==========

Topic: **Selenium Grid**

==========

1. Document prepared by: **Rajat Verma**
    a. https://www.linkedin.com/in/rajat-v-3b0685128/
    b. https://github.com/rajatt95
    c. https://rajatt95.github.io/

========== Testing - Selenium GRID ==========

Selenium Grid:

1. Selenium Grid is **a tool that distributes the tests across multiple physical or virtual machines so that we can execute scripts in parallel (simultaneously)**.
2. It accelerates the testing process across browsers and across platforms by giving us quick and accurate feedback.
3. **Selenium Grid** is a part of the Selenium Suite that specializes in running multiple tests across different browsers, operating systems, and machines in parallel.
4. It is achieved by routing the commands of remote browser instances where a server acts as a hub. A user needs to configure the remote server in order to execute the tests.
5. Selenium Grid uses a hub-node concept where you only run the test on a single machine called a **hub**, but the execution will be done by different machines called **nodes**.
6. It is a configuration that helps to run the test scripts on multiple machines
7. Hub-Node architecture
8. 1 Hub and many nodes; Nodes need to be connected to Hub
9. It supports Distributed test execution.
10. 

When to use Selenium Grid:

1. Run your tests against different browsers, operating systems, and machines all at the same time.
2. This will ensure that the application we are testing is fully compatible with a wide variety of Browser-OS combinations.
3. It saves time in the execution of test suites.
4. Infrastructure needs
    a. You can distribute your test cases over different machines and platforms.
5. Parallel execution
    a. Total - 50
        i. MAC - 15
        ii. Ubuntu - 15
        iii. WIN - 20

What are Hub and Node?

1. **Hub:**
    a. It is the central point where we load our scripts into.
    b. There should be one Hub in the Grid.
    c. The Hub can be launched only on a single machine.
    d. It is the **central point** that will receive all the test requests and distribute them to the right nodes.
    e. The machine containing the Hub is where the tests will be triggered, but, you'll see the Browser being automated on the Node.

---

2. **Node:**
    a. Nodes are the Selenium instances that will execute the tests that you loaded on the hub.
    b. There can be one or more nodes in a grid.
    c. Nodes can be launched on multiple machines with different platforms and browsers.
    d. The machines running the nodes need not be the same platform as that of the hub.
    e. Nodes can exist on the same as well as on different machines.
    f. On each node, there can be multiple browsers running parallel.
    g. One machine can have multiple nodes.

---

1. **Download**
    a. selenium-server-standalone-3.141.59.jar from
        i. https://www.selenium.dev/downloads/

NOTE:

1. By default, a node supports
    a. 5 instances of Firefox Browser
    b. 5 instances of Chrome Browser
    c. 1 instance of IE Browser
2. These number of browsers (5, 5, 1) are called **MaxInstance.**
    a. We can change this max session as well.

Working with Selenium Grid:

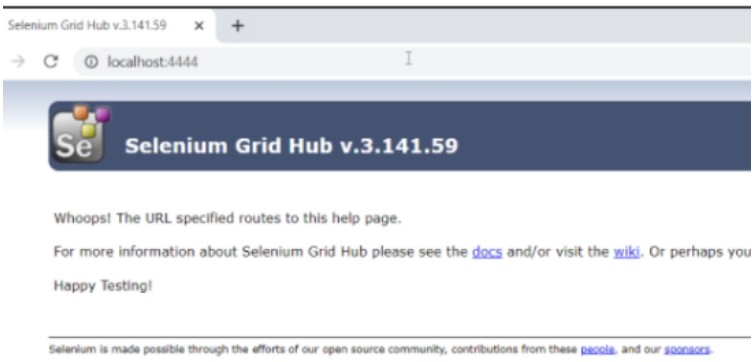
a. Configure the Hub
b. Configure the Nodes
c. Develop the Script
d. Test Execution
e. Analyze Result

Working with Selenium Grid (Steps):

1. Configure the Hub.
    a. 1st cmd ->
    java -jar selenium-server-standalone-3.141.59.jar -port 4444 -role hub

```
f:\Work_in_local_machine\Softwares\1_Selenium\Grid> java -jar selenium-server-standalone-3.141.59.jar -
ub
00:39:46.241 INFO [GridLauncherV3.parse] - Selenium server version: 3.141.59, revision: e82be7d358
00:39:46.515 INFO [GridLauncherV3.lambda$buildLaunchers$5] - Launching Selenium Grid hub on port 4444
2021-07-04 00:39:47.074:INFO::main: Logging initialized @1652ms to org.seleniumhq.jetty9.util.log.StdEr
00:39:48.278 INFO [Hub.start] - Selenium Grid hub is up and running
00:39:48.280 INFO [Hub.start] - Nodes should register to http://192.168.29.11:4444/grid/register/
00:39:48.283 INFO [Hub.start] - Clients should connect to http://192.168.29.11:4444/wd/hub
```

    b. Chrome -> http://localhost:4444/
        i. You should be able to see the UI for Selenium Grid



        ii.

iii.

2. Configure the nodes.
    a. **2nd cmd -> java -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://192.168.29.11:4444/grid/register/ -port 5555**
    b. **Chrome -> http://localhost:4444/grid/console**
        i. You should be able to see the Nodes attached to the Hub.
        ii. This URL is used to check whether nodes are connected to Hub or not
    c. **192.168.29.11** - IP of the machine on which Hub is running
    d. What is my IP?
        i. cmd -> ipconfig
        ii. IPv4 address -> **192.168.29.11**
    e.
3. Develop the script
    a. Project -> Build path -> Add dependency for Selenium Grid
4. Test execution
5. Analyze results

NOTE:

1. Capabilities that are passed in the Java code should get matched with 1 of the nodes registered with Hub.
2. If it does not find any matching node, then,
    a. org.openqa.selenium.**SessionNotCreatedException:**
        i. session not created: No matching capabilities found

---

Only with Chrome:

1. **java -Dwebdriver.chrome.driver=F:\Work_in_local_machine\Softwares\1_Selenium\Drivers\WIN\chromedriver.exe -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://192.168.29.11:4444/grid/register/ -port 6666 -browser browserName=chrome,maxInstances=10**

2. **java -Dwebdriver.chrome.driver=D:\Softwares\Selenium_Grid\chromedriver.exe -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://**192.168.29.190**:4444/grid/register/ -port 6666 -browser browserName=chrome,maxInstances=10**
3. **Chrome -> http://localhost:4444/grid/console**

a.

To configure more browsers and with no. of instances:
And,
How to increase concurrent test support for nodes (Chrome, Firefox, Edge):

1. java -Dwebdriver.chrome.driver=**F:\Drivers\WIN\chromedriver.exe** -Dwebdriver.gecko.driver=**F:\Drivers\WIN\geckodriver.exe** -Dwebdriver.edge.driver=**F:\Drivers\WIN\msedgedriver.exe** -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://**192.168.29.11**:4444/grid/register -port 5555 -**browser browserName=firefox,maxInstances=4 -browser browserName=MicrosoftEdge,maxInstances=8 -browser browserName=chrome,maxInstances=5 -maxSession 4**

2.
3. Chrome -> **http://localhost:4444/grid/console**

CODE - Start

```
package _03_SeleniumGrid;

import java.net.MalformedURLException;
import java.net.URL;

import org.openqa.selenium.Capabilities;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Platform;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.remote.BrowserType;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.remote.RemoteWebDriver;

public class _01_Remote_Test_Execution {
    private static final String Application_URL = "https://www.google.com/";
    private static final String HUB_URL = "http://192.168.29.11:4444/wd/hub";

    public static void main(String[] args) throws MalformedURLException, InterruptedException {

        DesiredCapabilities caps = setCapabilityForNode();
        // LOCAL
        // WebDriver driver = new ChromeDriver();
        // REMOTE
        RemoteWebDriver driver = new RemoteWebDriver(new URL(HUB_URL), caps);
        getBrowserInfo(driver);
        System.out.println("Navigating to: " + Application_URL);
        driver.get(Application_URL);

        System.out.println("driver.getCurrentUrl(): " + driver.getCurrentUrl());
        System.out.println("driver.getTitle(): " + driver.getTitle());

        do_Zoom(driver);
        System.out.println("Success");
        driver.quit();

    }

    private static void do_Zoom(RemoteWebDriver driver) throws InterruptedException {
```

```java
    private static void do_Zoom(RemoteWebDriver driver) throws InterruptedException {
        JavascriptExecutor js = (JavascriptExecutor) driver;

        System.out.println("Set Zoom level - 80%");
        js.executeScript("document.body.style.transform='scale(0.8)';");
        Thread.sleep(2000);

        System.out.println("Set Zoom level - 50%");
        js.executeScript("document.body.style.transform='scale(0.5)';");
        Thread.sleep(2000);

        System.out.println("Set Zoom level - 120%");
        js.executeScript("document.body.style.transform='scale(1.2)';");

        Thread.sleep(2000);
    }

    private static DesiredCapabilities setCapabilityForNode() {
        System.out.println("Setting capability for execution over Node");
        DesiredCapabilities caps = new DesiredCapabilities();
        // caps.setBrowserName(BrowserType.CHROME);
        // caps.setBrowserName(BrowserType.EDGE);
        caps.setBrowserName(BrowserType.FIREFOX);
        caps.setPlatform(Platform.WINDOWS);
        return caps;
    }

    protected static void getBrowserInfo(WebDriver driver) throws InterruptedException {
        Capabilities cap = ((RemoteWebDriver) driver).getCapabilities();
        String browserName = cap.getBrowserName().toLowerCase();
        String os = cap.getPlatform().toString();
        String version = cap.getVersion().toString();

        System.out.println("================================");
        System.out.println("Operating System: " + os);
        System.out.println("Browser: " + browserName.toUpperCase() + " - " + version);
        System.out.println("================================");
        Thread.sleep(2000);
    }
```

CODE - END

==========