**Selenium Grid Overview**

**What is Selenium Grid?  
Selenium Grid is a tool used for running tests in parallel across multiple machines and browsers. It helps speed up test execution and ensures compatibility across different environments.**

**Key Components of Selenium Grid:**

1. **Hub 🖥️**
   * **The central point that controls test execution.**
   * **Sends test requests to connected nodes.**
2. **Nodes 🖥️🖥️**
   * **Machines where actual tests run.**
   * **Can have different browsers and operating systems.**
   * **Connects to the Hub and executes tests.**

**Why Use Selenium Grid?  
✅ Parallel Execution – Run multiple tests at the same time.  
✅ Cross-Browser Testing – Test on Chrome, Firefox, Edge, etc.  
✅ Multiple OS Support – Run tests on Windows, macOS, Linux, etc.  
✅ Faster Testing – Reduces total execution time.**

**How It Works?**

1. **Start the Hub (Main Server).**
2. **Register Nodes (Test Machines) to the Hub.**
3. **Run test scripts; the Hub assigns them to Nodes.**
4. **Nodes execute tests on different browsers/environments.**

**Example Use Case:  
Imagine you need to test a website on Chrome, Firefox, and Edge at the same time. Instead of running tests one by one, Selenium Grid allows you to run them simultaneously on different machines—saving time!**

**Step-by-Step Guide to Setting Up Selenium Grid on Windows**

Selenium Grid allows running tests on multiple machines and browsers in parallel. Follow these simple steps to set up **Selenium Grid** on Windows.

**Object for Session:** Multiple browser (Chrome and Firefox) with multiple instances in parallel

**🔹 Prerequisites**

✔️ Install **Java JDK** (Check with java -version)  
✔️ Install the latest **Selenium Server** from Selenium Official Website

✔️ Need to designed cucumber page object model framework with testNg integration  
✔️ Download WebDriver binaries (**ChromeDriver, GeckoDriver, etc.**)

**🚀 Step 1: Download Selenium Server**

1️. Go to Selenium Downloads Page  
2️. Download the latest **Selenium Server JAR** file (e.g., selenium-server-4.x.x.jar)  
3️. Place the JAR file in a dedicated folder (e.g., C:\SeleniumGrid\)

**🚀 Step 2: Start Selenium Grid Hub**

1️Open **Command Prompt** and navigate to the Selenium Grid folder:

cd C:\SeleniumGrid

2️ Start the Selenium Grid Hub using the following command:

java -jar selenium-server-4.x.x.jar hub

3️ If successful, the hub will start on **http://localhost:4444/**

📌 **Verify:** Open a browser and go to **http://localhost:4444** to see the Grid UI.

**🚀 Step 3: Start a Node (Browser Instance)**

A node is a machine where tests will run.

**Option 1: Start a Local Node (Same Machine as Hub)**

1️ Open a **new Command Prompt** window.  
2️ Navigate to the Selenium Grid folder:

cd C:\SeleniumGrid

3️ Start the node using:

java -jar selenium-server-4.x.x.jar node --detect-drivers

4️ The node will connect to the **Hub (http://localhost:4444/)** and register available browsers.

📌 **Check:** The node should now appear in the Selenium Grid UI.

------------------------Added new steps---------------------

**Manually Specify WebDrivers (Alternative to PATH)**

Instead of relying on automatic detection, specify the WebDriver path manually.

Run the command with explicit driver paths:

**java -jar selenium-server-4.8.2.jar node --config drivers.toml**

📌 **Create a drivers.toml file** in the same directory (C:\SeleniumGrid\) with the following content:

[server]

port = 5555

[node]

detect-drivers = false

[[node.driver-configuration]]

display-name = "Chrome"

webdriver-executable = "C:\QAShastra\MyTrainings\Selenium\Selenium Grid> \\drivers\\chromedriver.exe"

max-sessions = 5

stereotype = '{"browserName": "chrome", "browserVersion": "latest"}'

[[node.driver-configuration]]

display-name = "Firefox"

webdriver-executable = "C:\QAShastra\MyTrainings\Selenium\Selenium Grid> \\drivers\\geckodriver.exe"

max-sessions = 3

stereotype = '{"browserName": "firefox", "browserVersion": "latest"}'

Then run the command again.

**4️Verify Grid Connection**

1. Start the **Hub**:

java -jar selenium-server-4.8.2.jar hub

1. Start the **Node**:

java -jar selenium-server-4.8.2.jar node --detect-drivers

1. Open **http://localhost:4444** in a browser.
   * You should see the **Node registered and available**.

-------------------------------------------Added new Steps---------------------------

**🚀 Step 4: Run Tests on the Grid**

1️ Update your Selenium **Test Script** to use **RemoteWebDriver**:

ChromeDriver

Junit -Chrome

DriverMange RemoteWe Driver()

TestRunner java -migrated for TestNG

2️ **Run the test** – It should execute on the configured Grid node!

**🚀 Step 5: Scaling Up (Multiple Nodes & Browsers)**

🔹 You can add more nodes (on different machines) using:

java -jar selenium-server-4.x.x.jar node --hub http://<Hub\_IP>:4444

🔹 To specify a browser, use:

java -jar selenium-server-4.x.x.jar node --browser "chrome"

**🎯 Summary**

✅ **Step 1:** Download and install Java & Selenium Server  
✅ **Step 2:** Start Selenium Grid Hub (java -jar selenium-server-4.x.x.jar hub)  
✅ **Step 3:** Start a Node (java -jar selenium-server-4.x.x.jar node --detect-drivers)  
✅ **Step 4:** Run Selenium tests using **RemoteWebDriver**  
✅ **Step 5:** Scale up by adding more nodes

Now, you have **Selenium Grid** running on Windows, allowing parallel execution of tests across multiple browsers & machines!