**

**Softech** **Solutions Inc.**

[www.softechnosolutionsgroup.com](http://www.softechnosolutionsgroup.com)

[saney.alam@softechnosolutionsgroup.com](mailto:saney.alam@softechnosolutionsgroup.com%20)

# A Complete Guide for Beginners

Selenium – Running Test on Different Browsers

***Lecture Notes***

Selenium – Running Test on Different Browsers

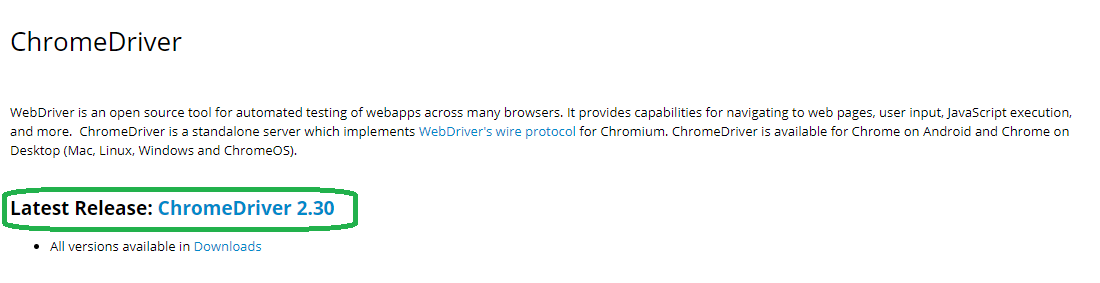
|  |  |
| --- | --- |
| ***Table of Contents Page*** | |
| ***Running Tests on Google Chrome Browser*** | ***02*** |
| ***Chrome Driver Server*** | ***04*** |
| ***Launching Chrome Browser using Selenium WebDriver*** | ***07*** |
| ***Running Tests on Internet Explorer Browser*** |  |
|  |  |

**Running Tests on Google Chrome Browser**

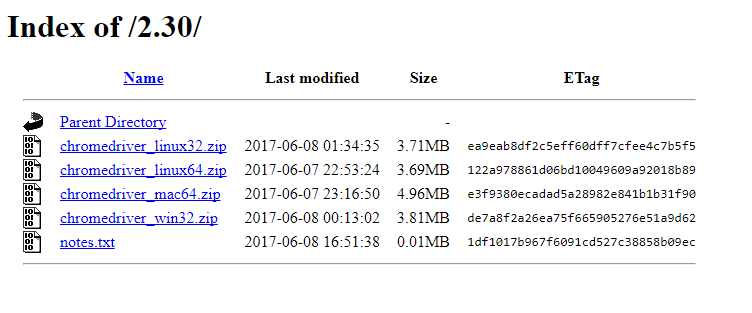
## Chrome browser implements the *WebDriver* protocol using an executable called ***ChromeDriver.exe***. This executable starts a server on your system. All your tests communicate to this server to run your tests.

## **Chrome Driver Server**

First we have to download the ***Chrome Driver Server*** from the Chromium project [**here**.](https://sites.google.com/a/chromium.org/chromedriver/) This website details out all the features that are available in the ***Chrome driver***. We are, at present, interested in downloading the Server. On the page you will find a section of latest release. That is where we will have a link to download the latest release of ***Chrome WebDriver***. Refer to the image below



Click on the link and you can follow it to download the Chrome driver. The main reason why I wanted to show you this page is to make you aware of the brilliant amount of documentation that is available for chrome driver.



Download the zip file based on the OS platform that you have. Unzip the zip file and keep it somewhere on a known location on your PC. Now let’s see how we can run tests in a Chrome Driver.

## **Launching Chrome Browser using Selenium WebDriver**

In order to launch ***Chrome*** browser we have to do two steps

1. Set a system property ***“****webdriver.chrome.driver****”*** *to the path of your*ChromeDriver.exe*file*
2. Instantiate a *ChromeDriver* class.

Here is a sample code to do that:

**public class LaunchingChrome {**

**public static void main(String[] args) {**

**String exePath = "C:\\Users\\abc\\Desktop\\Server\\chromedriver.exe";**

**System.setProperty("webdriver.chrome.driver", exePath);**

**WebDriver driver = new ChromeDriver();**

**driver.get("http://www.google.com");**

**}**

**}**

## **Assignment Operators**

So that’s it, now we can write any test that we want using chrome driver.

**Running Tests on IE Browser**

### Running tests in IE Explorer is quite easy. Internet Explorer can’t be launched directly, we have to communicate with the Internet explorer via Internet explorer driver. So what is an Internet Explorer driver and how do we get it.

### **Internet Explorer Driver Server**

### Internet explorer driver server is the link between your tests in Selenium and the Internet explorer browser. As selenium WebDriver has no native implementation of IE, we have to direct all the driver commands through IE driver server. IE driver server is an executable file that you need to have in one of the system path before starting your tests.

To download the server you can go “ http://selenium-release.storage.googleapis.com/index.html”. Just choose the latest version and download it based whether you are on the 32 bit or a 64 bit operating system.

Download the zip file and extract it in a known location on your computer.

## **Launching IE using Selenium WebDriver**

### Approach 1: Setting up the *webdriver.ie.driver* property

Selenium Webdriver has a class called *InternetExplorerDriver* that is used to launch and control IE browser. The code to launch IE Driver is exactly the same as if you were launching a *ChromeDriver* or a *FirefoxDriver*. All we need to do is specify the correct path to the *IEDriver* server and use the *InternetExplorerDriver* class. Like this

**public class LaunchingIE {**

**public static void main(String[] args) {**

**//Path to the folder where you have extracted the IEDriverServer executable**

**String service = "C:\\Users\\abc\\Desktop\\Server\\IEDriverServer.exe";**

**System.setProperty("webdriver.ie.driver", service);**

**InternetExplorerDriver driver = new InternetExplorerDriver();**

**driver.get("http://yahoo.com");**

**}**

**}**

System.setProperty is used to set a *webdriver.ie.driver* property to the path of IE driver server executable. This helps Selenium to start the IE Driver server and then launch IE Browser.

### Approach 2: Using the Service builder

Internet driver service is a class that helps you configures the IE driver server. Instead of specifying the executable path as a system property you can directly specify it in a Driver Service like this. In the below code we will be using *InternetExplorerDriverService.Builder* class. We can also specify the log file path, which will contain the IE driver logs. These logs can be helpful in debugging issues later on. Here is the full sample code.

### **Set up WebDriver Extension for***Safari***browser**

## **Exceptions in using GeckoDriver**

There multiple exceptions you can face by using the GeckoDriver:

### ***Error 1:***

***Exception****:* Exception in thread “main” org.openqa.selenium.WebDriverException: Failed to decode response from marionette (WARNING: The server did not provide any stacktrace information).

***Solution****:* There is some mismatch between the Gecko version or Firefox version or Selenium version. First try updating the Gecko version from[***Github***](https://github.com/mozilla/geckodriver/releases)***.***If you still face the error try updating the FF version as well***.***

### ***Error 2:***

***Exception***: Exception in thread “main” org.openqa.selenium.remote.UnreachableBrowserException: Error communicating with the remote browser. It may have died.

***Solution****:* This happens to me because I was using driver.quit(). So avoid using this and replace this statement with driver.close().