

Ques: WATS to verify Google webpage is displayed or not

Script:

-->To do verificationg using String class

1: equals(): This method is used to verify string but case sensitive and accepts complete string.

2: contains(): Case sensitive but accepts only few characters in sequence

3: equalsignorecase(): Not Case Sensitive but it accepts complete string

```
WebDriver driver = new FirefoxDriver();
driver.get("https://www.google.com/");
String str = driver.getTitle();
if(str.equals("Google"))
{
    System.out.println("Google Webpage displayed");
}
else
{
    System.out.println("Google Webpage not displayed");
}
driver.close();
```

-----  
AssignQues: WATS to verify facebook webpage displayed or not  
-----

\*\*\*WebDriver Methods\*\*\*

-->Its an interface which contains abstract methods which are used to perform action on browser.

Examp: Open,Close,Maximize,Minimize etc

-->It provides 11 abstract methods + 2 inherited methods

1: get(String url): It is used to open application using URL.

-->Always pass "fully qualified URL".

-->If we did not pass fully qualified url we get "InvalidArgumentException".

-----  
2: getTitle(): It is used to display the title of the webpage

Return type: String

-->It is used for validation purpose

-----  
3: getCurrentUrl(): It is used to display the url of the webpage

Return type: String

-->It is used for Validation Purpose

-----  
4: getPageSource():

-->It is used to capture the source code of a webpage

Return type: String

Script:

```
WebDriver driver = new ChromeDriver();
driver.get("https://www.instagram.com/");
//usage of getPageSource
String pagesource = driver.getPageSource();
System.out.println(pagesource);
```

-----  
5: close():

-->It is used to close the current/parent window(where driver control is present)

-->Return type: void  
-----

6: quit():

-->It is used to close all the windows opened by selenium(parent + child window)

-->return type: Void.

Script:

```
WebDriver driver = new ChromeDriver();
driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(15));
driver.get("https://www.shoppersstack.com/products_page/25");
driver.findElement(By.id("compare")).click();
driver.quit();
```

Note: close() and quit() should always be the last line of code.

-->Otherwise we get an exception saying "NoSuchSessionException"

-----  
Ques: Diff. between close() and quit()

Close()	quit()
---------	--------

- |                                |  |
|--------------------------------|--|
| 1: It will close parent window | 1: It will close parent + child window |
| 2: Return type: Void           | 2: Return type: Void                   |
| 3: It will not stop the server | 3: It will stop the server.            |
- 

7: manage(): It is used for managing purposes.

-->Return type: options

[Options is an interface in selenium]

-->It is used to manage:

- 1: window
- 2: timeouts
- 3: cookies

**\*\*Now 1: window**

-->return type is window[Interface in selenium]

- 1: maximize()
- 2: minimize()
- 3: fullscreen()
- 4: getSize()
- 5: setSize(Dimension)
- 6: getPosition()
- 7: setPosition(point)

1: maximize(): It is used to maximize the browser window.

Syntax: driver.manage().window().maximize();

2: minimize(): It is used to minimize the browser window

Syntax: driver.manage().window().minimize();

3: fullscreen(): It is used to make browser window fullscreen.

-->Here, the browser window title, url and others parts will be hidden  
Syntax: driver.manage().window().fullscreen();

4: getSize():

-->It is used to return height and width of the webpage  
-->It returns size in form of pixel format  
-->return type of getSize() is Dimension[class in selenium]

Syntax: Dimension size = driver.manage().window().getSize();  
syso(size);

5: setSize(Dimension):

-->It is used to set the size of the browser  
-->Here we need to create an object of Dimension class

```
driver.manage().window().setSize(new Dimension(int width,int height));
```

6: getPosition():

-->It is used to return position of the window  
-->Return type is Point[class in selenium]

Syntax: Point point=driver.manage().window().getPosition();  
syso(point);

7: setPosition():

-->It is used to set the position of the window.  
-->here we need to create an object of Point class.

Syntax: driver.manage().window().setPosition(new Point(int x, int Y));

-----  
Script:

```
WebDriver driver = new ChromeDriver();  
//      driver.manage().window().maximize();  
//      driver.get("https://www.amazon.in/");  
//      driver.manage().window().minimize();  
//      driver.manage().window().fullscreen();  
  
driver.get("https://www.amazon.in/");  
//Before Maximizing  
Dimension size = driver.manage().window().getSize();  
System.out.println(size);  
System.out.println(size.getHeight());  
System.out.println(size.getWidth());  
  
//Maximize the browser  
driver.manage().window().maximize();  
  
//After Maximizing  
System.out.println(driver.manage().window().getSize());  
System.out.println(driver.manage().window().getSize().getHeight());  
System.out.println(driver.manage().window().getSize().getWidth());
```

```
//Setting the size
driver.manage().window().setSize(new Dimension(300,300));

//getPosition
Point position = driver.manage().window().getPosition();
System.out.println(position);
System.out.println("X coordinate="+position.getX());
System.out.println("Y coordinate="+position.getY());

//setPosition
driver.manage().window().setPosition(new Point(300,300));
```

---

AssignQues: Write same script for Facebook App using all manage methods

---

8: getWindowHandle():

-->It is used to get the window ID/session ID of a webpage where driver control is present(Parent window)

-->Return type: String

Script:

```
//Launch the browser
WebDriver driver = new ChromeDriver();

//Maximize the browser window
driver.manage().window().maximize();

//Navigate to URL
driver.get("https://www.naukri.com/");

//Get the session ID
String parentId = driver.getWindowHandle();
System.out.println(parentId);
```

---

9: getWindowHandles():

--> It is used to get all the window ID of the webpages that are opened by selenium(Parent + Child window)

Return type: set<String>

Script:

```
WebDriver driver = new ChromeDriver();
driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(15));
driver.get("https://www.shoppersstack.com/products_page/25");
driver.findElement(By.id("compare")).click();

//get the session Id
Set<String> allIds = driver.getWindowHandles();
System.out.println(allIds);
```

---

Ques: Diff. between getWindowHandle() & getWindowHandles()

getWindowHandle()

getWindowHandles()

1: It is used to get the session ID of window

Parent window

2: Return type: String

1: It is used to give the session ID of Parent + child

2: Return type: set<String>

---

10: switchTo():

-->It is used to switch the driver control to

1: window()

2: frame()

3: alert()

-->Return type of switchTo() is TargetLocator(Interface in selenium)

---

11: navigate(): It is used to navigate to other application or within the application

-->Return type: Navigation[Interface in selenium]

-->Using navigate methods we can perform following actions:

1: to(String url)

Note: to(): it is the alternate method to open application using URL [Internally calls get()]

2: forward(): using this we can perform forward() action

3: back(): using this we can perform back() action

4: refresh(): using this we can refresh the page

Ques: WATS to perform foll actions:

1: Open Browser

2: Open Facebook App

3: Open Google App

4: Navigate back to facebook App

5: Navigate forward to Google App

6: Refresh the webpage

7: Close the browser

Script:

```
WebDriver driver = new FirefoxDriver();
driver.get("https://www.facebook.com/");
driver.get("https://www.google.com/");
driver.get("https://www.facebook.com/");
driver.get("https://www.google.com/");
```

Script:

```
driver.get("https://www.facebook.com/");
driver.navigate().to("https://www.google.com/");
driver.navigate().back();
driver.navigate().forward();
driver.navigate().refresh();
driver.close();
```

---

Ques: Diff. between get() and navigate()

get()

navigate()

1: It is used to open an application/a webpage

2: It will wait until the webpage is completely loaded.

1: It is used to open an application/a webpage

2: It will not wait until the webpage is

loaded

3: We can only open the application.  
operations.

3: We can perform back, forward, and refresh

4: Return type: Void

4: Return type: Navigation

---

AssignQues: WATS to perform Following actions:

1: Open Browser

2: Open Google App

3: Open Facebook App

4: Navigate back to google

5: Navigate to facebook

6: Refresh the browser

7: Change browser size to 300\*500

8: Change browser position to 200\*300

9: maximize the browser

10: set browser fullscreen

11: minimize the browser

12: close the browser

---

Ques: WATS to verify OrangeHRM login page displayed or not