



# SELENIUM WEBDRIVER BROWSER COMMANDS

*Lecture Notes*

A COMPLETE SELENIUM GUIDE

**Softech Solutions Inc.**

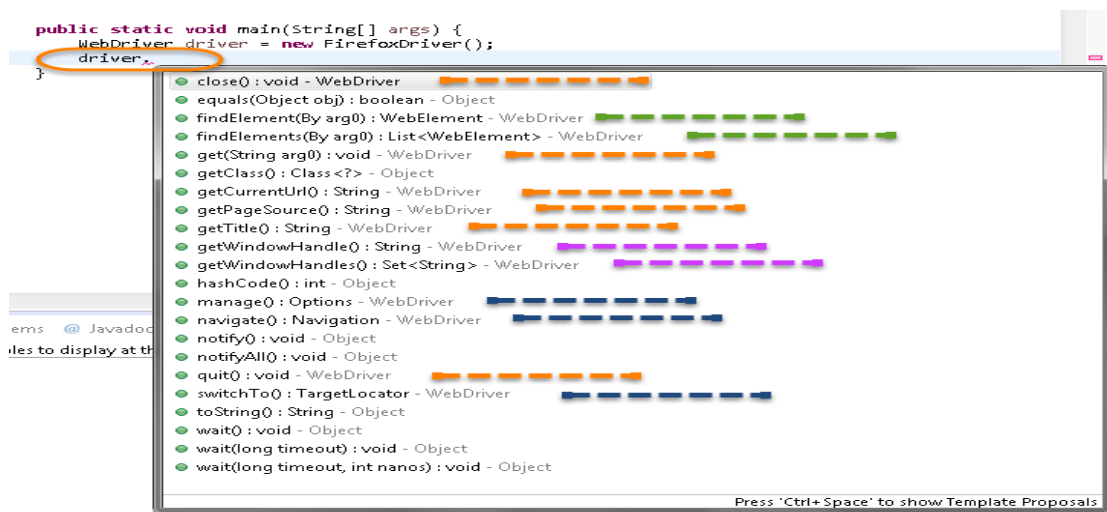
[www.softechsolutionsgroup.com](http://www.softechsolutionsgroup.com)

[saney.alam@softechsolutionsgroup.com](mailto:saney.alam@softechsolutionsgroup.com)

# Selenium WebDriver Browser Commands

The very first question which comes to my mind and has been asked in many interviews is **what is Selenium WebDriver?** Is it an Automation Tool? Is it a Class? Is it an Interface or what actually it is? To answer this question we need to understand the Advance Java OOPs concepts first and then we would be able to visualize the WebDriver Implementation. For the sake of simplicity, we will avoid this WebDriver Implementation topic for now and will cover this in later chapters. As of now we start with all the methods we get from WebDriver.

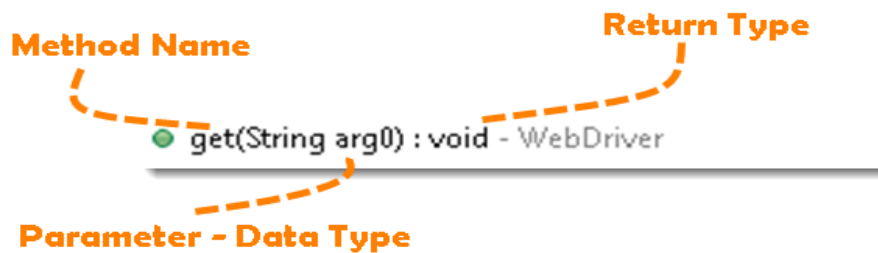
Now the next question is, How to access the methods of WebDriver? To check what all we have in WebDriver, create a driver object from WebDriver and press dot key. This will list down all the methods of WebDriver.



**Note:** Methods followed by Object keyword are the generic methods gets from Object Class in Java. You will find these methods for every object of java language.

- The suggestions marked in Blue Color are Nested Classes under WebDriver and will be covered in detail separately in the following chapters.
- The suggestions marked in Green Color are also Interfaces like WebDriver and will be covered in detail separately in the following chapters.
- The suggestions marked in Violet Color are similar methods like Orange but will be covered in detail separately in the following chapters.

Let's just start discussing the **Orange colored** methods of **Selenium WebDriver** but before that try to understand the syntax of the suggestions display by Eclipse for WebDriver.



**Method:** A Java method is a collection of statements that are grouped together to perform an operation.

- **Method Name:** *To access any method of any class, we need to create an object of class and then all the public methods will appear for the object.*
- **Parameter:** *It is an argument which is passed to a method as a parameter to perform some operation. Every argument must pass with the same data type. For e.g. **get(String arg0) : void**. This is asking for a String type argument.*
- **Return Type:** *Method can return a value or returning nothing (void). If the void is mentioned after the method, it means the method is returning no value. And if it is returning any value, then it must display the type of the value for e.g. **getTitle() : String**.*

Now it would be very easy to understand the WebDriver commands in the below chapter. The very first thing you like to do with Selenium is to Opening a new browser, Perform few tasks and Closing the browser. Below are the numbers of commands you can apply on the Selenium opened browser.

### Get Command

**get(String arg0) : void** – This method **Load** a new web page in the current browser window. Accepts String as a parameter and returns nothing.

**Command – driver.get(appUrl);**

Where *appUrl* is the website address to load. It is best to use a fully qualified URL.

```
driver.get("http://www.google.com");  
//Or can be written as  
String URL = "http://www.google.com";  
driver.get(URL);
```

### Get Title Command

**getTitle()** : String – This method fetches the Title of the current page. Accepts nothing as a parameter and returns a String value.

Command – **driver.getTitle();**

As the return type is String value, the output must be stored in String object/variable.

```
driver.getTitle();  
//Or can be used as  
String Title = driver.getTitle();
```

### Get Current URL Command

**getCurrentUrl()** : String – This method fetches the string representing the Current URL which is opened in the browser. Accepts nothing as a parameter and returns a String value.

Command – **driver.getCurrentUrl();**

As the return type is String value, the output must be stored in String object/variable.

```
driver.getCurrentUrl();  
//Or can be used as  
String CurrentUrl= driver.getCurrentUrl();
```

### Get Page Source Command

**getPageSource()** : String – This method returns the Source Code of the page. Accepts nothing as a parameter and returns a String value.

Command – **driver.getPageSource();**

As the return type is String value, the output must be stored in String object/variable.

```
driver.getPageSource();  
//Or can be used as  
String PageSource= driver.getPageSource();
```

### Close Command

**close()** : void – This method *Close* only the current window the **WebDriver** is currently controlling. Accepts nothing as a parameter and returns nothing.

Command – **driver.close();**

Quit the browser if it's the last window currently open.

```
driver.close();
```

### Quit Command

**quit() : void** – This method *Closes* all windows opened by the **WebDriver**. Accepts nothing as a parameter and returns nothing.

**Command – driver.quit();**  
Close every associated window.

```
driver.quit();
```