**Handle Multiple Browsers:**

On an HTML page a link can open up in a new window. This is quite common and can be found on our Test Practice Page here. Click on the very first button and it will launch another window and load http://toolsqa.com in the new window. In this chapter we will see How to Handle Multiple Windows using Selenium WebDriver. For this tutorial we will use our test page

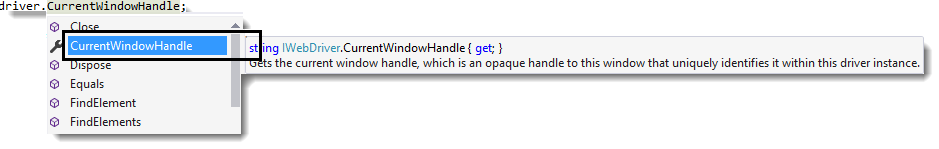
<http://toolsqa.com/automation-practice-switch-windows/>

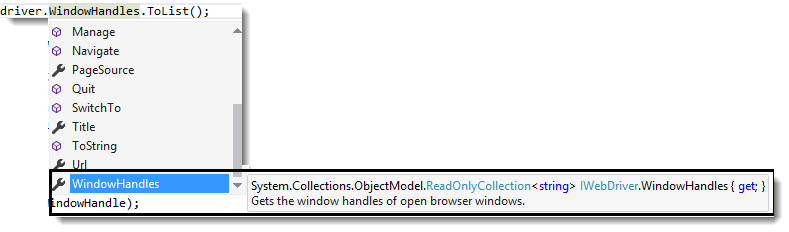
**Basics of Multiple Window:**

There is only one way you can get multiple windows via Selenium web driver, that is by clicking on a link which opens the page in a new browser window. Selenium web driver keeps a track of how many windows it opened during a session. Which means that it will not keep track of any browser window which is – Opened manually – Opened by a previous session of Selenium Webdriver By session of selenium WebDriver we mean the duration from the time we instantiate a WebDriver instance to the time we kill it via WebDriver.Quit or by manually killing the process. With this understanding lets first open some windows using Selenium WebDriver. To achieve this all we have to do is to click on the button on our practice page.

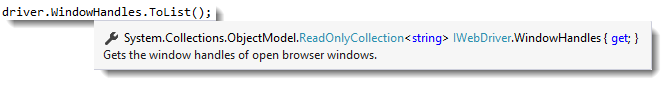
**How to Handle Window in Selenium CSharp?**

To uniquely identify an opened browser Selenium WebDriver keeps a map of Opened windows VS Window Handle. Window handle is a unique string value that uniquely identifies a Browser window on desktop. It is guaranteed that each browser will have a unique window handle. To get Window handle WebDriver interface provides two methods CurrentWindowHandle – WindowHandles





CurrentWindowHandle method return a string value and it returns the Window handle of current focused browser window. WindowHandles. This method returns a ReadOnlyCollection (You need to typecast this collection to List Collection by using ToList()) of all Window handles of all the browsers that were opened in the session.



**How to Handle Multiple Windows in Selenium CSharp?**

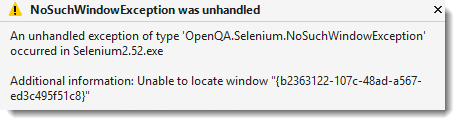
There is a concept of current focused window which means that all selenium webdriver commands will go to the focused window. By default the focus is always on the Parent window, please see the screenshot above. In order to shift focus from Parent Window to any child window we have to use the following command on WebDriver – driver.SwitchTo().Window(String WindowHandle); This command takes in a window handle and switches the driver context on that window. Once the Switch happens all the driver commands will go to the newly focused window. This is very important to understand, without switching to the desired window we will not be able to perform any action on that window. Now let’s see some code which iteratively moves across all the open windows and navigates to a particular page in all the open windows one by one.

**Closing all the Windows:**

There are basically two commands that we can use to close the opened browser windows WebDriver.close() and WebDriver.Quit().

WebDriver.Close() command will close the current window on which the focus is present. This can be used to close windows selectively. Just switch to the window that you want to close by using the correct Window handle and the call the WebDriver.Close command. That will close the current browser window.

Note: *After closing a window you have to explicity switch to another valid window before sending in any WebDriver commands. if you fail to do this you wil get following exception. org.openqa.selenium.NoSuchWindowException: Window not found. The browser window may have been closed.*



WebDriver.Quit() will close all the windows opened in the session. This command basically shuts down the driver instance and any further commands to WebDriver results in exception. That’s pretty much it that we have on handling multiple windows. Do let your comments flow in.