**Queens University Belfast**

Android Studio

CSC3054 / CSC7054

Working with the emulator

Working with the emulator

What if you wanted to test an app on a different phone, say the `Galaxy Nexus`? In this case, rather than run out and buy a new `Galaxy Nexus`, it is possible to run the `HelloAndroid` program on an emulated `Galaxy Nexus` using the `Android Emulator`. In order to do this an emulated phone instance should be created and used inside the emulator. These emulated phone instances are called `Android Virtual Devices,` or `AVDs`.

- Go back to the IDE and set up an `Android virtual device` corresponding to a `Galaxy Nexus`.

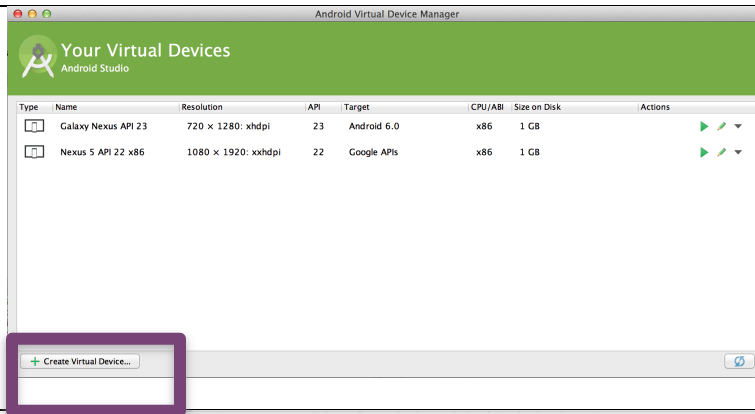- Go to the tool bar and launch the `AVD Manager`.
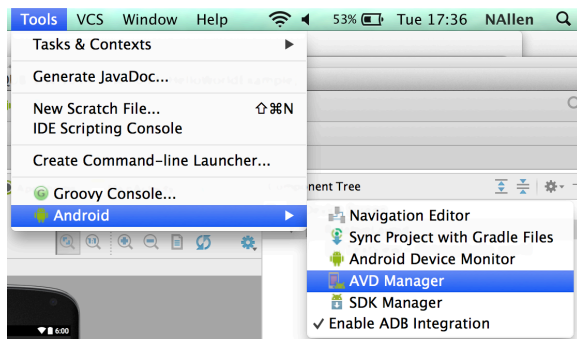
- Alternatively, look for the icon that shows an Android in front of a device, click on it

- A dialogue box will be displayed showing existing virtual devices. It will also allow you to make new ones.
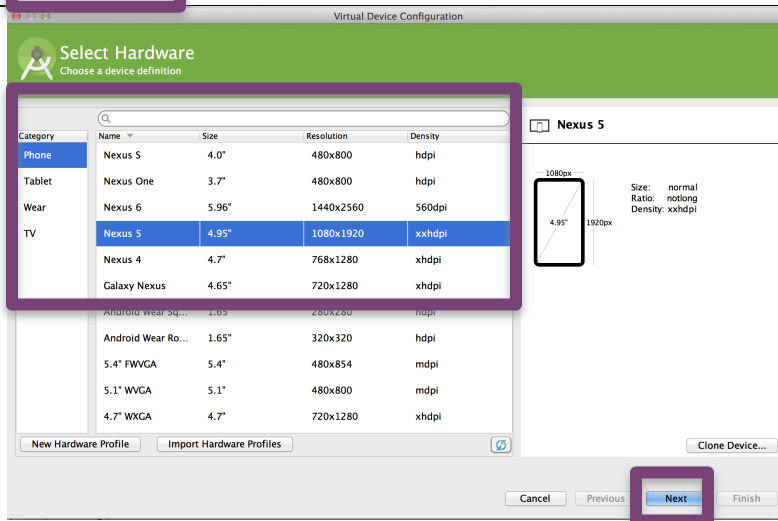
Create a new virtual device.

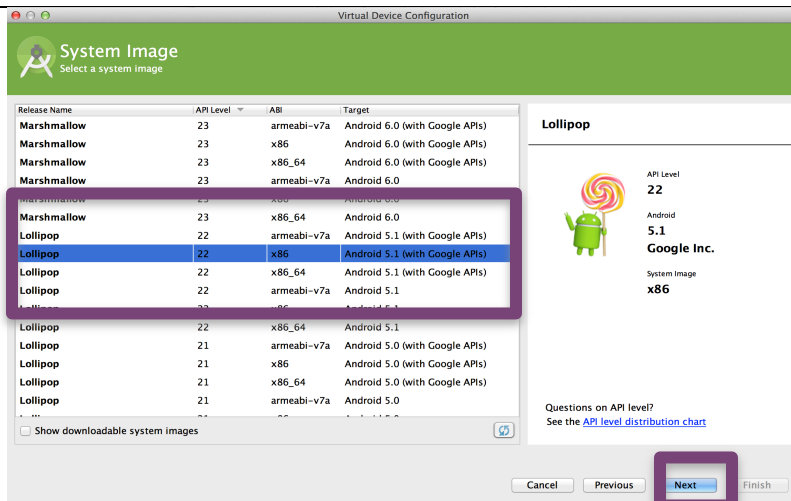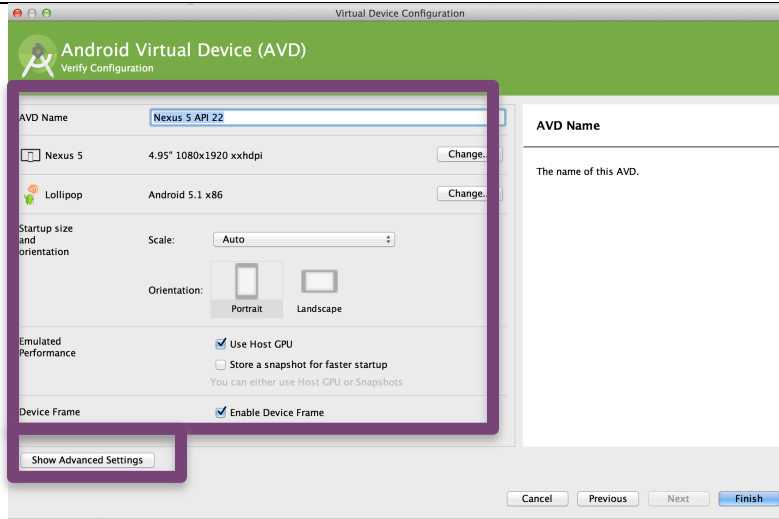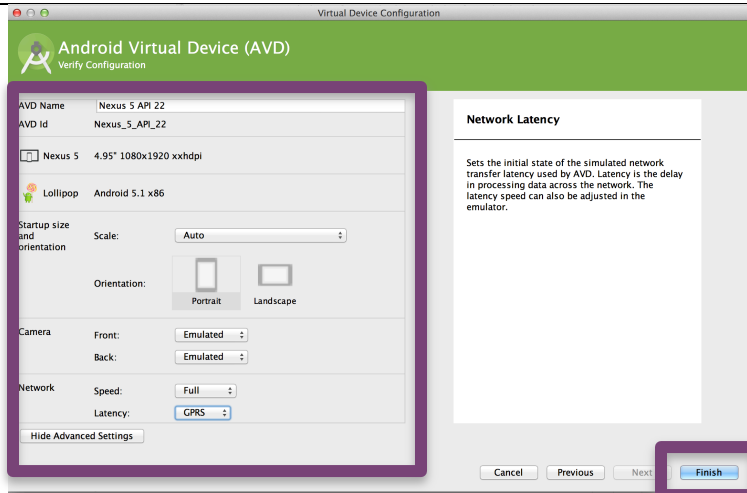| Step 1		Click on the `Create Virtual Device` button.
	Android Virtual Device Manager — Your Virtual Devices — Android Studio. Type / Name / Resolution / API / Target / CPU/ABI / Size on Disk / Actions. Galaxy Nexus API 23 — 720 × 1280: xhdpi — 23 — Android 6.0 — x86 — 1 GB. Nexus 5 API 22 x86 — 1080 × 1920: xxhdpi — 22 — Google APIs — x86 — 1 GB. Create Virtual Device...	
Step 2	Virtual Device Configuration — Select Hardware — Choose a device definition. Category / Name / Size / Resolution / Density. Phone / Tablet / Wear / TV. Nexus S — 4.0" — 480x800 — hdpi. Nexus One — 3.7" — 480x800 — hdpi. Nexus 6 — 5.96" — 1440x2560 — 560dpi. Nexus 5 — 4.95" — 1080x1920 — xxhdpi. Nexus 4 — 4.7" — 768x1280 — xhdpi. Galaxy Nexus — 4.65" — 720x1280 — xhdpi. Android Wear Sq... — 1.65 — 280x280 — hdpi. Android Wear Ro... — 1.65" — 320x320 — hdpi. 5.4" FWVGA — 5.4" — 480x854 — mdpi. 5.1" WVGA — 5.1" — 480x800 — mdpi. 4.7" WXGA — 4.7" — 720x1280 — xhdpi. Nexus 5 — Size: normal, Ratio: notlong, Density: xxhdpi. New Hardware Profile / Import Hardware Profiles / Clone Device... Cancel / Previous / **Next** / Finish	A new dialogue box will appear and select the `Nexus 5` device definition. Click `Next`.
Step 3	Virtual Device Configuration — System Image — Select a system image. Release Name / API Level / ABI / Target. Marshmallow — 23 — armeabi-v7a — Android 6.0 (with Google APIs). Marshmallow — 23 — x86 — Android 6.0 (with Google APIs). Marshmallow — 23 — x86_64 — Android 6.0 (with Google APIs). Marshmallow — 23 — armeabi-v7a — Android 6.0. Marshmallow — 23 — x86 — Android 6.0. Marshmallow — 23 — x86_64 — Android 6.0. Lollipop — 22 — armeabi-v7a — Android 5.1 (with Google APIs). Lollipop — 22 — x86 — Android 5.1 (with Google APIs). Lollipop — 22 — x86_64 — Android 5.1 (with Google APIs). Lollipop — 22 — armeabi-v7a — Android 5.1. Lollipop — 22 — x86_64 — Android 5.1. Lollipop — 21 — armeabi-v7a — Android 5.0 (with Google APIs). Lollipop — 21 — x86 — Android 5.0 (with Google APIs). Lollipop — 21 — x86_64 — Android 5.0 (with Google APIs). Lollipop — 21 — armeabi-v7a — Android 5.0. Lollipop — API Level 22, Android 5.1, Google Inc., System Image x86. Questions on API level? See the API level distribution chart. Show downloadable system images. Cancel / Previous / **Next** / Finish	Select a system image for the device. In this case use a `Lollipop` or `Android 5.1.1` image that also has the `Google APIs`. This means that it will include apps such as `Google Maps`, which will be used in this tutorial. Click `Next`.

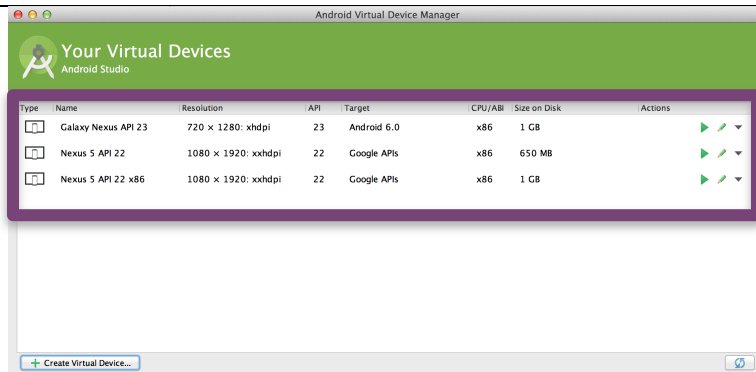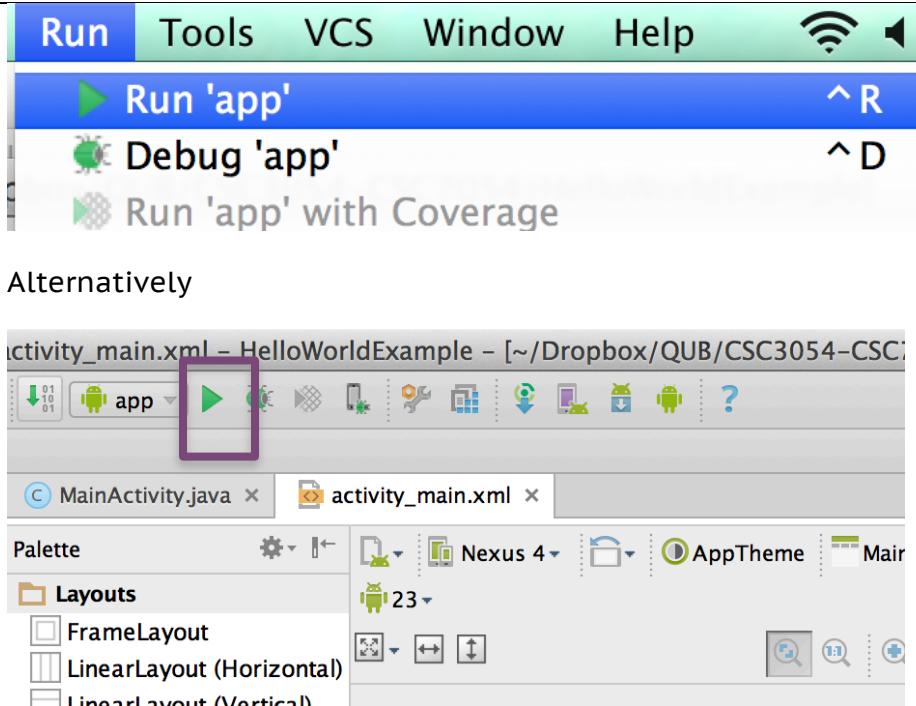|		
Step 4		Under `AVD Name`, give this new virtual device a name. Click on the `Show Advanced Settings` button.
Step 5		Another dialogue will appear. This allows a front or back facing camera to be included or changes the size of the device's SD card Press the `Finish` button.
Step 6		The new `AVD` will eventually show up in the list. `Click` and `start` the created `AVD`.

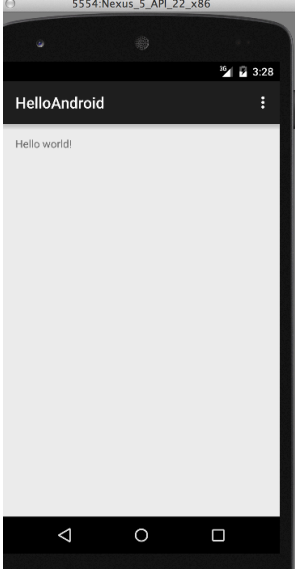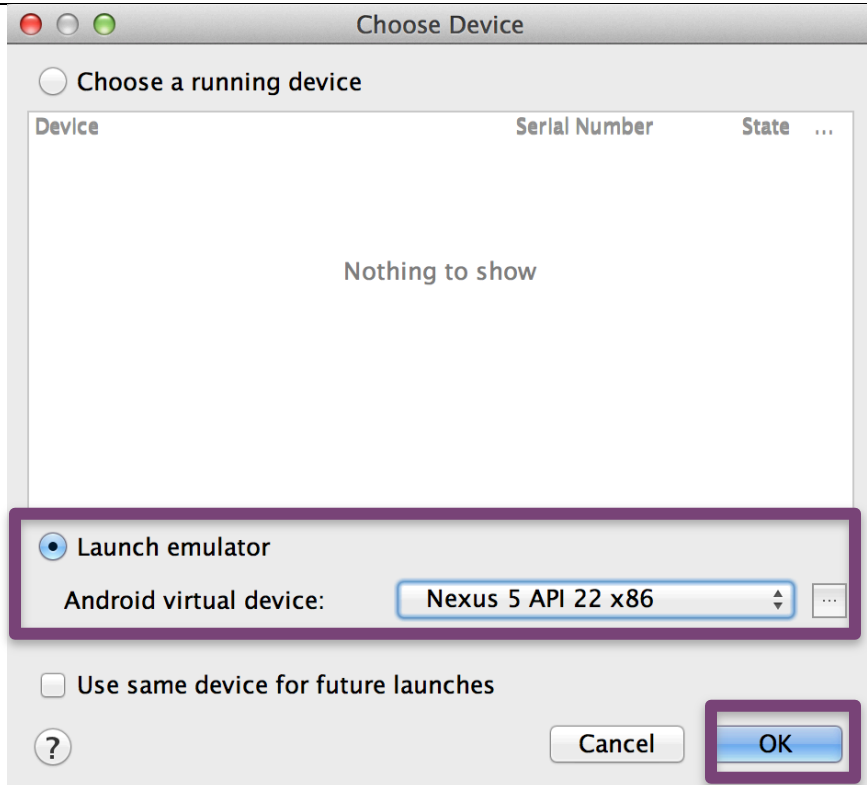|		
Step 7		Install and run `HelloAndroid` on the emulated phone, rather than on a physical device. Click `Run` and then `Run app`. Alternatively, select the `Run` icon.
Step 8		A dialog box will appear asking for a device to use. Select the virtual device and click `OK`.

Step 9			The IDE will install the application on the virtual device and then run it.

Why use an Emulator?

Advantages	Disadvantages
Emulators are cheaper than actual devices.	The Android emulator is really slow. This can be frustrating when rapidly experimenting and tuning an app
You don't have to buy all the devices that you may want to test on.	Some features are not well supported by the emulator. For instance, there's no support for Bluetooth connectivity and no support for connecting accessories to the emulator via USB cables.
The emulator allows you to easily configure hardware characteristics like the size of the SD card you have, the display size, and whether the device has a track ball.	Some software features aren't available by default in the emulator, so certain applications may not run on it.
Any modifications made are isolated to the emulated device so you don't have to worry that your testing will mess up your phone or its data	The emulator is not the device. It is not possible to really know how an app is going to look and perform on an actual device just by looking at how it performs in the emulator.