Test Automation & Advanced Selenium

Lesson 2: Introduction to Selenium

Capgemini

Lesson Objectives

- Introduction to Selenium
- Selenium : What it is?
- Landscape and Usage
- Overview of Selenium Core
- Overview Selenium Remote Control (Selenium 1.0)
- Overview of Selenium IDE
- Overview of Selenium Web Driver (Selenium 2.0)
- Overview of Selenium Grid
- Why Selenium?
- Selenium 3.0 Out Now!

2.1: Introduction to Selenium
Introduction to Selenium

- Selenium is a suite of browser automation tools for automating web browsers across a variety of platforms
- It is primarily used for automating web applications for testing purposes
- Selenium is the most popular open source automation testing tool for Web based applications
- The reason behind its success is its compatibility with all available Web browsers
- Selenium is not just a single tool but a suite of software's, each catering to different testing needs of an organization
- Test cases in selenium can be written in many popular programming languages supported by selenium like Java, C#, Ruby, Python etc.
- Selenium was developed by Jason R. Huggins, an engineer at ThoughtWorks

Introduction to Selenium

Since Selenium is not a single tool used for automating web application but it is a collection of different tools, it had contribution from different developers through it's growth and evolution. Primarily, Selenium was created by Jason Huggins in 2004, an engineer at ThoughtWorks.

He was working on a web application that required frequent testing. During testing he realized that repetitious manual testing of their application was becoming more and more inefficient. To resolve the issue he created a JavaScript program that would automatically control the browser's actions. He named this program as the "JavaScriptTestRunner."

Seeing potential in this idea to help automate other web applications, he made JavaScriptRunner open-source which was later re-named as Selenium Core.

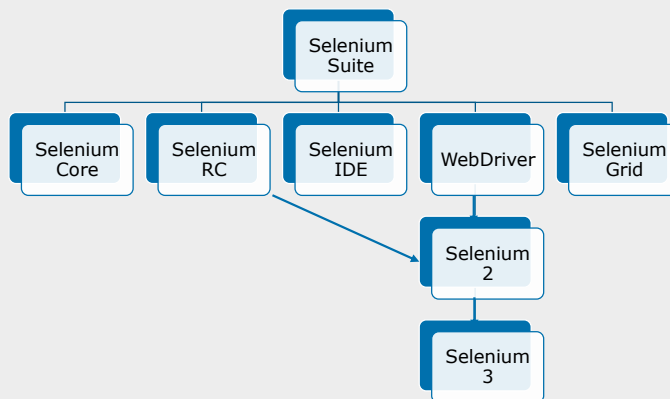2.1: Introduction To Selenium
Selenium : What it is?

- Selenium automates browsers - That's it!
- Selenium is a portable testing API for web applications
- Web Application: Pure HTML & JavaScript (No Flash, ActiveX, Silverlight etc.)
- Primarily, it is for automating web applications for testing purposes, but is certainly not limited to just that
- It is Open Source and Freeware
- Supports multiple browsers & OS
  - Browsers: Firefox, Chrome, IE, Safari, Opera, PhantomJS
  - OS: Windows, MAC, Linux, Android, iOS
- In process Headless execution

Landscape and Usage

Selenium is not just a single tool but a suite of software's, each catering to different testing needs of an organization. *It has four components:*

```
                          ┌──────────┐
                          │ Selenium │
                          │  Suite   │
                          └──────────┘
    ┌──────────┬──────────┬──────────┬──────────┐
┌──────────┐┌──────────┐┌──────────┐┌──────────┐┌──────────┐
│ Selenium ││ Selenium ││ Selenium ││WebDriver ││ Selenium │
│  Core    ││   RC     ││   IDE    ││          ││  Grid    │
└──────────┘└──────────┘└──────────┘└──────────┘└──────────┘
                                    │
                              ┌──────────┐
                              │ Selenium │
                              │    2     │
                              └──────────┘
                                    │
                              ┌──────────┐
                              │ Selenium │
                              │    3     │
                              └──────────┘
```

Add the notes here.

2.1: Introduction To Selenium
Overview of Selenium Core

- Selenium Core is a JavaScript-based test tool for Web applications
- Selenium Core tests run directly in a browser, just as real users do
  - Utility for running tests in web browser
  - Executes commands received from test script
  - Allows test scripts to run inside supported browsers
  - Works with Java script enabled browser
  - Works on a large selection of browsers and operating systems
  - Controls AUT (application under test) in other frame
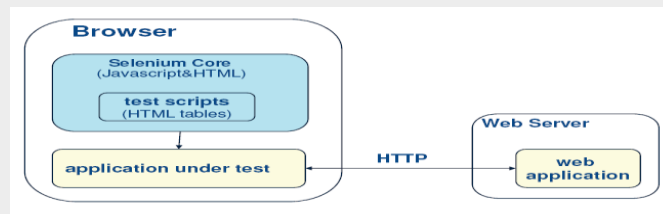
Overview of Selenium Core

Selenium was created by Jason Huggins in 2004. He created a JavaScript program that would automatically control the browser's actions. He named this program as the "JavaScriptTestRunner."
Looking at the potential in this idea to help automate other web applications, he made JavaScriptRunner open-source which was later re-named as Selenium Core.
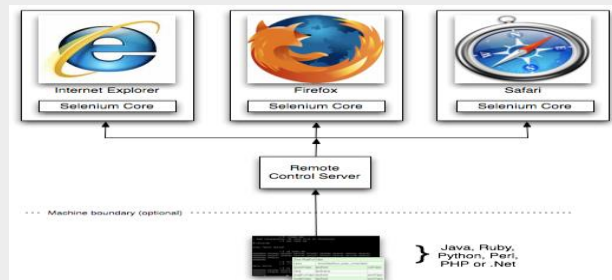
The Same Origin Policy Issue
The **Same Origin Policy** restricts any JavaScript code from accessing elements from a domain that is different from where it was launched. **Example :**
The HTML code in www.google.com uses a JavaScript program "HelloScript.js".
According to the same origin policy HelloScript.js can access pages within google.com such as google.com/mail, google.com/login, or google.com/signup. However, it cannot access pages from different sites such as bing.com/search or yahoo.com/search because they belong to different domains. This is the reason why prior to Selenium RC, testers needed to install local copies of both Selenium Core (a JavaScript program) and the web server containing the web application being tested so they would belong to the same domain.

2.1: Introduction To Selenium
Overview Selenium Remote Control (Selenium 1.0)

- Selenium Remote Control (RC) is a test tool that allows you to write automated web application UI tests in any programming language against any HTTP website using any mainstream JavaScript-enabled browser.
- Selenium RC comes in two parts:
  - A server which automatically launches and kills browsers, and acts as a HTTP proxy for web requests from them
  - Client libraries for your favorite computer language

Overview Selenium Remote Control (Selenium 1.0)

Due to the restriction imposed by the "Same Origin Policy", testers using Selenium Core had to install the application under test (AUT) and the web server on their own local system. Another engineer, Paul Hammant from ThoughtWork's decided to create a server that will act as an HTTP proxy to "trick" the browser into believing that Selenium Core and the web application being tested come from the same domain. This system became known as the Selenium Remote Control or Selenium 1.

2.1: Introduction To Selenium
Overview of Selenium IDE

- Firefox plug in, allows you to record, play back, edit, and debug tests in browser
- Allows you to record user actions on browser window
- Generate scripts from recorded user actions in most of the popular languages like Java, C#, Perl, Ruby etc. However to run them in an automated testing fashion you need to use Selenium Web Driver
- Selenium default scripts are html (added JavaScript) and that is the script we are going to use it in selenium IDE
- The reason for availability of other language is, user can get the scripts for Selenium WebDriver/RC
- It also has a context menu (right-click) integrated with the Firefox browser, which allows the user to pick from a list of assertions and verifications for the selected location

Overview of Selenium IDE

Shinya Kasatani of Japan created Selenium IDE, a Firefox extension that can automate the browser through a record-and-playback feature. He came up with this idea to further increase the speed in creating test cases. He donated Selenium IDE to the Selenium Project in 2006.

2.1: Introduction To Selenium
Overview of Selenium Web Driver (Selenium 2.0)

- Selenium WebDriver is an API designed to provide a simpler, more concise programming interface in addition to addressing some limitations in the Selenium-RC API
- Selenium 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 to it
- Selenium-WebDriver was developed to better support dynamic web pages where elements of a page may change without the page itself being reloaded
- WebDriver's goal is to supply a well-designed object-oriented API that provides improved support for modern advanced web-app testing problems

Overview of Selenium Web Driver (Selenium 2.0)

Simon Stewart created WebDriver circa 2006 when browsers and web applications were becoming more powerful and more restrictive with JavaScript programs like Selenium Core. It was the first cross-platform testing framework that could control the browser from the OS level.

Birth of Selenium 2

In 2008, the whole Selenium Team decided to merge WebDriver and Selenium RC to form a more powerful tool called Selenium 2, with WebDriver being the core. Currently, Selenium RC is still being developed but only in maintenance mode. Most of the Selenium Project's efforts are now focused on Selenium 2.

2.1: Introduction To Selenium
Overview of Selenium Grid

- Selenium Grid allows you to execute multiple tests at the same time on different machines and browsers in order to speed up testing
- It is essential to speed up the test execution in order to ensure maximum benefit from CI
- Running tests in parallel on a grid can improve the speed of your testing greatly Selenium Grid is comprised of one machine called a hub and multiple nodes for execution
- The more nodes you have, the more tests you can run simultaneously
- While you can make your own Selenium grid with a costly device lab or VMs, many have found moving Selenium grid to a third-party cloud is faster and more productive, allowing more tests to run on more OS systems and browsers simultaneously
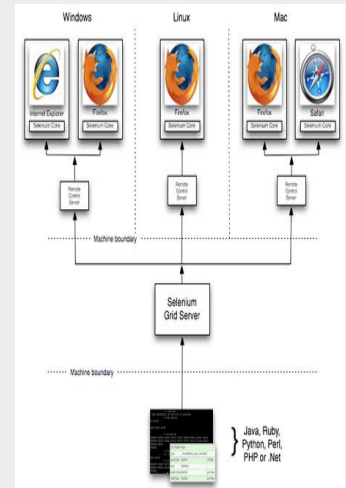
Overview of Selenium Grid

Developed by Patrick Lightbody, Selenium Grid addresses the need of minimizing test execution time as much as possible by facilitating test engineers to run the tests on multiple platforms & on multiple web browsers. Selenium Grid was initially called as "Hosted QA." It was capable of capturing browser screenshots during significant stages, and also of sending out Selenium commands to different machines simultaneously.

2.1: Introduction To Selenium
Selenium 3.0 Out Now!

- Released in October, 2016, Selenium 3.0 comes with bug fixes of Selenium 2.0 with more & improved stability
- The major change in Selenium 3.0 is removal of the original Selenium Core implementation and replacing it with one backed by WebDriver
- This will affect all users of the Selenium RC APIs
- In the new version, the Selenium Core is removed and the RC APIs have become legacy
- The other notable change has been that there is now a W3C specification for browser automation, based on the Open Source WebDriver
- This has yet to reach "recommendation" status, but the people working on it are now focusing on finishing the text and writing the implementations
- Good news! Selenium WebDriver APIs have no major changes
- GeckoDriver – a new WebDriver from Mozilla

2.1: Introduction To Selenium
Why Selenium?

- Cost-effective automation testing tool
- Supports various operating systems
- Supports multiple languages
- Supports range of browsers
- Effective handling of AJAX based web application
- Easy to implement frameworks based on OOP
- Supports implementation of different testing frameworks

Why Selenium?

Selenium is probably the best option for automated testing of Websites today. It is becoming increasingly popular and it is the first choice of automation testers as well as organizations for automating the testing of Web-based applications.

Following are some of the important reasons behind Selenium's growing popularity:
Cost-effective automation tool - Selenium is an open source testing tool and hence it serves for cost-effective automation testing. It is a free download and support is free too.
Supports various operating systems – In comparison with other automation testing tools, Selenium has the capability to operate on almost every operating system.
Supports multiple languages - Selenium supports multiple languages such as Python, Pearl, Ruby, PHP, .NET-C# and Java. You are required to be comfortable in just a single language in order to operate Selenium.
Supports range of browsers – Test scripts written using Selenium can be executed on range of browsers like Opera, Safari, Chrome, IE and Mozilla Firefox etc.

Summary

In this lesson, you have learnt

- Selenium is a free (open source) automated testing suite for web applications across different browsers and platforms
- Selenium is a collection of different tools
- Selenium IDE, a Firefox add-on that you can only use in creating relatively simple test cases and test suites
- Selenium RC allows users to use programming languages in creating complex tests
- WebDriver, allows test scripts to communicate directly to the browser, thereby controlling it from the OS level
- Selenium Grid is used with Selenium RC to execute parallel tests across different browsers and operating systems

Summary

Add the notes here.

Review Question

Question 1
- Select the component which is NOT part of Selenium suite.
  - Selenium IDE
  - Selenium RC
  - Selenium Grid
  - Selenium Web

Question 2: Fill in the Blanks
- WebDriver, unlike Selenium RC, does not rely on _____ for automation.

Review Question

Question 3: Fill in the Blanks
- _____ is the record and playback tool.

Question 4: True/False
- Selenium Grid is used to distribute your test execution on multiple platforms and environments.
- Selenium supports testing of only web based applications.
- Selenium is a set of tools that supports rapid development of test automation scripts for web based applications.