Topics for today’s discussion

i. What is selenium?

ii. Inadequacies in Selenium

iii. Why is Java preferred for Selenium?

iv. Integration of Selenium and Java with webDriver

v. Hands-on: Testing web application

## **What is Selenium?**

Selenium is a free (open-source) automated testing framework used to validate web applications across different browsers and platforms. You can use multiple programming languages like Java, C#, Python etc to create Selenium Test Scripts. Testing done using the Selenium tool is usually referred to as Selenium Testing.

Selenium Software is not just a single tool but a suite of software, each piece catering to different testing needs of an organization.

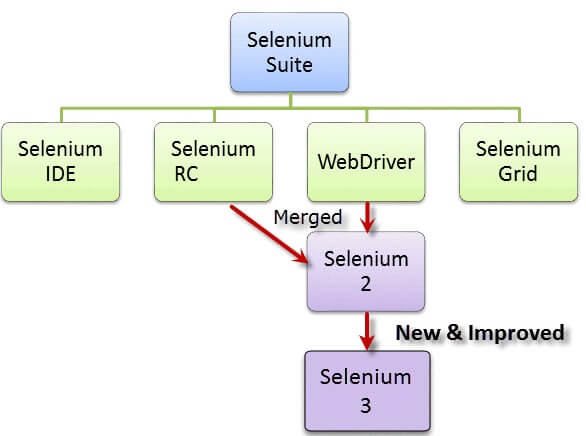
Here is the list of tools.

1.Selenium Integrated Development Environment (IDE)

2.Selenium Remote Control (RC)

3.WebDriver

4.Selenium Grid



Reference :

<https://www.guru99.com/introduction-to-selenium.html>

## **Brief Introduction Selenium Remote Control (Selenium RC):**

Selenium RC was the flagship testing framework of the whole Selenium project for a long time. This is the first automated web testing tool that allowed users to use aprogramming language they prefer. As of version 2.25.0, RC can support the following programming languages:

* [Java](https://www.guru99.com/java-tutorial.html)
* [C#](https://www.guru99.com/c-sharp-tutorial.html)
* [PHP](https://www.guru99.com/php-tutorials.html)
* Python
* Perl
* Ruby

## **Brief Introduction WebDriver :**

The WebDriver proves itself to be better than both Selenium IDE and Selenium RC**in** many aspects. It implements a more modern and stable approach in automating the browser's actions. WebDriver, unlike Selenium RC, does not rely on JavaScript for Automation. It controls the browser by directly communicating with it.

The supported languages are the same as those in Selenium RC.

* Java
* C#
* PHP
* Python
* Perl
* Ruby

## **Selenium Grid**

Selenium Grid is a tool used together with Selenium RC to run parallel **tests** across different machines and different browsers all at the same time. Parallel execution means running multiple tests at once.

Features:

* Enables simultaneous running of tests in multiple browsers and environments.
* Saves time enormously.
* Utilizes the hub-and-nodes concept. The hub acts as a central source of Selenium commands to each node connected to it.

## **Note on Browser and Environment Support**

Because of their architectural differences, Selenium IDE, Selenium RC, and WebDriver support different sets of browsers and operating environments.

|  | **Selenium IDE** | **WebDriver** |
| --- | --- | --- |
| BrowserSupport | Mozilla Firefox | Internet Explorer versions 6 to 11, both 32 and 64-bit   Microsoft Edge version 12.10240 & above ( partial support some functionalities under development)  Firefox 3.0 and above  Google Chrome 12.0. and above  Opera 11.5 and above  Android - 2.3 and above for phones and tablets  (devices & emulators)   iOS 3+ for phones (devices & emulators) and 3.2+ for tablets (devices & emulators)   HtmlUnit 2.9 and above |
| Operating System | Windows,Mac OS X,Linux | All operating systems where the browsers above can run. |

Note: Selenium WebDriver is termed as the successor of Selenium RC which has been deprecated & officially announced by SeleniumHQ.

**How to Choose the Right Selenium Tool for Your Need**

| Tool | Why Choose? |
| --- | --- |
| Selenium IDE | To learn about concepts on automated testing and Selenium, including:   * Selenese commands such as type, open, clickAndWait, assert, verify, etc. * Locators such as id, name, xpath, css selector, etc. * Executing customized JavaScript code using runScript * Exporting test cases in various formats. * To create tests with little or no prior knowledge in programming. * To create simple test cases and test suites that you can export later to RC or WebDriver. * To test a web application against Firefox and Chrome only. |
| Selenium RC | * To design a test using a more expressive language than Selenese * To run your test against different browsers (except HtmlUnit) on different operating systems. * To deploy your tests across multiple environments using Selenium Grid. * To test your application against a new browser that supports JavaScript. * To test web applications with complex AJAX-based scenarios. |
| WebDriver | * To use a certain programming language in designing your test case. * To test applications that are rich in AJAX-based functionalities. * To execute tests on the HtmlUnit browser. * To create customized test results. |
| Selenium Grid | * To run your Selenium RC scripts in multiple browsers and operating systems simultaneously. * To run a huge test suite, that needs to complete in the soonest time possible. |

**Inadequacies in Selenium:**

1. Selenium doesn't have an in-built object repository. In-built object repository helps in identify objects in the test scripts when the web application is being tested.

**Solution**: this feature can be included in selenium by integrated git

1. Selenium cannot perform image testing

**Solution**: need to integrate Sikuli

1. no official support as it is open-source software

**Solution**: community

1. Selenium cannot generate test report. It shows the comparison of the test results with the test objectives.

**Solution**: solution: need to integrate testNG tool

**Why is Java preferred for Selenium?**

1. Development environment

* IDE(code editor, compiler, debugger) + language specific elements such as Ant and Maven build tools. And testNG, JunitTest

1. Programming language

* JIT(just in time ) compilers improved JVMS, which makes it faster.
* (Note: even webDriver is also written in java)

1. Web applications

* Many web applications are made by java which involves serious business process and database access on the server

Integration of Selenium and Java with webDriver:

Selenium WebDriver:

**Selenium WebDriver** is a collection of open source APIs which are used to automate the testing of a web application. **Selenium WebDriver** tool is used to automate web application testing to verify that it works as expected. It supports many browsers such as Firefox, Chrome, IE, and Safari.

* Cross platform testing framework
* Direct communicate with browser
* No separate server required

Environment setup:

* Java(jdk and Jre)
* Selenium( selenium server standalone, selenium server,
* Chrome driver
* intellij idea / eclipse