WebDriver Interface

This is the next tutorial in the selenium-java series. In the last tutorial, we had studied about how to work with different browsers. In this tutorial we are going to take a look at webdriver interface!

What you will Learn:
Understand WebDriver Interface
Practice few functions
Interface reference pointing to a class

Understand WebDriver Interface:
Go to https://selenium.dev/selenium/docs/api/java/index.html. On the left hand side menu, whatever you see in *italic* is an interface. Notice that *WebDriver* too is in italic & hence is an interface. Click *WebDriver* link, the below page should open

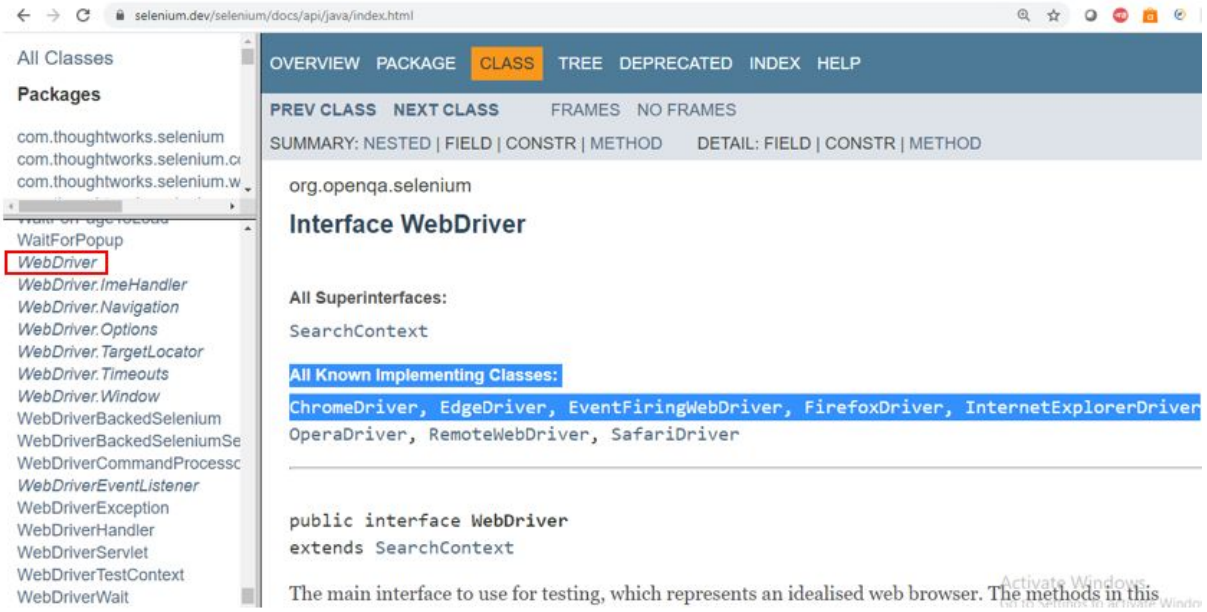
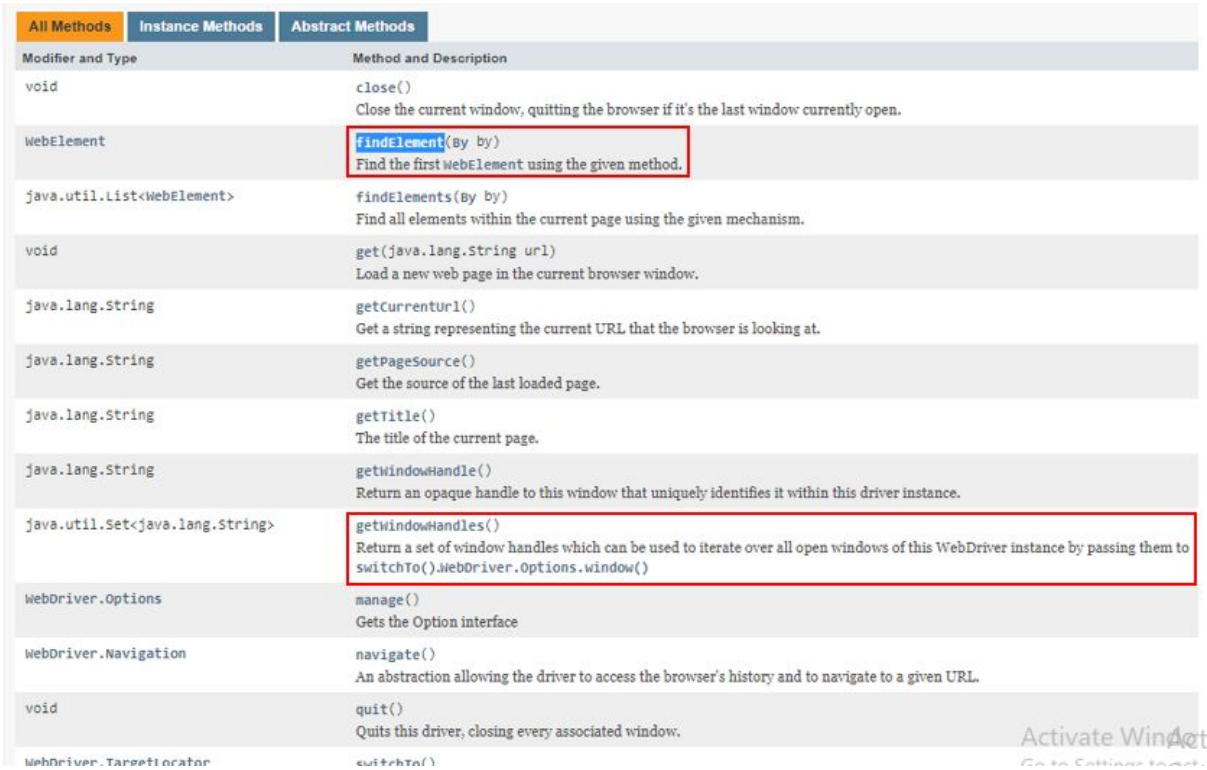


Figure 1

If you scroll down, you would notice 'Method Summary' section. This section shows lot of methods present in webdriver interface, like findElement(), get(), getWindowHandle() etc alongwith their description

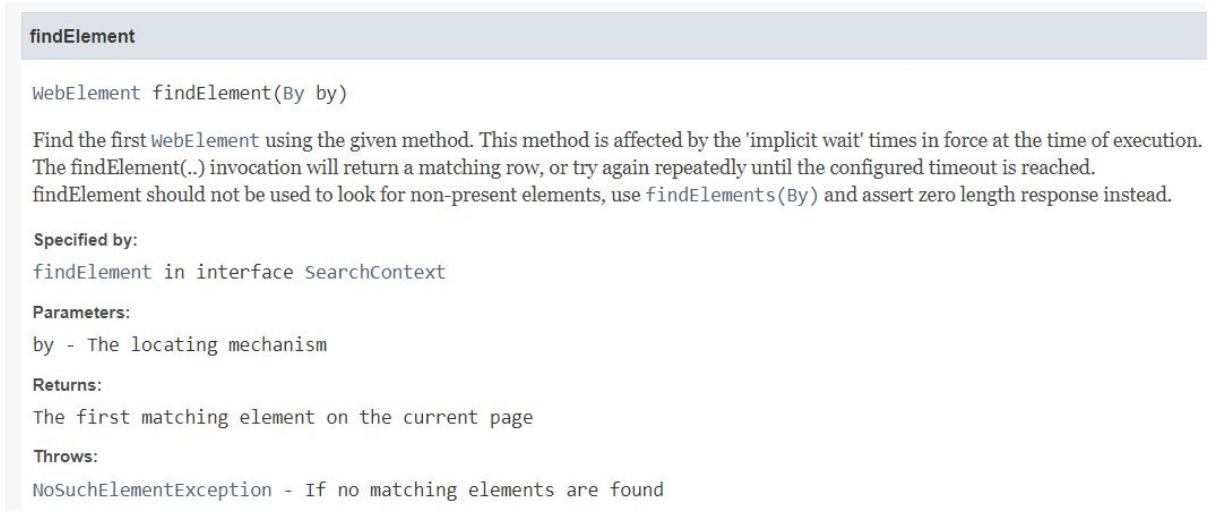


All Methods | Instance Methods | Abstract Methods

Modifier and Type	Method and Description
void	close() Close the current window, quitting the browser if it's the last window currently open.
WebElement	findElement(By by) Find the first WebElement using the given method.
java.util.List<WebElement>	findElements(By by) Find all elements within the current page using the given mechanism.
void	get(java.lang.String url) Load a new web page in the current browser window.
java.lang.String	getCurrentUrl() Get a string representing the current URL that the browser is looking at.
java.lang.String	getPageSource() Get the source of the last loaded page.
java.lang.String	getTitle() The title of the current page.
java.lang.String	getWindowHandle() Return an opaque handle to this window that uniquely identifies it within this driver instance.
java.util.Set<java.lang.String>	getWindowHandles() Return a set of window handles which can be used to iterate over all open windows of this WebDriver instance by passing them to switchTo().WebDriver.Options.window()
WebDriver.Options	manage() Gets the Option interface
WebDriver.Navigation	navigate() An abstraction allowing the driver to access the browser's history and to navigate to a given URL.
void	quit() Quits this driver, closing every associated window.
WebDriver.TargetLocator	switchTo()

Figure 2

Click findElement() method link (or any method link), the method documentation gets displayed



findElement

```
WebElement findElement(By by)
```

Find the first `WebElement` using the given method. This method is affected by the 'implicit wait' times in force at the time of execution. The findElement(..) invocation will return a matching row, or try again repeatedly until the configured timeout is reached.
findElement should not be used to look for non-present elements, use `findElements(By)` and assert zero length response instead.

Specified by:

`findElement in interface SearchContext`

Parameters:

`by - The locating mechanism`

Returns:

`The first matching element on the current page`

Throws:

`NoSuchElementException - If no matching elements are found`

Figure 3

As you can notice in figure 1, all the driver classes are implementing the same **WebDriver Interface.** So all these classes will also have these functions 'findElement', getTitle etc. The function names in all the implementing classes will be the same. So if you see over below, the 'get' function is present in all the classes. So the commands to interact with the browser will be the same, whether we have chrome, ie, ff, etc..

Practice few functions:

```
5  public class Browsers {
6
7      public static void main(String[] args) {
8
9          System.setProperty("webdriver.ie.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\IEDriverServer.exe");
10         InternetExplorerDriver ie = new InternetExplorerDriver();
11         ie.get("https://gmail.com");
12
13         System.setProperty("webdriver.gecko.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\geckodriver.exe");
14         FirefoxDriver fd = new FirefoxDriver();
15         fd.get("https://facebook.com");
16
17         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\chromedriver.exe");
18         ChromeDriver cd = new ChromeDriver();
19         cd.get("https://google.com");
```

Figure 4

Similarly, findelement() function is present in all the browser classes

```
5  public class Browsers {
6
7      public static void main(String[] args) {
8
9          System.setProperty("webdriver.ie.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\IEDriverServer.exe");
10         InternetExplorerDriver ie = new InternetExplorerDriver();
11         ie.get("https://gmail.com");
12         ie.findElement(by);
13
14         System.setProperty("webdriver.gecko.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\geckodriver.exe");
15         FirefoxDriver fd = new FirefoxDriver();
16         fd.get("https://facebook.com");
17         fd.findElement(by);
18
19         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\chromedriver.exe");
20         ChromeDriver cd = new ChromeDriver();
21         cd.get("https://google.com");
22         cd.findElement(by);
```

Figure 5

These driver classes corresponds to a browser. ChromeDriver will help you in executing the selenium scripts in chrome browser and similarly others. We simply create the objects of these driver classes and can start working with them. So technically WebDriver is an interface. Let us create a temp class.

Browsers.java | Temp.java

```
1
2  public class Temp {
3
4      public static void main(String[] args)
5
6      }
7
```

Figure 6

So far we have been creating objects like this: **ChromeDriver** cd = new ChromeDriver()

Browsers.java | Temp.java

```
1  import org.openqa.selenium.chrome.ChromeDriver;
2
3  public class Temp {
4
5      public static void main(String[] args) {
6          ChromeDriver cd = new ChromeDriver();
7      }
```

Figure 7

VERY IMPORTANT: **You can have the interface reference pointing to the class that implements that interface.** So we can re-write the same statement as:
WebDriver cd = new ChromeDriver()
The reason being, WebDriver is an interface & ChromeDriver is an implementing class. So the interface reference 'cd' will have access to all the functions of ChromeDriver class. Make sure to import org.openqa.selenium.WebDriver

```
1  import org.openqa.selenium.WebDriver;
2  import org.openqa.selenium.chrome.ChromeDriver;
3
4  public class Temp {
5
6      public static void main(String[] args) {
7          WebDriver cd = new ChromeDriver();
8      }
```

Figure 8

Similarly we can write lines#10 and 11

```
1  import org.openqa.selenium.WebDriver;
2  import org.openqa.selenium.chrome.ChromeDriver;
3  import org.openqa.selenium.firefox.FirefoxDriver;
4  import org.openqa.selenium.ie.InternetExplorerDriver;
5
6  public class Temp {
7
8      public static void main(String[] args) {
9          WebDriver cd = new ChromeDriver();
10         WebDriver fd = new FirefoxDriver();
11         WebDriver ie = new InternetExplorerDriver();
12
```

Figure 9

When we say **WebDriver driver = null**, it means that the driver reference is not pointing anywhere

Figure 10

Line#13 represents that in the below figure. Now, based on our choice of the browser, we can point the same driver reference to any class (lines#15, 16, 17)

```
1  import org.openqa.selenium.WebDriver;
2  import org.openqa.selenium.chrome.ChromeDriver;
3  import org.openqa.selenium.firefox.FirefoxDriver;
4  import org.openqa.selenium.ie.InternetExplorerDriver;
5
6  public class Temp {
7
8      public static void main(String[] args) {
9          WebDriver cd = new ChromeDriver();
10         WebDriver fd = new FirefoxDriver();
11         WebDriver ie = new InternetExplorerDriver();
12
13         WebDriver driver = null;
14
15         driver = new ChromeDriver();
16         driver = new FirefoxDriver();
17         driver = new InternetExplorerDriver();
18     }
```

Figure 11

Comment lines#9-11

```
1  import org.openqa.selenium.WebDriver;
5
6  public class Temp {
7
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
```

Figure 12

Add lines#13-15

```
1  import org.openqa.selenium.WebDriver;
5
6  public class Temp {
7
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\chromedriver.exe");
14         System.setProperty("webdriver.gecko.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\geckodriver.exe");
15         System.setProperty("webdriver.ie.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\IEDriverServer.exe");
```

Figure 13

So our script should look like as seen below

```
1  import org.openqa.selenium.WebDriver;
5
6  public class Temp {
7
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\chromedriver.exe");
14         System.setProperty("webdriver.gecko.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\geckodriver.exe");
15         System.setProperty("webdriver.ie.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\IEDriverServer.exe");
16
17         WebDriver driver = null;
18
19         driver = new ChromeDriver();
20         driver = new FirefoxDriver();
21         driver = new InternetExplorerDriver();
```

Figure 14

Run, we should see all the 3 browsers launch

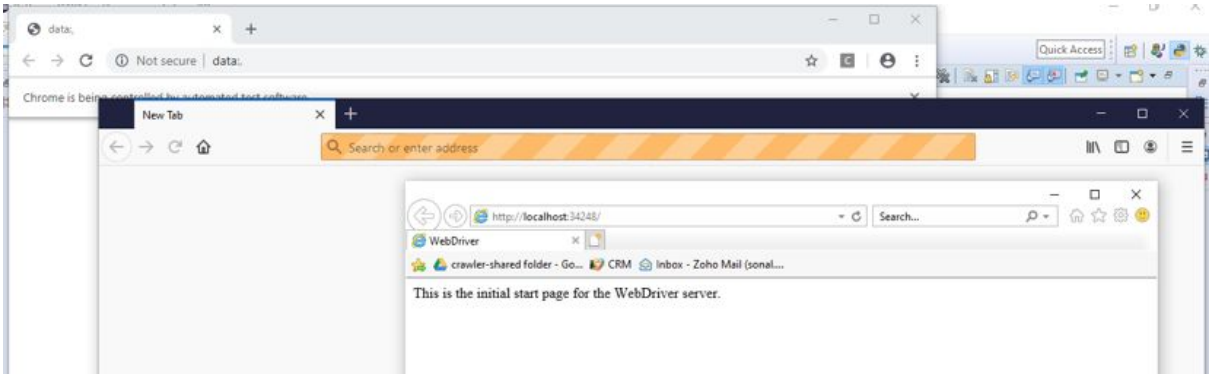


Figure 15

So using interface concept, we can use the same driver reference dynamically for any browser. We can use driver.close() to close all the browsers automatically (lines#20, 22, 24). Run the below script and check

```
6  public class Temp {
7
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\De
14         System.setProperty("webdriver.gecko.driver", "C:\\Users\\DELL\\Des
15         System.setProperty("webdriver.ie.driver", "C:\\Users\\DELL\\Deskto
16
17         WebDriver driver = null;
18
19         driver = new ChromeDriver();
20         driver.close();
21         driver = new FirefoxDriver();
22         driver.close();
23         driver = new InternetExplorerDriver();
24         driver.close();
```

Figure 16

We can now navigate to any website in all 3 browsers, see below.

```
6  public class Temp {
7
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C
14         System.setProperty("webdriver.gecko.driver", "C:
15         System.setProperty("webdriver.ie.driver", "C:\\U
16
17         WebDriver driver = null;
18
19         driver = new ChromeDriver();
20         driver.get("https://google.com");
21
22         driver = new FirefoxDriver();
23         driver.get("https://google.com");
24
25         driver = new InternetExplorerDriver();
26         driver.get("https://google.com");
```

Figure 17

Run the script, you will see the 3 browsers launch & navigate to google.com

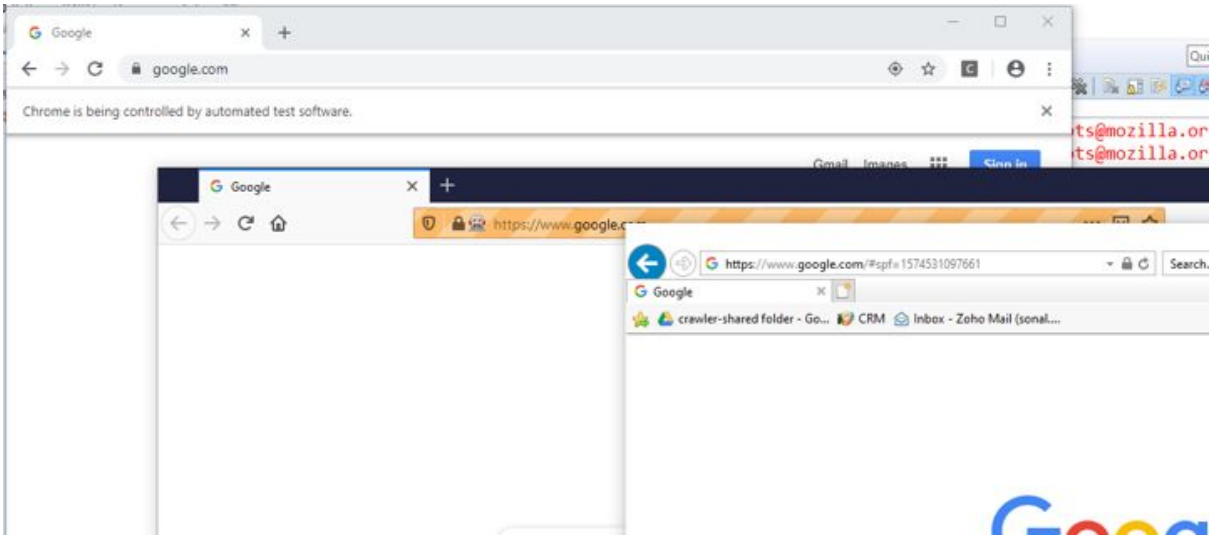


Figure 18

Comment FF & IE code. Let us fetch the title & current url of the website, lines 21-22

```
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C:
14         System.setProperty("webdriver.gecko.driver", "C:\
15         System.setProperty("webdriver.ie.driver", "C:\\Us
16
17         WebDriver driver = null;
18
19         driver = new ChromeDriver();
20         driver.get("https://google.com");
21         System.out.println(driver.getTitle());
22         System.out.println(driver.getCurrentUrl());
23
24         /*driver = new FirefoxDriver();
25         driver.get("https://google.com");
26
27         driver = new InternetExplorerDriver();
28         driver.get("https://google.com");*/
```

Figure 19

Run, see below o/p

```
Google
https://www.google.com/
```

Figure 20

Comment lines 21-22. We will now get the page source

```
19          driver = new ChromeDriver();
20          driver.get("https://google.com");
21          //System.out.println(driver.getTitle());
22          //System.out.println(driver.getCurrentUrl());
23          System.out.println(driver.getPageSource());
```

Figure 21

Run, see below o/p

```
<terminated> Temp [Java Application] C:\Program Files\Java\jre1.8.0_191\bin\java
var Ve,Ue,Ye,Ze,Xe;_.Te=function(a){a=Strin
Ue=function(a,b,c){if(null==b)c.push("null"
!isNaN(b)?String(b):"null");break;case "boo
_.$e=function(a){switch(a){case 200:case 20
_.af=function(){};_.af.prototype.w=null;var
var df;df=function(){};_.t(df,_.af);df.prot
_.cf=new df;
_.ff=function(a){if(a.Wa&amp;&amp;"function
 .hf=function(a,b){if(a.forEach&amp;&amp;"f
```

Figure 22

Comment pagesource function. From google.com, let us try to navigate to another website yahoo.com. Run the script and check the behaviour

```
8      public static void main(String[] args) {
9          //WebDriver cd = new ChromeDriver();
10         //WebDriver fd = new FirefoxDriver();
11         //WebDriver ie = new InternetExplorerDriver();
12
13         System.setProperty("webdriver.chrome.driver", "C
14         System.setProperty("webdriver.gecko.driver", "C:
15         System.setProperty("webdriver.ie.driver", "C:\\U
16
17         WebDriver driver = null;
18
19         driver = new ChromeDriver();
20         driver.get("https://google.com");
21         driver.navigate().to("https://yahoo.com");
22         //System.out.println(driver.getTitle());
23         //System.out.println(driver.getCurrentUrl());
24         //System.out.println(driver.getPageSource());
```

Figure 23

Let us now navigate back from yahoo to google. Run the script and check the behaviour

```
18
19         driver = new ChromeDriver();
20         driver.get("https://google.com");
21         driver.navigate().to("https://yahoo.com");
22         driver.navigate().back();
```

Figure 24

Let us now close the current browser (that is opened by selenium)

```
18
19         driver = new ChromeDriver();
20         driver.get("https://google.com");
21         driver.navigate().to("https://yahoo.com");
22         driver.navigate().back();
23         driver.close();
```

Figure 25

So this was all about the importance of WebDriver interface. Thank you for reading!