Selenium with Java – Introduction to Selenium Lecture 1

What is Selenium?

Selenium is a powerful open-source framework for automating web browser testing easily**.** Selenium is a powerful tool for controlling web browsers through programs. It is functional for all browsers (**Chrome, Firefox, Internet Explorer** etc.), works on all major OS (**Windows, Linux, Mac**), and its scripts are written in various languages (i.e., **Python, Java, C#, Ruby** etc.)

Selenium is a widely used tool for testing web-based applications that checks if they are doing as expected. Selenium is also platform-independent, so it can provide distributed testing using the Selenium Network. Selenium is very extensible and can be integrated with other tools and frameworks like TestNG, JUnit, Cucumber, etc.

Importance of Selenium in Testing

Manual testing can be time-consuming and prone to human errors. Selenium Automation allows tests to be executed quickly and accurately, reducing the likelihood of human mistakes and ensuring consistent test results.

Selenium allows developers and testers to automate the testing of web applications across different browsers and platforms.

The main Features of Selenium are as follows:

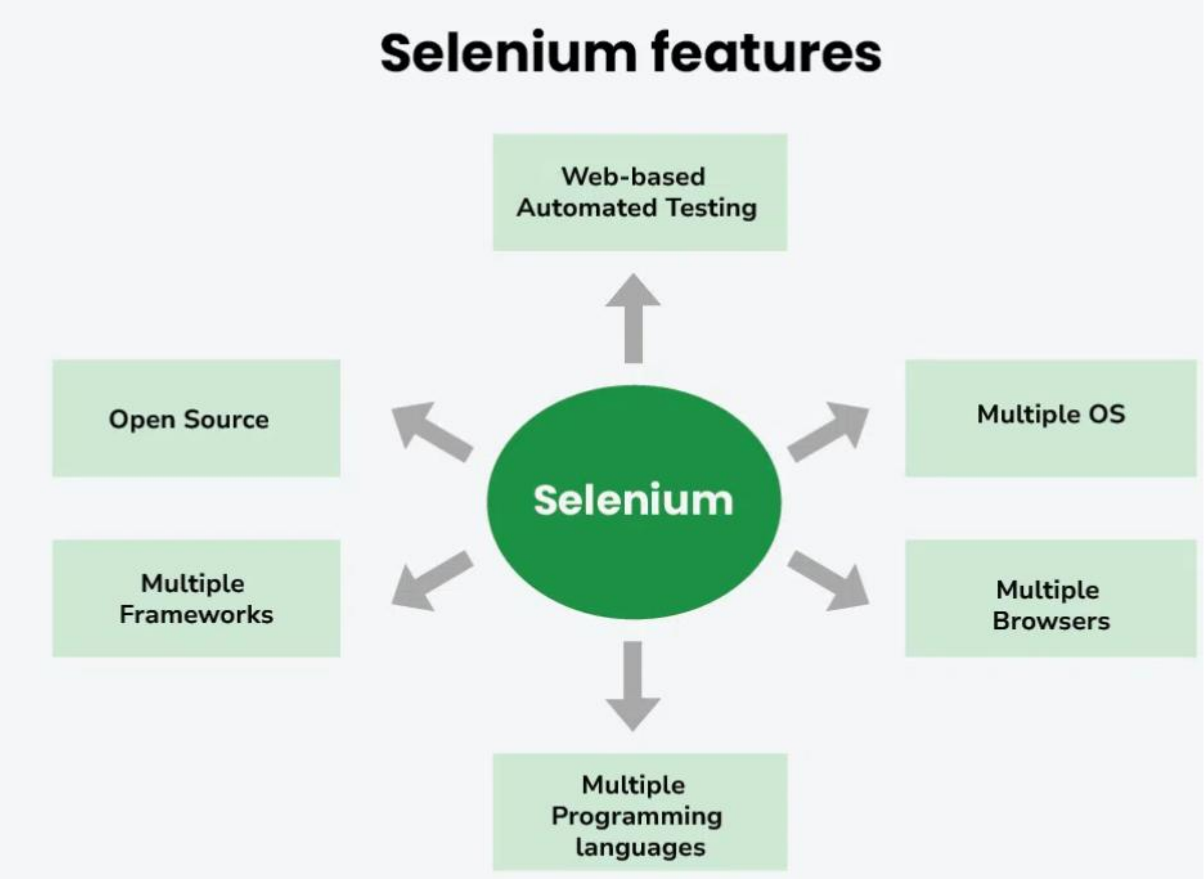
- Open Source platform for use.
- Easy Identification and Use of Web Elements while using the selenium.
- Performance and Speed are more as compared to the other tools.
- Dynamic Web Elements are present for easy use.
- **Language Support:** Selenium allows you to create test scripts in different languages like Ruby, Java, PHP, Perl, Python, JavaScript, and C#, among others.
- **Browser Support:** Selenium enables you to test your website on different browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, Internet Explorer (IE), etc.
- **Scalability:** Automated testing with Selenium can easily scale to cover a wide range of test cases, scenarios, and user interactions. This scalability ensures maximum test coverage of the application’s functionality.
- **Reusable Test Scripts:** Selenium allows testers to create reusable test scripts that can be used across different test cases and projects. This reusability saves time and effort in test script creation and maintenance.
- **Parallel Testing:** Selenium supports parallel test execution, allowing multiple tests to run concurrently. This helps reduce the overall testing time, making the development process more efficient.

- **Documentation and Reporting:** Selenium provides detailed test execution logs and reports, making it easier to track test results and identify areas that require attention.
- **User Experience Testing**: Selenium can simulate user interactions and behavior, allowing testers to assess the user experience and ensure that the application is intuitive and user-friendly.
- **Continuous Integration and Continuous Deployment (CI/CD**): Selenium can be integrated into CI/CD pipelines to automate the testing of each code change. This integration helps identify and address issues earlier in the development cycle, allowing for faster and more reliable releases.

Selenium Features:

The main Features of Selenium are as follows

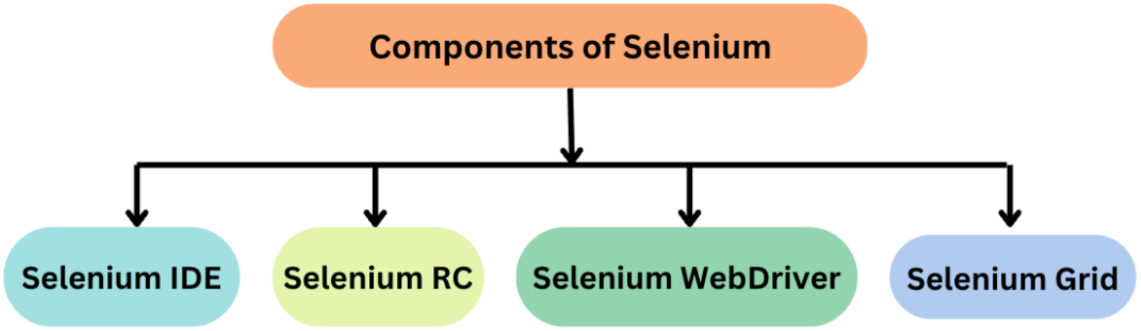
- Multi-Browser Support.
- Multi-Language Compatibility support.
- Open Source platform for use.
- Portability (Easily accessible in any OS)
- Reusability of any codes and element is possible in the tool



Selenium Suite has 4 components namely:

- Selenium IDE.
- Selenium RC.
- Selenium WebDriver.

- Selenium Grid.



Selenium IDE:

Selenium IDE(Integrated Development Environment) is a Chrome and Firefox plugin. The primary use of a Selenium IDE is to record user interactions such as clicks, selections etc. in the browser and plays them back as automated tests. No prior coding knowledge required.

It is used as prototyping tool so we cannot perform advance testing.

Selenium RC (Remote Control):

The predecessor to WebDriver, Selenium RC enabled automated web testing using a server to communicate with browsers.

Selenium RC had to work with browsers using a JavaScript-based “proxy” mechanism due to that it is not able to work with modern web applications.

The use of a JavaScript proxy added overhead and affected the speed and performance of test execution.

Selenium RC required separate “drivers” for each browser, making maintenance and compatibility challenging as browsers continued to update and evolve.

Selenium WebDriver:

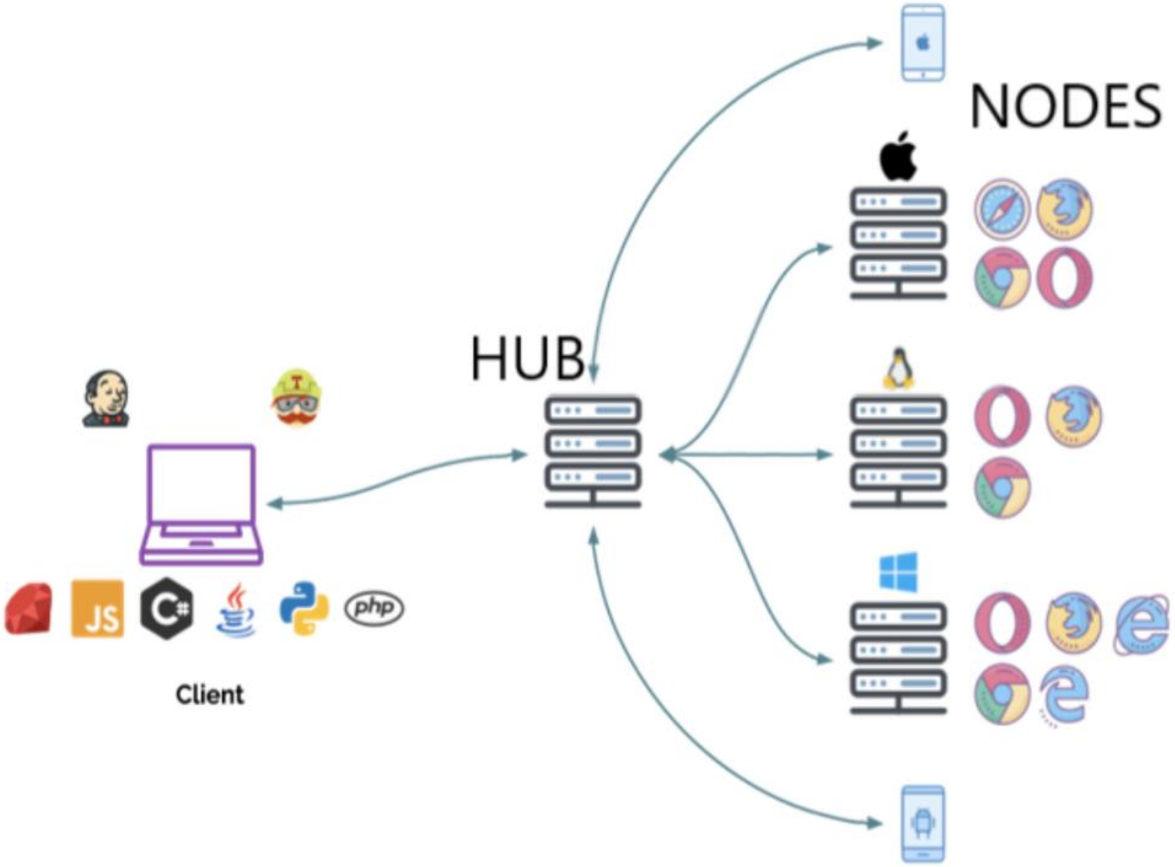
Selenium WebDriver is a powerful and enhanced version of Selenium RC, which was developed to overcome the limitations of Selenium RC.

WebDriver communicates with browsers directly with the help of browser-specific native methods, thereby completely eliminating the need of Selenium RC.

WebDriver works closely with Selenium IDE and Selenium Grid resulting in reliable test execution at speed and scale.

Selenium Grid:

A tool that enables parallel test execution across multiple browsers and environments, reducing overall test execution time. This is done by routing commands to remote web browser instances, where one server acts as the hub. This hub routes test commands that are in JSON format to multiple registered Grid nodes.



Disadvantages of Selenium web driver:

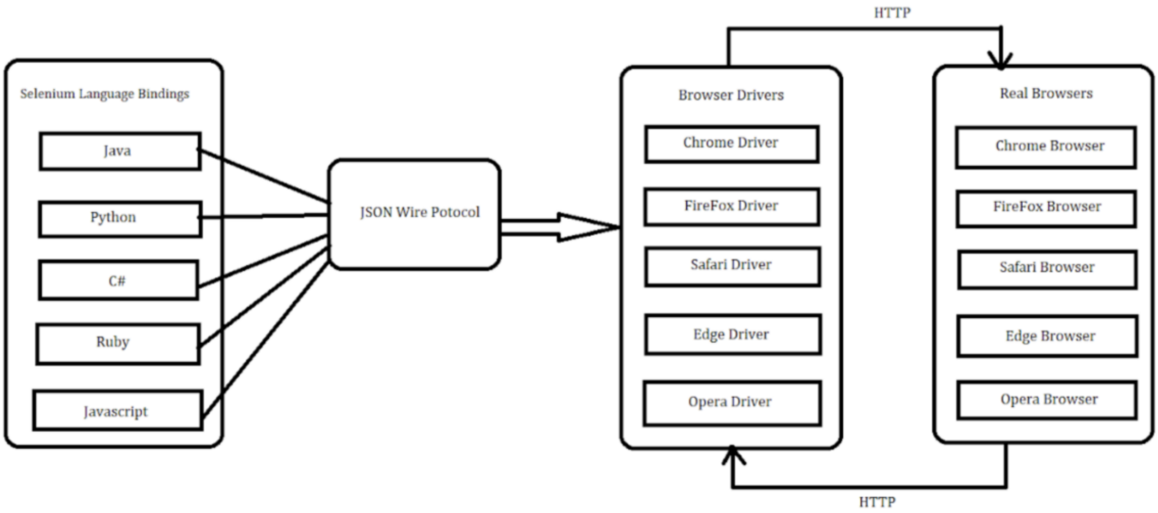
- It supports only web based application and does not support window-based application.
- It does not provide facility for data driven testing
- It does not have reporting feature

By using third party tools, we overcome these disadvantages:

- Windows operation –AutoIt
- Data Driven Testing – Apache POI

- Reporting –Extent Report

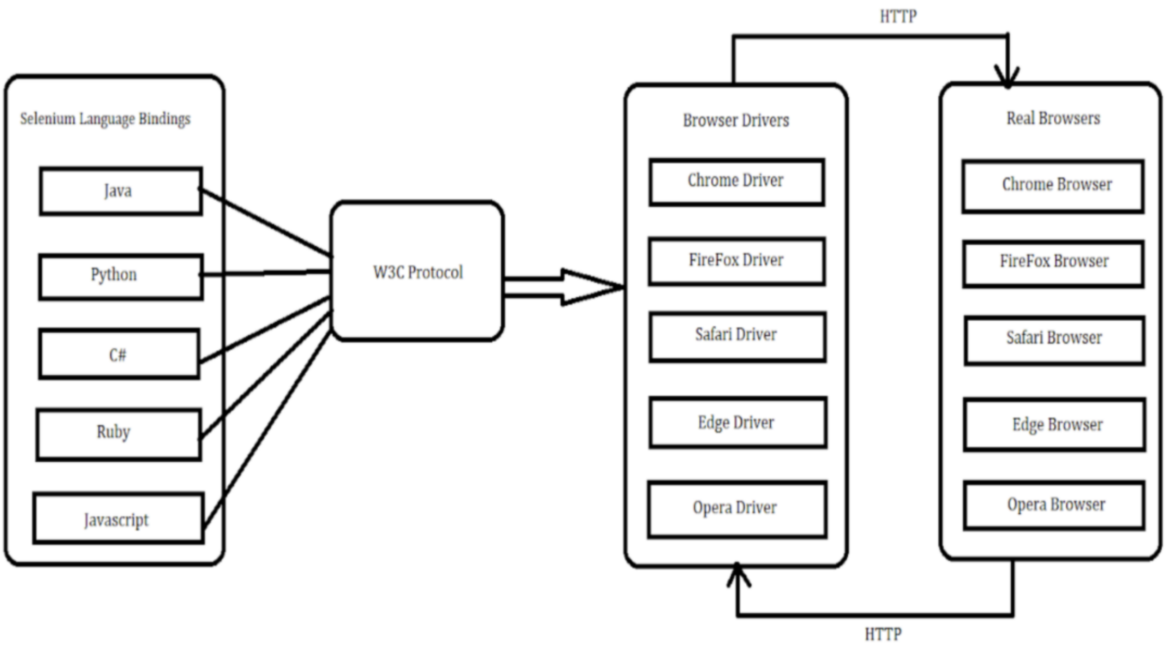
Architecture of Selenium WebDriver (Selenium 3)



- **Selenium Client library:** Selenium provides support to multiple libraries such as Ruby, Python, Java, etc as language bindings
- **JSON wire protocol over HTTP**: JSON is an acronym for JavaScript Object Notation. It is an open standard that provides a transport mechanism for transferring data between client and server on the web.
- **Browser Drivers:** Selenium browser drivers are native to each browser, interacting with the browser by establishing a secure connection. Selenium supports different browser drivers such as ChromeDriver, GeckoDriver, Microsoft Edge WebDriver, SafariDriver, and InternetExplorerDriver.
- **Browsers:** Selenium provides support for multiple browsers like Chrome, Firefox, Safari, Internet Explorer etc.

Architecture of Selenium 4

The architecture of Selenium 4 is similar to Selenium 3, however it uses W3C protocol instead of JSON wire protocol for communication between Client Libraries and Browser Drivers.



W3C protocol was introduced because all the web browsers followed the W3C standards and also all the browser drivers followed the W3C standards. To standardise the communication, JSON wire protocol was replaced by W3C in Selenium 4.

This helped in better communication with the browsers, stability, and common code (i.e. no browser specific code required). Due to W3C there is a direct transfer of information between client and server.

- In Selenium 4, several new methods have been added to the Actions class: clickAndHold(WebElement), contextClick(WebElement), doubleClick(WebElement), release(), click(WebElement)
- In Selenium 4, you can use the relative locators **above, below, toLeftOf, toRightOf,** and **near**.
- Better Window/Tab Management in Selenium 4
- Improved Documentation

Selenium 4	Selenium 3
Selenium 4 uses W3C standard protocol	Selenium 3 used JSON wire protocol
Chrome Driver class extends chromium driver class	Chrome Driver class extended Remote webdriver class
Optimised Selenium Grid with enhanced GUI and support for Docker	No Support for docker

Enhanced Selenium IDE with improved GUI and cloud based selenium grid	Selenium IDE just available as a firefox add-on
Testers need not start the Hub and Node jars everytime they perform automation testing using Selenium Grid.	Testers always had to start Hub and Node jars which was a difficult task in selenium 3.

Selenium Web Driver Installation:

- Download & install Java https://www.oracle.com/java/technolog...
- Download and configure Eclipse or any Java IDE of your choice https://www.eclipse.org/downloads/
- Download Selenium Web Driver Java Client https://www.selenium.dev/downloads/
- Download Chrome Driver https://chromedriver.chromium.org/dow...
- Selenium Java library dependency can be downloaded from Maven repository website using link given below https://mvnrepository.com/