#What is selenium grid?

-selenium grid is just like a configuration if you want to run the test cases on multiple OS and multiple browsers or

different system with the help of the remote web driver and multiple desired capabilities we can execute our test cases.

Selenium-Grid allows you to run your tests on different machines against different browsers in parallel. That is, running multiple tests at the same time against different machines running different browsers and operating systems. Essentially, Selenium-Grid support distributed test execution. It allows for running your tests in a distributed test execution environment.

#what things are needed for selenium grid to run the test cases?

-we have to create servers one hub and node.

1.hub is just like a server or that machine where you have written all the test scripts and hub will connect to node

through JSON Wire protocol/HTTP request.

2.Node is just like a server or that machine or VM where you want to run your test cases on multiple OS as well multiple

browsers.

-Only one hub to be create.(hub will be create on that machine where you have written all the test scripts)

-Multiple nodes can be create.(node will be create on that machine or VM where you want to run your test cases

on multiple OS as well browsers.)

#How to create a Hub or start the hub server?

1. We have to download selenium server standlone jar file in hub machine.

2. Go to cmd just go into that directory or folder where you have kept the selenium server standlone jar file in hub machine.

STEP ->

1.Go to selenium server standlone jar file directory or folder

2. Type java -jar and path of selenium server standlone file with extension -role hub,s (java -jar selenium-server

standalone-3.8.1.jar -role hub) and press enter button.

3.Then you should see in cmd two lines <Nodes should register to http://192.168.0.3:4445/grid/register and selenium Grid hub is up and running now you can assume, you have successfully created hub server in hub machine.

4.Here you can check this hub is really up or not just open browser copy this URL //192.168.0.3:4445 and paste it in browser

then if you are getting Selenium Grid Hub v.3.8.1 page, now you can assume, you have successfully created hub server in hub machine.

NOTE -> If you are getting exception during STEP 2 (java.net.BindException: Address already in use: bind) then you have to use another port or use this command (java -jar selenium-server standalone-3.8.1.jar -role hub -port 4445) and press enter button.

#How to create a Node or start the node server?

1.We have to download selenium server standlone jar file in node machine.

2.We have to download any browser or where you want to excute your test cases

3.We have to download related browser driver

4.Go to cmd just go into that directory or folder where you have kept the selenium server standlone jar file in node machine.

STEP ->

1.Go to selenium server standlone jar file directory or folder

2.Type

java -Dwebdriver.chrome.driver="C:\Users\guptaav\Downloads\chromedriver\_win32\chromedriver.exe" -jar selenium-server-standalone-3.8.1.jar -role node -hub http://10.141.233.96:4444/grid/register/

3.Then you should see in cmd four lines Launching a Selenium Grid node, Selenium Grid node is up and ready to register to the hub, Registering the node to the hub: http://192.168.0.3:4445/grid/register and The node is registered to the hub and ready to use, now you can assume, you have successfully created node server in node machine.

4.Then you should see in hub server "Registered a node http://192.168.0.3:5555"(Just copy this url and paste in the browser)

#We have to write a code in eclipse platform for launching the browser with the help of DesiredCapabilities and new RemoteWebDriver and code in below.

DesiredCapabilities cap=new DesiredCapabilities();

cap.setPlatform(Platform.WIN10);

cap.setBrowserName("chrome");

WebDriver driver=new RemoteWebDriver(new URL("http://192.168.0.3:4445/wd/hub"), cap);//here you have to give that URL that is recived on hub Machine.

driver.get("http://www.google.com");

System.out.println(driver.getTitle());