Appium Setup on Mac:

1. **Download and install java jdk**

<https://www.oracle.com/in/java/technologies/javase-jdk11-downloads.html> - latest version

<https://www.oracle.com/in/java/technologies/javase/javase-jdk8-downloads.html> - old version

Note: You may need to create account in Oracle to download java(as per latest policy)

1. **Download and install Android studio**

<https://developer.android.com/studio?gclid=Cj0KCQjwl9GCBhDvARIsAFunhsle5gtbsqJvTVZE11EEsQSRlXF0ZfuGOS03W_CKAExBW8cmh659R5MaAnQtEALw_wcB&gclsrc=aw.ds>

Note: At the time of preparing the doc, Android version is: 4.1.3

1. **Setup Environment variables in bash profile**

use commands to open bash\_profile

Go to terminal -> touch .bash\_profile or open .bash\_profile

Ex:   
export JAVA\_HOME=$(/usr/libexec/java\_home)

export ANDROID\_HOME=/Users/apple/Library/Android/sdk

export PATH=$PATH:$ANDROID\_HOME/tools

export PATH=$PATH:$ANDROID\_HOME/tools/bin

export PATH=$PATH:$ANDROID\_HOME/platform-tools

export PATH=$PATH:$ANDROID\_HOME/tools/lib

export PATH=$PATH:${JAVA\_HOME}/bin  
  
Note: Rvm path  
[[ -s "$HOME/.rvm/scripts/rvm" ]] && . "$HOME/.rvm/scripts/rvm" # Load RVM function' >> ~/.bash\_profile

**4)Install Homebrew:**  
open terminal and type below command  
/bin/bash -c "$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)"  
  
Note: Homebrew installs the stuff you need that Apple (or your Linux system) didn’t.

Homebrew installs packages to their own directory

**5)Install Node js and npm:**  
- NPM is a package manager for Node.js packages

- Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux  
You can download and install:

<https://nodejs.org/en/download/>

Note: to check node and npm is installed or not  
use below on terminal:   
node -v  
npm -v

**6) Install Appium via npm (to run Appium tests from terminal)**  
  
npm install -g appium - It will download latest version of Appium

npm install -g appium@1.18.1 **-**  to download specific version of Appium

npm install wd

**Note:** If you face permission issue’s, use below commands

id -un #Gives you your user name (Use it as owner below.)

sudo chown -R [owner] /usr/local/lib/node\_modules

**7)Download Appium Desktop(to inspect elements)**  
[**https://github.com/appium/appium-desktop/releases/**](https://github.com/appium/appium-desktop/releases/)

**8)Install Xcode:**  
Go to app store -> Xcode -> Provide Apple credentials to download  
  
9) **Important packages to run test on iOS devices:**  
**a) Carthage:** **brew install carthage**  
- Carthage is intended to be the simplest way to add frameworks to your Cocoa application.

- Carthage builds your dependencies and provides you with binary frameworks, but you retain full control over your project structure and setup. Carthage does not automatically modify your project files or your build settings  
 **b)** **Install libimobiledevice: brew install libimobiledevice**

- In order to run our tests on real devices, we need to install libimobiledevice dependency

**c)Install Authorize-ios: npm install -g authorize-ios**

[authorize-ios](https://github.com/appium/authorize-ios) is a utility which pre-authorizes to run UIAutomation scripts on iOS devices. We need this tool for running our tests on real devices.

**d)Install ios-deploy: brew install ios-deploy**

To deploy our apps to our devices by using ios-deploy.  We can install and debug apps from the command line without using Xcode

**e)Install ideviceinstaller**[ideviceinstaller](https://github.com/libimobiledevice/ideviceinstaller) helps us to install, uninstall, upgrade, archive, restore and enumerate installed or archived apps. We need this utility to run tests on real devices. In order to install it, run below commands:

**- sudo xcode-select -r**

**- brew install ideviceinstaller**

**g)Install ios\_webkit\_debug\_proxy: brew install ios-webkit-debug-proxy**In order to access web views on real iOS devices, we need to install this tool. Run the below command:

**10. Editor:**

Download and install VS code:

[**https://code.visualstudio.com/download**](https://code.visualstudio.com/download)

**11.** get App package and app activity  
adb shell dumpsys window | grep -E 'mCurrentFocus'