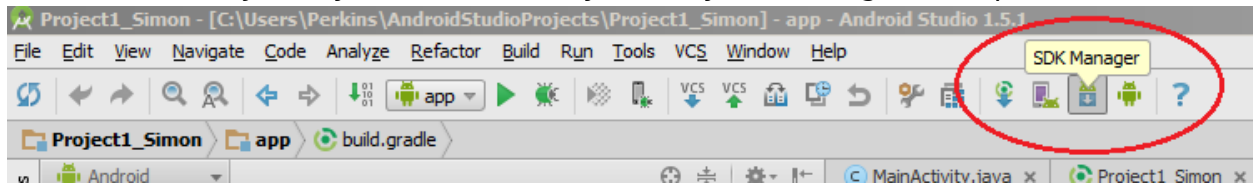


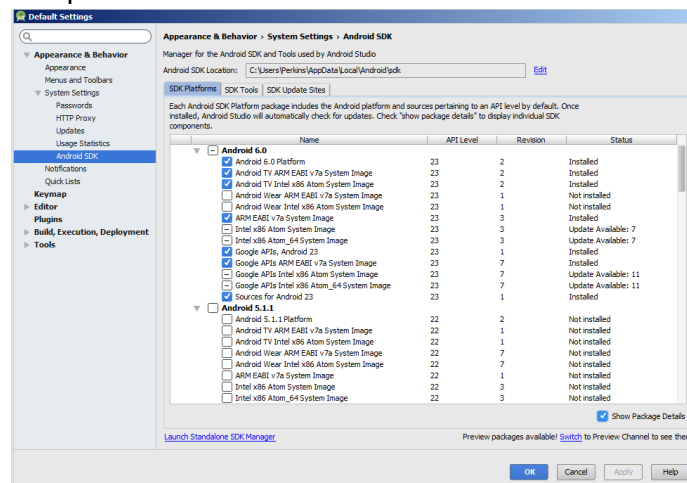
Install and Configure Development Environment

Install latest Android Studio and SDK <http://developer.android.com/sdk/index.html>
(Based on IntelliJ so if you use that or Pycharm you are in good shape)



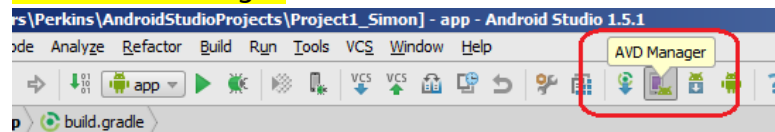
Go to SDK manager

- Note the number of versions of Android, 1 for every major release and each device is running one of these versions
- tools and extras, at least your phone version. You target particular versions of Android for your apps, make it as wide as possible to make it appeal to as many users using as many different versions as possible. But keep in mind that each new release added new features, if you make it backwardly compatible to far (1 for instance) then you cannot use apis that were introduced in 3. Here I am using 6



go to SDK tools and show off build tools and the critical Google USB Driver needed for Nexus devices (Windows only)

Go to AvdManager



AVD manager – manage virtual devices to test your app on can have as many as you want different sizes and APIs. Used to be slow, very fast now.

Can debug on Emulator or physical device. Emulator is much faster but is simulated and may be missing hardware (gps, accelerometer etc..) AS has a process called

adb (Android Debug Bridge) which communicates with emulated and physical devices.

Connect Physical device

Enable developer mode on device (Nexus –Toggle on "USB Debugging" in the "Developer Options" area of Settings.). If you do not see "Developer Options", go into "About device" in Settings and tap on the "Build number" entry seven times, which will unlock "Developer Options".

configure and use USB device driver for windows if necessary (via Windows device manager)

Enable Debugging

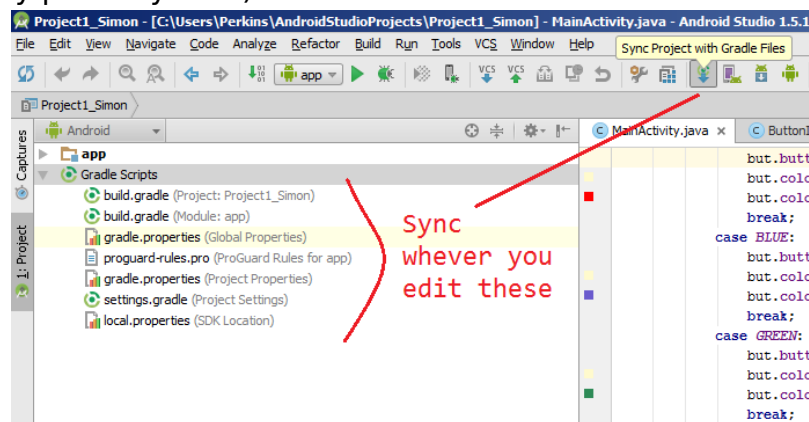
Demo on Nexus 7

Go to Debug Options

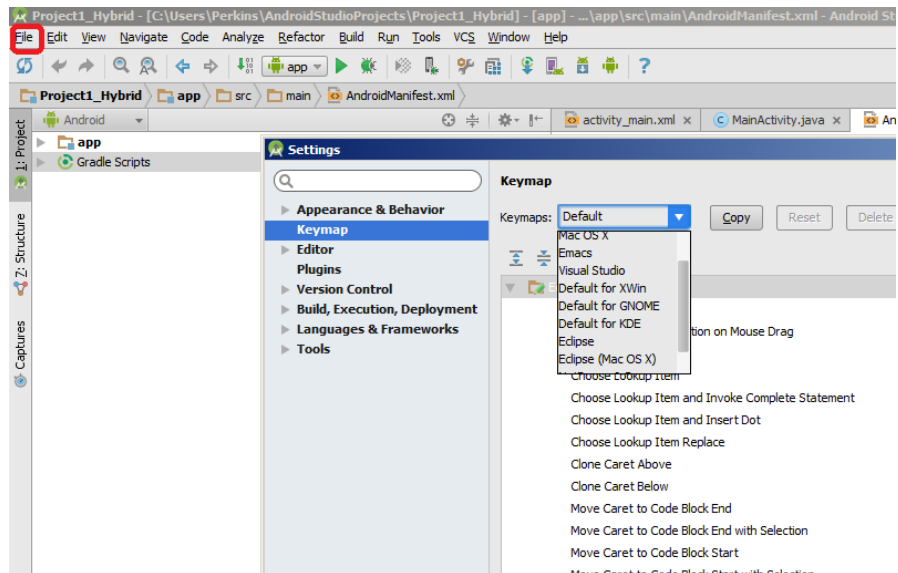
Stay Awake (otherwise you have to constantly unlock your app when debugging)

Usb Debugging

Whenever you change a system file resync your gradle files. It will rebuild your project (possibly painfully slow)

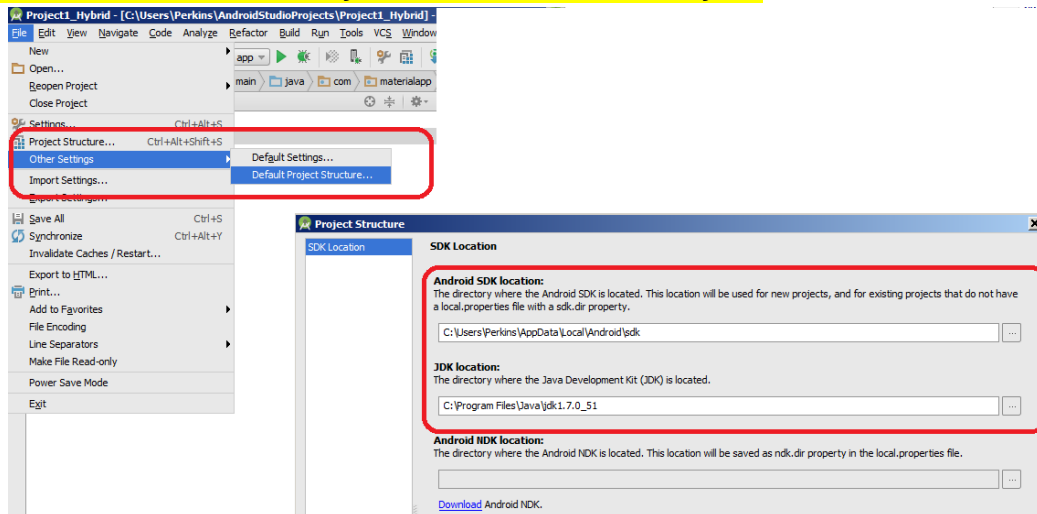


Keyboard shortcuts (productivity boost) can use defaults or configure how you like

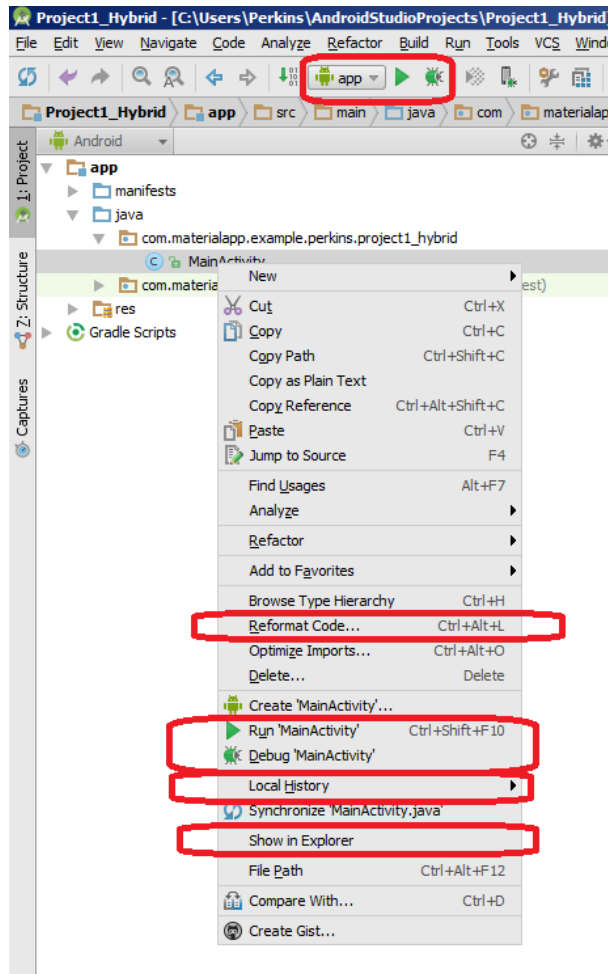


I like AS defaults, here are the ones I use a lot they are in a word file on scholar (Hotkeys I use). Here they are in a note I keep floating nearby

And where is the AS SDK and JDK? AS installs its own JRE.



Useful File Stuff



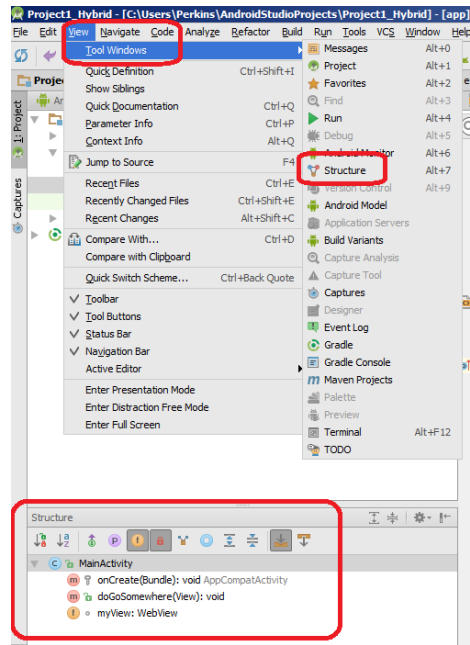
Manifest and Resources (much more later)

Manifest defines your app, what **permissions** it needs (read by google play and used to ask you “do you want to allow this app to...” and what the startup UI is, what other components are defined and used.

Res folder

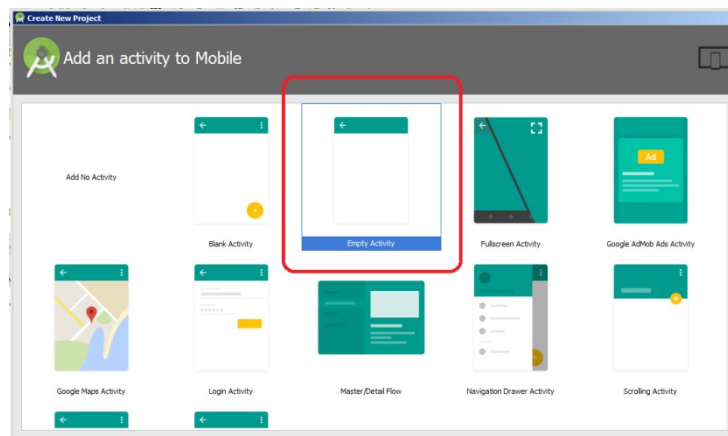
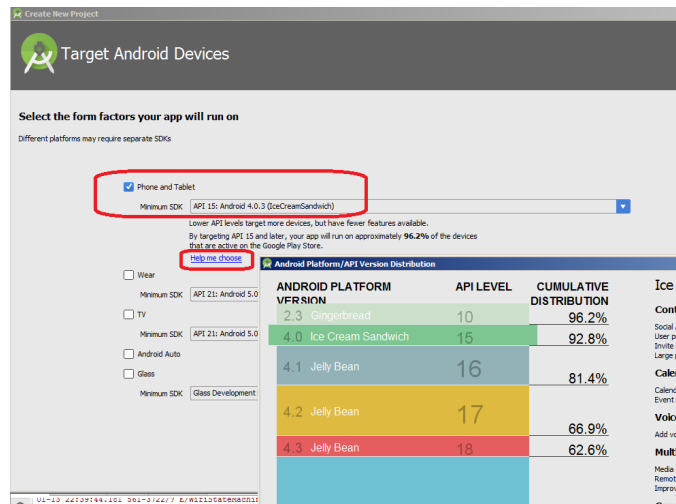
Holds xml layout files (they describe UI)
Strings you will use, colors, drawables, themes

Use Structures view to see object hierarchy



Development

New Project
Package
Minimum sdk



Show

Build, Run

Debug

Show

logcat (alt-6 to restore)

Breakpoints (including conditional, right click on red breakpoint and add a boolean condition)

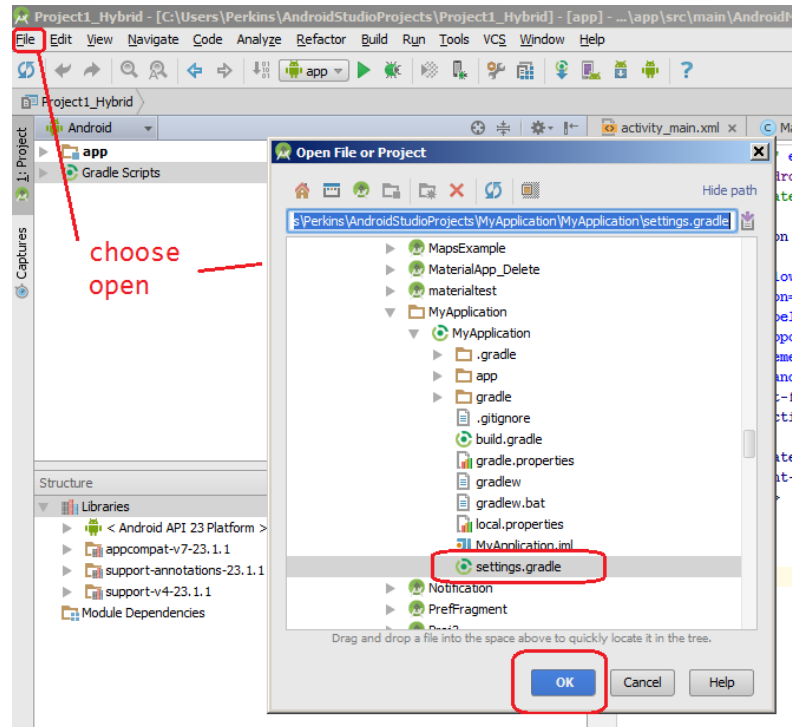
Show ctrl-j to make tag and log statement

show uninstall, reinstall

clean

close project

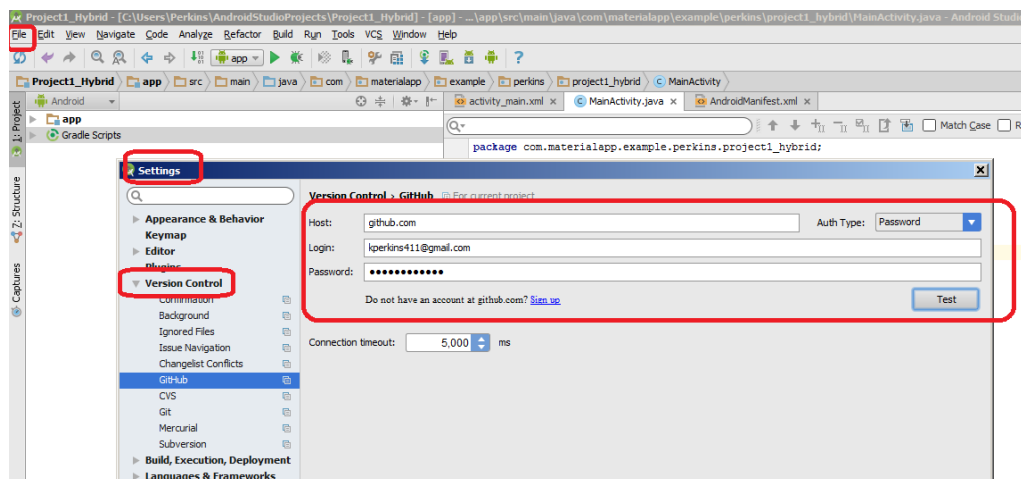
Find (show in folder)
Zip it
Delete orig project
Unzip
Import project (via settings.gradle)



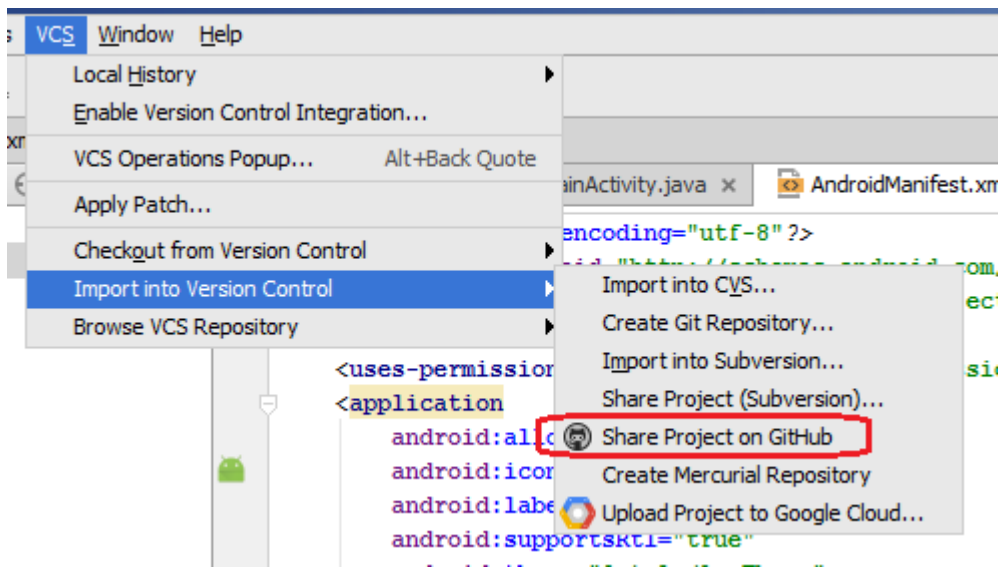
Ready to go

Source Control- I use git and GitHub

First set it up

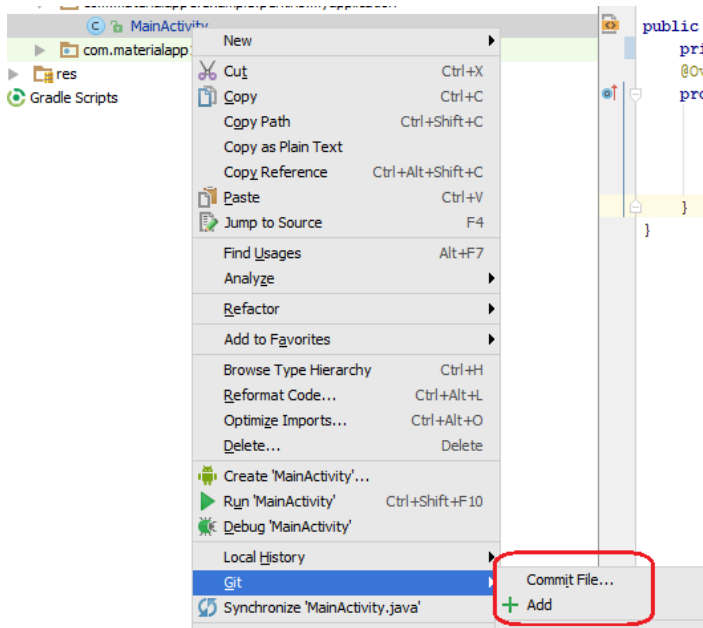


Share on Github

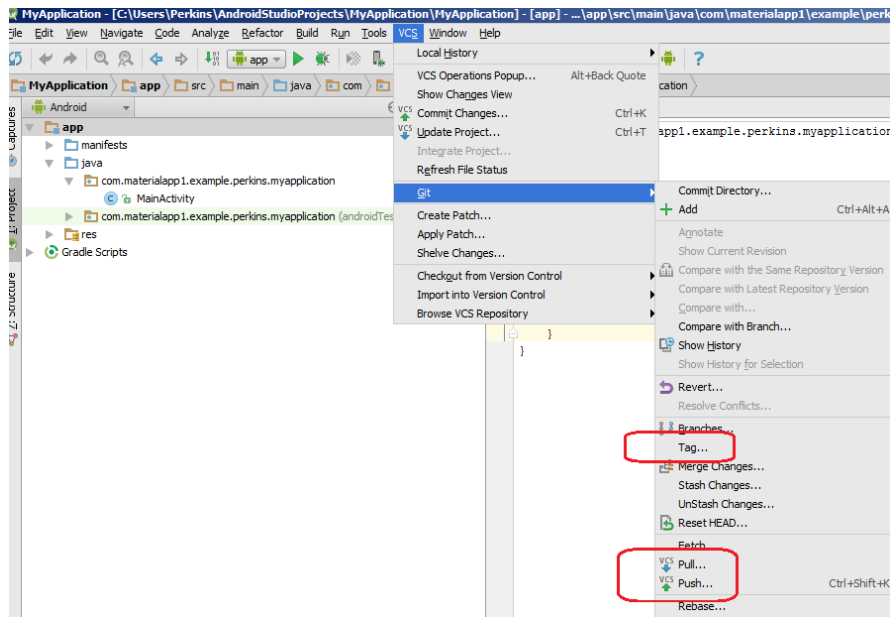


Will push it on up to github if your account is setup

Now change a file and it turns blue in the Project Window, so commit it locally.



To push up, or tag



Tips:

If everything goes wonky kill AS (should kill adb.exe) and restart. If adb.exe remains running kill it and then restart AS.

If you cannot connect to device over usb cable suspect the cable! Try another, better yet try one that you have verified with another device