|  |  |
| --- | --- |
| D:\PAL\Logo_4_21_15\Logo_4_21_15\Primary Logo\png_files\EPAM_LOGO_Full_Color_RGB.png | CDP Automated Testing Mentoring Program  2018Q2 |

Module 7: Selenium WD Advanced with JavaScript: Actions, JavaScript Executor, Selenium Server + Selenium Grid

## home task

1. Convert your project from previous hometasks (Selenium WebDriver, hometask #1, #2) – add at least 2 new Actions implementations (for example: 1 for mouse actions and 1 for keyboard).
2. Add at least 2 implementation of JavaScript Executor usage (for example: implement element highlighter, JS Executor based clicker or element finder);
3. Set up Selenium Grid with at least two nodes and run all your test suites on the grid.

## home task

As a basis for Module 7 task I took my home task from Module 5 for WebdriverJS part.

I separated the code into 2 folders:

* actions\_executors
* grid

To run and view result for actions and executors part:

* copy code from repository https://github.com/Alexey-Kachemirov/js\_atm.git
* open ‘module7/actions\_executors’ folder
* install project from copied folder running ‘npm install’
* download ‘selenium-server-standalone-2.37.0.jar’ and copy in the project folder
* copy file with credentials in the project folder
* run command from saml-idp project
* start bat file ‘run.bat’
* start js file index.js

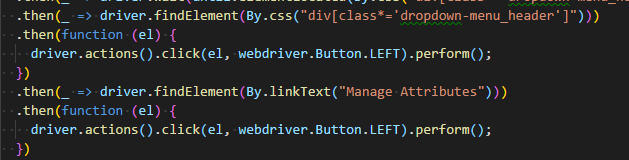
To run and view result grid part:

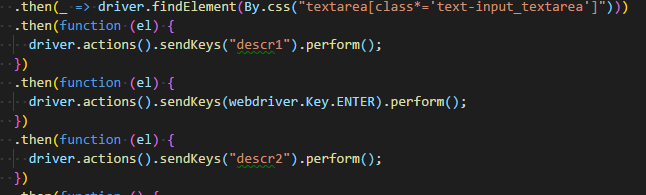
* copy code from repository https://github.com/Alexey-Kachemirov/js\_atm.git
* open ‘module7/grid’ folder
* install project from copied folder running ‘npm install’
* download ‘selenium-server-standalone-3.14.0.jar’ and copy in the project folder
* copy file with credentials in the project folder
* run command from saml-idp project
* start bat file ‘hub\_3.14.0’ to create hub
* start bat file ‘se-node-chrome-3.14.0.bat’ to create node for Chrome browser
* start bat file ‘se-node-ie-3.14.0.bat’ to create node for IE browser
* start js file ‘index.js’ – it will start test in chrome as chrome parameters is set in code by default but can be changed. Detailed description see below.

1. I converted some commands to actions, for example

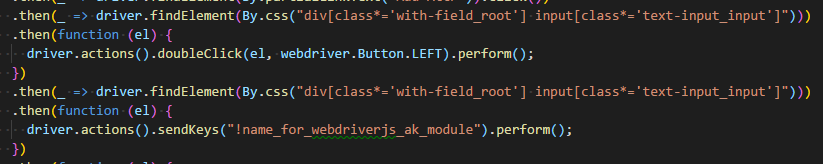
* click
* doubleClick
* perform
* sendKeys

driver.actions().click(el, webdriver.Button.LEFT).perform()

  
driver.actions().sendKeys(webdriver.Key.ENTER).perform()



driver.actions().doubleClick(el, webdriver.Button.LEFT).perform()

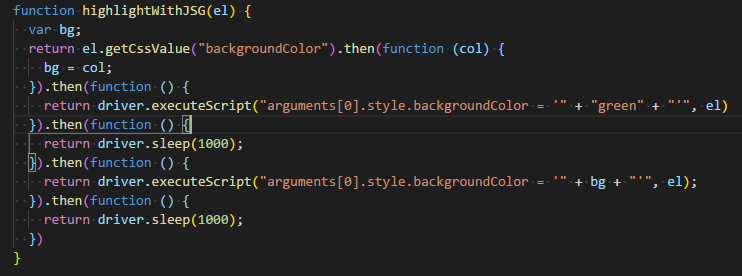


Overall changed and updated 9 actions

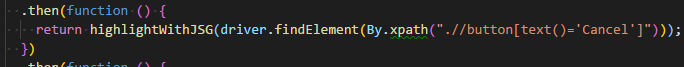
1. I added several JS executors: 2 executors with element highlighter and scrolling.

For example, highlight specific button in green color

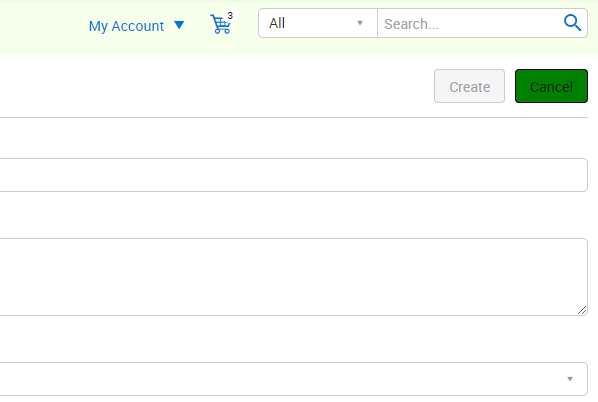
For element highlighter I added a function,



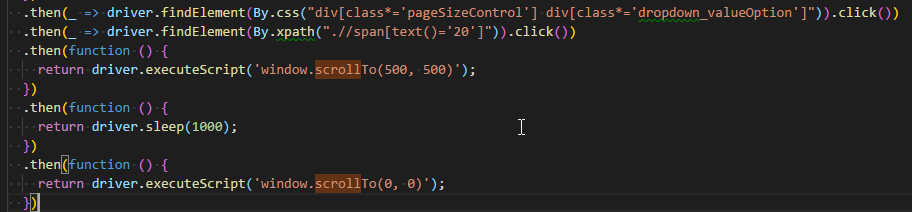
then applied this function in the step



and viewed result in UI

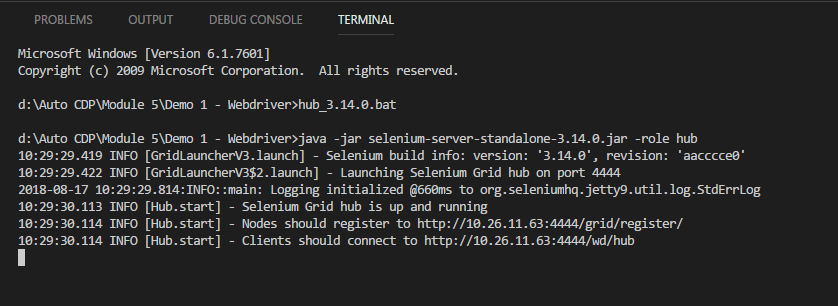


For scrolling I just added executeScript command

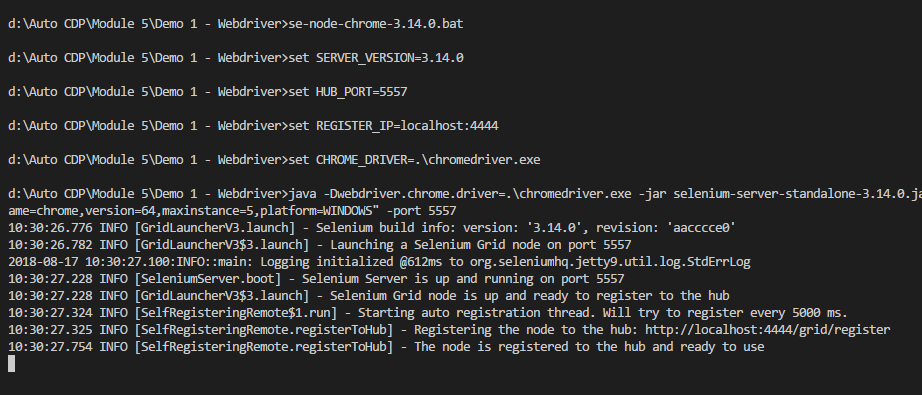


1. Firstly I downloaded latest Selenium Standalone Server from the official download page. I downloaded version 3.14.0 and put into the folder with project.

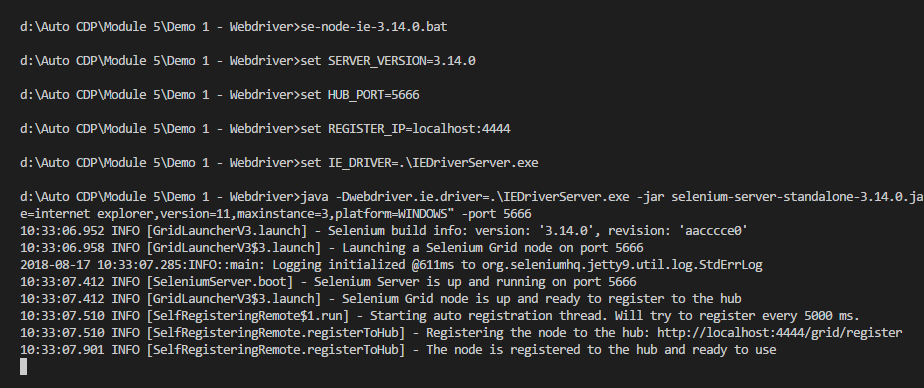
Then I started hub instance:



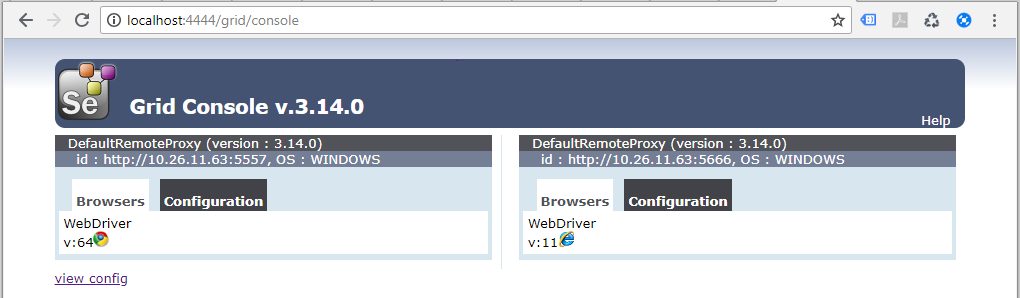
Added node for chrome:



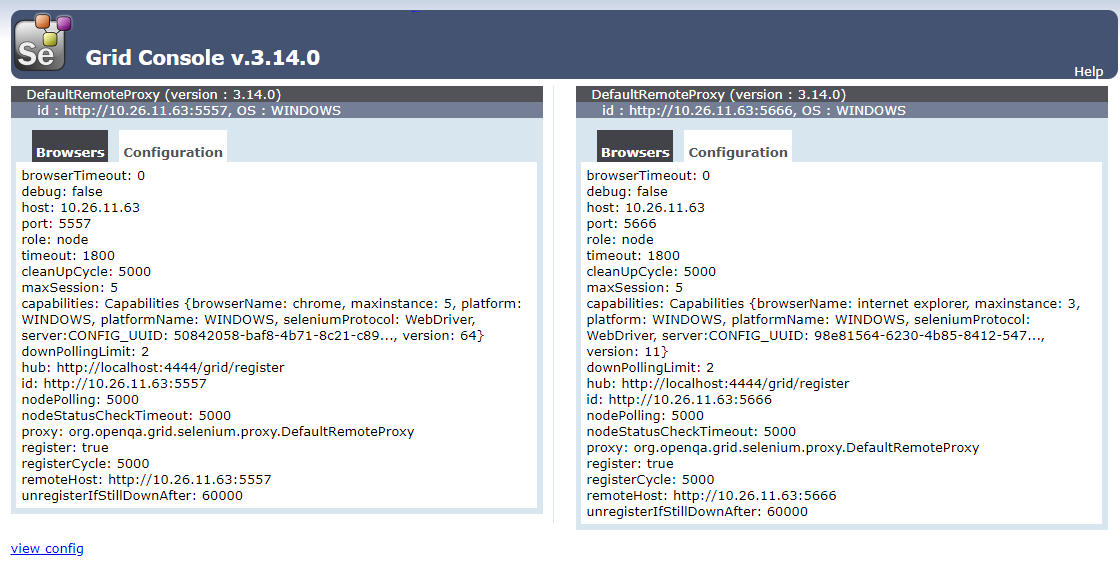
And node for IE:



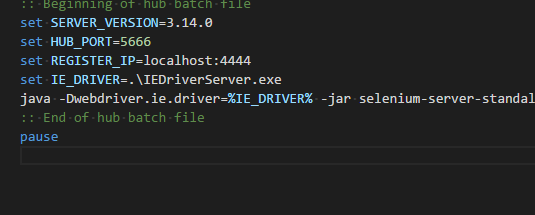
Then check grid state in browser by address <http://localhost:4444/grid/console>



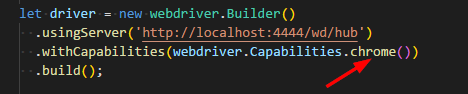
Also we can check configuration tab



where all set parameters are displayed so user can created nodes with different parameters, e.g. change port. For IE I set port 5666.



To start test we need to set proper browser:



Finally I started my test running command

node index.js

During test run I check grid state and I saw that browser icon is disabled during this run

