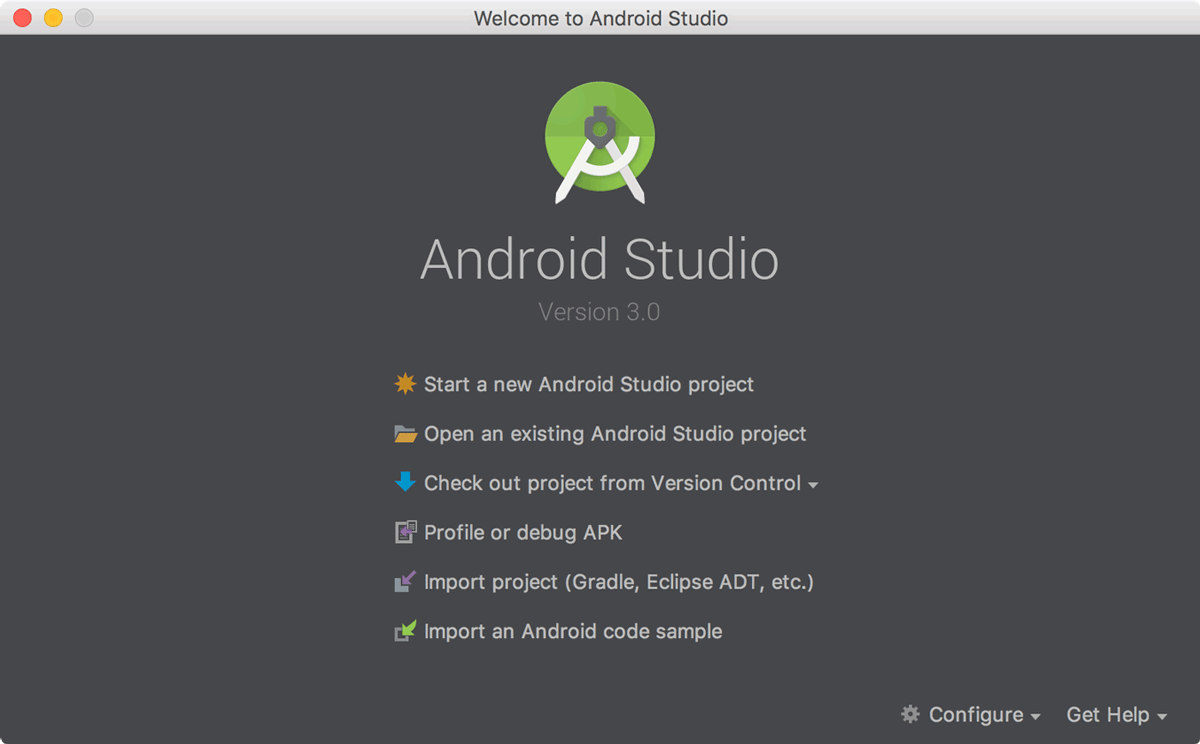
Create an Android Project

This lesson shows you how to create a new Android project with Android Studio and describes some of the files in the project.

First, be sure you have installed the latest version of Android Studio. [Download Android Studio here](https://developer.android.com/studio/).

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.

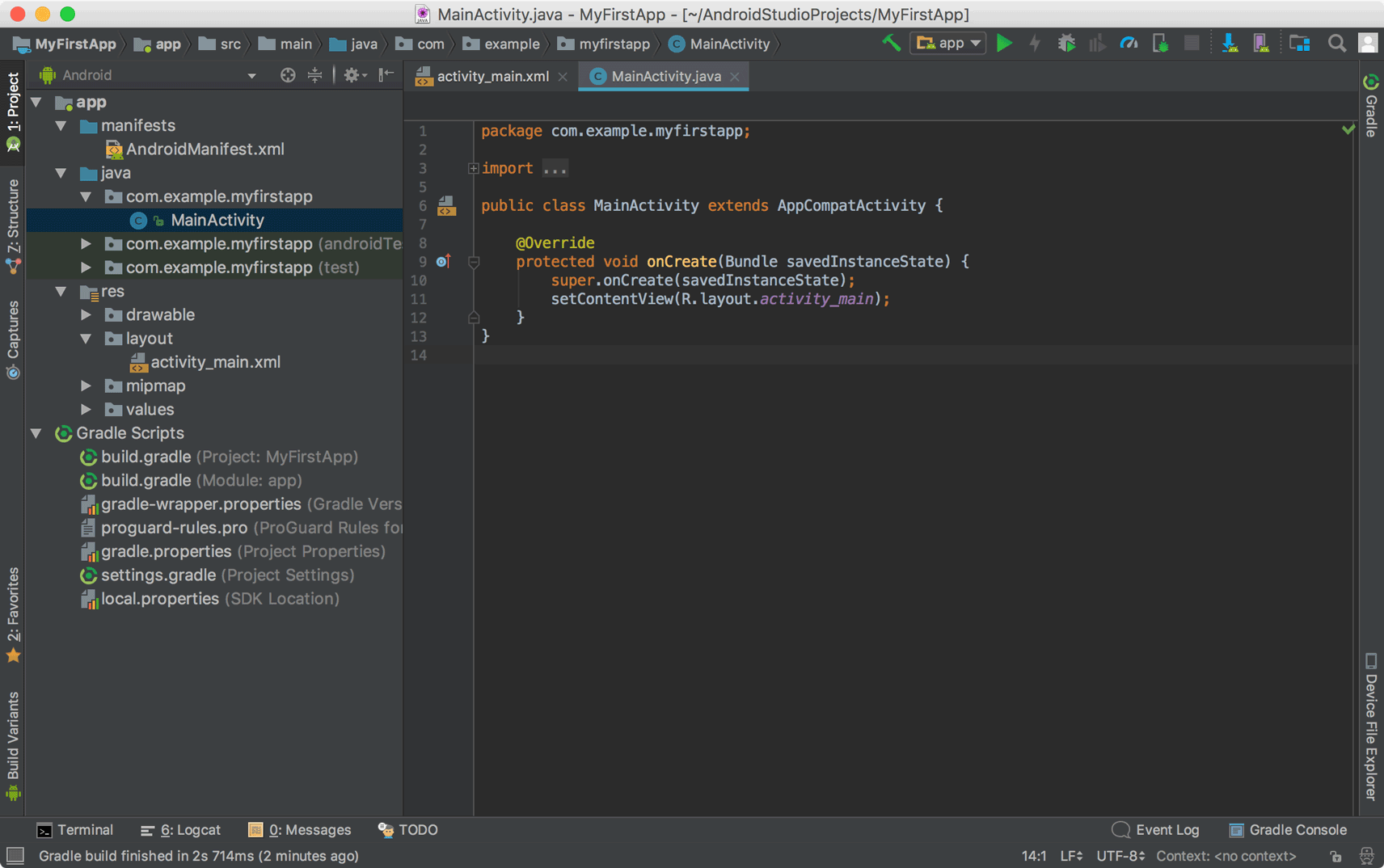
Or if you have a project opened, select **File > New Project**.

1. In the **Create New Project** window, enter the following values:
   * **Application Name**: "My First App"
   * **Company Domain**: "example.com"

You might want to change the project location. Also, if you want to write a Kotlin app, check the **Include Kotlin support** checkbox. Leave the other options as they are.

1. Click **Next**.
2. In the **Target Android Devices** screen, keep the default values and click **Next**.
3. In the **Add an Activity to Mobile** screen, select **Empty Activity** and click **Next**.
4. In the **Configure Activity** screen, keep the default values and click **Finish**.

After some processing, Android Studio opens the IDE.



Now take a moment to review the most important files.

First, be sure the **Project** window is open (select **View > Tool Windows > Project**) and the **Android** view is selected from the drop-down list at the top of that window. You can then see the following files:

**app > java > com.example.myfirstapp > MainActivity**

This is the main activity (the entry point for your app). When you build and run the app, the system launches an instance of this [Activity](https://developer.android.com/reference/android/app/Activity.html) and loads its layout.

**app > res > layout > activity\_main.xml**

This XML file defines the layout for the activity's UI. It contains a [TextView](https://developer.android.com/reference/android/widget/TextView.html) element with the text "Hello world!".

**app > manifests > AndroidManifest.xml**

The [manifest file](https://developer.android.com/guide/topics/manifest/manifest-intro.html) describes the fundamental characteristics of the app and defines each of its components.

**Gradle Scripts > build.gradle**

You'll see two files with this name: one for the project and one for the "app" module. Each module has its own build.gradle file, but this project currently has just one module. You'll mostly work with the module's build.gradle file to configure how the Gradle tools compile and build your app

# Run Your App

## Run on a real device

Set up your device as follows:

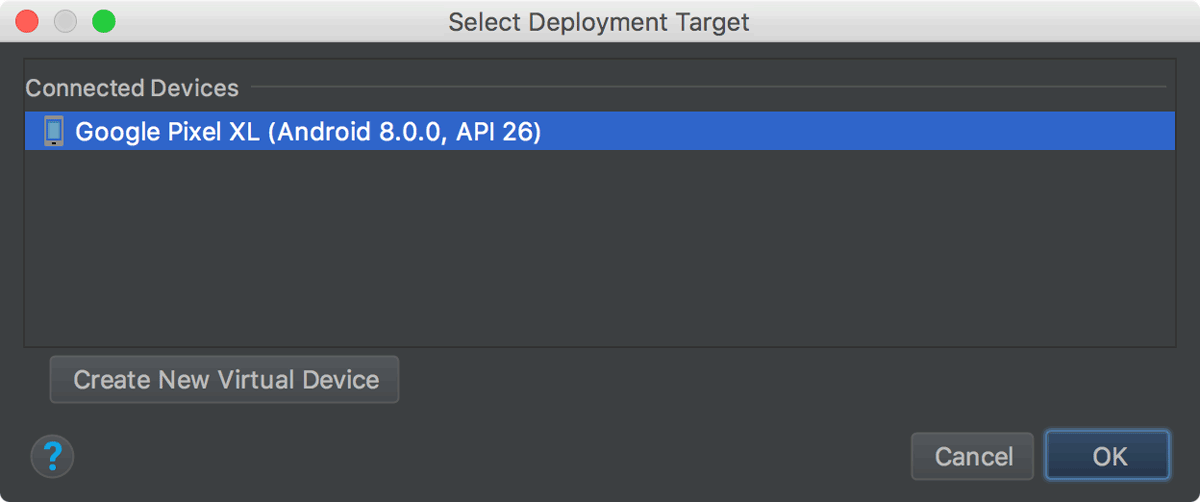
1. Connect your device to your development machine with a USB cable. If you're developing on Windows, you might need to [install the appropriate USB driver](https://developer.android.com/studio/run/oem-usb.html) for your device.
2. Enable **USB debugging** in the **Developer options** as follows.

First, you must enable the developer options:

* 1. Open the **Settings** app.
  2. (Only on Android 8.0 or higher) Select **System**.
  3. Scroll to the bottom and select **About phone**.
  4. Scroll to the bottom and tap **Build number** 7 times.
  5. Return to the previous screen to find **Developer options** near the bottom.

Open **Developer options**, and then scroll down to find and enable **USB debugging**.

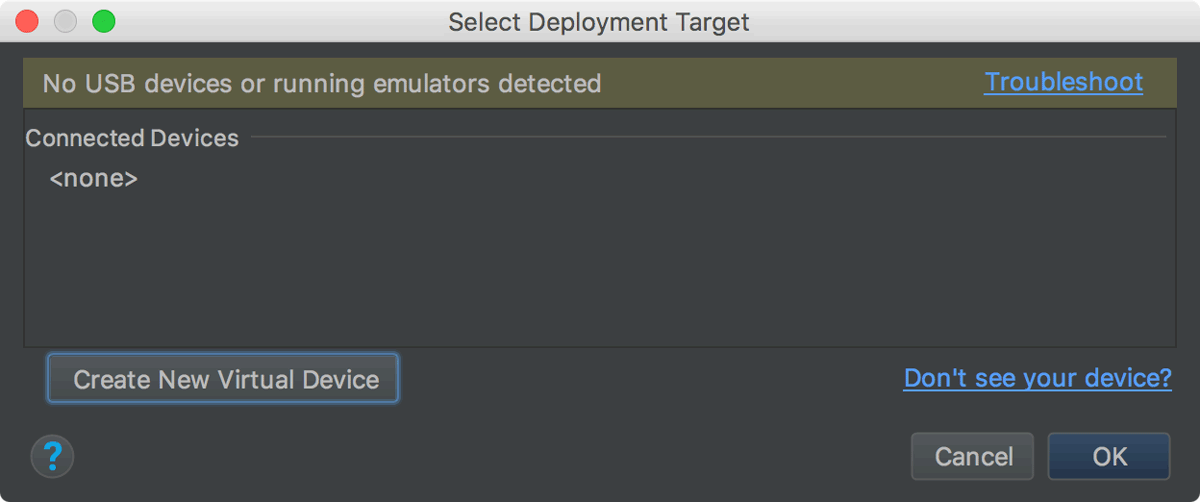
Run the app on your device as follows:

1. In Android Studio, click the **app** module in the **Project** window and then select **Run > Run** (or click **Run** https://developer.android.com/studio/images/buttons/toolbar-run.png in the toolbar).
2. In the **Select Deployment Target** window, select your device, and click **OK**.

Android Studio installs the app on your connected device and starts it. You should now see "Hello World!" displayed in the app running on your device.

## Run on an emulator (emulator is virtual device)

Run the app on an emulator as follows:

1. In Android Studio, click the **app** module in the **Project** window and then select **Run > Run** (or click **Run** https://developer.android.com/studio/images/buttons/toolbar-run.png in the toolbar).
2. In the **Select Deployment Target** window, click **Create New Virtual Device**.
3. In the **Select Hardware** screen, select a phone device, such as Pixel, and then click **Next**.
4. In the **System Image** screen, select the version with the highest API level. If you don't have that version installed, a **Download** link is shown, so click that and complete the download.
5. Click **Next**.
6. On the **Android Virtual Device (AVD)** screen, leave all the settings alone and click **Finish**.
7. Back in the **Select Deployment Target** dialog, select the device you just created and click **OK**.