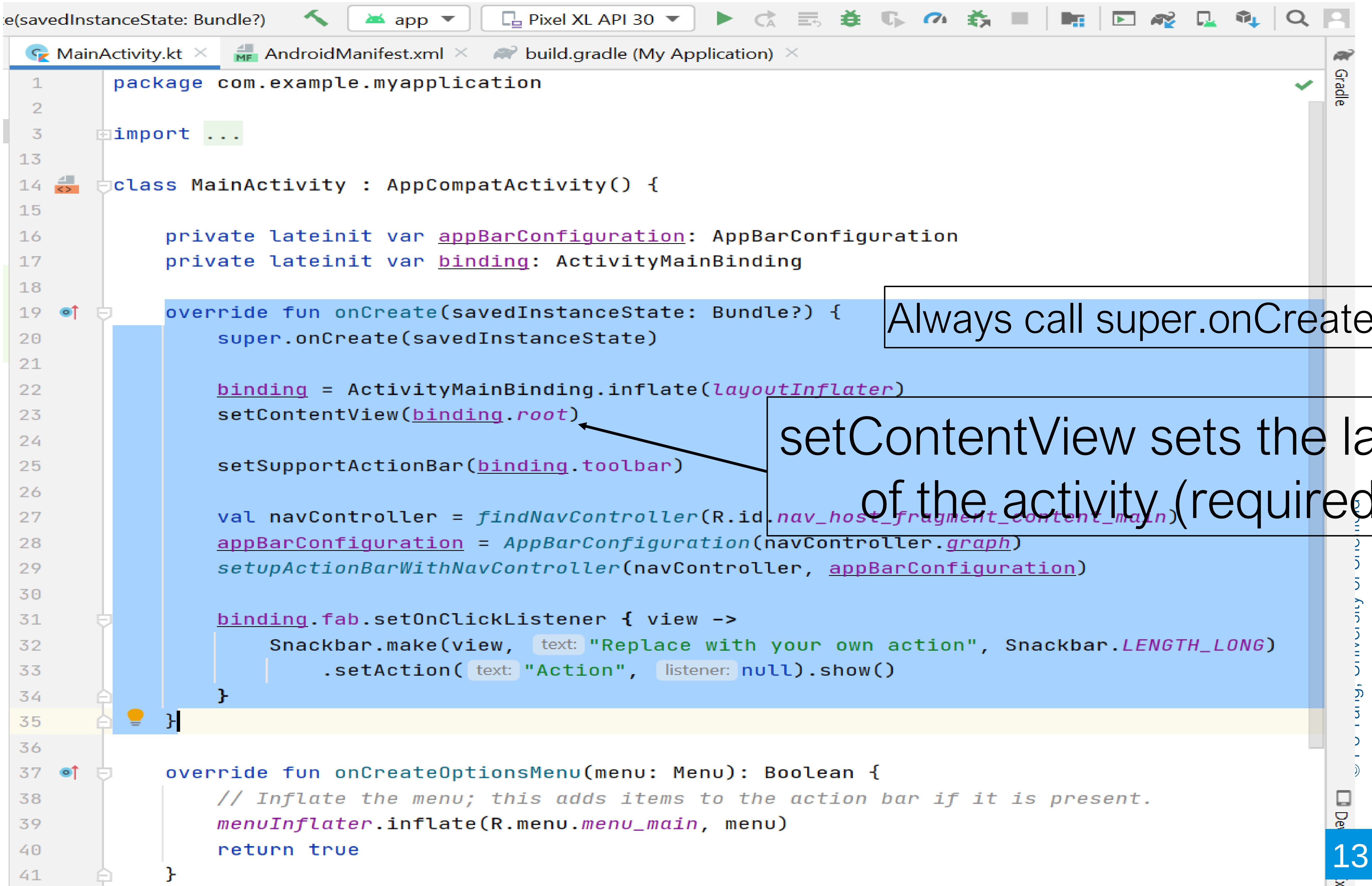


```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="My Application"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.MyApplication">
        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:label="My Application"
            android:theme="@style/Theme.MyApplication.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

saves status when app is uninstalled

Intent to launch this activity when icon clicked

MainActivity



e(savedInstanceState: Bundle?) app Pixel XL API 30

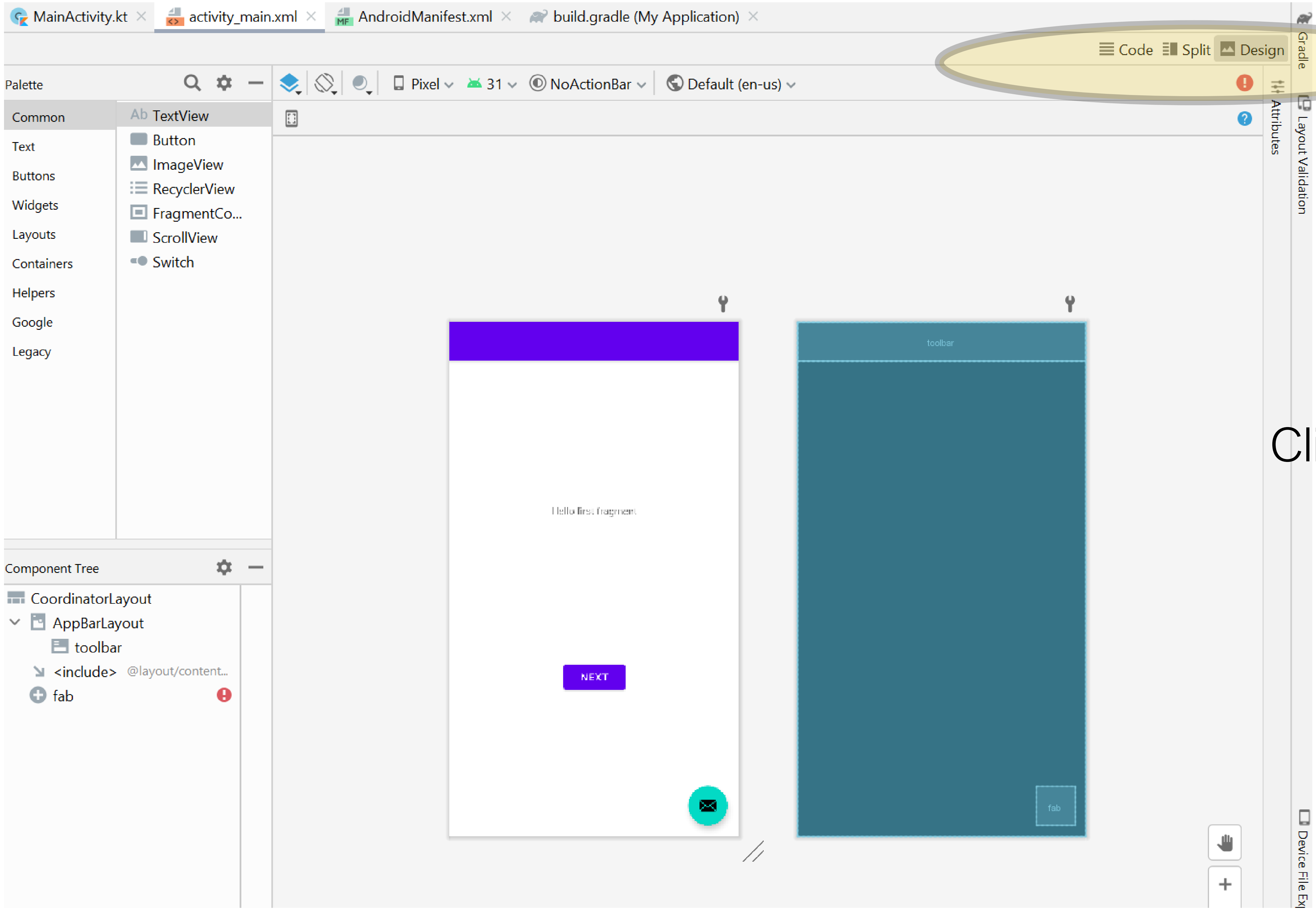
MainActivity.kt AndroidManifest.xml build.gradle (My Application)

```
package com.example.myapplication

import ...

class MainActivity : AppCompatActivity() {

    private lateinit var appBarConfiguration: AppBarConfiguration
    private lateinit var binding: ActivityMainBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setSupportActionBar(binding.toolbar)

        val navController = findNavController(R.id.nav_host_fragment_content_main)
        appBarConfiguration = AppBarConfiguration(navController.graph)
        setupActionBarWithNavController(navController, appBarConfiguration)

        binding.fab.setOnClickListener { view ->
            Snackbar.make(view, text: "Replace with your own action", Snackbar.LENGTH_LONG)
                .setAction( text: "Action", listener: null).show()
        }
    }

    override fun onCreateOptionsMenu(menu: Menu): Boolean {
        // Inflate the menu; this adds items to the action bar if it is present.
        menuInflater.inflate(R.menu.menu_main, menu)
        return true
    }
```

Always call super.onCreate

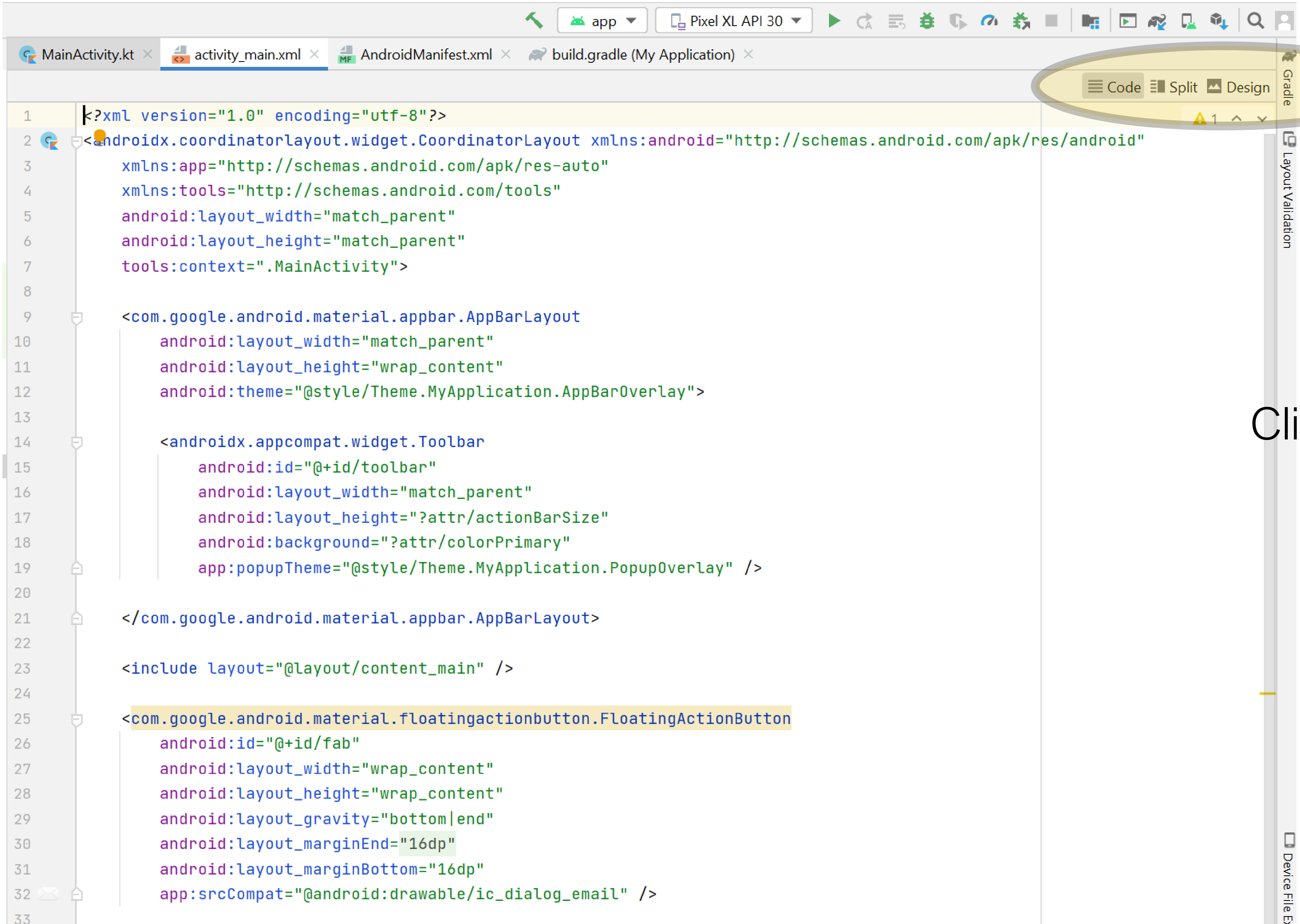
setContentView sets the la
of the activity (required

Layout - design view



Click here

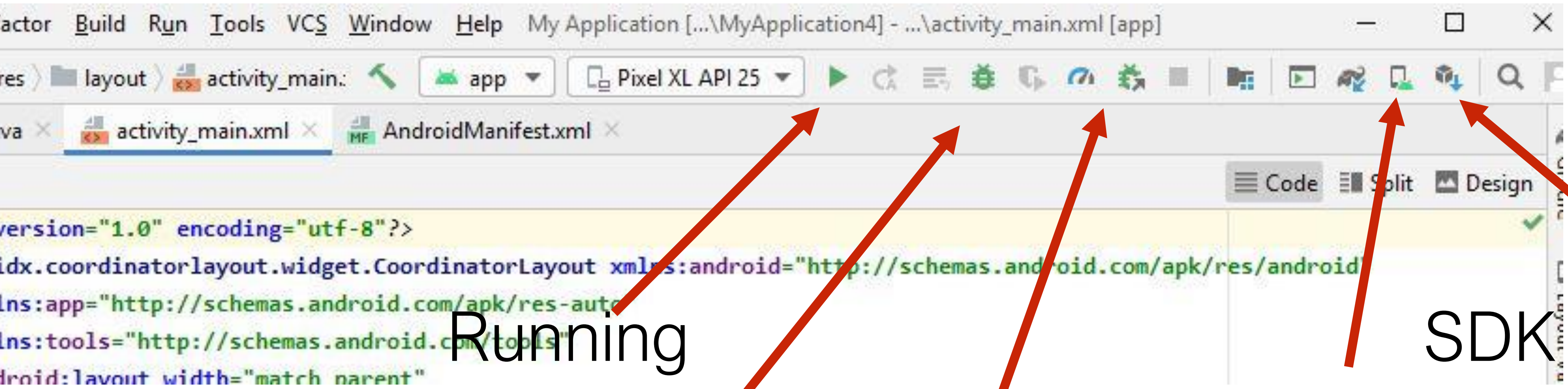
Layout — Text View



```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay" />

    </com.google.android.material.appbar.AppBarLayout>

    <include layout="@layout/content_main" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|end"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        app:srcCompat="@android:drawable/ic_dialog_email" />
```

Click here

Running and Debugging


actor Build Run Tools VCS Window Help My Application [...\MyApplication4] - ...\activity_main.xml [app]
res layout activity_main.
app
Pixel XL API 25
va activity_main.xml AndroidManifest.xml
Code Split Design

```
version="1.0" encoding="utf-8"?>
idx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
lns:app="http://schemas.android.com/apk/res-auto"
lns:tools="http://schemas.android.com/tools"
droid:layout_width="match_parent"
```

Running

Debugging

Attach debugger
to Android Process

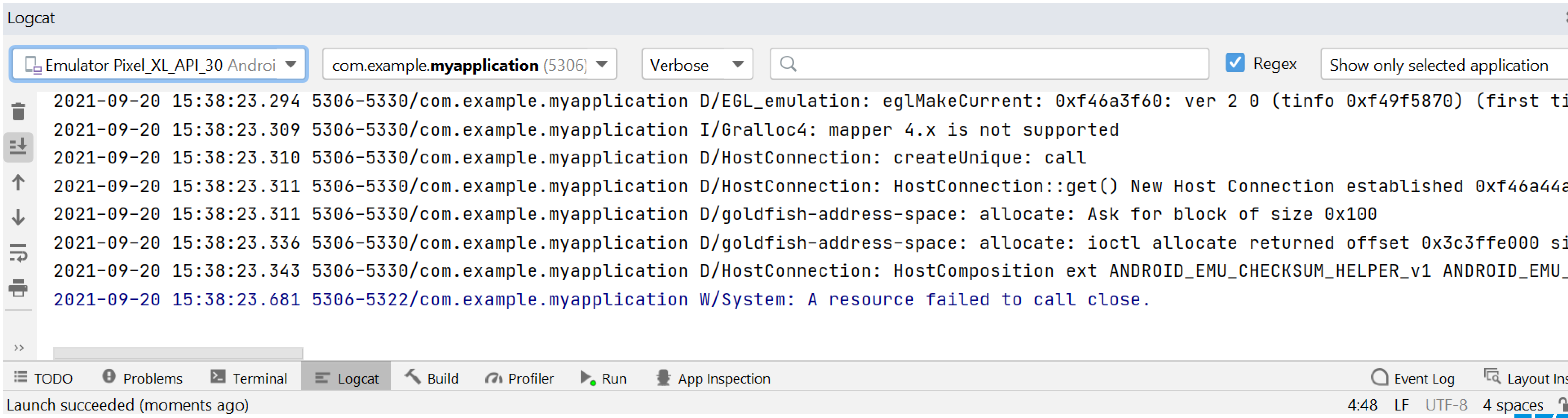
SDK Manager

AVD Manager

The University Of Sheffield.

Running

- Will require an AVD or real device to run on (see later)
- Android Monitor tab at the bottom will show the logs
  - You can use Log.d/w/e/i (TAG, String); to log there (as you would do System.out.println in Java)

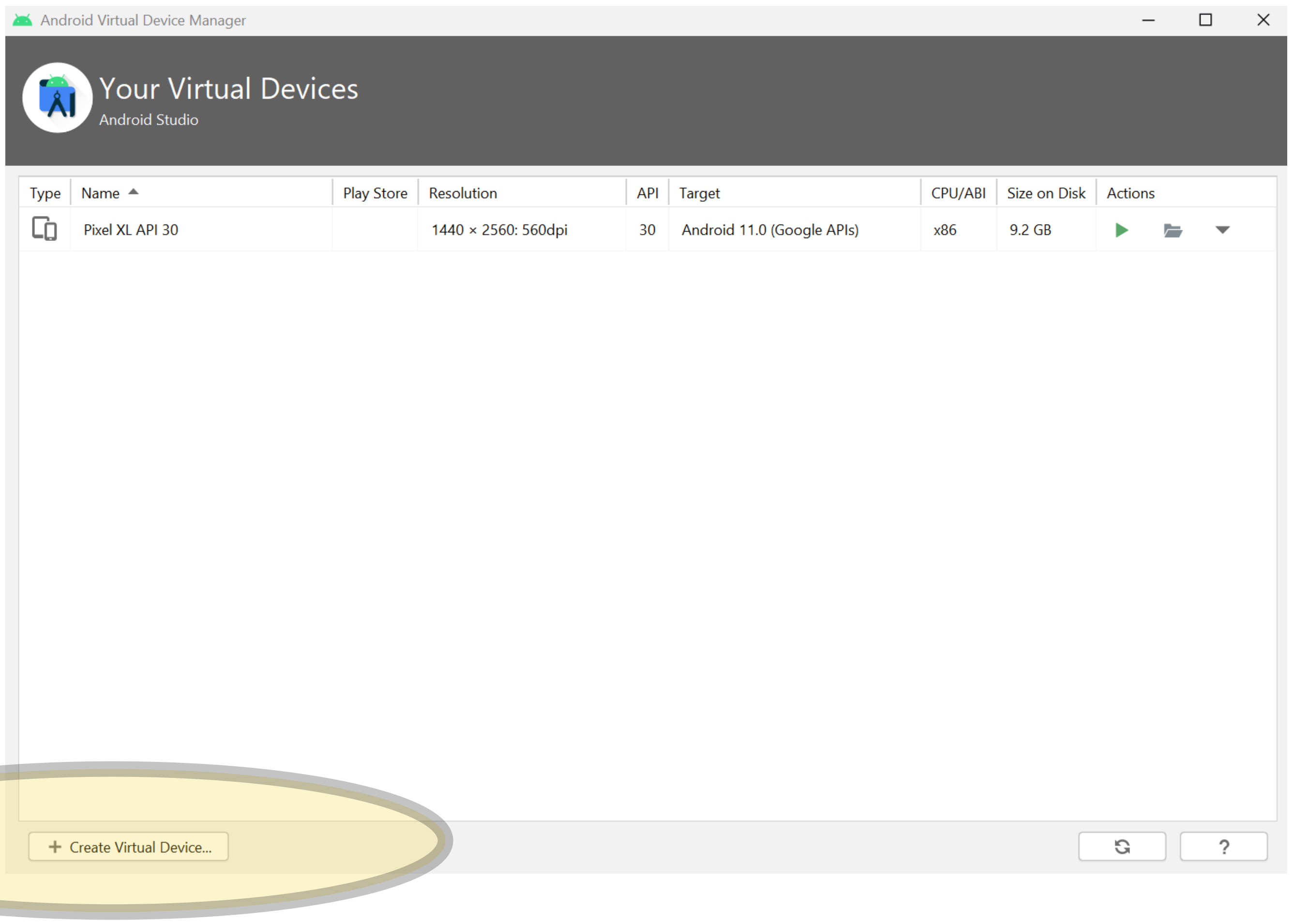


```
2021-09-20 15:38:23.294 5306-5330/com.example.myapplication D/EGL_emulation: eglMakeCurrent: 0xf46a3f60: ver 2 0 (tinfo 0xf49f5870) (first ti
2021-09-20 15:38:23.309 5306-5330/com.example.myapplication I/Gralloc4: mapper 4.x is not supported
2021-09-20 15:38:23.310 5306-5330/com.example.myapplication D/HostConnection: createUnique: call
2021-09-20 15:38:23.311 5306-5330/com.example.myapplication D/HostConnection: HostConnection::get() New Host Connection established 0xf46a44a
2021-09-20 15:38:23.311 5306-5330/com.example.myapplication D/goldfish-address-space: allocate: Ask for block of size 0x100
2021-09-20 15:38:23.336 5306-5330/com.example.myapplication D/goldfish-address-space: allocate: ioctl allocate returned offset 0x3c3ffe000 si
2021-09-20 15:38:23.343 5306-5330/com.example.myapplication D/HostConnection: HostComposition ext ANDROID_EMU_CHECKSUM_HELPER_v1 ANDROID_EMU_
2021-09-20 15:38:23.681 5306-5322/com.example.myapplication W/System: A resource failed to call close.
```

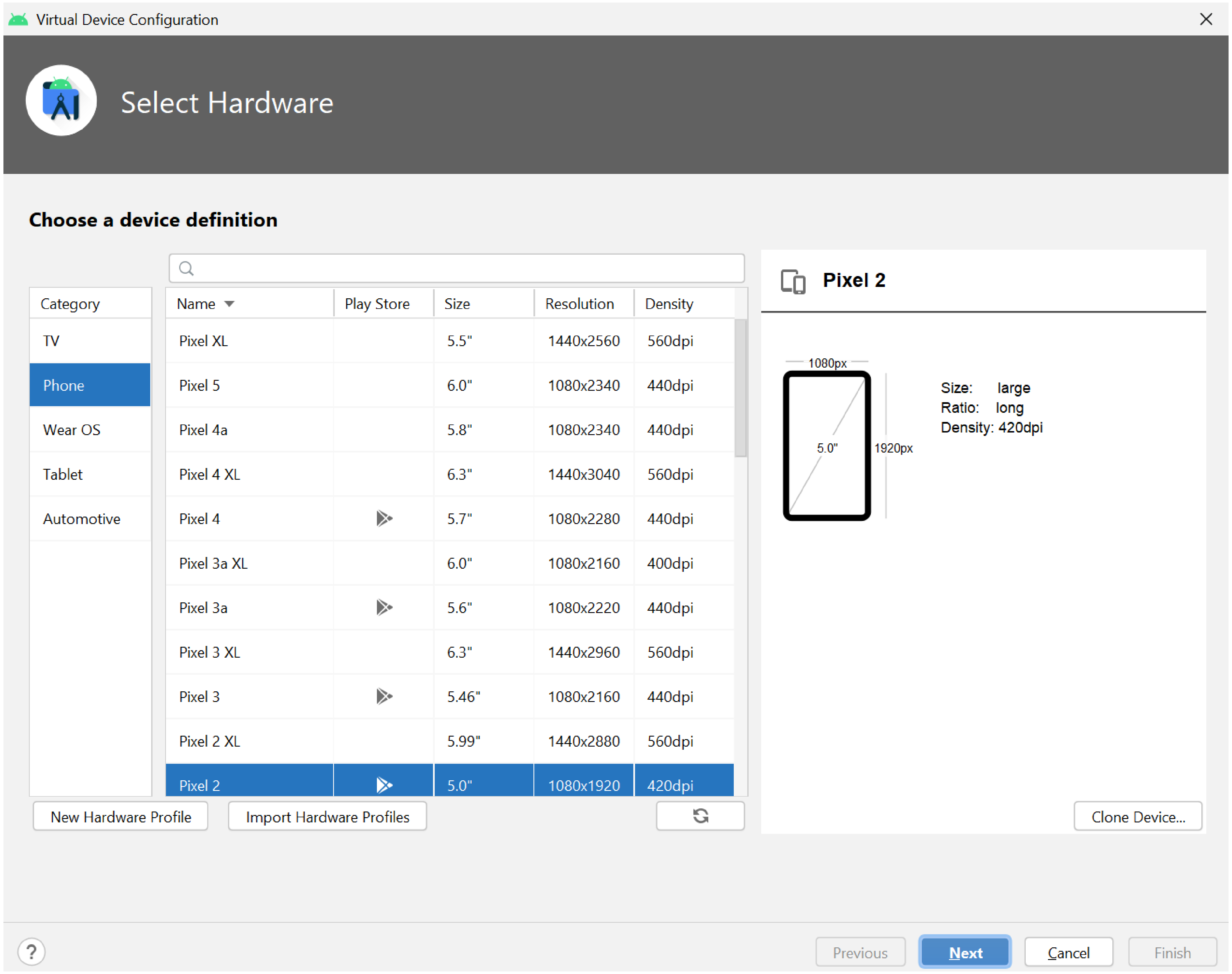
Launch succeeded (moments ago)

Creating an AVD

- Android Virtual Device (AVD)
- Your code can either work on a real device or you can use an AVD
  - click on AVD button or click run and the following menu will appear
  - click on "Create New Virtual Device"



Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Pixel XL API 30		1440 × 2560: 560dpi	30	Android 11.0 (Google APIs)	x86	9.2 GB	



Virtual Device Configuration

Select Hardware

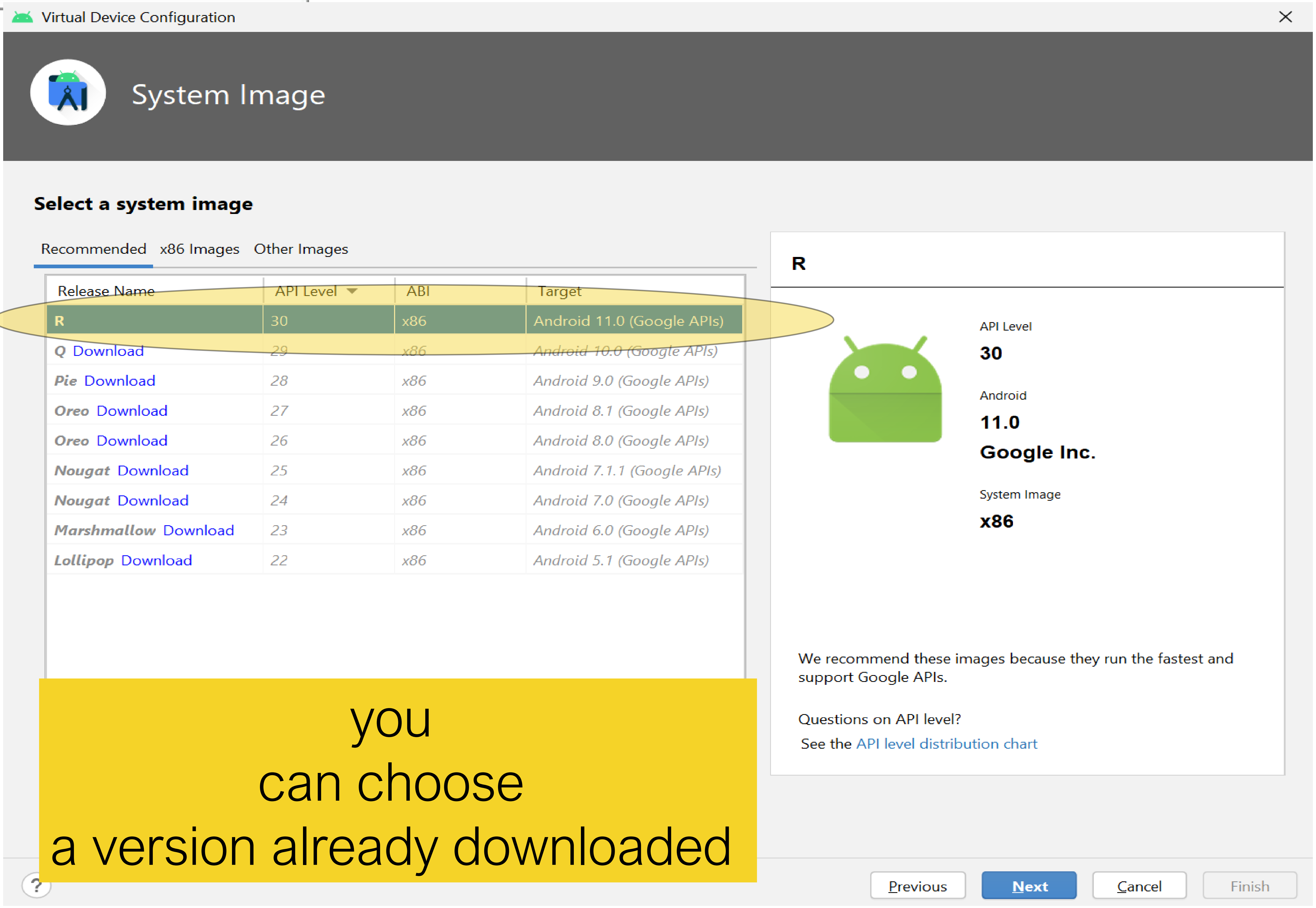
Choose a device definition

Category
TV
Phone
Wear OS
Tablet
Automotive

Name	Play Store	Size	Resolution	Density
Pixel XL		5.5"	1440x2560	560dpi
Pixel 5		6.0"	1080x2340	440dpi
Pixel 4a		5.8"	1080x2340	440dpi
Pixel 4 XL		6.3"	1440x3040	560dpi
Pixel 4	▶	5.7"	1080x2280	440dpi
Pixel 3a XL		6.0"	1080x2160	400dpi
Pixel 3a	▶	5.6"	1080x2220	440dpi
Pixel 3 XL		6.3"	1440x2960	560dpi
Pixel 3	▶	5.46"	1080x2160	440dpi
Pixel 2 XL		5.99"	1440x2880	560dpi
Pixel 2	▶	5.0"	1080x1920	420dpi

Pixel 2

Size: large
Ratio: long
Density: 420dpi



Virtual Device Configuration

System Image

Select a system image

Recommended | x86 Images | Other Images

Release Name	API Level	ABI	Target
R	30	x86	Android 11.0 (Google APIs)
Q Download	*29*	*x86*	*Android 10.0 (Google APIs)*
Pie Download	*28*	*x86*	*Android 9.0 (Google APIs)*
Oreo Download	*27*	*x86*	*Android 8.1 (Google APIs)*
Oreo Download	*26*	*x86*	*Android 8.0 (Google APIs)*
Nougat Download	*25*	*x86*	*Android 7.1.1 (Google APIs)*
Nougat Download	*24*	*x86*	*Android 7.0 (Google APIs)*
Marshmallow Download	*23*	*x86*	*Android 6.0 (Google APIs)*
Lollipop Download	*22*	*x86*	*Android 5.1 (Google APIs)*

R

API Level
30

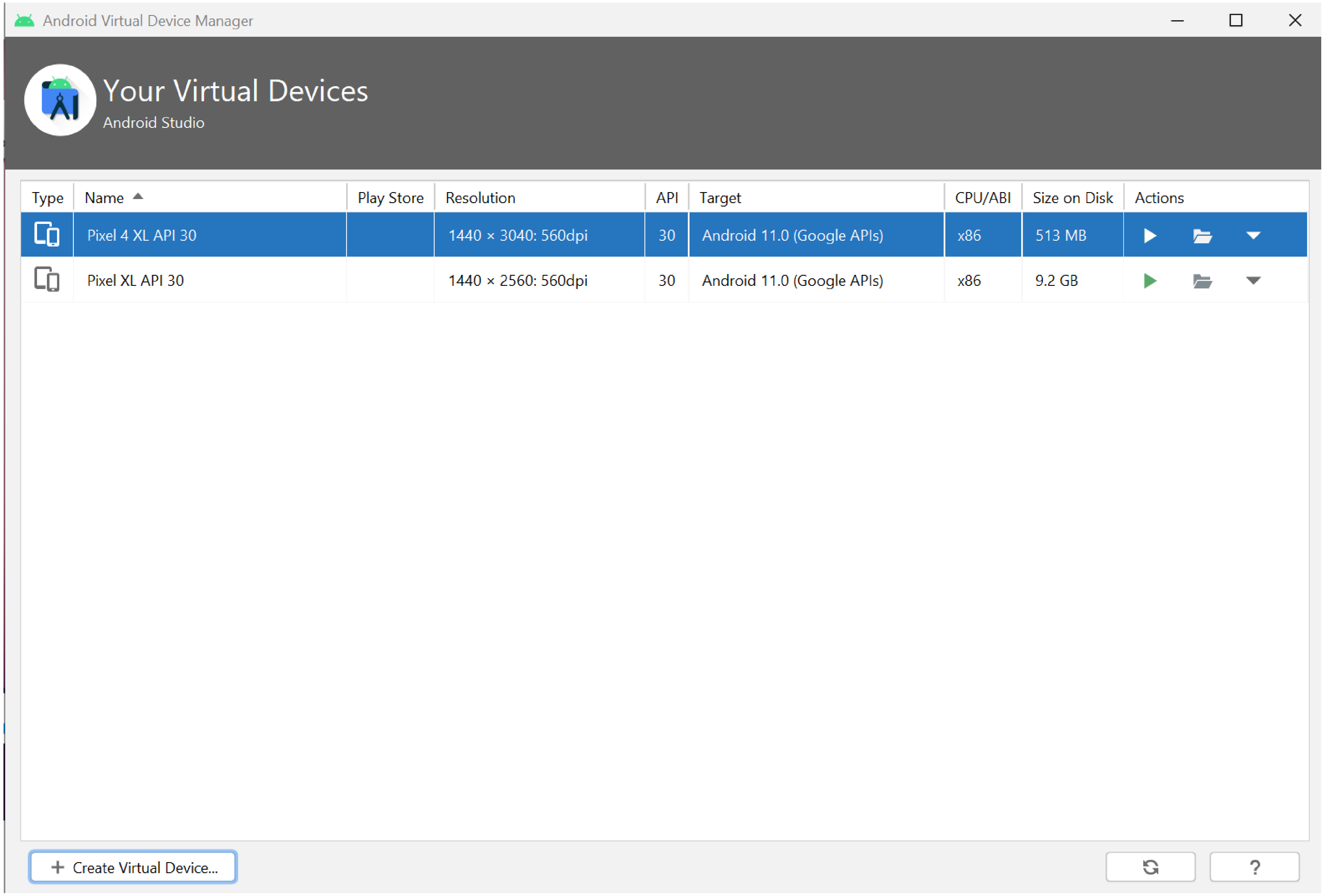
Android
11.0
Google Inc.

System Image
x86

We recommend these images because they run the fastest and support Google APIs.

Questions on API level?
See the API level distribution chart

you
can choose
a version already downloaded



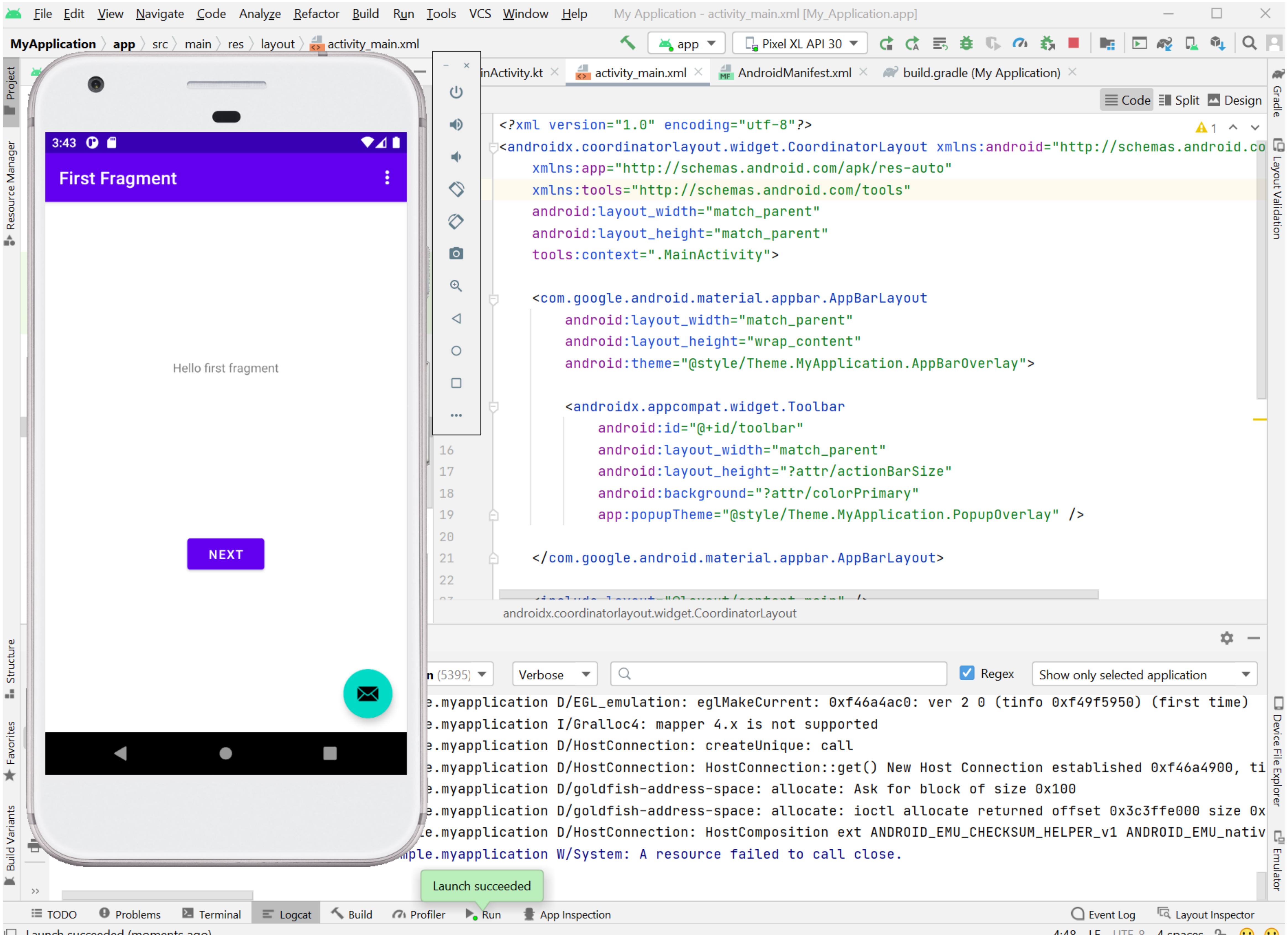
Android Virtual Device Manager

Your Virtual Devices
Android Studio

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Pixel 4 XL API 30		1440 × 3040: 560dpi	30	Android 11.0 (Google APIs)	x86	513 MB	
	Pixel XL API 30		1440 × 2560: 560dpi	30	Android 11.0 (Google APIs)	x86	9.2 GB	

The University Of Sheffield.

The AVD will boot like a normal Android device and then will show your app



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas.android.co
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/Theme.MyApplication.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/Theme.MyApplication.PopupOverlay" />

    </com.google.android.material.appbar.AppBarLayout>
```

```
e.myapplication D/EGL_emulation: eglMakeCurrent: 0xf46a4ac0: ver 2 0 (tinfo 0xf49f5950) (first time)
e.myapplication I/Gralloc4: mapper 4.x is not supported
e.myapplication D/HostConnection: createUnique: call
e.myapplication D/HostConnection: HostConnection::get() New Host Connection established 0xf46a4900, ti
e.myapplication D/goldfish-address-space: allocate: Ask for block of size 0x100
e.myapplication D/goldfish-address-space: allocate: ioctl allocate returned offset 0x3c3ffe000 size 0x
e.myapplication D/HostConnection: HostComposition ext ANDROID_EMU_CHECKSUM_HELPER_v1 ANDROID_EMU_nativ
mple.myapplication W/System: A resource failed to call close.
```

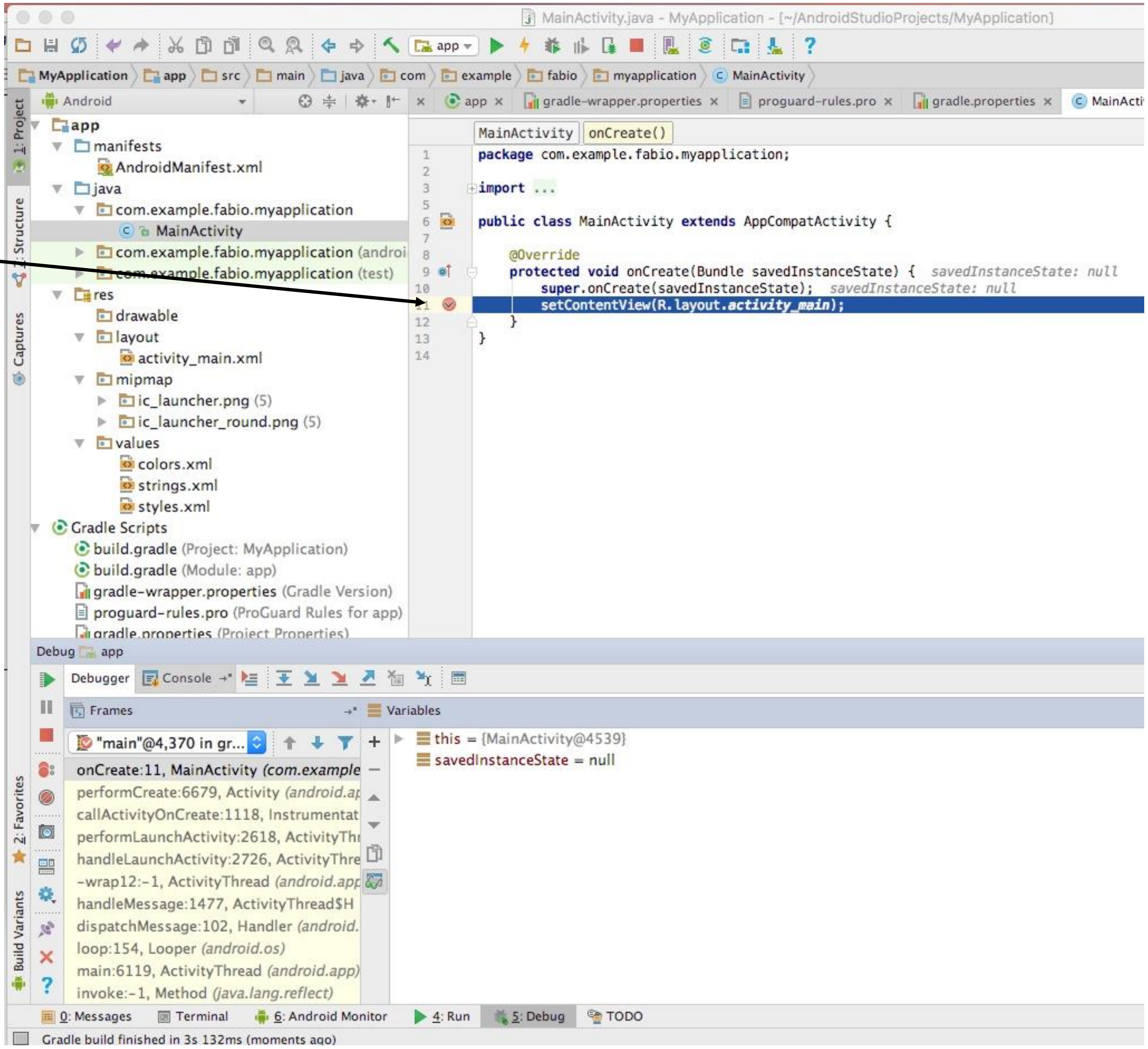
Launch succeeded (moments ago)

The University Of Sheffield.

Debugging

Set Break Points (Click here)

Inspect Stack Trace and variables



MainActivity.java - MyApplication - [~/AndroidStudioProjects/MyApplication]

MyApplication > app > src > main > java > com > example > fabio > myapplication > MainActivity

```
app
  manifests
    AndroidManifest.xml
  java
    com.example.fabio.myapplication
      MainActivity
    com.example.fabio.myapplication (androi
    com.example.fabio.myapplication (test)
  res
    drawable
    layout
      activity_main.xml
    mipmap
      ic_launcher.png (5)
      ic_launcher_round.png (5)
    values
      colors.xml
      strings.xml
      styles.xml
Gradle Scripts
  build.gradle (Project: MyApplication)
  build.gradle (Module: app)
  gradle-wrapper.properties (Gradle Version)
  proguard-rules.pro (ProGuard Rules for app)
  gradle.properties (Project Properties)
```

```java
package com.example.fabio.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {  savedInstanceState: null
        super.onCreate(savedInstanceState);  savedInstanceState: null
        setContentView(R.layout.activity_main);
    }
}
```

Debug app

Frames

```
"main"@4,370 in gr...
onCreate:11, MainActivity (com.example
performCreate:6679, Activity (android.ap
callActivityOnCreate:1118, Instrumentat
performLaunchActivity:2618, ActivityThr
handleLaunchActivity:2726, ActivityThre
-wrap12:-1, ActivityThread (android.app
handleMessage:1477, ActivityThread$H
dispatchMessage:102, Handler (android.
loop:154, Looper (android.os)
main:6119, ActivityThread (android.app)
invoke:-1, Method (java.lang.reflect)
```

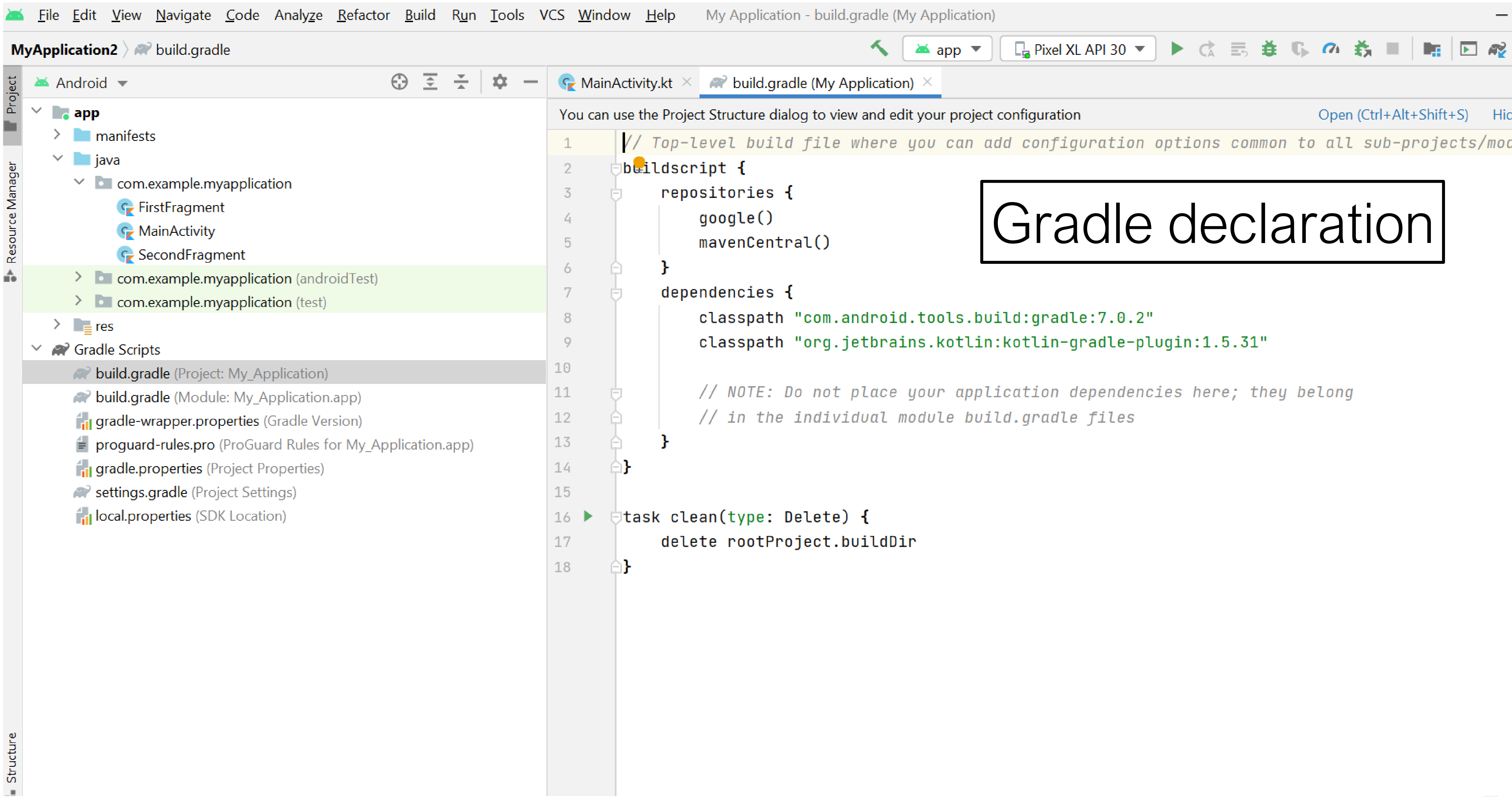
Variables

```
this = {MainActivity@4539}
savedInstanceState = null
```

Gradle build finished in 3s 132ms (moments ago)

Gradle

- Gradle is a build system similar to Maven or Ant
- It is extensible and flexible
- In AndroidStudio it is used to declare the parts of an app, its constraints and to link the external libraries needed by the app
- NOTE:
  - the Gradle file partially overwrites Manifest.xml


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
My Application - build.gradle (My Application)
MyApplication2
build.gradle
app
Pixel XL API 30
Android
MainActivity.kt
build.gradle (My Application)
Project
Resource Manager
Structure

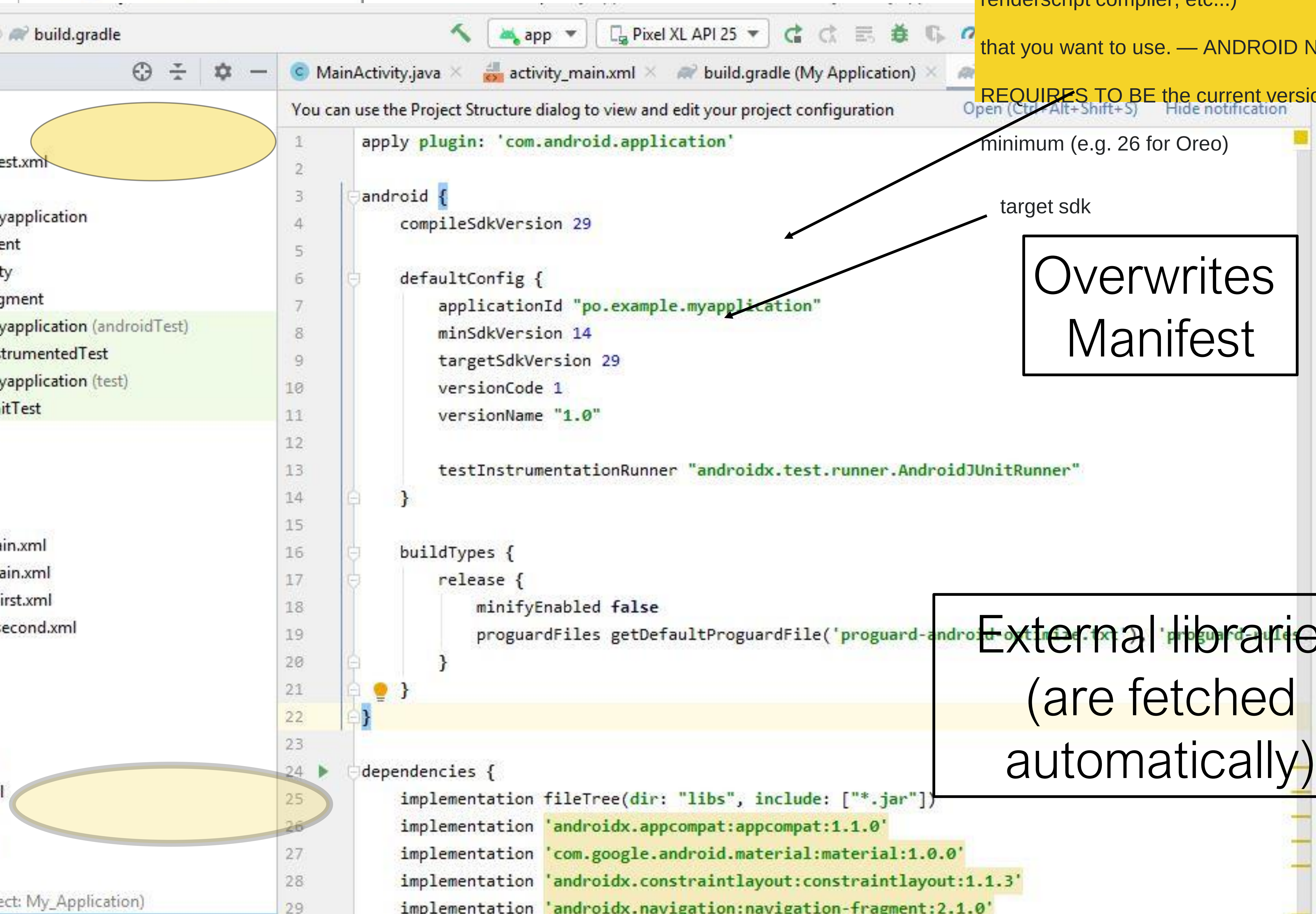
```
app
  manifests
  java
    com.example.myapplication
      FirstFragment
      MainActivity
      SecondFragment
    com.example.myapplication (androidTest)
    com.example.myapplication (test)
  res
Gradle Scripts
  build.gradle (Project: My_Application)
  build.gradle (Module: My_Application.app)
  gradle-wrapper.properties (Gradle Version)
  proguard-rules.pro (ProGuard Rules for My_Application.app)
  gradle.properties (Project Properties)
  settings.gradle (Project Settings)
  local.properties (SDK Location)
```

You can use the Project Structure dialog to view and edit your project configuration Open (Ctrl+Alt+Shift+S) Hic

```
// Top-level build file where you can add configuration options common to all sub-projects/moc
buildscript {
    repositories {
        google()
        mavenCentral()
    }
    dependencies {
        classpath "com.android.tools.build:gradle:7.0.2"
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:1.5.31"

        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

Gradle declaration

App Gradle Module



build.gradle | app | Pixel XL API 25

MainActivity.java | activity_main.xml | build.gradle (My Application)

est.xml
yapplication
ent
ty
gment
yapplication (androidTest)
trumentedTest
yapplication (test)
itTest
in.xml
ain.xml
irst.xml
econd.xml
ect: My_Application)

You can use the Project Structure dialog to view and edit your project configuration — Open (Ctrl+Alt+Shift+S) — Hide notification

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 29

    defaultConfig {
        applicationId "po.example.myapplication"
        minSdkVersion 14
        targetSdkVersion 29
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules
        }
    }
}

dependencies {
    implementation fileTree(dir: "libs", include: ["*.jar"])
    implementation 'androidx.appcompat:appcompat:1.1.0'
    implementation 'com.google.android.material:material:1.0.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    implementation 'androidx.navigation:navigation-fragment:2.1.0'
```

is the version of the compilers (aapt, dx, renderscript compiler, etc...) that you want to use. — ANDROID NOW REQUIRES TO BE the current version as minimum (e.g. 26 for Oreo)

target sdk

Overwrites Manifest

External libraries (are fetched automatically)