

Step 1: Install Java JDK

Set environment variables

JAVA_HOME

PATH

java -version

Step 2: Install Homebrew

Install Homebrew (it's a package management system that makes it easier to install software on Mac OS machines).

Please find Homebrew and follow the instructions on this website:

⇒ <https://brew.sh/>

Step 3: Install node js and npm

brew install node

nodejs.org > download LTS version> open .pkg and follow wizard to install

node -v

Npm -v

Step 4: Install Appium : npm install -g appium

Install Appium Desktop (optional)

Step 5: Get XCode

Xcode is the development and debug environment on Mac.

Provides required tools, files for dev/automation for Mac and iOS apps

Download Xcode:

⇒ <https://developer.apple.com/xcode/>

Step 6: Install XCode command line tools

Open terminal and run
`xcode-select --install`

Step 7: Create Apple ID

Goto Xcode - preferences - account
Add Apple ID

Step 8: Install Carthage

`brew install Carthage`

In order to launch WebDriverAgent, your macOS will need to have Carthage installed
(Not needed for automation on android)

Step 9: Install WebDriverAgent

WebDriverAgent is a WebDriver version for iOS. It serves for controlling iOS devices and apps remotely. You can find it in the following directories:

- If you installed Appium Desktop:
`/Applications/Appium.app/Contents/Resources/app/node_modules/appium/node_modules/appium-webdriver`
- If you installed Appium via Node.js:
`/usr/local/lib/node_modules/appium/node_modules/appium-webdriver`

To install:

- In the Terminal window, go to the directory specified above and run the following command:

```
mkdir -p Resources/WebDriverAgent.bundle  
./Scripts/bootstrap.sh -d
```

- Open the WebDriverAgent.xcodeproj project in Xcode (you can find the project in one of the directories specified above).

For

the **WebDriverAgentLib** and **WebDriverAgentRunner** targets, go to the **General** tab and select the **Automatically manage signing** check box there, and then select your development team.

- Build the project to verify that everything works fine:

```
xcodebuild -project WebDriverAgent.xcodeproj -scheme  
WebDriverAgentRunner -destination 'id=udid' test
```

Step 10: Open WebDriverAgent.xcodeproj in Xcode

Project name : WebDriverAgent.xcodeproj

Location :

For Appium Desktop

/Applications/Appium.app/Contents/Resources/app/
node_modules/appium/node_modules/appium-xcuitest-driver/
WebDriverAgent

For Appium Installed through node
/usr/local/lib/node_modules/appium/node_modules/appium-
xcuitest-driver/WebDriverAgent

<https://github.com/appium/appium/issues/12586>

For WebDriverAgentLib and WebDriverAgentRunner targets, go
to general tab and select "Automatically manage signing", and
then select your Development Team

Xcode may fail to create a provisioning profile for the
WebDriverAgentRunner target:

Failed to create a provisioning profile
No profile for '.....' were found

manually change the bundle id for the target by going into the
"Build Settings" tab, and changing the "Product Bundle
Identifier" from com.Facebook.WebDriverAgentRunner to
something that Xcode will accept:

Going back to the "General" tab for the WebDriverAgentRunner
target, you should now see that it has created a provisioning
profile and all is well:

Clean - Build - Run

Build the project to verify that everything works fine:

```
xcodebuild -project WebDriverAgent.xcodeproj -scheme  
WebDriverAgentRunner -destination 'id=udid' test
```

udid = device UDID

Example :

Lets say udid = A9F18306-6B22-442B-99F4-F217A5EE8D38

xcodebuild -project WebDriverAgent.xcodeproj -scheme

WebDriverAgentRunner -destination

'id=A9F18306-6B22-442B-99F4-F217A5EE8D38' test

To get the udid of the real device connected

npm install -g ios-deploy

ios-deploy -c

OR

instruments -s devices

Step 11: Create a project and add desired capabilities for automation on iOS

deviceName

platformName

platformVersion

app

Open Eclipse

Create a new maven project or use any existing maven project

Add maven dependencies

appium

selenium