

# Android for .NET Developers Series

## Getting Started

### Setting Up Your Environment

Jim Wilson

[jimw@jwhh.com](mailto:jimw@jwhh.com)

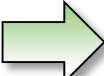
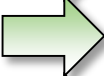
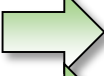


@hedgehogjim

<http://facebook.com/hedgehogjim>



**pluralsight**  
hardcore developer training

# Outline

-  What do we need?
-  Android Virtual Devices (a.k.a. Emulators)
-  Debugging using a real device
-  Special considerations for Android 4.2.2 and newer
-  Stay awake

# What do we need

## ➡ Android Developer Tools (ADT) Bundle

- ❑ Eclipse Integrated Dev Environment (IDE)
- ❑ ADT Plugin
  - ❑ Adds Android development features to Eclipse
- ❑ Android SDK Tools
- ❑ Android Platform Tools
- ❑ Android Platform
- ❑ Android emulator Images

## ➡ Java Development Kit (JDK)

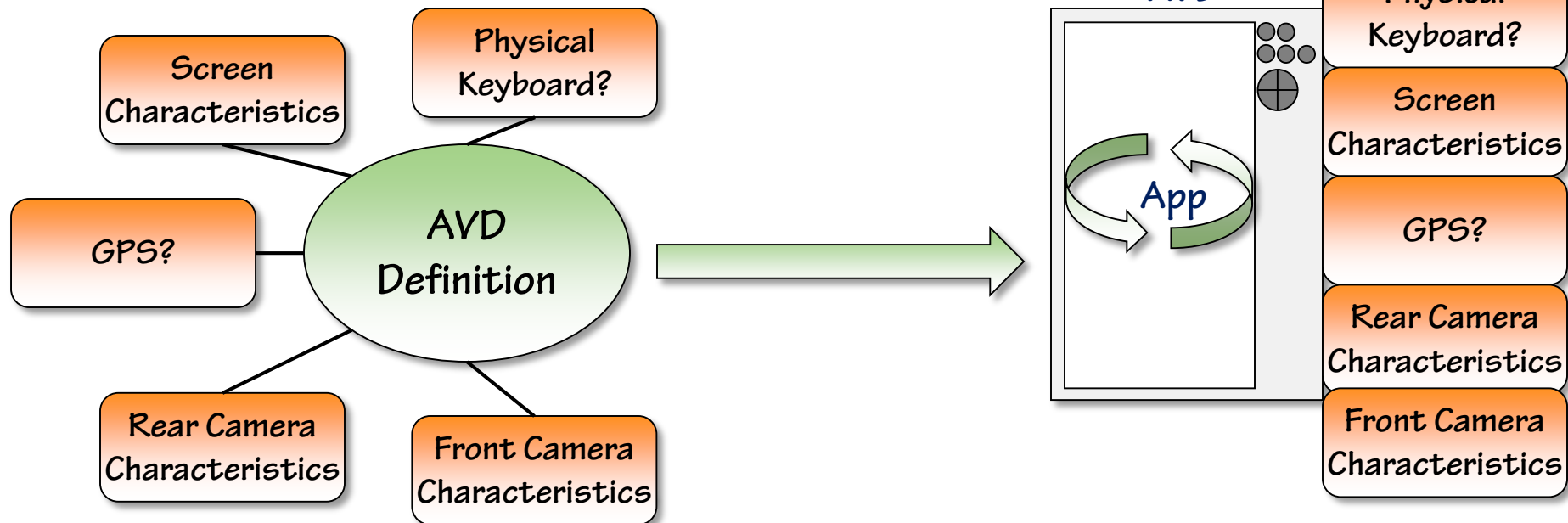
- ❑ Java compiler
- ❑ Java libraries



*Be sure to check  
Android SDK Requirements  
For required JDK version*

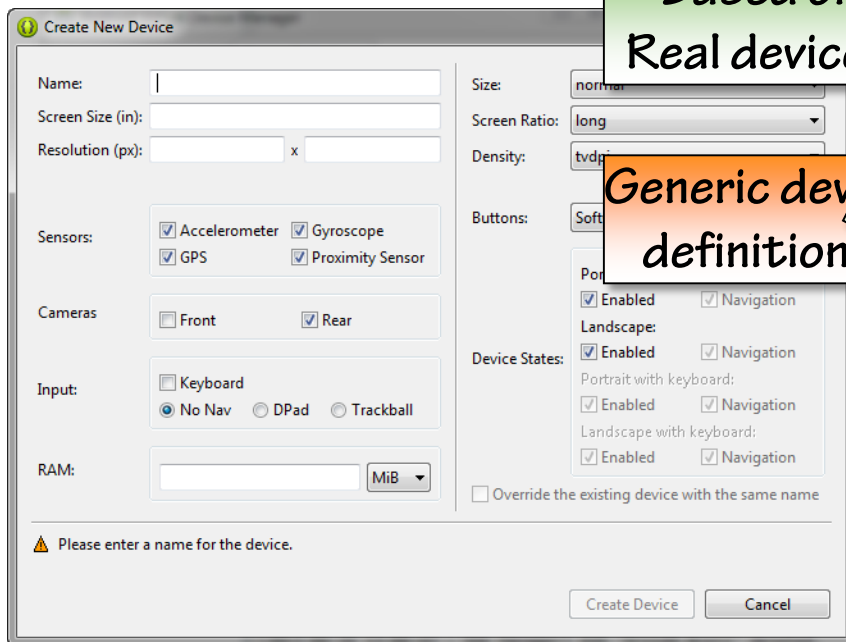
# Android Virtual Devices (Emulators)

- ➔ **AVDs/Emulators allow you to test apps on your desktop**
- ➔ An AVD definition describes characteristics of a device
  - ❑ Can be based on a real on-the-market device or generic set of characteristics
- ➔ You create an AVD image from a definition
  - ❑ You can make modifications to some definition characteristics
  - ❑ Many images can be created from the same definition
  - ❑ You can run your code on an AVD image



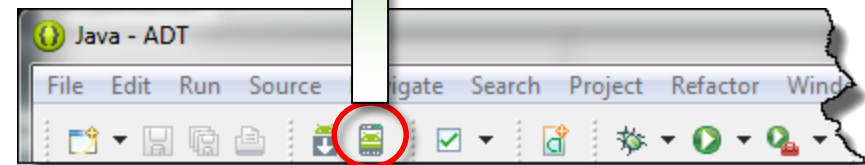
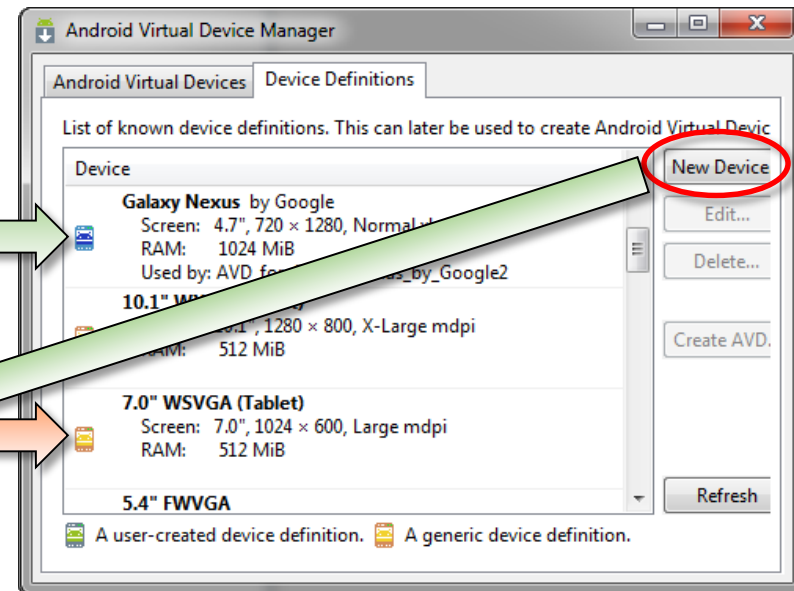
# Android Virtual Device Manager

- ➔ Use Android Virtual Device Manager to create and access AVDs
- ➔ Provides access to device definitions & AVDs
  - ❑ Preconfigured starting definitions
  - ❑ Can create definitions from scratch
- ➔ Can launch directly from Eclipse toolbar



Based on  
Real devices

Generic device  
definitions



# Creating Android Virtual Devices

Create new Android Virtual Device (AVD)

AVD Name: AVD\_for\_Galaxy\_Nexus\_by\_Google

Device: Galaxy Nexus (4.65", 720 × 1280: xhdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☒ Hardware keyboard present

Skin: ☒ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: ~~1024~~ 512 VM Heap: 64

Internal Storage: 200 MiB

SD Card: ☒ Size: 32 MiB ☐ Browse...

Emulation Options: ☐ Snapshot ☐ Use Host GPU

☐ Override the existing AVD with the same name

On Windows, emulating RAM greater than 768M may fail depending on the system load. Try progressively smaller values of RAM if the emulator fails to launch.

OK Cancel

Select desired definition

Android Virtual Device Manager

Device Definitions

List of known device definitions. This can later be used to create Android Virtual Device

Device
Galaxy Nexus by Google Screen: 4.7", 720 × 1280, Normal mdpi RAM: 1024 MiB Used by: AVD_for_Galaxy_Nexus_by_Google
10.1" WXGA (Tablet) Screen: 10.1", 1280 × 800, X-Large mdpi RAM: 512 MiB
7.0" WSVGA (Tablet) Screen: 7.0", 1024 × 600, Large mdpi RAM: 512 MiB
5.4" FWVGA

New Device  
Clone...  
Delete...  
Create AVD.  
Refresh

A user-created device definition. A generic device definition.

# Debugging using a real device

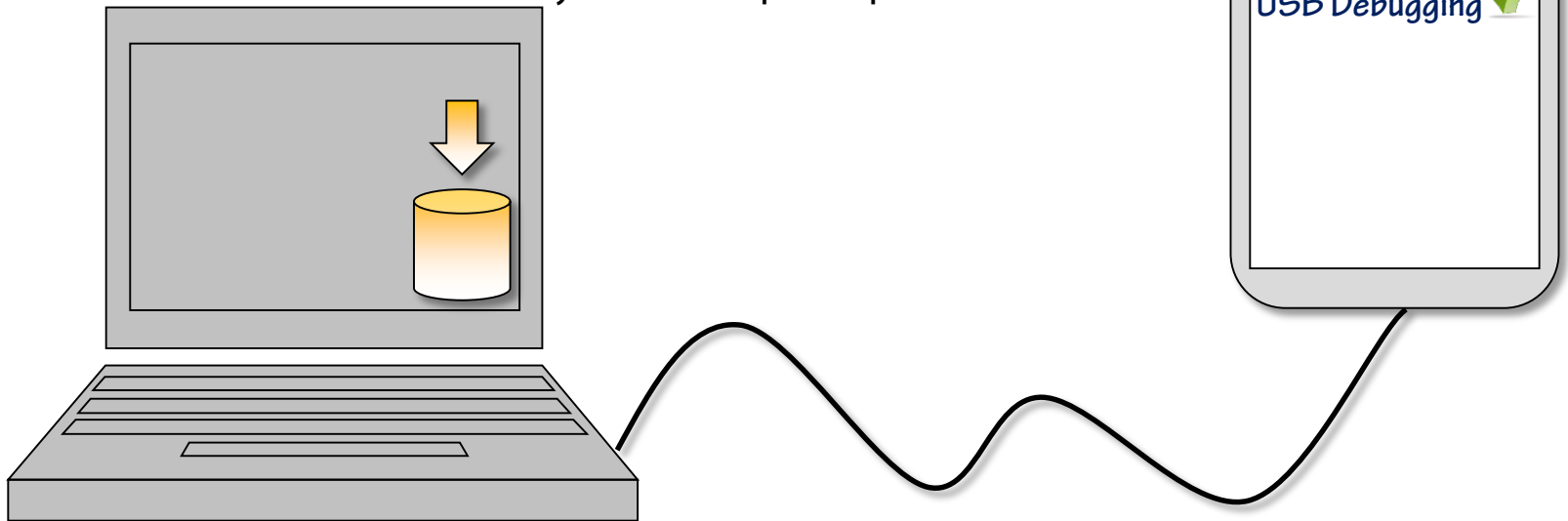
➡ **Debugging using a real device must be part of your testing**

➡ AVDs are not perfect replications of real devices

- ❑ Some features are difficult to simulate
- ❑ Some features behave slightly different

➡ **Debugging on a real device requires extra setup**

- ❑ Enable USB debugging on your phone
- ❑ Install USB drivers on your desktop computer



# Enable USB debugging on a device

- ➔ Enabling USB debugging is easy ... well ... sort of
- ➔ Simply check the “USB debugging” entry on the appropriate screen
- ➔ The location of the checkbox varies by Android version

- On most pre-Android 4.0 devices
  - Settings > Applications > Debugging
- On most Android 4.0 and 4.1 devices
  - Settings > Developer Options
- And then there's Android 4.2 and newer devices...



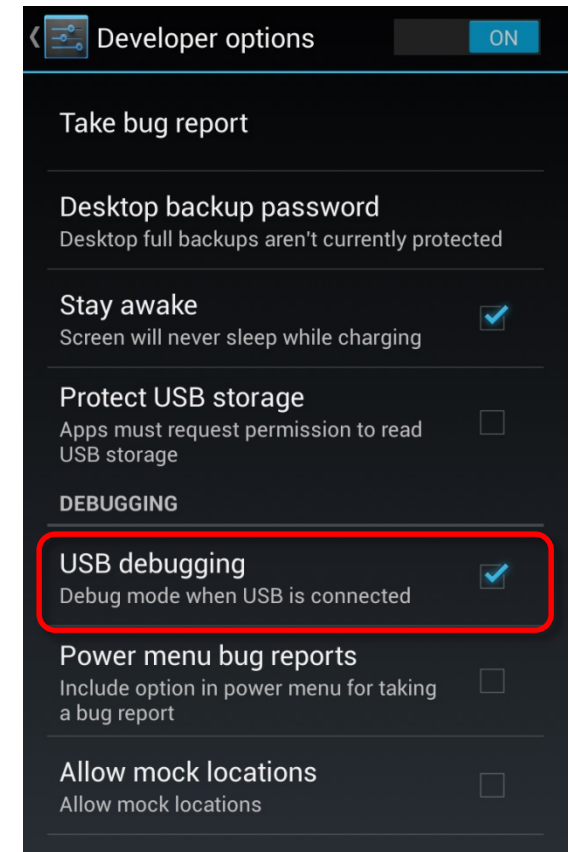
< 4.0 <=



Applications

Debugging

Developer Options





# Enable USB debugging on Android 4.2 device

➡ Starting in Android 4.2, Developer Options is hidden

➡ You must specifically make Developer Options available

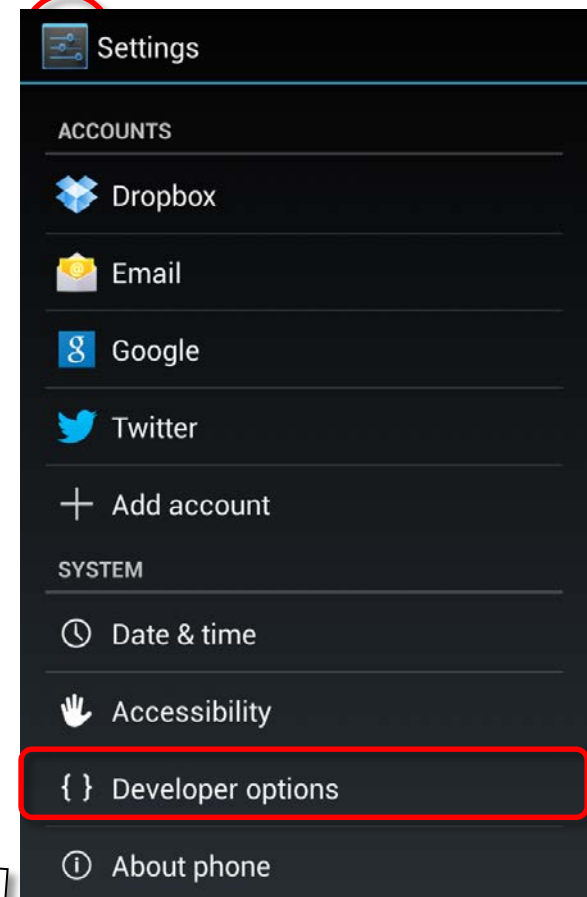
- ❑ Open Settings menu
- ❑ Select "About phone"
- ❑ Scroll until "Build number" is visible
- ❑ Tap on the "Build number" 7 times

➡ Once enabled, the same as 4.0 & 4.1 devices

- ❑ Settings > Developer options



About Phone



# Install USB drivers on your desktop

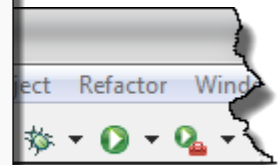
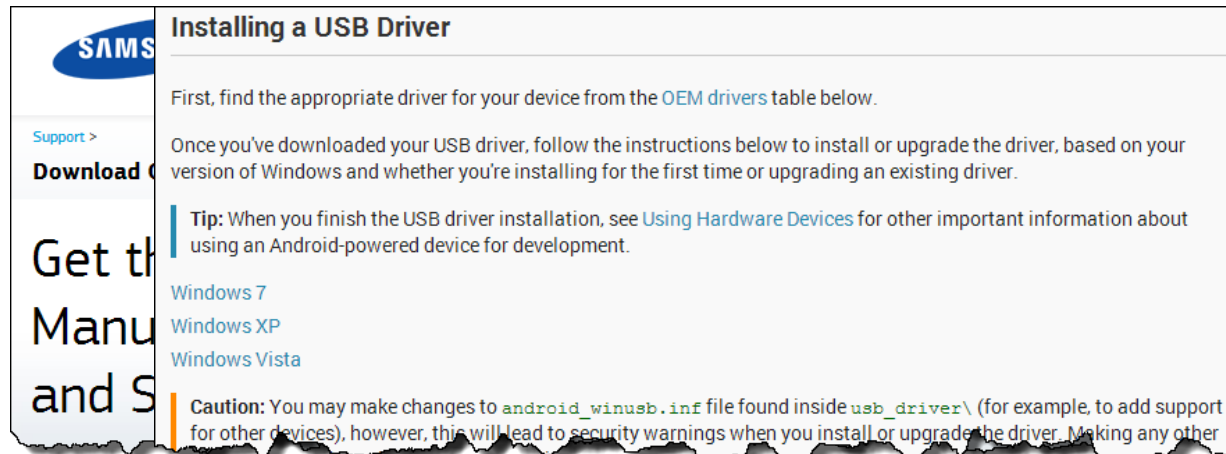
➡ The USB driver enables dev tools to communicate with device

➡ You must download the appropriate USB driver for your device

- ❑ Most “Android Developer Phones” use the Google USB Driver
  - ❑ Available through SDK Manager
  - ❑ Examples of phones this applies to are Nexus 1 and Nexus s
- ❑ The Galaxy Nexus relies on a Samsung provided driver
  - ❑ Download from: <http://bit.ly/13GS2ML>
- ❑ For all other phones, you must download the manufacturer drivers
  - ❑ List of manufacturer download sites: <http://bit.ly/15Va7Lc>

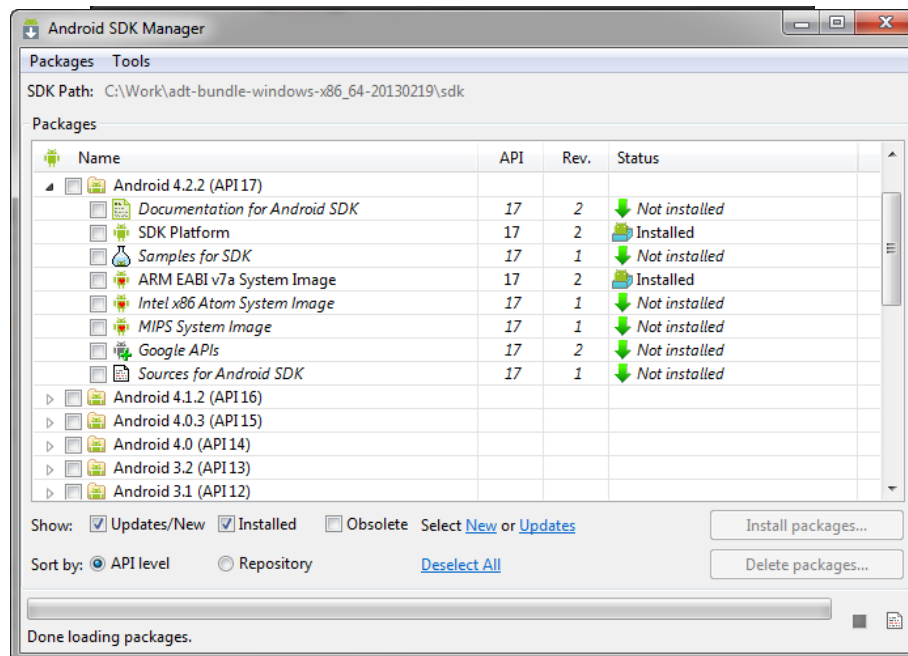
➡ Install the USB driver much like any other

- ❑ Specific instructions for each version of Windows at <http://bit.ly/11RhKy6>



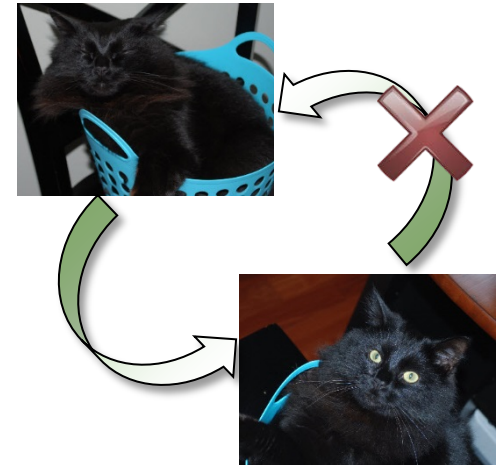
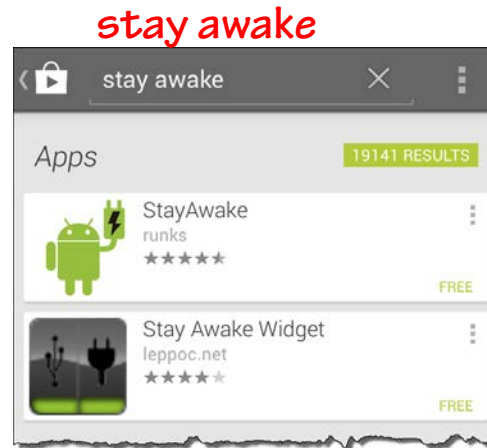
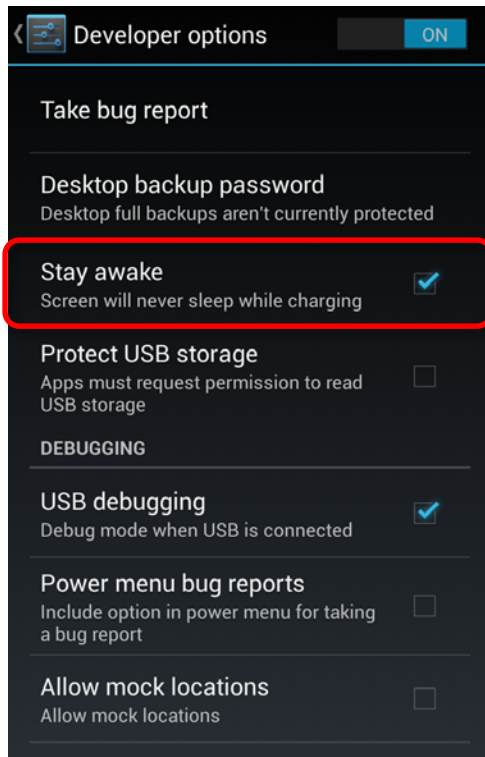
# Special considerations for Android 4.2.2 or newer

- ➡ Android 4.2.2 increased the security on USB debugging
- ➡ Device prompts you the first time it's connected to a desktop
  - ❑ Will present an RSA key for that desktop
  - ❑ You must tell the device to accept the key to be able to debug
- ➡ You must be sure that you have an up-to-date version of the tools
  - ❑ Needs to be Android Platform Tools r16.0.1 or newer
  - ❑ If you installed the ADT Bundle in the past, use SDK Manager to update



# Stay Awake

- ➔ A helpful debugging setting is to keep the device awake
- ➔ Device going to sleep can be annoying during debugging
- ➔ Can set device to stay awake when connected to USB
- ❑ Many devices have setting on Developer Options screen
  - ❑ If no device option, can download app that provides this behavior



# Summary

 **Android Developer Tools (ADT) Bundle is the core of development**

 **Need to also install JDK**

- Be sure you download and install the correct version

 **Android Virtual Devices allow testing of apps on desktop**

- AVD definitions represent device configurations
- Must make an image from a definition to run your code

 **Testing your code on a real device requires some extra setup**

- Enable USB debugging on device
  - Remember that option is hidden on Android 4.2 and newer
- Install appropriate USB driver on desktop

 **Enable Stay Awake feature to ease debugging effort**