**Xpath Expressions versus CSS Selectors**

<https://selenium-by-arun.blogspot.com/2017/05/xpath-expressions-versus-css-selectors.html>

[**XPath Helper plugin - Using this plugin in Selenium for locating the GUI elements in Chrome browser**](http://selenium-by-arun.blogspot.com/2017/06/xpath-helper-plugin-using-this-plugin.html)

<https://selenium-by-arun.blogspot.com/2017/06/xpath-helper-plugin-using-this-plugin.html>

[**Relative XPath Helper - Using this plugin in Selenium for generating relative XPath locators in Chrome browser**](http://selenium-by-arun.blogspot.com/2017/06/relative-xpath-helper-using-this-plugin.html)

<https://selenium-by-arun.blogspot.com/2017/06/relative-xpath-helper-using-this-plugin.html>

333. [**last( ) - XPath Function**](http://selenium-by-arun.blogspot.com/2017/04/333-last-xpath-function.html)  
334. [**position( ) - XPath function**](http://selenium-by-arun.blogspot.com/2017/04/334-position-xpath-function.html)  
335. [**contains( ) - XPath function**](http://selenium-by-arun.blogspot.com/2017/04/335-contains-xpath-function.html)  
336. [**starts-with( ) XPath Function**](http://selenium-by-arun.blogspot.com/2017/04/336-starts-with-xpath-function.html)

340. [**Using node( ) wild card in XPath Statements**](http://selenium-by-arun.blogspot.com/2017/04/340-using-node-wild-card-in-xpath.html)  
341. [**Using \* wild card in XPath Statements**](http://selenium-by-arun.blogspot.com/2017/04/341-using-wild-card-in-xpath-statements.html)

342. [**Using | to combine XPath Statements**](http://selenium-by-arun.blogspot.com/2017/04/342-using-to-combine-xpath-statements.html)

### 338. XPath Operators

<https://selenium-by-arun.blogspot.com/2017/04/338-xpath-operators.html>

### 343. XPath Optimization

<https://selenium-by-arun.blogspot.com/2017/04/343-xpath-optimization.html>

Cấu hình selenium grid cho việc chạy kiểm thử song song trên các browser.

Selenium Grid

Selenium-grid cho phép tester test với nhiều browser trên nhiều thiết bị (hệ điều hành) một cách đồng thời. Selenium sử dụng lý thuyết hub-node nơi bạn chỉ cần chạy test trên một máy duy nhất gọi là hub, nhưng tác vụ test sẽ được xử lý trên nhiều máy khác được liên kiết với hub được gọi là các node.

Hub nói một cách tổng quát - nó được xem như một bộ não trung tâm cho mọi hoạt động, và Selenium Grid Hub cũng hoạt động theo concept này. Selenium grid hub là điểm các đoạn testcase được khởi chạy và thì thực thi trên các node. Các node có thể là bất kì thiết bị nào chạy trên bất kì hệ điều hành nào (ios, window, linux … ). Khi một hub gọi một đoạn code test, các node thực thi đoạn code như một bản sao của hub.

Cài đặt Selenium Grid Hub

Bước 1: Tải thư viện .jar selenium grid

Tải selenium-standalone-server.jar https://www.selenium.dev/downloads/ trên cả máy hub và máy node.

Bước 2: Khởi chạy hub bằng câu lênh:

java -jar <selenium\_stanalone\_server\_file\_name\_here> -role hub -host <hub\_ip\_here>

<selenium\_stanalone\_server\_file\_name\_here> : tên của file jar vừa tải về.

<hub\_id\_here>: địa chỉ của máy chủ (hub).

**Example: java -jar selenium-server-standalone-3.141.59.jar -role hub -host localhost**



*Lưu ý: máy phải cài Java JDK và cấu hình variable environment: JAVA\_HOME (đường dẫn đến thư mục cài JDK) và PATH (đường dẫn đến thư mục bin của JDK)*

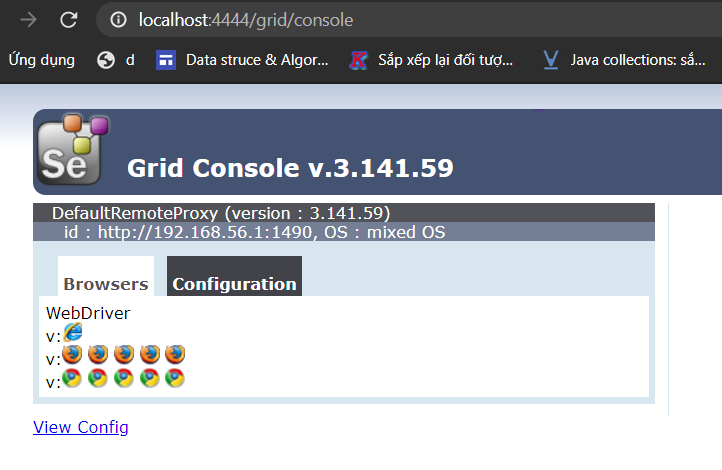
Bước 3: Khởi chạy các node

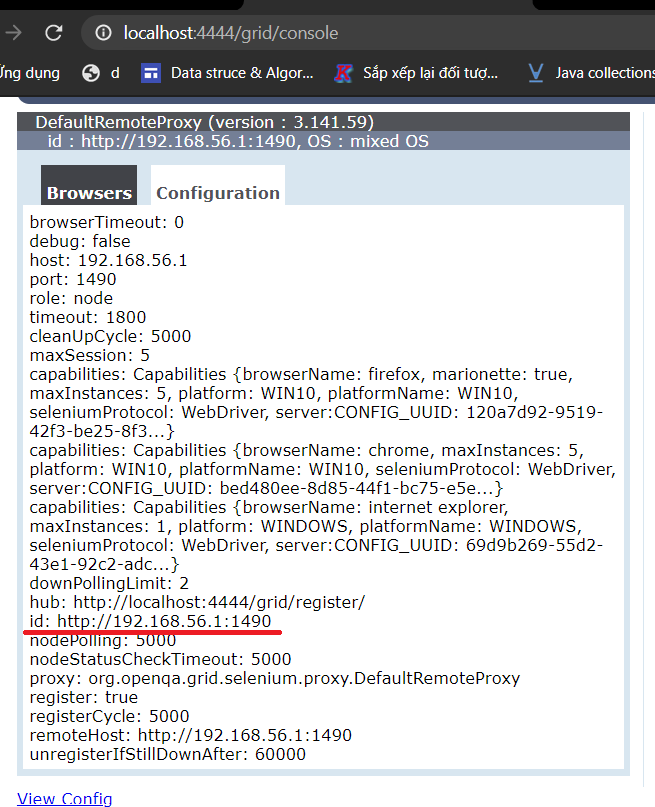
java -Dwebdriver.chrome.driver="chromedriver.exe" -Dwebdriver.internetexplorer.driver="IEDriverServer.exe" -Dwebdriver.gecko.driver="geckodriver.exe" -jar "selenium-server-standalone-3.141.59.jar" -role node -hub <hub\_register\_uri>

<hub\_register\_uri>: đường dẫn đăng ký node của máy hub.

**Example:**

**java -Dwebdriver.chrome.driver="chromedriver.exe" -Dwebdriver.internetexplorer.driver="IEDriverServer.exe" -Dwebdriver.gecko.driver="geckodriver.exe" -jar "selenium-server-standalone-3.141.59.jar" -role node -hub "http://localhost:4444/grid/register/"**





Ở tab **Configuration,** id chính là địa chỉ của máy node được đăng ký với máy hub, máy hub sẽ kết nối với máy node thông qua id này.

Bước 4: Viết class config cho cả project

public class SetupDriver {

private WebDriver driver = null;

private String browser = null;

private String baseUrl = null;

private String os = null;

private String node = null;

public SetupDriver(String os, String browser, String baseUrl, String node) throws MalformedURLException {

this.browser = browser;

this.os = os;

this.baseUrl = baseUrl;

this.node = node;

Platform platform = Platform.fromString(os.toUpperCase());

if(browser.equalsIgnoreCase("chrome")) {

ChromeOptions chromeOptions = new ChromeOptions();

chromeOptions.setCapability("platform", platform);

this.driver = new RemoteWebDriver(new URL(node + "/wd/hub"), chromeOptions);

} else if (browser.equalsIgnoreCase("firefox")) {

FirefoxOptions firefoxOptions = new FirefoxOptions();

firefoxOptions.setCapability("platform", platform);

this.driver = new RemoteWebDriver(new URL(node + "/wd/hub"), firefoxOptions);

} else if (browser.equalsIgnoreCase("ie11")) {

InternetExplorerOptions ieOption = new InternetExplorerOptions();

ieOption.setCapability("platform", platform);

this.driver = new RemoteWebDriver(new URL(node + "/wd/hub"), ieOption);

}

this.driver.manage().timeouts().implicitlyWait(60, TimeUnit.SECONDS);

this.driver.manage().window().maximize();

this.driver.get(baseUrl);

}

public String getOs() {

return this.os;

}

public String getBrowser() {

return this.browser;

}

public String getBaseUrl() {

return this.baseUrl;

}

public String getNode() {

return this.node;

}

public WebDriver getDriver() {

return this.driver;

}

}

Bước 5: Gọn config trong các class chạy testcase

public class GoogleSearchTest {

public WebDriver driver;

@BeforeClass(alwaysRun = true)

@Parameters({"os", "browser", "url", "node"})

public void setUp(String os, String browser, String url, String node) throws MalformedURLException {

System.out.println(os + browser + url + node);

SetupDriver setupTestDriver = new SetupDriver(os, browser, url, node);

driver = setupTestDriver.getDriver();

}

}

Bước 6: Viết file .xml để chạy test case trên nhiều driver cùng lúc

<!--src/test/resources/launchers/google.xml-->

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name="Google Search Tests" verbose="1" parallel="tests">

<parameter name="url" value="https://www.google.com/"/>

<parameter name="os" value="windows"/>

<parameter name="node" value="http://192.168.56.1:1490"/>

<test name="Tests on Firefox" preserve-order="true" group-by-instances="true">

<parameter name="browser" value="firefox"/>

<classes>

<class name="tiktzuki.Junit.GoogleSearchTest"/>

</classes>

</test>

<test name="Tests on Chrome" preserve-order="true" group-by-instances="true">

<parameter name="browser" value="chrome"/>

<classes>

<class name="tiktzuki.Junit.GoogleSearchTest"/>

</classes>

</test>

<!--

<test name="Tests on IE11" preserve-order="true" group-by-instances="true">

<parameter name="browser" value="ie11"/>

<classes>

<class name="tiktzuki.Junit.GoogleSearchTest"/>

</classes>

</test> -->

</suite>

https://the-internet.herokuapp.com/