**What is Selenium?**

Selenium is a powerful tool for controlling web browsers through programs and performing browser automation. It is functional for all browsers, works on all major OS and its scripts are written in various languages i.e Python, Java, C#, etc, we will be working with Python

### **Why learn Selenium Python?**

* **Open Source and Portable** – Selenium is an open source and portable Web testing Framework.
* **Combination of tool and DSL** – Selenium is combination of tools and DSL (Domain Specific Language) in order to carry out various types of tests.
* **Easier to understand and implement** – Selenium commands are categorized in terms of different classes which make it easier to understand and implement.
* **Less burden and stress for testers** – As mentioned above, the amount of time required to do testing repeated test scenarios on each and every new build is reduced to zero, almost. Hence, the burden of tester gets reduced.
* **Cost reduction for the Business Clients** – The Business needs to pay the testers their salary, which is saved using automation testing tool. The automation not only saves time but gets cost benefits too, to the business.

**Pre requisites before working on Selenium Python Automation**

1. Python

2. Pycharm

3. Specific Drivers for different Browsers of Selenium

Once u download, extract them in folder…zip se unzip

|  |  |
| --- | --- |
| **Chrome**: | <https://chromedriver.chromium.org/downloads> |
| **Edge**: | <https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/> |
| **Firefox**: | <https://github.com/mozilla/geckodriver/releases> |
| **Safari**: | <https://webkit.org/blog/6900/webdriver-support-in-safari-10/> |

**4. Set up Selenium in your system**

**Installation of selenium in Pycharm**

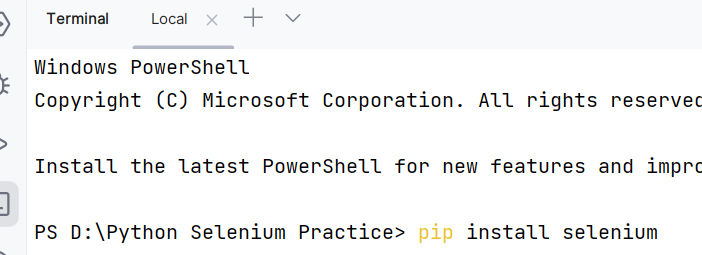
To install Selenium on your python project, follow these steps:

1. Open File > Settings > Project from the PyCharm toolbar
2. Select the project you want to install Selenium on
3. Click the tab "Python Interpreter" within the project tab
4. Click the + symbol to add a new library to the project
5. Type "Selenium" into the new library search box, select the "Selenium" library and confirm by clicking "install package"
6. When the installation has finished, close the project settings window.

**Installation of selenium in terminal(pycharm)**

Open Terminal/Cmd and Write Command as written Below

pip install selenium



**Installation of selenium in cmd**

python -m pip install selenium

from selenium import webdriver

driver = webdriver.Chrome("C:\\Users\yusuf\\Downloads\\chromedriver\_win32 (1)")  
driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login")

# Selenium Basics – Components, Features, Uses and Limitations

# Components-of-Selenium

#### **Selenium IDE**

Selenium IDE (Integrated Development Environment) is the major tool in the Selenium Suite. It is a complete integrated development environment (IDE) for Selenium tests. It is implemented as a Firefox Add-On and as a Chrome Extension. It allows for recording, editing and debugging of functional tests. It was previously known as Selenium Recorder.

Scripts may be automatically recorded and edited manually providing autocompletion support and the ability to move commands around quickly. Scripts are recorded in Selenese, a special test scripting language for Selenium. Selenese provides commands for performing actions in a browser (click a link, select an option) and for retrieving data from the resulting pages.

#### **Selenium RC (Remote control)**

Selenium Remote Control (RC) is a server, written in Java, that accepts commands for the browser via HTTP. RC makes it possible to write automated tests for a web application in any programming language, which allows for better integration of Selenium in existing unit test frameworks. To make writing tests easier, Selenium project currently provides client drivers for PHP, Python, Ruby, .NET, Perl and Java. The Java driver can also be used with JavaScript (via the Rhino engine). An instance of selenium RC server is needed to launch html test case – which means that the port should be different for each parallel run. However, for Java/PHP test case only one Selenium RC instance needs to be running continuously.

#### Selenium Web Driver

Selenium WebDriver is the successor to Selenium RC. Selenium WebDriver accepts commands (sent in Selenese, or via a Client API) and sends them to a browser. This is implemented through a browser-specific browser driver, which sends commands to a browser and retrieves results. Most browser drivers actually launch and access a browser application (such as Firefox, Google Chrome, Internet Explorer, Safari, or Microsoft Edge); there is also an HtmlUnit browser driver, which simulates a browser using the headless browser HtmlUnit.

Selenium WebDriver does not need a special server to execute tests. Instead, the WebDriver directly starts a browser instance and controls it. However, Selenium Grid can be used with WebDriver to execute tests on remote systems (see below). Where possible, WebDriver uses native operating system level functionality rather than browser-based JavaScript commands to drive the browser. This bypasses problems with subtle differences between native and JavaScript commands, including security restrictions.

It is module – contains the functions or classes. Classes contains different types methods.

It is available in selenium package.

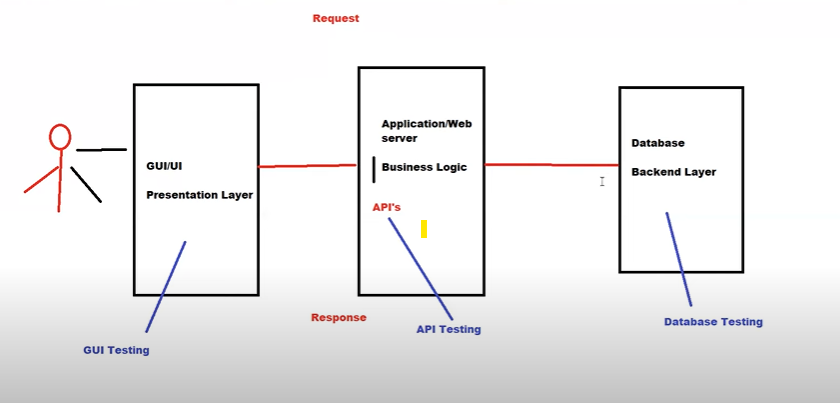
For eg each browser there are different classes

Chrome browser = Chrome () class

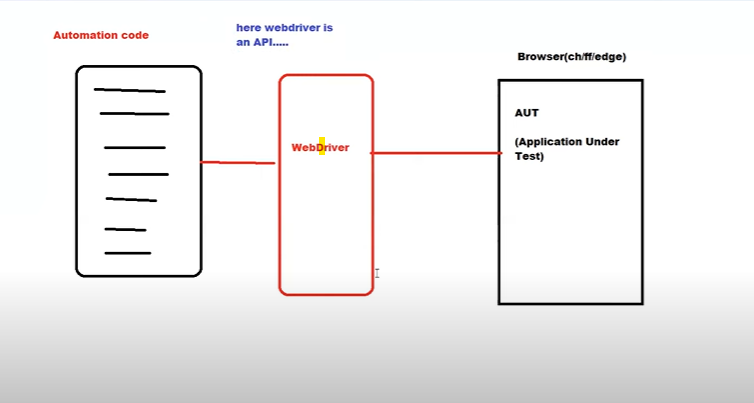
Firefox browser = Firfeox () class

Edge browser = Edge() class

Architecture of Application/Website

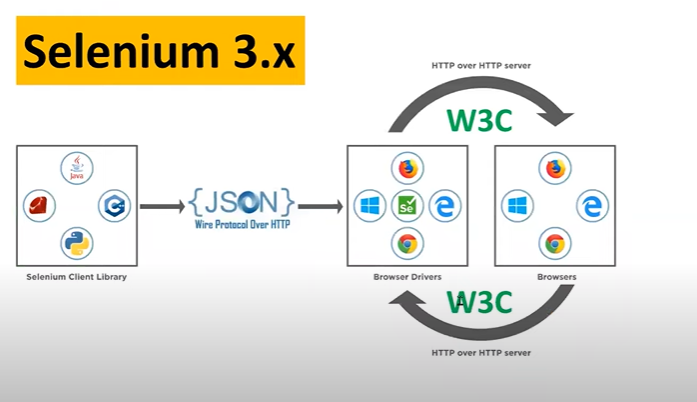


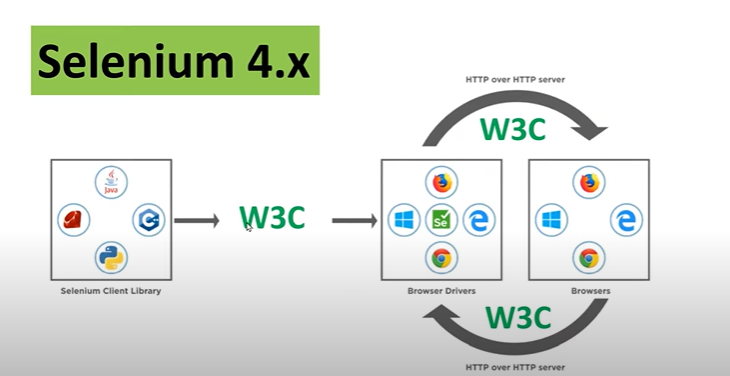
Webdriver is an API(Application interface Language)

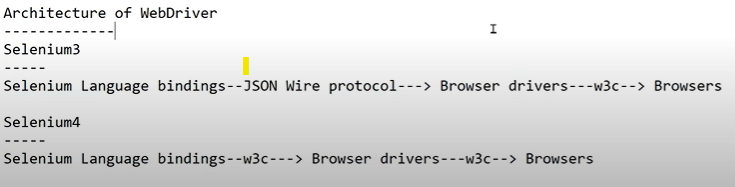


Clclassass and methods

Architecture Of Selenium Webdriver







#### **Selenium GRID**

Selenium Grid is a server that allows tests to use web browser instances running on remote machines. With Selenium Grid, one server acts as the hub. Tests contact the hub to obtain access to browser instances. The hub has a list of servers that provide access to browser instances (WebDriver nodes), and lets tests use these instances. Selenium Grid allows running tests in parallel on multiple machines and to manage different browser versions and browser configurations centrally (instead of in each individual test).

The ability to run tests on remote browser instances is useful to spread the load of testing across several machines and to run tests in browsers running on different platforms or operating systems. The latter is particularly useful in cases where not all browsers to be used for testing can run on the same platform.

## **Features**

* **Open Source and Portable** – Selenium is an open source and portable Web testing Framework.
* **Combination of tool and DSL** – Selenium is combination of tools and DSL (Domain Specific Language) in order to carry out various types of tests.
* **Easier to understand and implement** – Selenium commands are categorized in terms of different classes which make it easier to understand and implement.
* **Reduce test execution time** – Selenium supports parallel test execution that reduce the time taken in executing parallel tests.
* **Lesser resources required** – Selenium requires lesser resources when compared to its competitors like UFT, RFT, etc.
* **Supports Multiple Programming Languages** – C#, Java, Python, PHP, Ruby, Perl, and JavaScript
* **Supports Multiple Operating Systems** – Android, iOS, Windows, Linux, Mac, Solaris.
* **Supports Multiple Browsers** – Google Chrome, Mozilla Firefox, Internet Explorer, Edge, Opera, Safari, etc.
* **Parallel Test Execution** – It also supports parallel test execution which reduces time and increases the efficiency of tests.

## **Applications**

Selenium WebDriver 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. However, using the Selenium WebDriver, we can automate testing for web applications only.

* **Open Source and Portable** – Selenium is an open source and portable Web testing Framework.
* **Less burden and stress for testers** – As mentioned above, the amount of time required to do testing repeated test scenarios on each and every new build is reduced to zero, almost. Hence, the burden of tester gets reduced.
* **Cost reduction for the Business Clients** – The Business needs to pay the testers their salary, which is saved using automation testing tool. The automation not only saves time but gets cost benefits too, to the business.
* **Increased test coverage** – With the uses of Selenium, testing time gets reduced and hence the tester can do more testing on other test scenarios at the same time.
* **Reduce test execution time** – Selenium supports parallel test execution that reduce the time taken in executing parallel tests.

## Limitations

* **No support for desktop applications** – Selenium does not support testing for desktop applications.
* **Expertise** – Selenium requires expertise of your team — and resources to manage.
* **Maintenance and Scalability** – Selenium is a maintenance-heavy framework — and is difficult to scale as one grows.
* **Open Source Forums** – Since Selenium is open source software, one has to rely on community forums to get your technical issues resolved.
* **No support for REST and SOAP Platforms** – We can’t perform automation tests on web services like SOAP or REST using Selenium.
* **No Reporting capability** – Selenium does not have any inbuilt reporting capability, one has to rely on plug-ins like JUnit and TestNG for test reports.
* **Image Testing** – It is not possible to perform testing on images. One needs to integrate Selenium with Sikuli for image testing.

# Applications and Uses of Selenium WebDriver

Selenium Webdriver is a powerful tool for controlling web browser through program. It is functional for all browsers, works on all major OS and its scripts are written in various languages i.e Python, Java, C#, etc. Selenium Webdriver is a primary automation tool used by developers all around the world.  
This article revolves around **Major Applications of Selenium WebDriver.** Mastering Selenium will help you automate your day to day tasks like controlling your tweets, Whatsapp texting and even just googling without actually opening a browser in just 15-30 lines of python code.

#### Applications and Uses of Selenium Webdriver

Selenium WebDriver 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. However, using the Selenium WebDriver, we can automate testing for web applications only.

* **Open Source and Portable** – Selenium is an open source and portable Web testing Framework.
* **Combination of tool and DSL** – Selenium is combination of tools and DSL (Domain Specific Language) in order to carry out various types of tests.
* **Easier to understand and implement** – Selenium commands are categorized in terms of different classes which make it easier to understand and implement.
* **Less burden and stress for testers** – As mentioned above, the amount of time required to do testing repeated test scenarios on each and every new build is reduced to zero, almost. Hence, the burden of tester gets reduced.
* **Cost reduction for the Business Clients** – The Business needs to pay the testers their salary, which is saved using automation testing tool. The automation not only saves time but gets cost benefits too, to the business.
* **Increased test coverage** – With the uses of Selenium, testing time gets reduced and hence the tester can do more testing on other test scenarios at the same time.
* **Reduce test execution time** – Selenium supports parallel test execution that reduce the time taken in executing parallel tests.
* **Lesser resources required** – Selenium requires lesser resources when compared to its competitors like UFT, RFT, etc.
* **Supports Multiple Programming Languages** – C#, Java, Python, PHP, Ruby, Perl, and JavaScript
* **Supports Multiple Operating Systems** – Android, iOS, Windows, Linux, Mac, Solaris.
* **Supports Multiple Browsers** – Google Chrome, Mozilla Firefox, Internet Explorer, Edge, Opera, Safari, etc.
* **Parallel Test Execution** – It also supports parallel test execution which reduces time and increases the efficiency of tests.
* **A flexible language** – Once the test cases are prepared, they can be executed on any operating system like Linux, Macintosh, etc.
* **No installation Required –** Selenium web driver does not require server installation, test scripts interact directly with the browser.

Test Case Example

1. Open the Browser(chrome,ff,edge)

2. Open the URL(

3. Enter Username

4. Enter Password

5. Click on Login

6. Maximize the window

7. Minimize the window

8. Close the window

import time  
from selenium import webdriver  
from selenium.webdriver.common.by import By  
options = webdriver.ChromeOptions()  
options.add\_experimental\_option("detach",True)  
browser = webdriver.Chrome(options=options)  
username = "Username"  
password = "Password"  
time.sleep(5)  
url = 'https://www.facebook.com/'  
browser.get(url)  
  
browser.find\_element(By.NAME, "email").send\_keys(username)  
browser.find\_element(By.NAME, "pass").send\_keys(password)  
time.sleep(5)  
*# browser.find\_element(By.ID,"u\_0\_5\_1V")*browser.maximize\_window()  
time.sleep(5)  
browser.close()

Selenium 3



Selenium 4

