Lesson 0 - Your First Android App

Introduction (Read this first)

You will complete this lesson outside of lesson time.

The instructions given in this lesson will show you how to do a “hello world” Android app. Assuming that you have a fresh installation of Android Studio, a significant amount of time is consumed in downloading components from the internet.

You should allocate 45 mins to 1 hour, together with a fast and cheap (hopefully free) internet connection for this task. Some downloads may take some time, so plan something else to do during that period.

Building android apps is computationally intensive. Your laptop can get warm very quickly and could possibly overheat (which I experienced). Do take the appropriate steps.

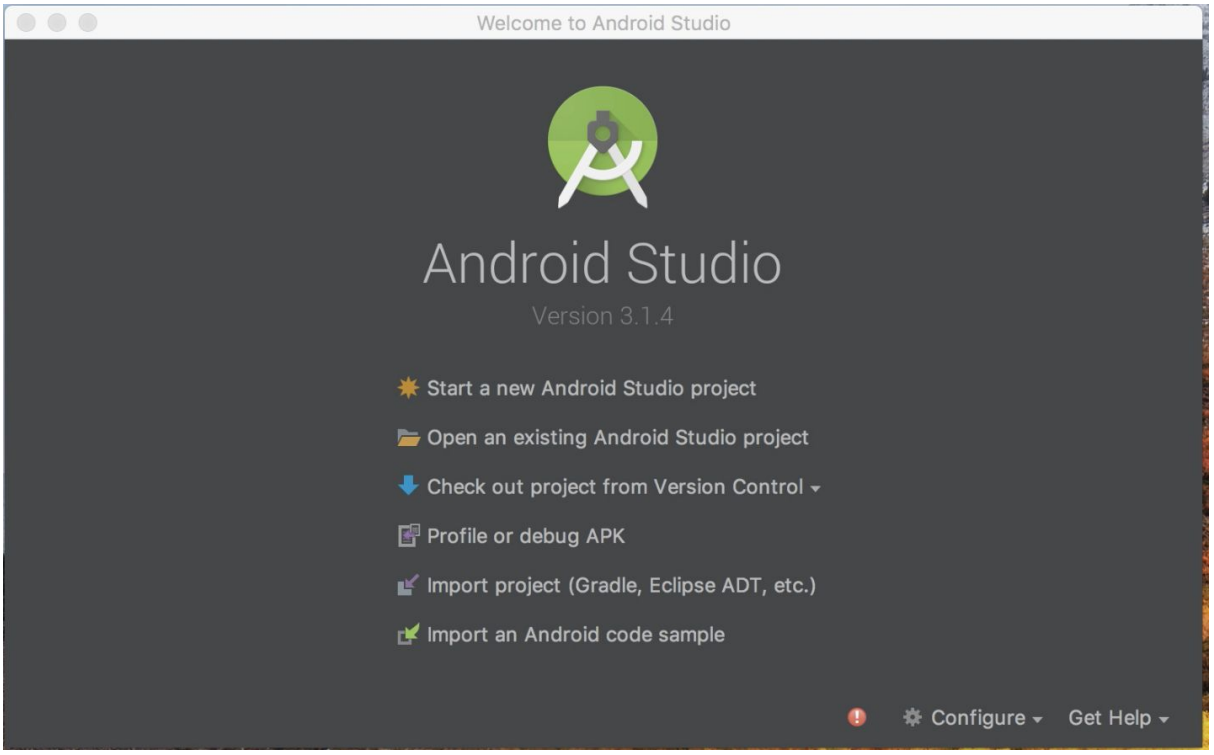
This tutorial was written with a freshly installed Android Studio in a new MacBook. Hence, what you see on your computer could be different, just respond accordingly to the instructions on the user interface.

Try to get an android phone if you do not have one. You may find that testing your app on a physical android mobile phone is much faster than using the emulator.

Create a “hello world” app

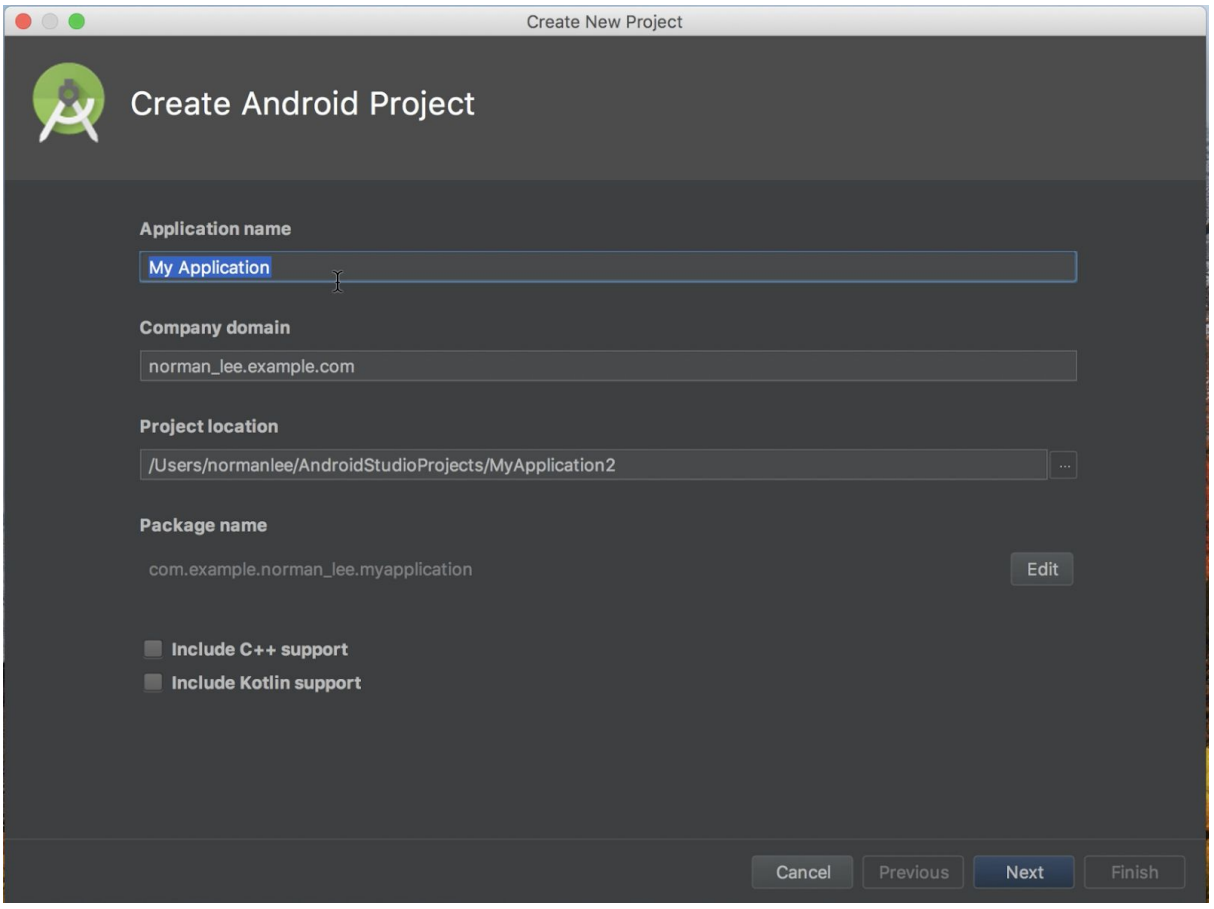
Start a new android studio project

Launch Android Studio from your computer and select **Start A New Android Studio Project**



Give your project a name

Give your project an **Application name** and state where you want it to be saved in the **Project location**. For starters, you may accept the defaults and carry on.

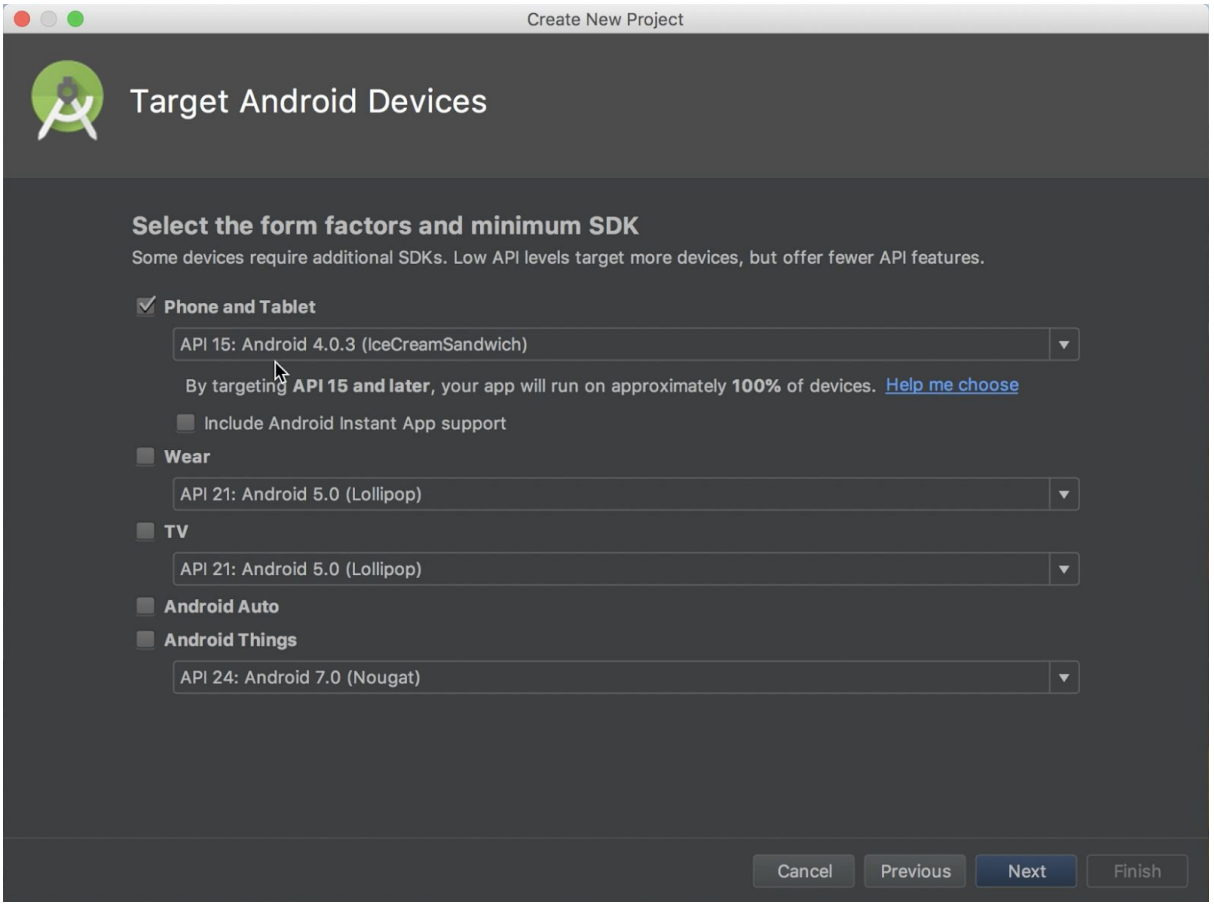


State what devices you want your android app to run on

Ensure **Phone and Tablet** is ticked. You will next have to decide the minimum **API Level** that you want to deploy your android app on.

The **API level** is the version of the android operating system. The lower the API level, the older the version, but the more devices that your app can run on. Explore the options. (You can read more about it here: https://developer.android.com/about/dashboards/)

For starters, accept the defaults.



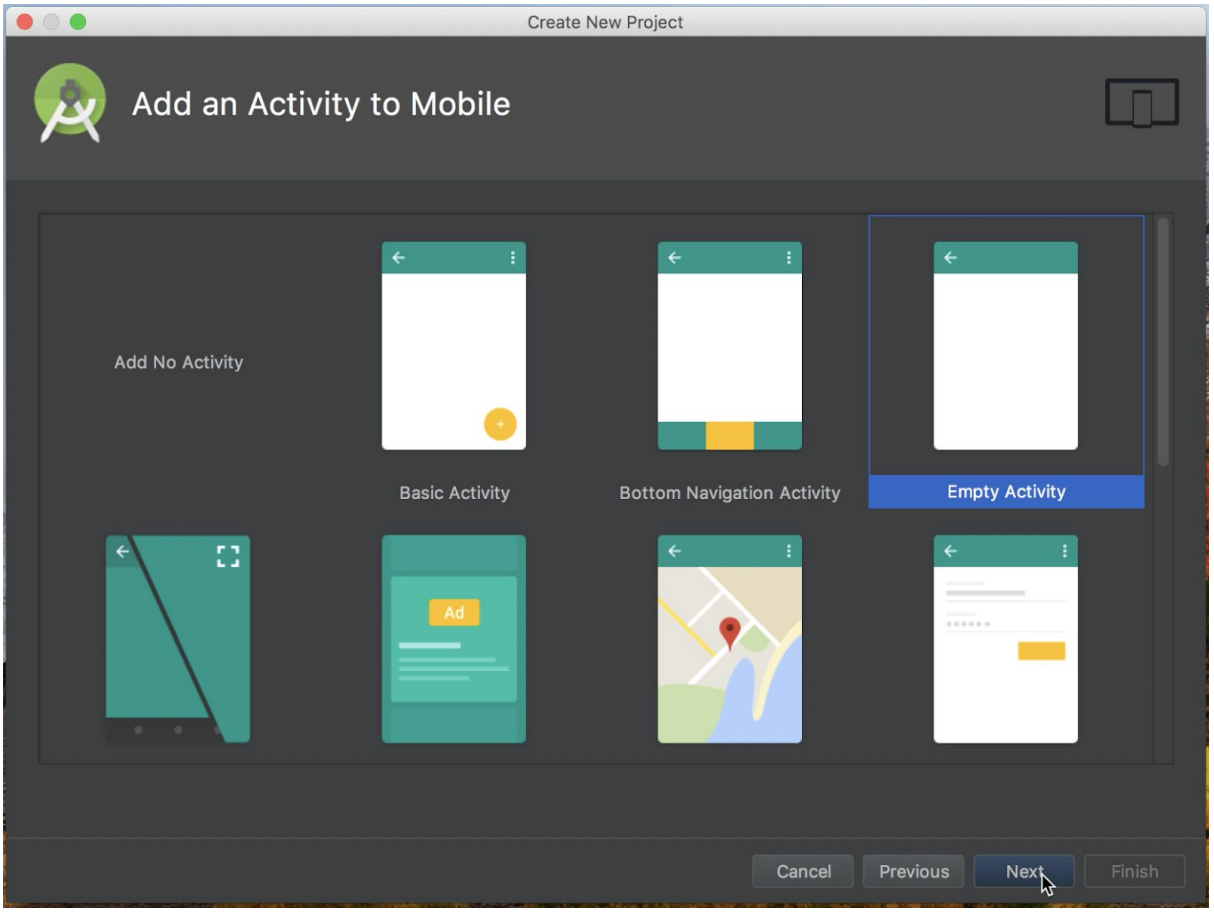
Start with an Activity

You will now have to add an **Activity**. The Android documentation defines an **Activity** as:

An activity is a single, focused thing that the user can do

In other words, it is a screen where the user interface (UI) resides on, for the user to interact with.

Make sure **Empty Activity** is selected, as it is the option with the least amount of code. The other options come pre-loaded with some android app features, and they are useful when you are more familiar with android app programming.



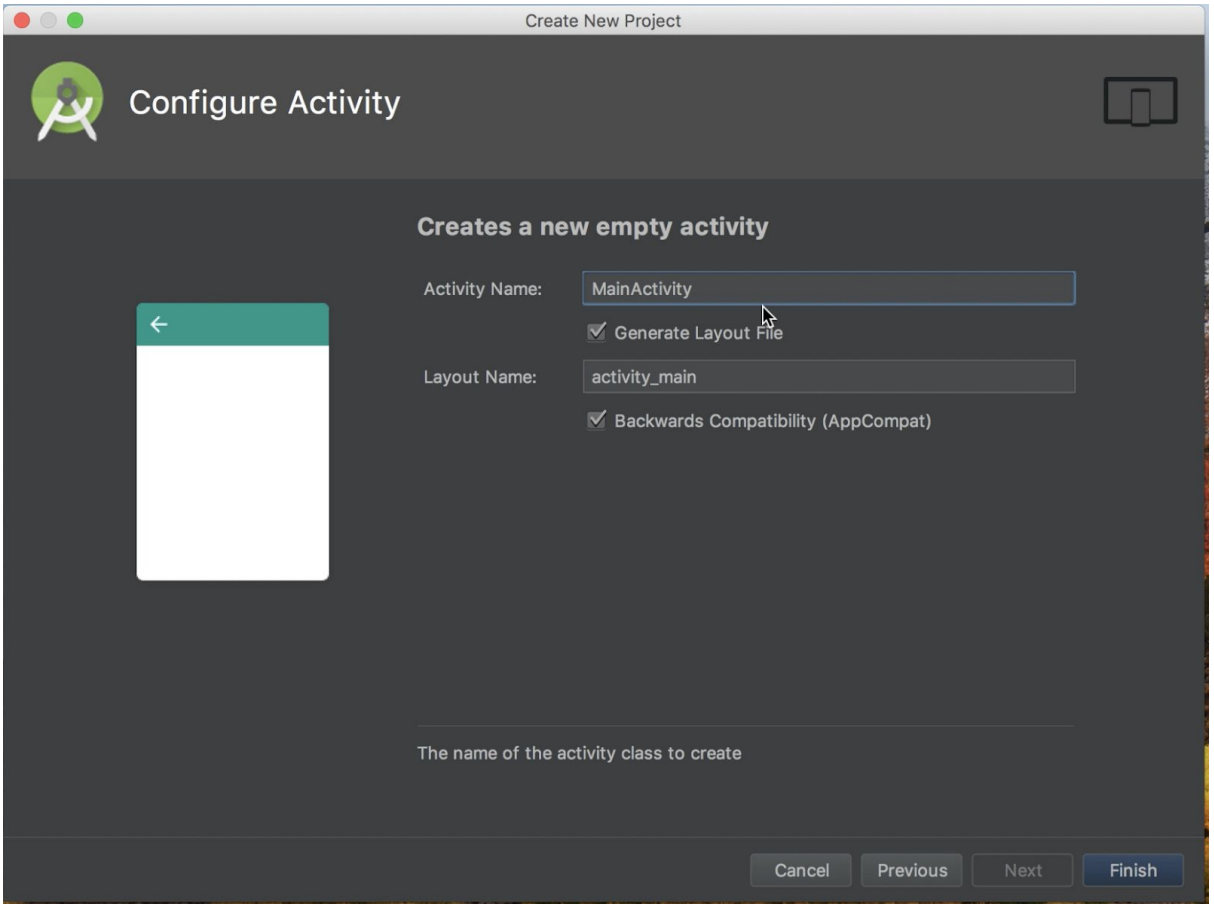
Name your activity

Your android app project will typically have many activities, and the entry point (i.e. the first activity that your user will see) is called **MainActivity**. Hence, just accept the defaults.

This creates two files:

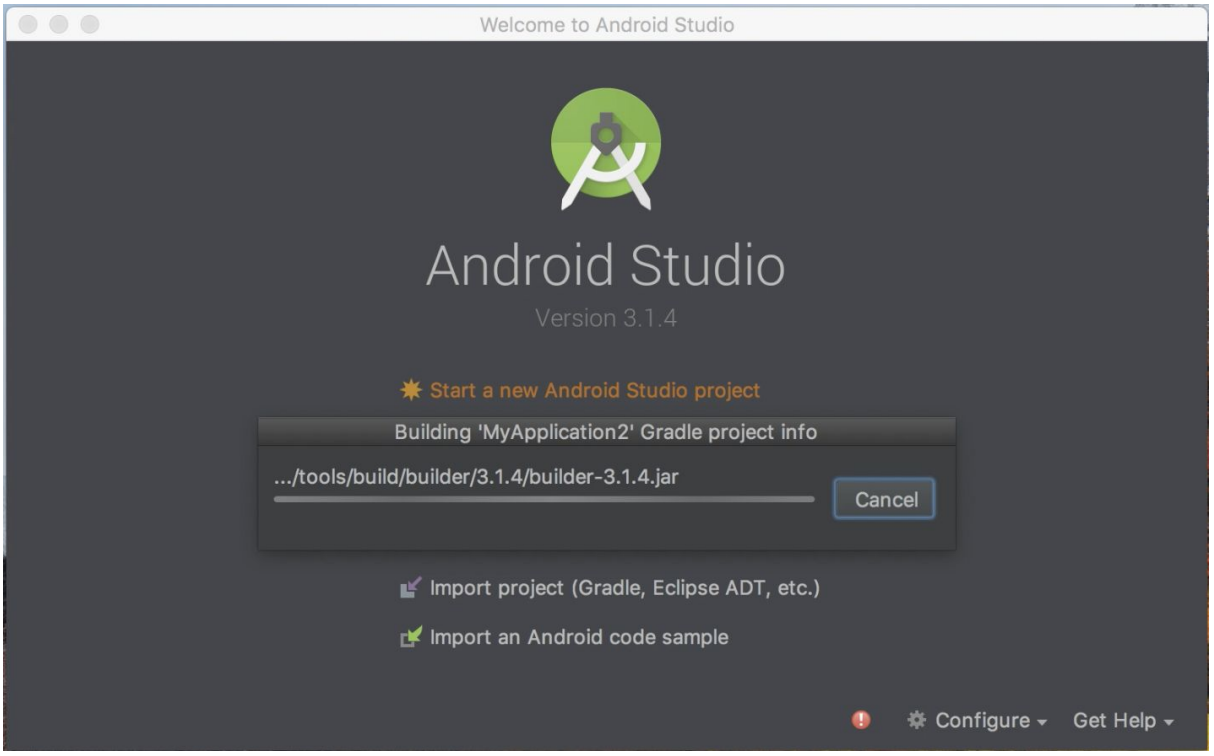
- An xml file named `activity_main.xml`
- A java class in a file named `MainActivity.java`

The layout of the UI is specified in the XML file and the logic (e.g. what happens when a button is pressed) is coded in the java file.



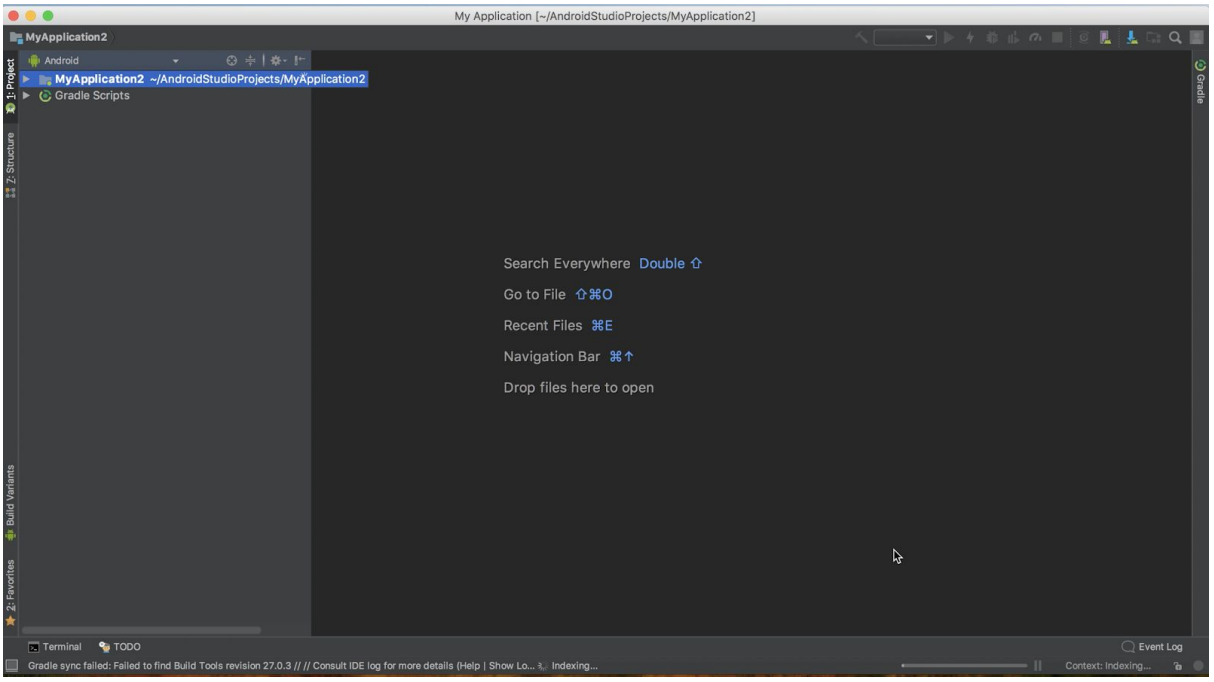
Wait for a while and install some stuff

While android studio is getting your project ready, the screen should look like this.

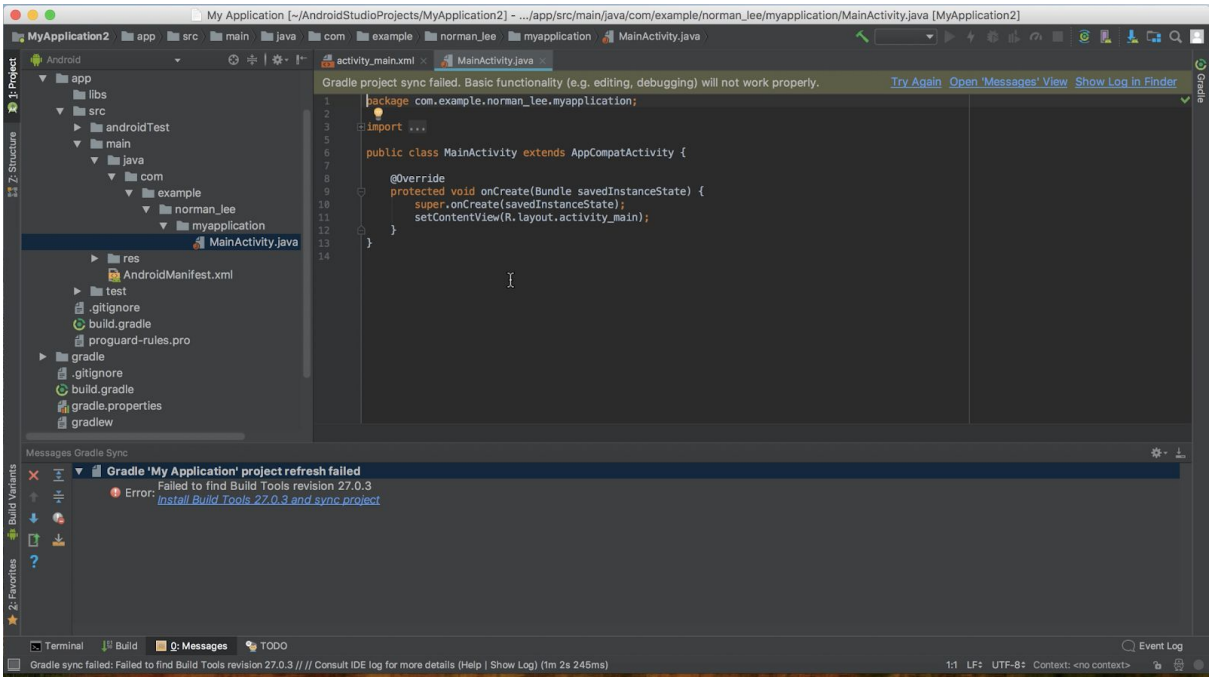


When the installation is done the screen should look like this.

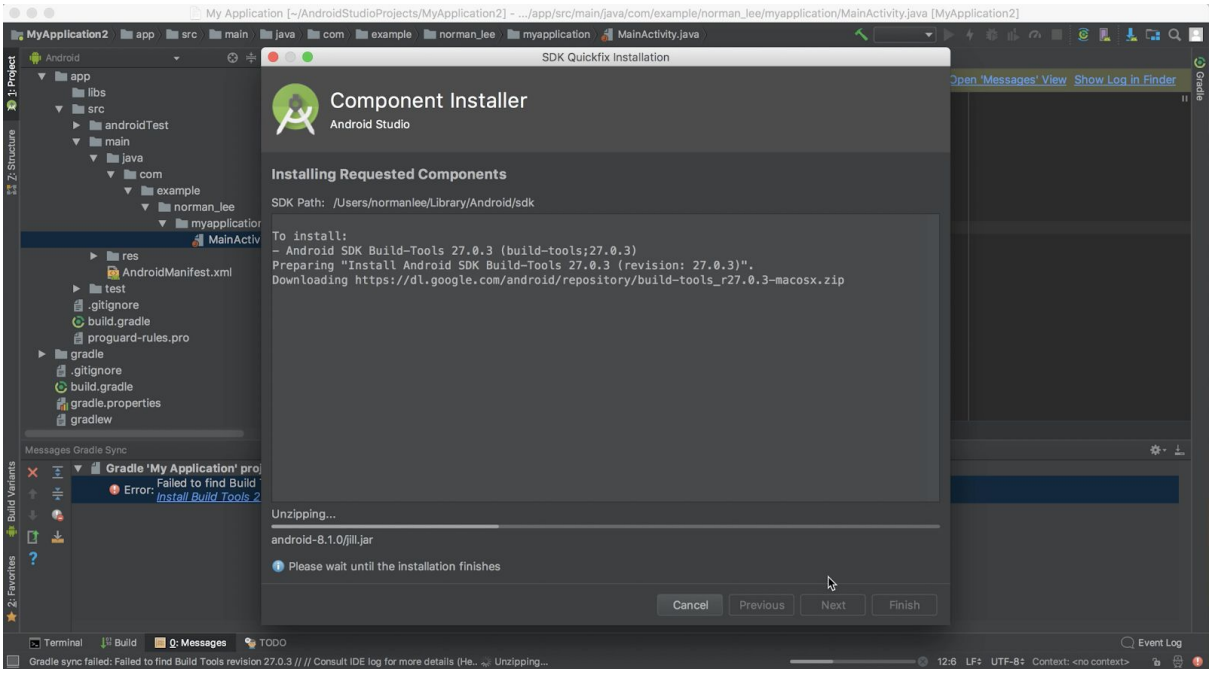
Click on the arrow beside **MyApplication** to expand the folder tree.



After expanding the folder tree, you will see some messages. At this stage, Android Studio does not have all the components necessary to build your app. Click on **Install Build Tools** and wait for a while.



After you click **Install Build Tools**, you will see this display. Click **Finish** when the download is completed.



Examine the folder structure

You'll notice that the folder tree now looks different and will look like the image below. This is the correct folder tree to see.

You can select different views (red box below). What you should select is the **Android** view.

There are two folders, **app** (see point 1 below) and **res** (see point 3 below).

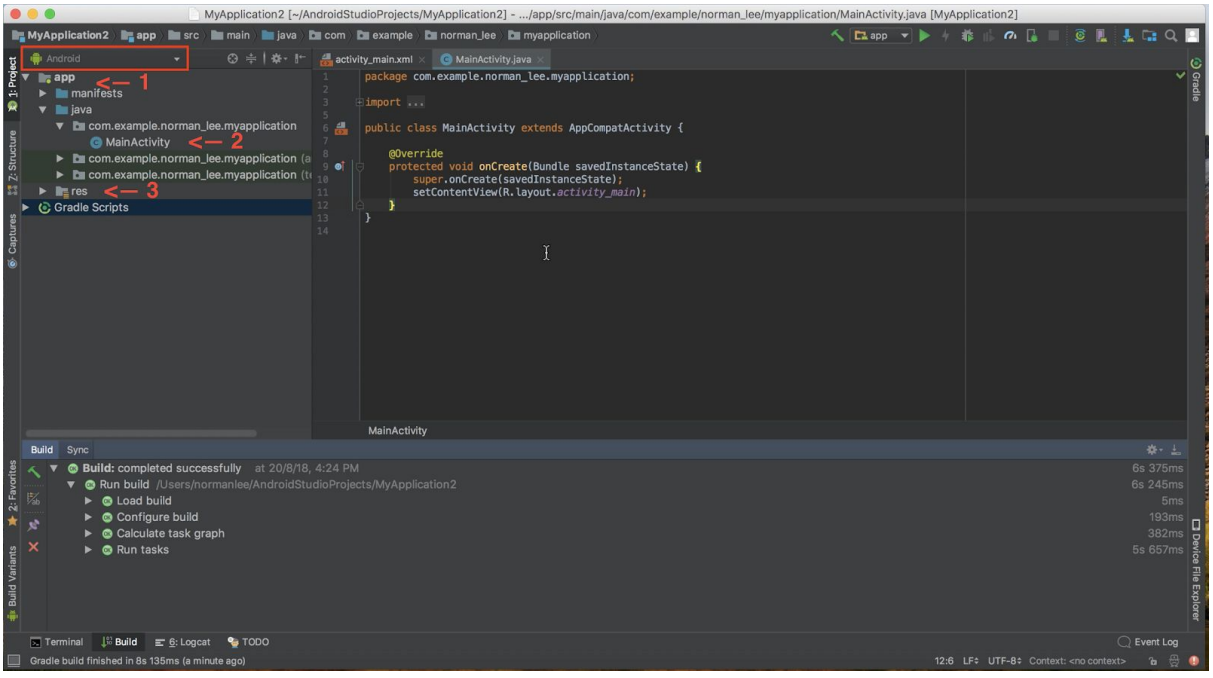
Recall earlier that when an **activity** is created, you get two files:

- An xml file named `activity_main.xml`
- A java class in a file named `MainActivity.java`

The **app** folder contains the java code and the java code `MainActivity.java` is found in the package folder within the java folder (see point 2).

The **res** folder (see point 3) contains resources for the app. This includes xml files, images and icons.

Expand the **res** folder and look for `activity_main.xml` file under the **layout** subfolder.



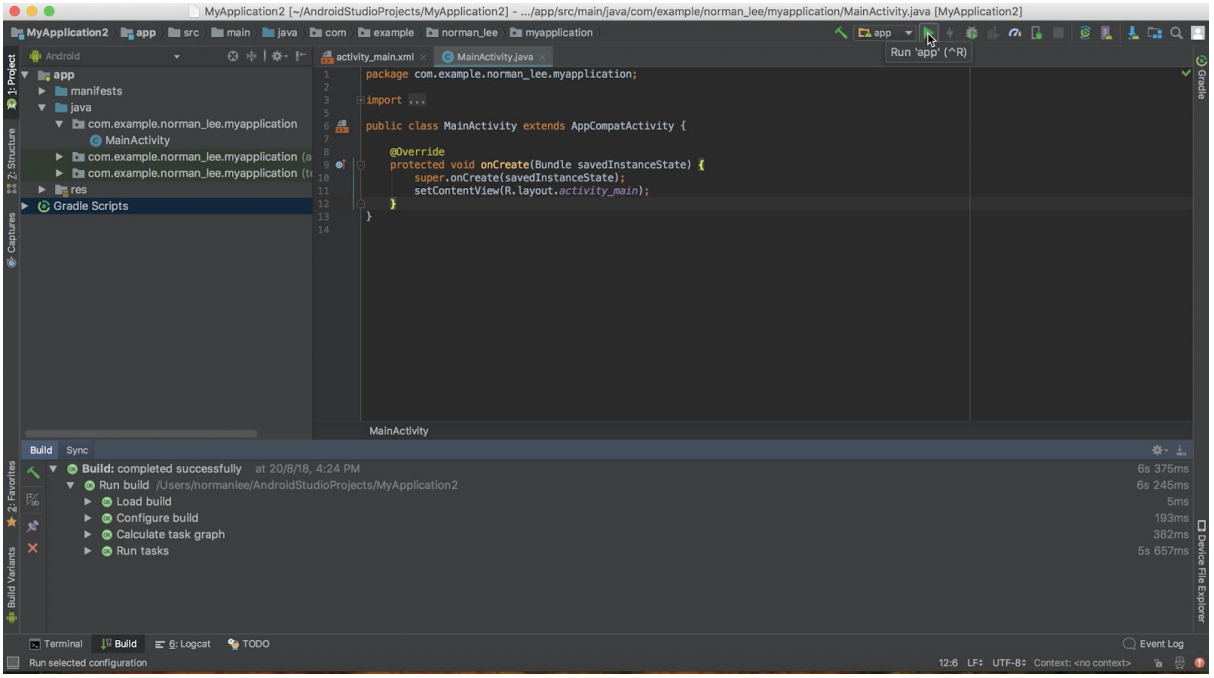
Test your App

If you have an android phone, follow the instructions:
https://developer.android.com/studio/run/device

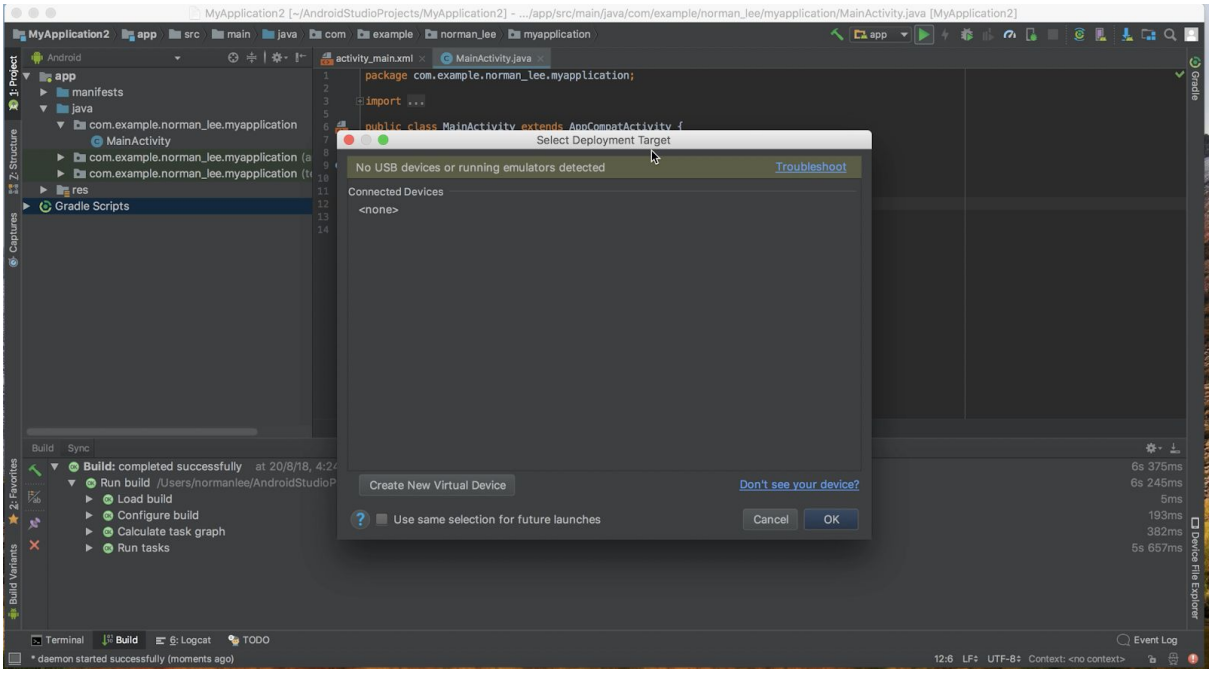
If you wish to use the emulator, continue with the rest of this guide. You will need at least 30 minutes more from now if you are doing a fresh installation.

Installing the Emulator

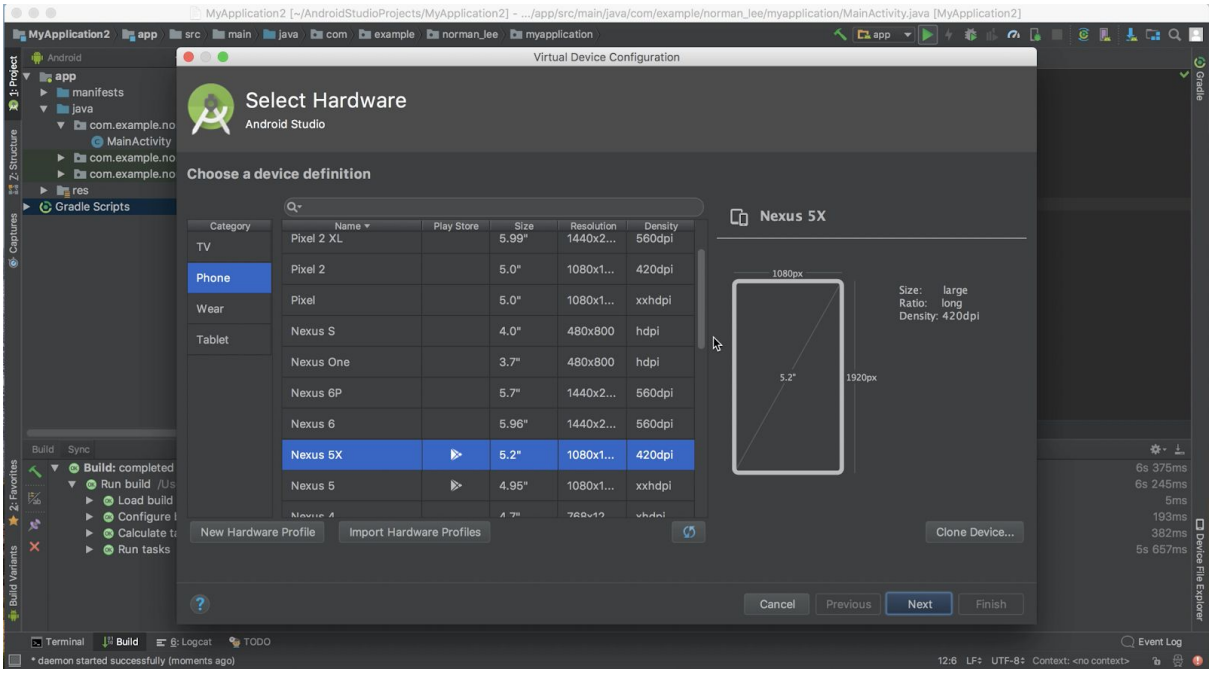
Go to the Run button and click it. Make sure “app” is shown beside it.



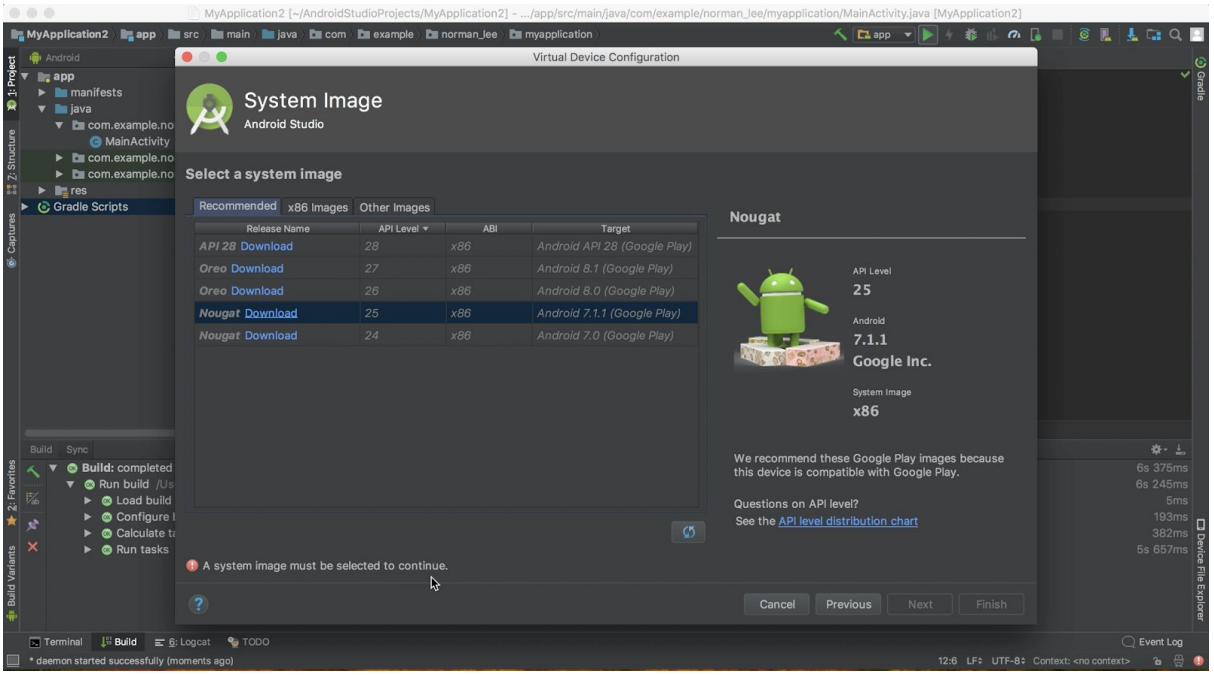
In the following window you should select **Create New Virtual Device**.



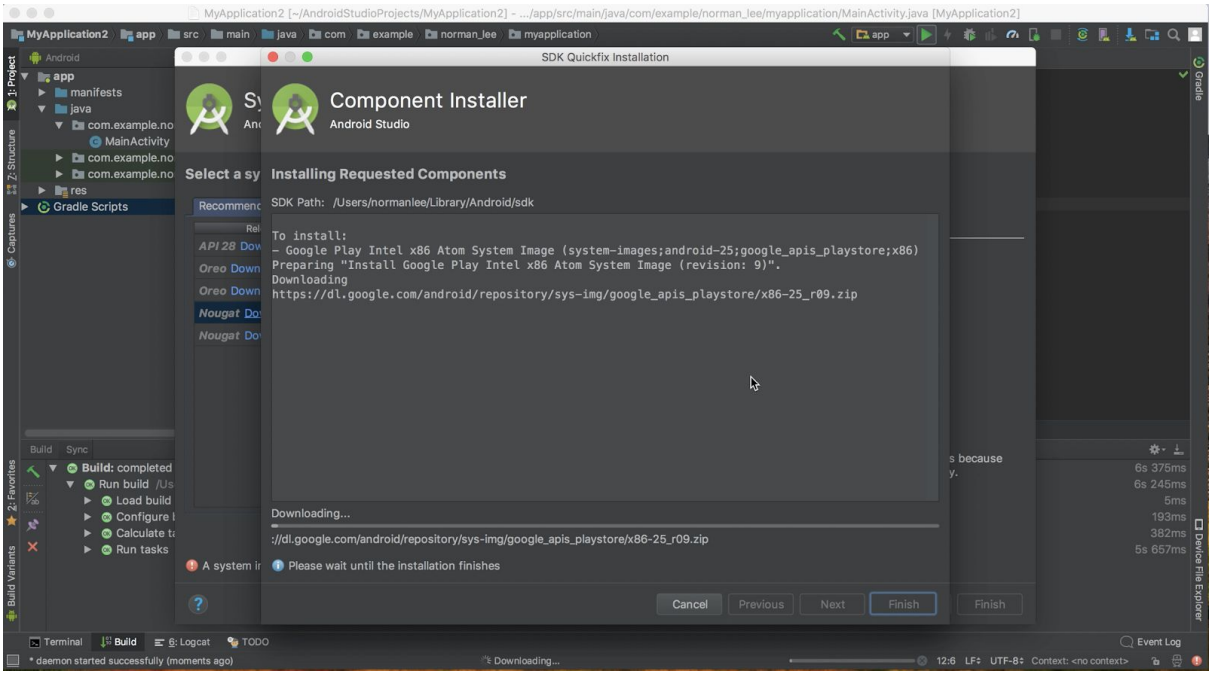
Many different devices are available for you with different screen sizes. You can explore by scrolling up and down. Just accept the defaults by clicking next.



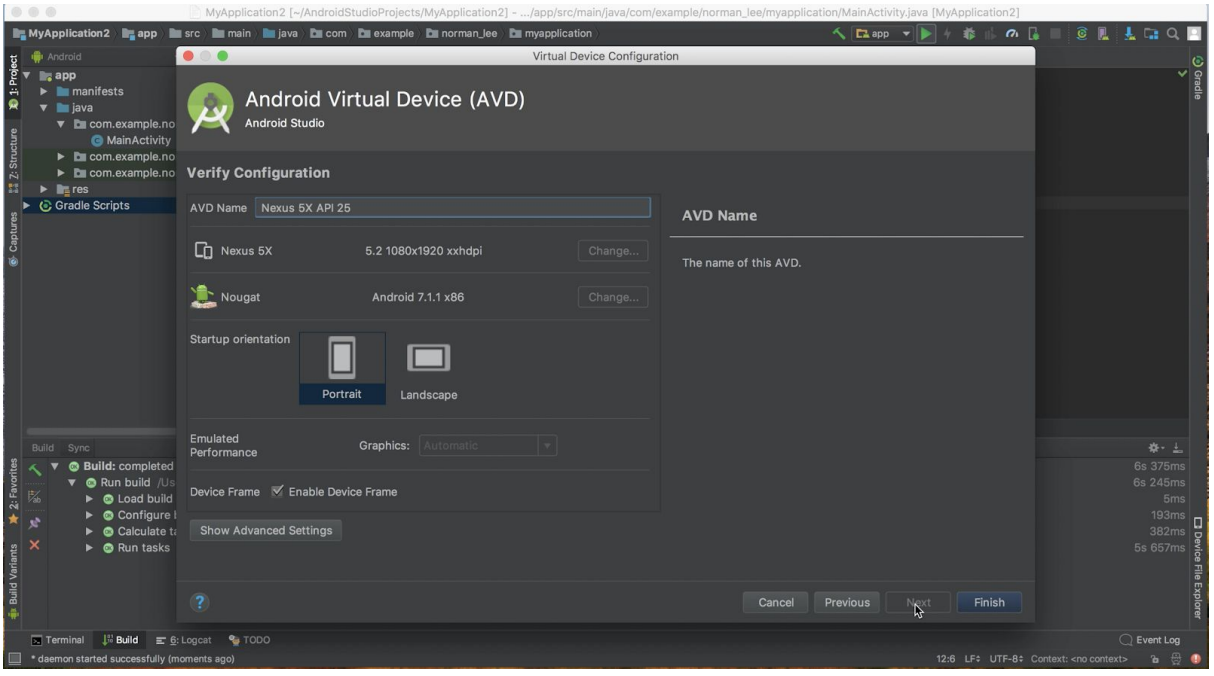
You are brought to this screen, where you have a choice of API levels (remember the API level is the version of the android OS). Choose API Level 25, we are not likely to need higher levels. After selecting the API Level, click on the Download link.



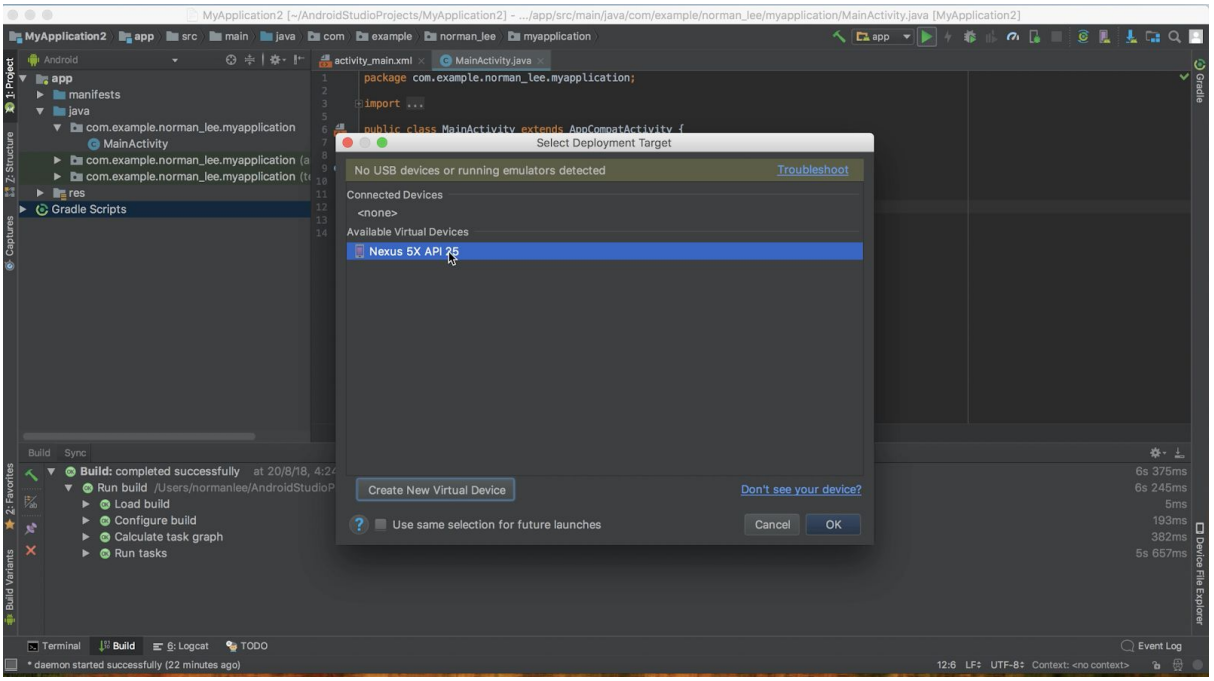
The following window pops up, and I had to wait for 20 minutes for the download to complete. If this is the case for you, find something else to do in the meantime. Click **Finish** when the download completes.



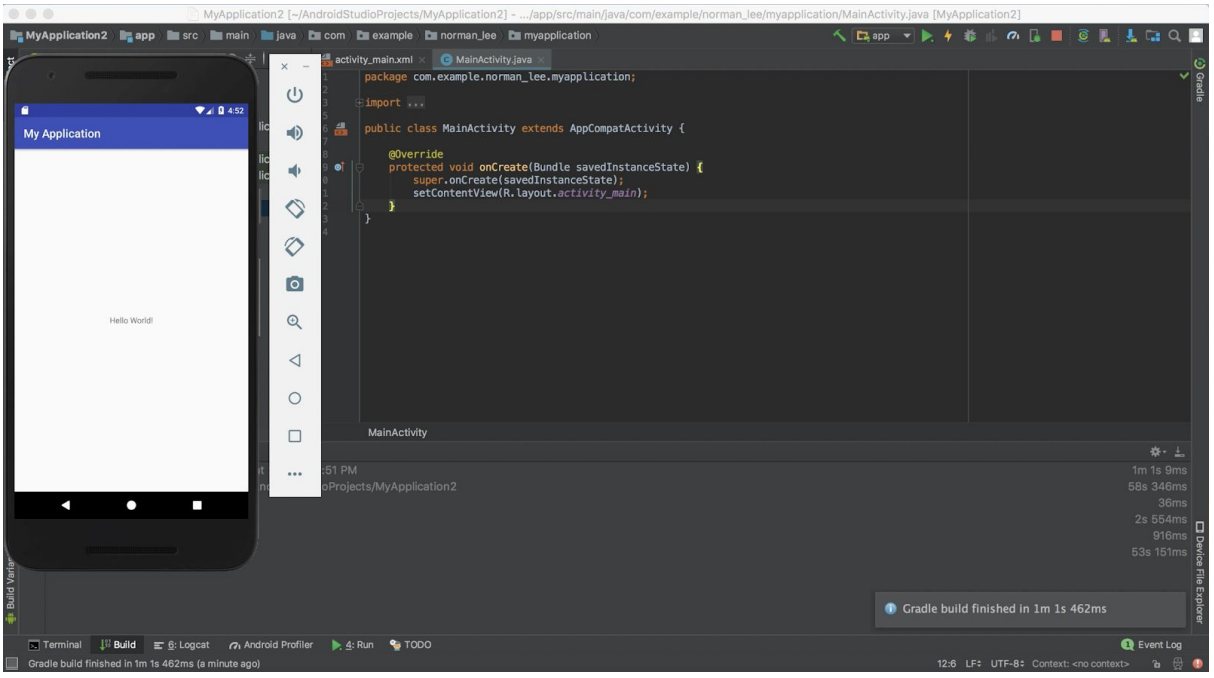
Next you are prompted for the settings of the emulator. Just accept the defaults and click Finish.



Now, an emulator is available in the window. Select it and click ok.



The emulator should now display the app. Congratulations, you have just completed your very first android app!



Further Reading

- Android Developer Fundamentals Concepts Section 1.1 - this was written in 2016 and changes have occurred since then, but it still gives a good overview of the process https://google-developer-training.gitbooks.io/android-developer-fundamentals-course-concepts/content/en/Unit%201/11_c_create_your_first_android_app.html