**Setup and Execution Guidelines**

Table of Contents

[Prerequisites: 2](#_Toc353822021)

[Set up: 2](#_Toc353822022)

[Script Execution: 8](#_Toc353822023)

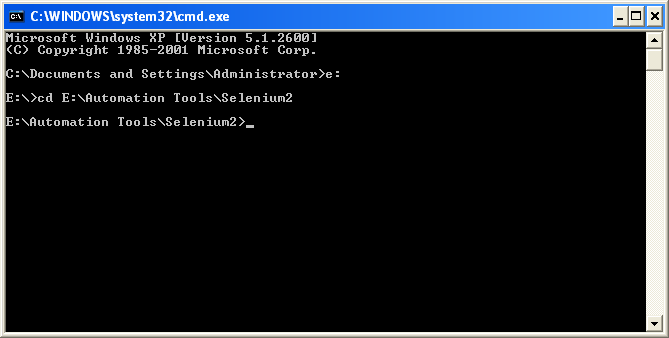
[Report: 10](#_Toc353822024)

# Prerequisites:

1. Python should be installed and set up path in your machine.
2. Download xmlrunner.py from below url: <http://pypi.python.org/pypi/XmlTestRunner/0.16654>
3. Download and install “setuptools 0.6c11” from :<http://pypi.python.org/pypi/setuptools>
4. Install selenium 2 in your system using command: > easy\_install selenium
5. Java should be installed and setup path in your machine.
6. Download “IEDriverServer.exe” from here: <https://selenium.googlecode.com/files/IEDriverServer_Win32_2.31.0.zip>and placed it on local machine.
7. Download “chromedriver.exe” from location: <https://chromedriver.googlecode.com/files/chromedriver2_win32_0.7.zip> and extract it on local machine.
8. Download “selenium-server-standalone-2.30.0.jar” from “<http://docs.seleniumhq.org/download/>”.

# Set up:

1. Download the latest package from email and extract on your machine.
2. Open “/Config/setting.ini” and change the **runMode** accordingly.
3. **Launching Hub:** Follow below mentioned steps to run the hub:
4. Open command prompt
5. Goto your selenium server jar file location:



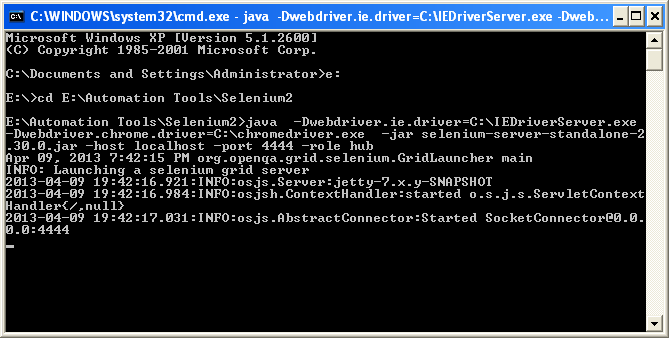
1. Run below command to launch the hub:

“*java -Dwebdriver.ie.driver=<path to IEDriverServer.exe -Dwebdriver.chrome.driver=<path to chromedriver.exe > -jar selenium-server-standalone-2.30.0.jar -host localhost -port 4444 -role hub*”

Example:

*java -Dwebdriver.ie.driver=C:\IEDriverServer.exe -Dwebdriver.chrome.driver=C:\chromedriver.exe -jar selenium-server-standalone-2.30.0.jar -host localhost -port 4444 -role hub*

1. You should see below screen:



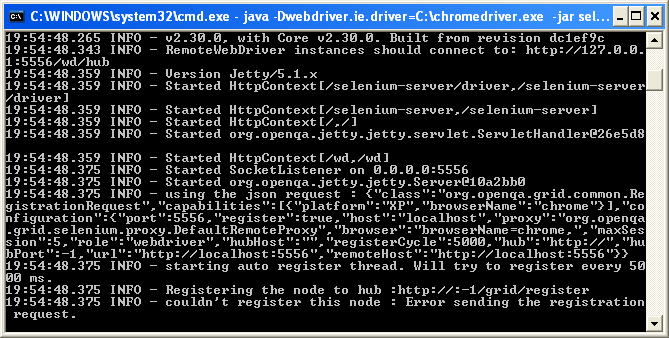
1. **Launching Node 1:** open command prompt go to you selenium server jar file and execute command:

“java -Dwebdriver.ie.driver= *<path to IEDriverServer.exe>* -jar selenium-server-standalone-2.30.0.jar -host localhost -port 5555 -role webdriver -hub http://localhost:4444/grid/register -browser browserName=iehta,platform=WINDOWS”



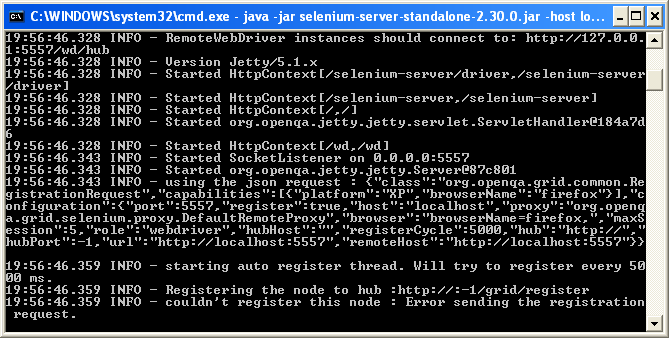
1. **Launching Node 2:** open another command prompt go to you selenium server jar file and execute command:

“java -Dwebdriver.chrome.driver= *<path to chromedriver.exe>* -jar selenium-server-standalone-2.30.0.jar -host localhost -port 5556 -role webdriver -hub http://*localhost*:4444/grid/register -browser browserName=chrome, platform=WINDOWS”:

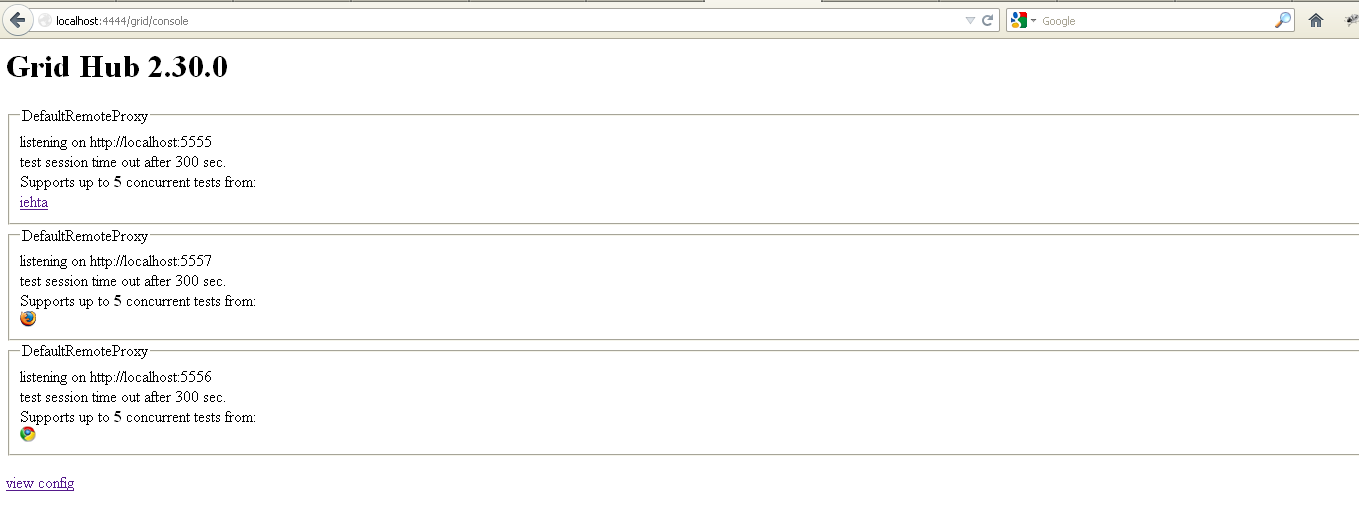


1. **Launching Node 3:** open another command prompt go to you selenium server jar file and execute command:

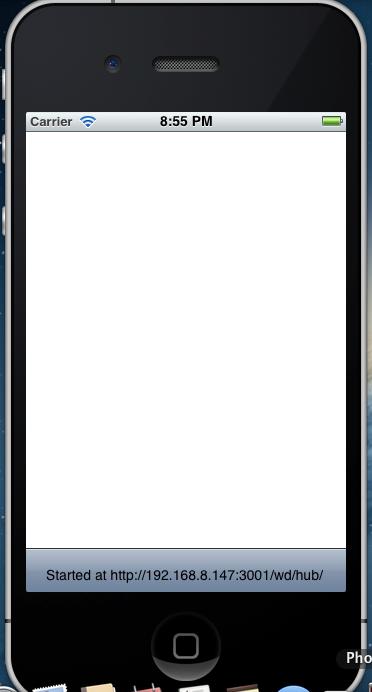
“java -jar selenium-server-standalone-2.30.0.jar -host localhost -port 5557 -role webdriver -hub http://*localhost*:4444/grid/register -browser browserName=firefox, platform=WINDOWS”:



1. Open “localhost:4444/grid/console on browser and you should get below screen:



1. **Launch Node on iPhone Simulator:** Download iWebDriver using the svn checkout from here: <https://code.google.com/p/selenium/source/checkout>
   1. Open “\iphone\iWebDriver.xcodeproj” project in Xcode.
   2. Select build configuration to Simulator / iPhone OS 5.0 / iWebDriver and run the project.
   3. You would see iOS simulator on the screen:



* 1. Next, Goto the Setting>>iWebDriver on iPhone simulator.
  2. Enter the IP of HUB machine at “HOST” field.
  3. Enter the port of HUB machine at PORT field.



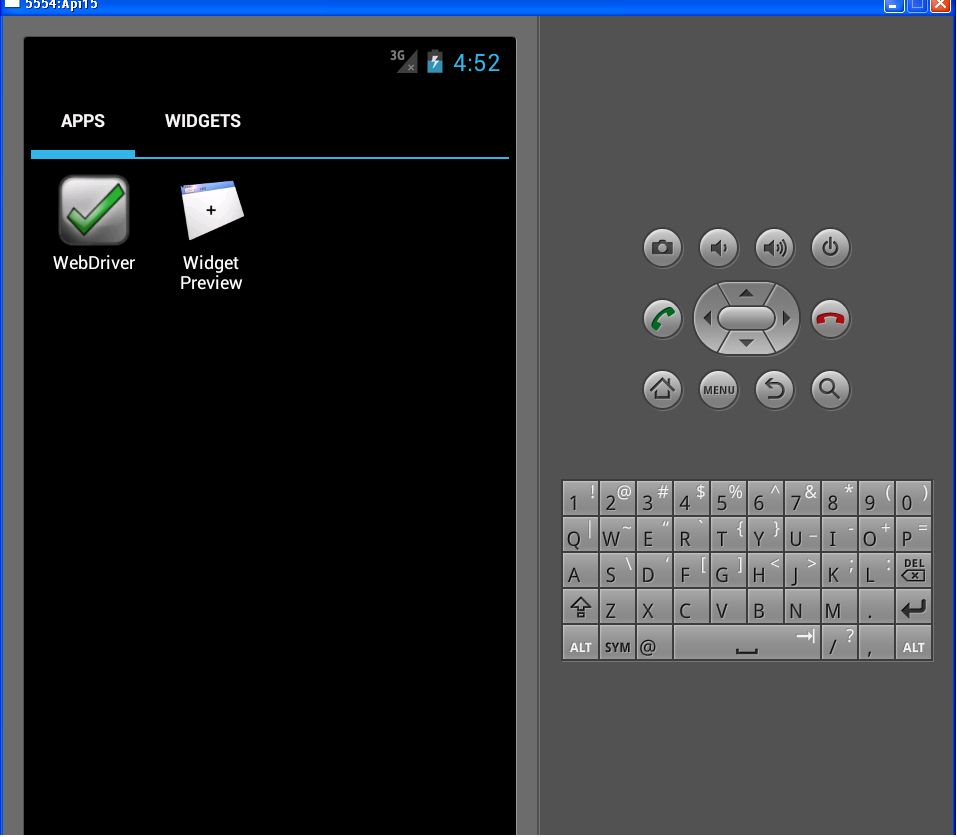
* 1. It automatically registers itself on the Hub providing the capability to run tests on the

iOS Safari browser.

1. **Launch Node on Android Emulator:** Download android server apk from : <http://code.google.com/p/selenium/downloads/detail?name=android-server-2.21.0.apk>and placed this into “\android-sdk-windows\platform-tools” location.
2. Create an android emulator of API 15 which must have SD card memory and launch the same.
3. Open command prompt and goto “\android-sdk-windows\platform-tools” location and run below command to install the server on emulator:

*adb install android-server-2.21.0.apk*

1. You will see webdriver installed in emulator:

**

1. Open cmd and run below command
   1. pip install flynnid

*Refer this link:* [*https://github.com/davehunt/flynnid*](https://github.com/davehunt/flynnid)

1. Once flynnid has been installed, open cmd and run following command to launch the node:

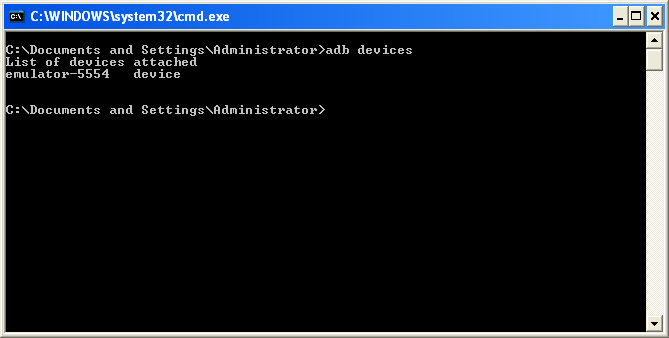
*flynnid --hubhost=<hub’s system IP address> --hubport=<hub port> --nodehost =<node’s syatme IP address> --nodeport=5559 --browsername=android --browserver=4.0.3 --platform=ANDROID*

*example: flynnid --hubhost=192.168.1.181 --hubport=4444 –nodehost=192.168.1.181 --nodeport=5559 --browsername=android --browserver=2.3.3 --platform=ANDROID*

1. Open one more command prompt and run below command:

***adb devices***

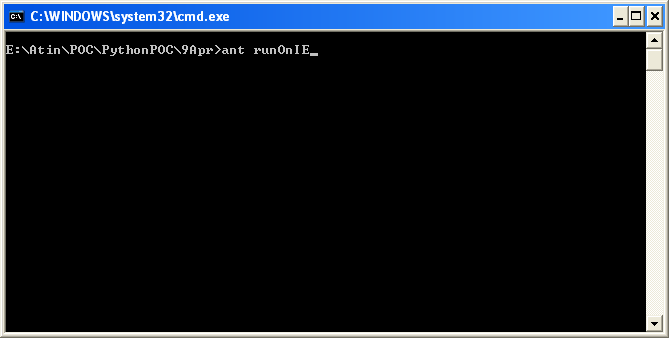
This command will provide the serial ID of the device.

**

Update the same serial ID in build.xml file at line no.11.

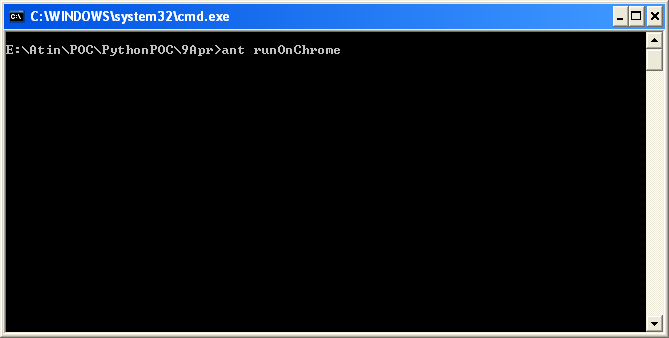
# Script Execution:

1. Open command prompts and go to your script directory and run below command to execute the tests on IE browser:
   * ant runOnIE

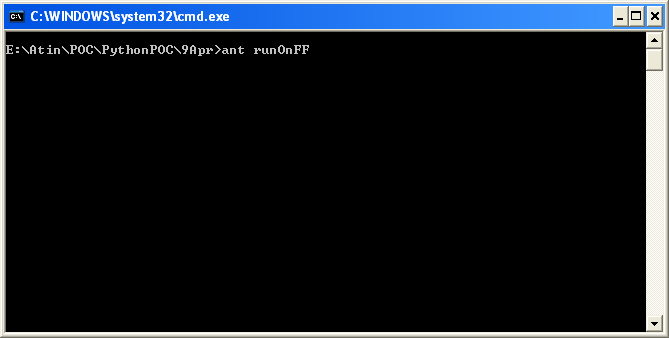


IE browser would gets launched and tests starts running on browsers.

1. Open command prompts and go to your script directory and run below command to execute the tests on chrome browser:
   * ant runOnChrome



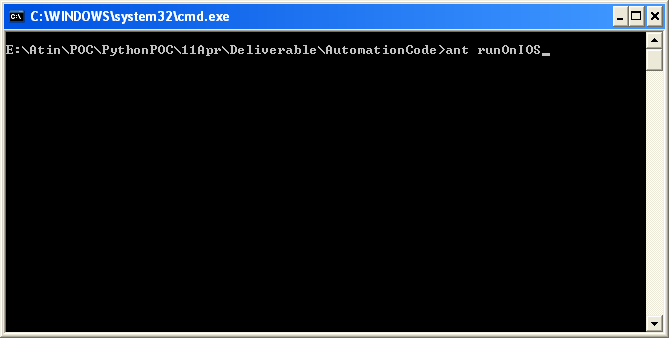
1. Open command prompts and go to your script directory and run below command to execute the tests on Firefox browser:
   * ant runOnFF



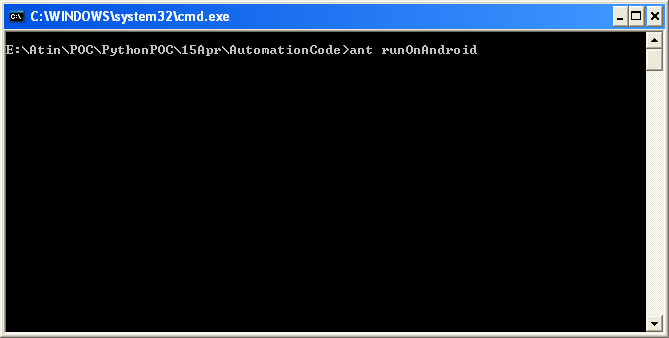
1. use below command to run tests on all three browser in parallel:
   * ant runAll or ant

All three browsers would invoke and tests execution would be started.

1. Open command prompts and go to your script directory and run “**ant runOnIOS**” command to the test on iPhone simulator.



1. Open command prompts and go to your script directory and run “**ant runOnAndroid**” command to the test on android emulator:



# Report:

Go to your project location and open **Reports** directory you should see three xml files are generated one with suffix with “chrome”, “iehta” and “firefox” ,open xml files to view the reports in XML format.

