

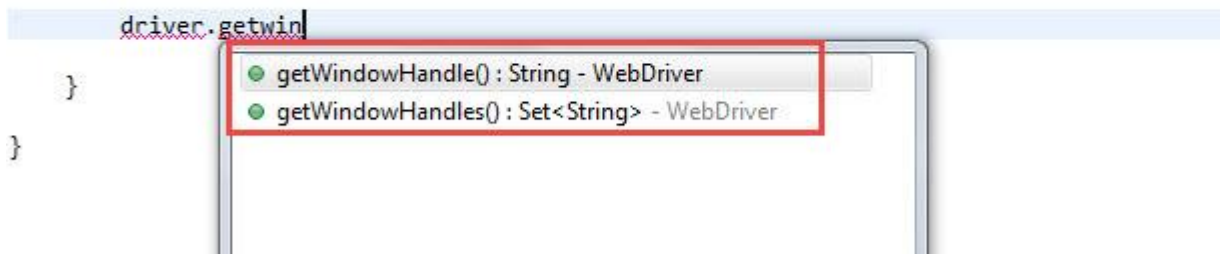
Handling multiple browsers in Selenium WebDriver

Handle Windows in Selenium Webdriver

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

- *getWindowHandle()*
- *getWindowHandles()*

```
for(int i = 0; i < 3; i++)  
{  
    clickElement.click();  
    Thread.sleep(3000);  
}
```



getWindowHandle() method return a string value and it returns the Window handle of current focused browser window. *getWindowHandles()* method returns a set of all Window handles of all the browsers that were opened in the session. In this case it will return 4 windows handles because we have 4 windows open. Here is the code to print out window handles on console of eclipse.

Solution:

```
public static void main(String[] args) throws InterruptedException  
{  
  
    WebDriver driver = new FirefoxDriver();  
    driver.get("http://toolsqa.wpengine.com/automation-practice-switch-windows/");  
    String parentWindowHandle = driver.getWindowHandle();  
    System.out.println("Parent window's handle -> " + parentWindowHandle);  
    WebElement clickElement = driver.findElement(By.id("button1"));
```

```

for(int i = 0; i < 3; i++)
{
    clickElement.click();
    Thread.sleep(3000);
}

```

```

Set<String> allWindowHandles = driver.getWindowHandles();

```

```

for(String handle : allWindowHandles)
{
    System.out.println("Window handle - > " + handle);
}
}
}

```

Handling Multiple Windows in Selenium WebDriver

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 – ***WebDriver.SwitchTo().window(String windowHandle)***; This command takes in a window handle and switches the driver context on that window.

```

public static void main(String[] args) throws InterruptedException {
    WebDriver driver = new FirefoxDriver();
    driver.get("http://toolsqa.wpengine.com/automation-practice-switch-
windows/");
    String parentWindowHandle = driver.getWindowHandle();
    System.out.println("Parent window's handle -> " + parentWindowHandle);
    WebElement clickElement = driver.findElement(By.id("button1"));
}
}

```

```

        for(int i = 0; i < 3; i++)
        {
            clickElement.click();
            Thread.sleep(3000);
        }
        Set<String> allWindowHandles = driver.getWindowHandles();
        for(String handle : allWindowHandles)
        {
            System.out.println("Switching to window - > " + handle);
            System.out.println("Navigating to google.com");
            driver.switchTo().window(handle); //Switch to the desired window
first and then execute commands using driver
            driver.get("http://google.com");
        }
    }
}

```

Closing all the Windows

There are basically two commands that we can use to close the opened browser windows. – ***WebDriver.close()*** – ***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 then call the `WebDriver.close` command. That will close the current browser window. **Note: After closing a window you have to explicitly switch to another valid window before sending in any WebDriver commands. if you fail to do this you will get following exception. org.openqa.selenium.NoSuchWindowException: Window not found. The browser window may have been closed.** In the code below we will close the parent window and then explicitly move focus to the last window in the list.

```

public static void main(String[] args) throws InterruptedException {
    WebDriver driver = new FirefoxDriver();

    driver.get("http://toolsqa.wpengine.com/automation-practice-switch-
windows/");

    String parentWindowHandle = driver.getWindowHandle();
}

```

```
System.out.println("Parent window's handle -> " + parentWindowHandle);  
WebElement clickElement = driver.findElement(By.id("button1"));
```

```
for(int i = 0; i < 3; i++)  
{  
    clickElement.click();  
    Thread.sleep(3000);  
}
```

```
Set<String> allWindowHandles = driver.getWindowHandles();  
String lastWindowHandle = "";  
for(String handle : allWindowHandles)  
{
```

```
    System.out.println("Switching to window - > " + handle);  
    System.out.println("Navigating to google.com");  
    driver.switchTo().window(handle); //Switch to the desired window  
first and then execute commands using driver  
    driver.get("http://google.com");  
    lastWindowHandle = handle;  
}
```

```
//Switch to the parent window  
driver.switchTo().window(parentWindowHandle);  
//close the parent window  
driver.close();  
//at this point there is no focused window, we have to explicitly switch  
back to some window.
```

```
driver.switchTo().window(lastWindowHandle);  
driver.get("http://toolsqa.wpengine.com");}
```

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.