**<http://www.software-testing-tutorials-automation.com/2013/08/how-to-download-and-install-selenium.html>**

**How to download selenium and install Selenium Webdriver with Eclipse and Java Step By Step**

**Download selenium webdriver** and **install selenium webdriver** is easy. You need to download selenium jar files. Then configure downloaded selenium jar files in eclipse. Actually there is nothing to install except JDK. Let me describe you step by step process of download, installation and configuration of web driver software and other required components. You can view my post about "[**What is selenium webdriver**](http://www.software-testing-tutorials-automation.com/2013/07/what-is-selenium-webdriver.html)" if you wants to know difference between WebDriver and selenium RC software tool.

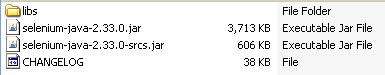
**(Note : I am suggesting you to take a tour of** [**Basic selenium commands tutorials with examples**](http://www.software-testing-tutorials-automation.com/2013/07/list-of-selenium-commands-with-examples.html) **before going ahead for webdriver. It will improve your basic knowledge and helps you to create webdriver scripts very easily. )**

**Steps To Setup and configure Selenium Webdriver With Eclipse and Java**  
  
(Note : You can [View More Articles On WebDriver](http://www.software-testing-tutorials-automation.com/2014/01/learn-selenium-webdriver-online-free.html) to learn it step by step)  
**Step 1 : Download and install Java in your system**  
First of all you need to install JDK (Java development kit) software in your system. So your next question will be "how can i download java" [**VIEW THIS ARTICLE**](http://www.software-testing-tutorials-automation.com/2015/09/steps-to-download-and-install-javajdk.html) to know how to download and install Java(JDK) software.  
  
**Step 2 : Download and install Eclipse**  
[**Download**](http://www.eclipse.org/downloads/) Eclipse for Java Developers and extract save it in any drive. It is totally free. You can run 'eclipse.exe' directly so you do not need to install Eclipse in your system.

**Step 3 : Download WebDriver Jar Files.**  
Selenium webdriver supports many languages and each language has its own client driver. Here we are configuring selenium 2 software with java so we need 'webdriver Java client driver'. [**Click hTMere**](http://docs.seleniumhq.org/download/) to go on WebDriver Java client driver download page for webdriver download file. On that page click on 'Download' link of java client driver as shown in bellow image.

[](http://2.bp.blogspot.com/-A0FtoMe1ldk/UfcZF3pwyzI/AAAAAAAAAN0/RyWsob1NBeI/s1600/Download+and+install+Selenium+Java+Client+Driver.PNG)

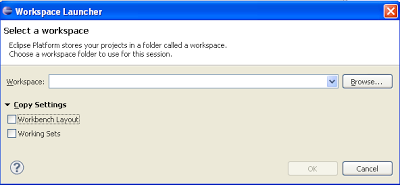
(language-specific client driver's version is changing time to time so it may be different version when you will visit download page. )  
  
Downloaded 'webDriver Java client driver' will be in zip format. Extract and save it in your system at path D:\selenium-2.33.0. There will be 'libs' folder, 2 jar files and change log in unzipped folder as shown in bellow figure. We will use all these files for configuring webdriver in eclipse.

[](http://3.bp.blogspot.com/-olTgvlMzltQ/UfceGcFxfQI/AAAAAAAAAOE/zXzcvJGI8UY/s1600/Java+client+driver+for+webdriver.PNG)

**Step 4 : Start Eclipse and configure it with selenium 2 (webdriver)**

* **Select WorkSpace on eclipse start up**

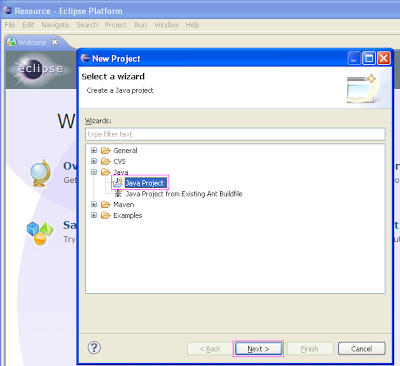
Double click on 'eclipse.exe' to start eclipse software application. First time when you start eclipse software application, it will ask you to select your workspace where your work will be stored as shown in bellow image. Create new folder in D: drive with name 'Webdriverwork' and select it as your workspace. You can change it later on from 'Switch Workspace' under 'file' menu of eclipse.

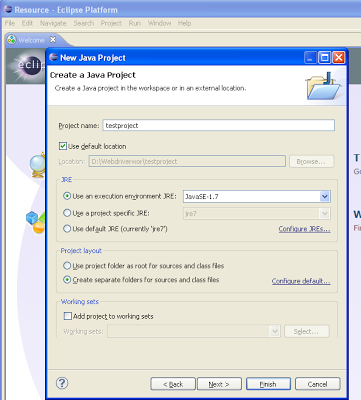
[](http://1.bp.blogspot.com/-2SCYUY2rh7M/UfclHnun07I/AAAAAAAAAOU/stX4nflbavA/s1600/Selecting+workspace+in+eclipse.PNG)

After selecting workspace folder, Eclipse will be open.

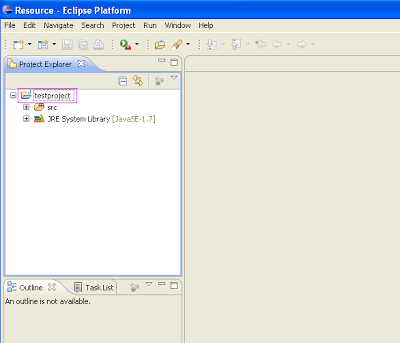
* **Create new project**

Create new java project from **File > New > Project > Java Project** and give your project name 'testproject' as shown in bellow given figures. Click on finish button.

[](http://4.bp.blogspot.com/-Ul8G7Lx-wFY/Ufcpam__tcI/AAAAAAAAAOs/_hMhtVFXmvw/s1600/Create+new+webdriver+project+in+eclipse.PNG)

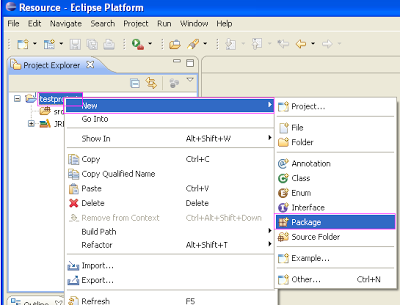
[](http://4.bp.blogspot.com/-fy4b6Px8RpQ/UfcqRC9da7I/AAAAAAAAAO4/7mjA6d8vARo/s1600/Create+new+java+project+in+eclipse.PNG)

Now your new created project 'testproject' will display in eclipse project explorer as bellow.

[](http://2.bp.blogspot.com/-HyDOHu6Xj6w/UffjOhoXHAI/AAAAAAAAAPI/ZOTf4bDqhYo/s1600/Webdriver+project+on+project+explorer.PNG)

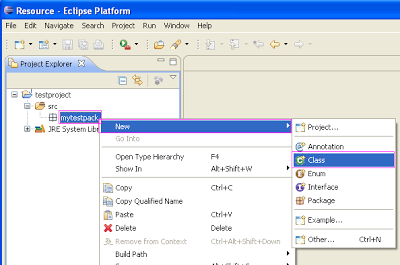
* **Create new package**

Right click on project name 'testproject' and select **New > Package**. Give your package name = 'mytestpack' and **click on finish button**. It will add new package with name 'mytestpack' under project name 'testproject'.

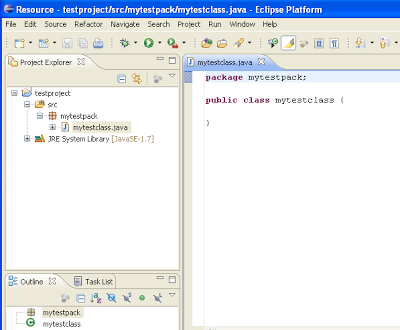
[](http://4.bp.blogspot.com/-EHtWmHIrymg/UfflsSS7hcI/AAAAAAAAAPY/UOdNMNESwVM/s1600/Creating+new+package+in+eclipse.PNG)

* **Create New Class**

Right click on package 'mytestpack' and select New > Class and set class name = 'mytestclass' and **click on Finish button**. It will add new class 'mytestclass' under package 'mytestpack'.

[](http://1.bp.blogspot.com/-1Kw2z_lAv9Q/UffnfB7hewI/AAAAAAAAAPo/4G6Et02K-t4/s1600/Creating+new+class+for+webdriver.PNG)

Now your Eclipse window will looks like bellow.

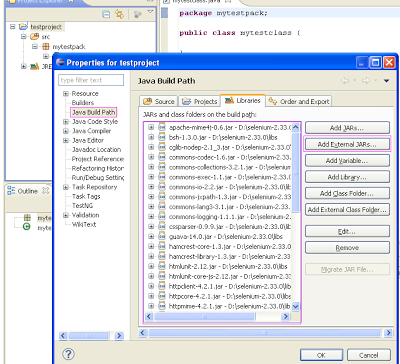
[](http://2.bp.blogspot.com/-j8oxKNdynS4/UffoJvA45yI/AAAAAAAAAPw/oCChCa83Hko/s1600/webdriver+configuration+with+eclipse.PNG)

* **Add external jar file to java build path**

Now you need to add selenium webdriver's jar files in to java build path.

* **Right click on** project 'testproject' > **Select** Properties > **Select** Java build path > **Navigate to** Libraries tab
* **Click on** add external JARs button > **select** both .jar files from D:\selenium-2.33.0.
* **Click on** add external JARs button > **select** all .jar files from D:\selenium-2.33.0\libs

Now your testproject's properties dialogue will looks like bellow.

[](http://1.bp.blogspot.com/-yReLcdTp0zw/UffrniiKE0I/AAAAAAAAAQI/0RLupUU4yl8/s1600/Adding+webdriver+external+jar+files+in+eclipse.PNG)

That's all about configuration of WebDriver software with eclipse. Now you are ready to write your test in eclipse and run it in WebDriver.  
  
You can Read My Post about [**how to write and run your first test in WebDriver**](http://www.software-testing-tutorials-automation.com/2013/09/create-and-run-first-webdriver-script.html).

download selenium webdriver, install webdriver, download webdriver selenium, selenium testing, selenium testing tool, how to download selenium webdriver, what is selenium webdriver, webdriver download, selenium webdriver download, selenium automation, selenium download, selenium install, install selenium webdriver, install selenium webdriver in eclipse, eclipse and selenium, java and selenium, how to install a server, selenium driver, how to setup selenium webdriver, download webdriver selenium, selenium webdriver tutorial java

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2013/07/what-is-selenium-webdriver.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2013/09/create-and-run-first-webdriver-script.html)

**Create And Run First Webdriver Script With Eclipse**

**Run First Webdriver Script**

You need to [**install webdriver with eclipse**](http://www.software-testing-tutorials-automation.com/2013/08/how-to-download-and-install-selenium.html) to run your script in webdriver. After installation of webdriver software testing tool, You need to write a java code in eclipse for your test case of software web application. Let me give you one simple example of creating simple webdriver script for software web application. First of all you need to create the package called

"Testing\_Pack" under your project and then create new class file 'mytestclass.java' under that package. Now copy-paste bellow given code In your 'mytestclass.java' file.

package Testing\_Pack;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class mytestclass {

public static void main(String[] args) {

WebDriver driver = new FirefoxDriver();

driver.get("http://only-testing-blog.blogspot.in");

String i = driver.getCurrentUrl();

System.out.println(i);

driver.close();

}

}

Now you are ready to run your script from Run menu as shown bellow.

[http://3.bp.blogspot.com/-hWKNpro86NU/UjEmh-KlcEI/AAAAAAAAAV0/K1IBG__M1bw/s400/Run+webdriver+first+script.PNG](http://3.bp.blogspot.com/-hWKNpro86NU/UjEmh-KlcEI/AAAAAAAAAV0/K1IBG__M1bw/s1600/Run+webdriver+first+script.PNG)

**Script Explanation**

Above script will

* open Firefox browser.
* Then **driver.get** syntax will open 'http://only-testing-blog.blogspot.in/' software application in firefox browser.
* Then **driver.getCurrentUrl()** will get the current page URL and it will be stored in variable 'i'. You can do same thing using [**"storeLocation" command**](http://www.software-testing-tutorials-automation.com/search/label/storeLocation%20Command) in selenium IDE
* And at last, it will print value of variable in console as shown bellow.

[http://4.bp.blogspot.com/-kODy7OB-XmM/UjJlYsSFEaI/AAAAAAAAAWE/5ofuZwc4Bpk/s400/webdriver+first+test+case+execution.PNG](http://4.bp.blogspot.com/-kODy7OB-XmM/UjJlYsSFEaI/AAAAAAAAAWE/5ofuZwc4Bpk/s1600/webdriver+first+test+case+execution.PNG)

So this is the simple webdriver script example. You can create it for your own software web application too by replacing URL in above script. We will learn more scripts in detail in my upcoming posts.  
  
[VIEW MORE TUTORIALS ON WEBDRIVER](http://www.software-testing-tutorials-automation.com/p/selenium-webdriver.html)

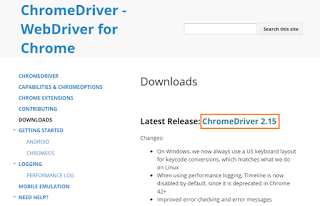
# Running Selenium Webdriver Test In Google Chrome

You can read my [**THIS POST**](http://www.software-testing-tutorials-automation.com/2014/01/locating-web-element-by-classname-in.html) to know **how to run selenium webdriver test** of software application in Mozilla Firefox browser. As you know, webdriver software testing tool support Google Chrome browser too. You need to do something extra for **launching webdriver test in in Google chrome browser**. Let me describe you **steps to launch webdriver test in Google Chrome** for software web application.

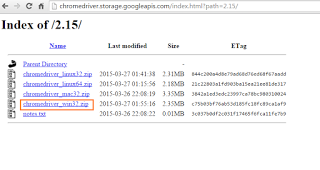
Create selenium webdriver data driven framework from scratch [@This Page](http://www.software-testing-tutorials-automation.com/2014/07/create-data-driven-framework-for.html).

**Download ChromeDriver server**

First of all, download latest version of ChromeDriver server for webdriver software testing tool. You can download it directly from [**https://sites.google.com/a/chromium.org/chromedriver/downloads**](https://sites.google.com/a/chromium.org/chromedriver/downloads)**.** Current latest version for win32 is "ChromeDriver 2.15" as shown bellow.

[](http://3.bp.blogspot.com/-1O5Wo3veJ8o/VViM3tzLfVI/AAAAAAAABlw/Ig6IcrwnhtM/s1600/chrome%2Bdriver.png)

Click on "ChromeDriver 2.15" link. It will take you to download page.

[](http://3.bp.blogspot.com/-yKURJLAue5s/VViNTclf32I/AAAAAAAABl4/8M4BlxU5F58/s1600/download%2Bchrome%2Bdriver.png)

Click on link shown above to download chrome driver zip file. On completion of download, extract zip file to D:/ drive.

Now in eclipse, create new project with name = 'chromeproject' and create new class with name = 'chromebrowser'. Don't forget to select 'public static void main(String[] args)' method during new java class creation. Write code as shown bellow in class file.

package mytestpack;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class chromebrowser {

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver", "D:\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

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

if (driver.findElement(By.xpath("//input[@name='q']")).isEnabled()) {

System.out.println("Google search text box Is enabled.");

driver.findElement(By.xpath("//input[@name='q']")).sendKeys("WebDriver Test successful.");

driver.findElement(By.xpath("//button[@name='btnG']")).click();

System.out.println("Google search completed.");

} else {

System.out.println("Google search test box Is Not enabled.");

}

driver.close();

}

}

Look in to above example. I set path for Google chrome driver as System.setProperty("webdriver.chrome.driver", "D:\\chromedriver\_win32\\chromedriver.exe");

So now, your software web application's test will run in Google chrome browser. Learn how to execute webdriver test in Internet Explorer browser in [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html).

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2013/09/create-and-run-first-webdriver-script.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html)

# How To Run Selenium WebDriver Test Script In Internet Explorer With Steps

Frequently we need to **run selenium WebDriver test** in different browsers like **run selenium in internet explorer**, Google Chrome, Opera, Mozilla Firefox etc.. for browser compatibility software testing. Selenium WebDriver software testing tool has separate driver for each browser. **Selenium WebDriver has InternetExplorerDriver** for **IE browser**. Earlier, I have already described [**HOW TO RUN WEBDRIVER TEST SCRIPT IN FIREFOX**](http://www.software-testing-tutorials-automation.com/2014/01/locating-web-element-by-classname-in.html) and [**HOW TO RUN TEST SCRIPT IN GOOGLE CHROME BROWSER**](http://www.software-testing-tutorials-automation.com/2013/09/running-selenium-webdriver-test-in.html). Now let's learn **how to run WebDriver test In IE** for software web application.

## Configuration Steps To Run selenium Test In  internet explorer

**Step 1 : Download IEDriverServer.exe**

For running **selenium internet explorer** test, you need IEDriverServer.exe. You can download It from It from bellow given links.

* Go to [**THIS PAGE**](http://www.software-testing-tutorials-automation.com/2014/07/selenium-webdriver-frameworks-required.html) and click on IEDriverServer.zip link.

Alternate download location

* You can download It from [**THIS PAGE**](http://www.seleniumhq.org/download/) too as shown In bellow Image.

[](http://2.bp.blogspot.com/-5y0pUM8dGX4/VXwfKFhApMI/AAAAAAAABpg/jKWl716Xdlo/s1600/download%2BIEDriverServer%2Bexe.png)

Save IEDriverServer.exe In your **D:** drive.

**Step 2 : Configure IE Browser To Resolve Expected Errors**

Before running your software automation **selenium internet explorer** test, You need to configure your IE browser to resolved bellow given 2 common errors which people are facing occasionally.

**Error 1** : Exception in thread "main" org.openqa.selenium.remote.SessionNotFoundException: Unexpected error launching Internet Explorer. Protected Mode settings are not the same for all zones. Enable Protected Mode must be set to the same value (enabled or disabled) for all zones. (WARNING: The server did not provide any stacktrace information).

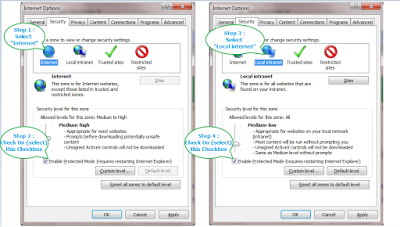
**Error 2** : Exception in thread "main" org.openqa.selenium.remote.SessionNotFoundException: Unexpected error launching Internet Explorer. Browser zoom level was set to 200%. It should be set to 100% (WARNING: The server did not provide any stacktrace information).

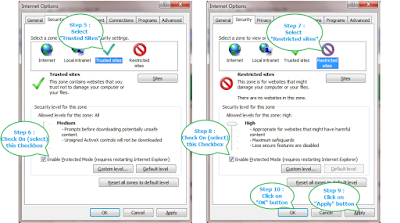
Solution to resolve above errors Is as bellow.

**Solution To Resolve Error 1 : Enable protected mode for all zones**

You need to enable protected mode for all zones from Internet Options -> Security tab. To enable protected mode for all zones

1. Open Internet Explorer browser.
2. Go to menu **Tools** -> **Internet Options**.
3. Click on **Security** tab.
4. Select **Internet** from "Select a zone to view or change security settings" and **Select(check)** check box "Enable Protected Mode" from In the "Security level for this zone" block as shown In bellow Images.
5. Apply same thing for all other 3 zones -> **Local Internet, Trusted Sites** and **Restricted Sites** as shown In bellow Images.

[](http://4.bp.blogspot.com/-l8rN9W7rKU4/VXw31BaULdI/AAAAAAAABpw/HOEeX3TJ6cc/s1600/enable%2Bprotected%2Bmodes%2BIn%2BIE%2Bfor%2Bselenium.png)

[](http://3.bp.blogspot.com/-S47wvlbiFzc/VXw39X4Fo8I/AAAAAAAABp4/ioqvdqsL4-o/s1600/enable%2Bprotected%2Bmodes%2BIn%2BIE%2Bfor%2Bwebdriver.png)

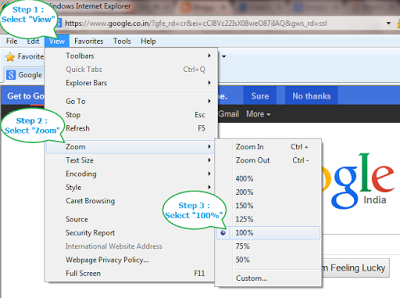
This setting will resolve error related to "Protected Mode settings are not the same for all zones."  
  
Also you can resolve this error on run time. View [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-enable.html) for detailed description and example.

**Solution To Resolve Error 2 : Set IE browser zoom level to 100%**

By default, IE browser's zoom level will be 100%. But If someone has changed It to 200% or any other level then you will face error during webdriver software test execution In IE browser and your test will fail. You need to set IE browser's zoom level 100%.

To Set IE browser's zoom level 100%

1. Open Internet Explorer browser.
2. Go to menu **View** -> **Zoom** -> **Select 100%** as shown In bellow Image.

[](http://1.bp.blogspot.com/-FaVewkAdXFg/VXw6vznhO8I/AAAAAAAABqE/Zg94T1ca_qo/s1600/set%2BIE%2Bbrowser%2Bzoom%2Blevel%2B100%2B%2525%2Bfor%2Bselenium%2Bwebdriver.png)

This setting will resolve error related to "Browser zoom level was set to 200%. It should be set to 100%"  
  
Also you can resolve this error on run time. View [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-set-ie-browser.html) for detailed description and example.

**Step 3 : Execute First WebDriver Test In Internet Explorer browser**

Now you are all set to execute first selenium webdriver In IE browser. I have created sample test to run test In IE browser. Execute bellow given sample calc test In eclipse and verify Its output In console.

IEBrowserSampleTest.java

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

public class IEBrowserSampleTest {

public static void main(String[] args) throws Exception {

**// Set path of IEDriverServer.exe.**

**// Note : IEDriverServer.exe should be In D: drive.**

System.setProperty("webdriver.ie.driver", "D://IEDriverServer.exe");

**// Initialize InternetExplorerDriver Instance.**

WebDriver driver = new InternetExplorerDriver();

**// Load sample calc test URL.**

driver.get("http://only-testing-blog.blogspot.com/2014/04/calc.html");

**// Execute sample calc test.**

driver.findElement(By.xpath("//input[@id='1']")).click();

driver.findElement(By.xpath("//input[@id='plus']")).click();

driver.findElement(By.xpath("//input[@id='6']")).click();

driver.findElement(By.xpath("//input[@id='equals']")).click();

String result = driver.findElement(By.xpath("//input[@id='Resultbox']")).getAttribute("value");

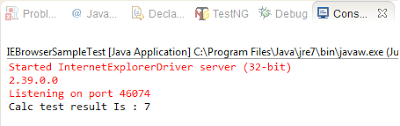
System.out.println("Calc test result Is : " + result);

driver.close();

}

}

At the end of execution, console output will be looks like bellow.

[](http://4.bp.blogspot.com/-wwZi4leeYtk/VXw-ny4MtXI/AAAAAAAABqY/MBnR61BGQ-8/s1600/IEdrivertest%2Bconsole%2Btest%2Bexecution%2Boutput.png)

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/05/selenium-webdriver-advanced-tutorials.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-enable.html)

**IEDriver - How To Resolve "Enable Protected Mode For All Zones" Error RunTime**

Earlier we learnt how to execute WebDriver software test In Internet Explorer browser using InternetExplorerDriver In previous post. We have also learnt how to resolved "Protected Mode settings are not the same for all zones. **Enable Protected Mode must be set to the same value** (enabled or disabled) for all zones." **error** by setting "Enable Protected Mode" check box value same

for all zones manually In Internet Explorer browser.

Manual browser setting will remain limited to your computer only means If same software test will be executed on other computer then your test will fail If IE browser's "Enable Protected Mode" settings not configured In that computer. To resolve this Issue, We need something In our software automation test script which can configure all these settings Itself on run time. Means It should handle all zones protected mode security settings Itself when you run your test script on any machine. Is It possible? Answer Is "Yes".

We can do It using DesiredCapabilities class of selenium WebDriver software testing Interface. Bellow given code will set IEDriver capability to Ignore "Enable Protected Mode" settings and will allow you to run your test without any error regardless whatever your settings are there for "Enable Protected Mode" In IE browser.

**// Set capability of IE driver to Ignore all zones browser protected mode settings.**

DesiredCapabilities caps = DesiredCapabilities.internetExplorer();

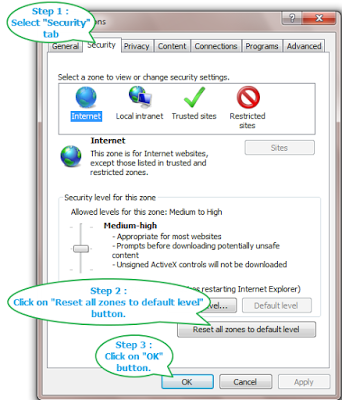
caps.setCapability(**InternetExplorerDriver.INTRODUCE\_FLAKINESS\_BY\_IGNORING\_SECURITY\_DOMAINS**,**true**);

**// Initialize InternetExplorerDriver Instance using new capability.**

WebDriver driver = new InternetExplorerDriver(caps);

To experiment It practically, I have created simple calc software test script as bellow. Before executing this example software test script, You need to reset all zones to default level as bellow.

1. Open IE browser.
2. Go to **Tools** -> **Internet Options**.
3. Select **Security** tab.
4. Click on "**Reset all zones to default level**" button and then click on "**OK**" button as shown In bellow Image.

[](http://4.bp.blogspot.com/-zeRFD2t_hXo/VX0Mm2U5CPI/AAAAAAAABqs/3HFOm4b7Q50/s1600/Reset%2Ball%2Bzones%2Bto%2Bdefault%2Blevel.png)

Now "Enable Protected Mode" check box status will be different for all zones. Now execute bellow given test In eclipse and verify result In console.

package Testing\_Pack;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

import org.openqa.selenium.remote.DesiredCapabilities;

public class IEEnableProtected {

public static void main(String[] args) throws Exception {

**// Set path of IEDriverServer.exe**

**// Note : IEDriverServer.exe should be In D: drive.**

System.setProperty("webdriver.ie.driver", "D://IEDriverServer.exe");

**// Set capability of IE driver to Ignore all zones browser protected mode settings.**

DesiredCapabilities caps = DesiredCapabilities.internetExplorer();

caps.setCapability(InternetExplorerDriver.INTRODUCE\_FLAKINESS\_BY\_IGNORING\_SECURITY\_DOMAINS,true);

**// Initialize InternetExplorerDriver Instance using new capability.**

WebDriver driver = new InternetExplorerDriver(caps);

driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);

**// Load sample calc test URL**

driver.get("http://only-testing-blog.blogspot.in/2014/04/calc.html");

**// Execute sample calc test.**

driver.findElement(By.xpath("//input[@id='1']")).click();

driver.findElement(By.xpath("//input[@id='plus']")).click();

driver.findElement(By.xpath("//input[@id='6']")).click();

driver.findElement(By.xpath("//input[@id='equals']")).click();

String result = driver.findElement(By.xpath("//input[@id='Resultbox']")).getAttribute("value");

System.out.println("Calc test result Is : " + result);

driver.close();

}

}

Above software test will run without any error regardless whatever settings are there for "Enable Protected Mode". Read next post to know how to set IE zoom level to 100% on runtime.

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-set-ie-browser.html)

**IEDriver - How To Resolve Set IE browser Zoom Level To 100% Error On RunTime**

If your IE browser zoom level Is not set to 100% then you will get an error like "Browser zoom level was set to XYZ%. It should be set to 100% ".  Earlier we learnt In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html) -> How to resolve this error by setting IE browser's zoom level to 100% manually before running software test automation script. Now If you need

to run your software test script In different computers then everywhere you need to set browser's zoom level every time manually before running your software web application's test script. Otherwise your software automation test script will fail.

As described In my previous post, We can resolve "Enable Protected Mode For All Zones" error run-time. So can we do same thing for browser zoom level? Answer Is Yes. We can apply some extra code In our software automation test script to set browser zoom level 100% run-time automatically.

Here we will use same class DesiredCapabilities of WebDriver Interface which we have used In previous post to resolve error. Here we will set browser's capability to Ignore zoom setting and disable the native events. Then we will set browser zoom level to 100%. Bellow given code snippet will perform above given all operations and set browser zoom level 100%.

**// Set desired capabilities to Ignore IEDriver zoom level settings and disable native events.**

DesiredCapabilities caps = DesiredCapabilities.internetExplorer();

caps.setCapability("EnableNativeEvents", false);

caps.setCapability("ignoreZoomSetting", true);

**// Initialize InternetExplorerDriver Instance using new capability.**

WebDriver driver = new InternetExplorerDriver(caps);

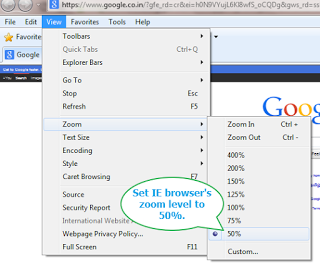
driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);

**// Press CTRL + 0 keys of keyboard to set IEDriver Instance zoom level to 100%.**

driver.findElement(By.tagName("html")).sendKeys(Keys.chord(Keys.CONTROL, "0"));

To experiment above code snippet practically,

* Set your IE browser's zoom level to 50% as shown In bellow Image

[](http://3.bp.blogspot.com/-lN2pGSPq844/VX1EXOLK4SI/AAAAAAAABq8/IfcqYB4rDPI/s1600/set%2BIE%2Bbrowser%2Bzoom%2Blevel%2B50%2B%2525%2Bfor%2Bselenium%2Bwebdriver.png)

* And then run bellow given test script In eclipse.

package Testing\_Pack;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.Keys;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

import org.openqa.selenium.remote.DesiredCapabilities;

public class IESetZoom {

public static void main(String[] args) throws Exception {

**// Set path of IEDriverServer.exe**

**// Note : IEDriverServer.exe should be In D: drive.**

System.setProperty("webdriver.ie.driver", "D://IEDriverServer.exe");

**// Set desired capabilities to Ignore IEDriver zoom level settings and disable native events.**

DesiredCapabilities caps = DesiredCapabilities.internetExplorer();

caps.setCapability("EnableNativeEvents", false);

caps.setCapability("ignoreZoomSetting", true);

**// Initialize InternetExplorerDriver Instance using new capability.**

WebDriver driver = new InternetExplorerDriver(caps);

driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);

**// Press CTRL + 0 keys of keyboard to set IEDriver Instance zoom level to 100%.**

driver.findElement(By.tagName("html")).sendKeys(Keys.chord(Keys.CONTROL, "0"));

**// Load sample calc test URL**

driver.get("http://only-testing-blog.blogspot.in/2014/04/calc.html");

**// Execute sample calc test.**

driver.findElement(By.xpath("//input[@id='1']")).click();

driver.findElement(By.xpath("//input[@id='plus']")).click();

driver.findElement(By.xpath("//input[@id='6']")).click();

driver.findElement(By.xpath("//input[@id='equals']")).click();

String result = driver.findElement(By.xpath("//input[@id='Resultbox']")).getAttribute("value");

Thread.sleep(5000);

System.out.println("Calc test result Is : " + result);

driver.close();

}

}

Above test will be executed without any error. This Is the best way to resolve Set IE browser Zoom Level To 100% error. It will apply zoom level settings automatically on each machine where this software test script will be executed. So you no need to worry about It. Next post will show you how to handle SSL certificate error in IE browser.

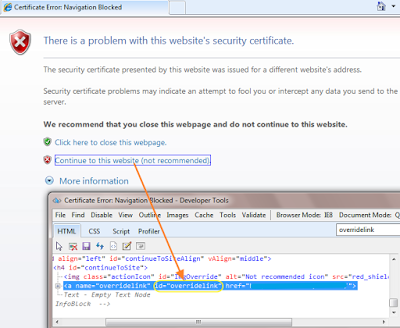
[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-enable.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/08/how-to-handle-ssl-certificate-error-in.html)

# How To Handle SSL Certificate Error In IE Browser For Selenium WebDriver Test

Earlier we learnt how to **handle SSL certificate error** by creating custom profile In **selenium WebDriver software test** when you run It In Firefox browser as described In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/03/how-to-handle-ssl-certificate-error-in.html). **IE browser** do not have any such feature to create and run software test In custom profile. So you need to do something different for **IE browser** to resolve certificate related error when you run WebDriver software automation test.

As you know, we can resolve error "Enable protected mode for all zones" as described In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-enable.html) and Set IE browser Zoom Level To 100% Error as described In previous post. Now let's see how to resolved SSL certificate error In IE browser for selenium WebDriver software automation test.

When you see SSL certificate error In IE browser, Your screen will looks like bellow.

[](http://4.bp.blogspot.com/-cfodLeWokAg/VX-BwO_azTI/AAAAAAAABsM/JrAFPf4BCeI/s1600/ssl%2Bcertificate%2Berror%2BIn%2BIE%2Bbrowser.png)

There Is one option link with text "Continue to this website (not recommended)." If somehow we can click on this link then original software website page will be loaded and our test script can go further for execution. If you view link In HTML mode using F12 then you will realize that link has ID called "overridelink". We can click on that link using driver.navigate() method with Javascript as bellow.

**//To click on "Continue to this website (not recommended)." link to load original website.**

driver.navigate().to("javascript:document.getElementById('overridelink').click()");

This solution Is worked for me In many website's tests. Full example demonstration Is as bellow.

package Testing\_Pack;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

public class SSLErrorInIE {

public static void main(String[] args) {

**// Set path of IEDriverServer.exe**

**// Note : IEDriverServer.exe should be In D: drive.**

System.setProperty("webdriver.ie.driver", "D://IEDriverServer.exe");

WebDriver driver = new InternetExplorerDriver();

driver.manage().timeouts().implicitlyWait(15, TimeUnit.SECONDS);

driver.get("**URL of SSL error site**");

**//To click on "Continue to this website (not recommended)." link to load original website.**

driver.navigate().to("javascript:document.getElementById('overridelink').click()");

}

}

This way we can resolve SSL certificate error In IE browser. View [**THIS ARTICLE**](http://www.software-testing-tutorials-automation.com/2015/08/handle-ssl-certificate-error-in-google.html) to know how to handle SSL cretificate error in google chrome browser.

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-set-ie-browser.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/08/fire-ie-selenium-tool-to-find-xpath-css.html)

**Fire-IE-Selenium Tool To Find XPath, CSS Path Or Any Other Locator For IE Only WebSites**

There are thousands of software websites which are allowed to **access only via Internet Explorer browser**. Such **IE only software websites** are not allowed to access through any other browsers like Firefox or google chrome so you can access such sites In Internet Explorer browser only. Supposing you need to **automate such IE only software website using selenium WebDriver**. Is there any challenge?

Earlier In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html), we learnt how to run selenium webdriver software test In IE browser. Biggest challenge to automate such IE Only websites Is to **find XPath** or any other element locator of software web page elements as IE browser do not have any add-on or extension which can provide us **XPath or CSSPath** directly. Generally we are using Firebug-FirePath Firefox add-on combination (As described In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/07/steps-to-get-element-xpathcss-using.html)) or Element locator for WebDriver Firefox add-on (As described In [**THIS POST**](http://www.software-testing-tutorials-automation.com/2015/03/install-webdriver-element-locator-add.html)) to get XPath of element. But these Firefox addons are useless as website Is accessible only In IE browser.

**Download Fire-IEBrowser1.4.xlsm**

There Is one tool called "[**fire-ie-selenium**](https://code.google.com/p/fire-ie-selenium/)" which can help us to get element locators for IE only websites. It Is Fire-IEBrowser1.4.xlsm file using which we can get XPath or CSSselector of any element. To get this tool,

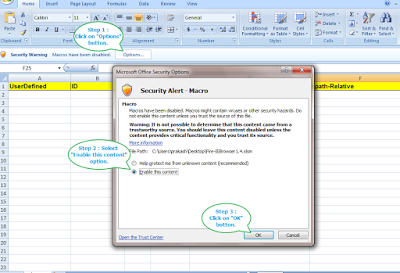
1. Go to [**THIS PAGE**](http://www.software-testing-tutorials-automation.com/2014/07/selenium-webdriver-frameworks-required.html)
2. Download **Fire-IEBrowser1.4.zip** file.
3. Extract zip folder. You will get Fire-IEBrowser1.4.xlsm file from unzipped folder.

**Using Fire-IEBrowser1.4.xlsm To Get XPath Or CssSelector**

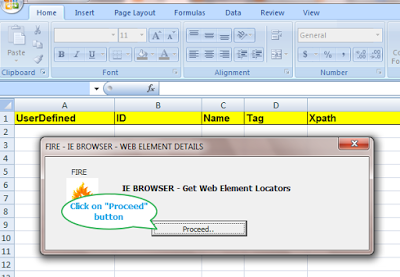
Let's take simple example of Google website to get XPath and CssSelector path of google search box and google search button.

**Note** : Tool Is being crashed with some software websites and there Is not any online support for this tool. So use It If works with your website.

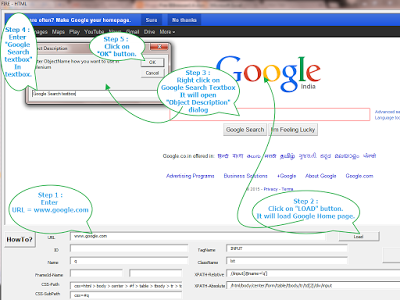
* Open Fire-IEBrowser1.4.xlsm file.
* Click on **Options** button. It will open **Microsoft Office Security Options** dialog as shown In bellow Image.
* Select **Enable this content** radio option from dialog and click on **OK** button.

[](http://4.bp.blogspot.com/-KwhH8vhFXP8/VX1qMiQ4oQI/AAAAAAAABrM/NV9gLfXbH0c/s1600/open%2BFire-IEBrowser%2Btool.png)

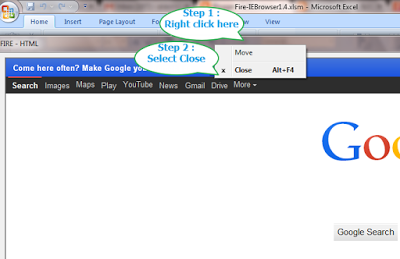
* It will open FIRE IE BROWSER - WEB ELEMENT DETAILS dialog with **Proceed** button as shown In bellow Image. Click on **Proceed**button.

[](http://3.bp.blogspot.com/-31FaPNGgfx0/VX1reXRyJqI/AAAAAAAABrU/bgnO7yIWDNE/s1600/open%2BFire-IEBrowser%2Btool%2Bproceed.png)

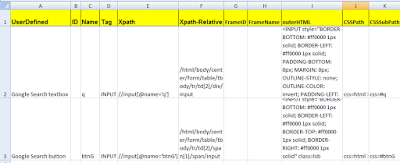
* It will open FIRE - HTML window as shown In bellow Image.
* There will be **URL** text field with **LOAD** button. Enter www.google.com In URL text box and click on **LOAD** button as shown In bellow Image. It will open Google WebSite Home page.

[](http://1.bp.blogspot.com/-rb8U4T5uBnU/VX1x05aQQAI/AAAAAAAABrk/r0yVmXcjJ-A/s1600/get%2Bxpath%2Busing%2BFire-IEBrowser.png)

* Right click on **Google Search** textbox as shown In above Image. It will Open **Object Description** dialog. Enter "Google Search textbox" In text box and click on **OK** button as shown In above Image. It will save XPath, CSS Path and other details of Google Search text box In Fire-IEBrowser1.4.xlsm file's sheet.
* Perform above step for **Google Search** button and save It with name "Google Search button".
* Close FIRE - HTML window by right clicking at top of FIRE - HTML window and selecting Close as shown bellow Image. It will close FIRE - HTML window.

[](http://4.bp.blogspot.com/-4g_g1y7QF_k/VX11k6KtdTI/AAAAAAAABrw/pZjiWVouW0g/s1600/Close%2Bfire%2Bhtml%2Bwindow.png)

Now your Fire-IEBrowser1.4.xlsm file's sheet will looks like bellow.

[](http://4.bp.blogspot.com/-e75U4jlFqu4/VX12UrRMoKI/AAAAAAAABr4/sCXQ3-feBtw/s1600/xpath%2Band%2Bcss%2Bpath%2Bfinder%2Bfor%2BIE%2Bonly%2Bwebsite.png)

You can see In above Image, Both element's Name, Tag Name, XPath, X-Path-Relative, OuterHTML, CSSPath and CSSSubPath are stored In sheet. Same way you can get all these detail of any element and use In Selenium WebDriver software test script for IE only or any WebSite.

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/08/how-to-handle-ssl-certificate-error-in.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/05/selenium-webdriver-advanced-tutorials.html)

**Selenium WebDriver Advanced Tutorials Part 7**

This section of webdriver tutorial Includes MySQL Database Testing With Selenium WebDriver software testing tool tutorials, IEDriver tutorials for software testing, ChromeDriver tutorials and few add-on related tutorials for software testing process.

Learn all bellow given tutorials step by step. It will help you to implement in your real project.  
  
**MySQL Database Testing With Selenium WebDriver**

1. [**Download And Install MySQL For Database Testing Using Selenium**](http://www.software-testing-tutorials-automation.com/2015/05/download-and-install-mysql-for-database.html)
2. [**MySQL - Creating Data Tables To Store Data**](http://www.software-testing-tutorials-automation.com/2015/06/mysql-creating-data-tables-to-use-in.html)
3. [**Fetching Data From MySQL Database Table For WebDriver Test**](http://www.software-testing-tutorials-automation.com/2015/06/fetching-data-from-mysql-database-table.html)
4. [**Fetching Data From MySQL Database Table With Where Clause**](http://www.software-testing-tutorials-automation.com/2015/06/using-where-clause-to-get-data-from.html)
5. [**Update Record In Database For Selenium WebDriver Test**](http://www.software-testing-tutorials-automation.com/2015/06/update-record-in-database-for-selenium.html)

**IEDriver**

1. [**Running WebDriver Test In Internet Explorer Browser**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-run-selenium-webdriver-test.html)
2. [**Resolve "Enable Protected Mode For All Zones" Error RunTime**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-enable.html)
3. [**Resolve Set IE browser Zoom Level To 100% Error On RunTime**](http://www.software-testing-tutorials-automation.com/2015/07/iedriver-how-to-resolve-set-ie-browser.html)
4. [**How To Handle SSL Certificate Error In IE Browser For WebDriver**](http://www.software-testing-tutorials-automation.com/2015/08/how-to-handle-ssl-certificate-error-in.html)
5. [**Use Fire-IE-Selenium Tool To Locate Element For IE Only WebSites**](http://www.software-testing-tutorials-automation.com/2015/08/fire-ie-selenium-tool-to-find-xpath-css.html)

**ChromeDriver**

1. [**Handle SSL Certificate Error In Google Chrome For WebDriver Test**](http://www.software-testing-tutorials-automation.com/2015/08/handle-ssl-certificate-error-in-google.html)

**Other Tutorials**

1. [**Install Firebug And Firepath Add-Ons In Firefox To Get XPath**](http://www.software-testing-tutorials-automation.com/2015/07/steps-to-install-firebug-and-firepath.html)
2. [**Attach Firebug and FirePath Add-On To Firefox Driver**](http://www.software-testing-tutorials-automation.com/2015/07/how-to-attach-firebug-and-firepath-add.html)
3. [**Get Element XPath/CSS Using Firebug And FirePath Add-On**](http://www.software-testing-tutorials-automation.com/2015/07/steps-to-get-element-xpathcss-using.html)
4. [**Finding Broken Links/Images From Page Using WebDriver**](http://www.software-testing-tutorials-automation.com/2015/08/how-to-find-broken-linksimages-from.html)
5. [**Selenium : Loading Google Chrome Driver With Extensions**](http://www.software-testing-tutorials-automation.com/2016/05/selenium-loading-google-chrome-driver.html)
6. [**How To Get Browser And OS Details On Run-Time In Selenium WebDriver**](http://www.software-testing-tutorials-automation.com/2016/05/how-to-get-browser-and-os-details-on.html)
7. [**How To Capture And Verify Tool tip Text In Selenium WebDriver Example**](http://www.software-testing-tutorials-automation.com/2016/05/how-to-capture-and-verify-tool-tip-text.html)
8. [**Selenium : How To Avoid Page Loading on Test Execution**](http://www.software-testing-tutorials-automation.com/2016/05/selenium-how-to-avoid-page-loading-on.html)
9. [**Selenium : How To Capture JS Errors In Google Chrome**](http://www.software-testing-tutorials-automation.com/2016/05/selenium-how-to-capture-js-errors-in.html)

[**<< PREVIOUS**](http://www.software-testing-tutorials-automation.com/2015/03/selenium-webdriver-advanced-tutorials.html)**||**[**NEXT >>**](http://www.software-testing-tutorials-automation.com/2015/07/oop-concepts-and-advanced-java.html)