**How to Scroll a Page in Selenium Webdriver:**

A Scrollbar lets us to move around the screen in horizontal or vertical direction if the current page content does not fit the visible area of the screen. It is used to move the window up and down [vertical scroll bar] or move the window left and right side [horizontal scroll bar].

Selenium Web driver does not require scroll to perform actions as it manipulates DOM. But in certain web pages, elements only become visible once the user have scrolled to them. In such cases scrolling may be necessary.

We use JavaScriptExecutor interface that helps to execute JavaScript methods through Selenium Webdriver.

//Java script executor syntax

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript("script",arguments);

* Script – The JavaScript that needs to execute.
* Arguments – The arguments to the script. It's optional.
* Returns – One of Boolean, Long, String, List, WebElement, or null.

**We can scroll by the page in below ways:**

1. Scroll to the bottom of the page. [Vertical scroll]
2. Scroll to the top of the page. [Vertical scroll]
3. Scroll until the element is found.
4. Scroll by an Element’s Pixel.
5. Scroll to the right of the page. [horizontal scroll]
6. Scroll to the left of the page. [horizontal scroll]

JavaScript methods that helps to scroll the document by the specified way.

window.scrollBy

window.scrollTo

**Scroll to the bottom of the page:**

**import** org.openqa.selenium.JavascriptExecutor;

**import** org.openqa.selenium.WebDriver;

**public** **class** ScrollWebPage {

**public** **static** **void** scrollToBottom(WebDriver driver) {

JavascriptExecutor jse = (JavascriptExecutor) driver;

jse.executeScript("window.scrollTo(0, document.body.scrollHeight)");

}

}

@Test(priority=1)

**public** **void** scrollingToBottomofAPage() {

ScrollWebPage.*scrollToBottom*(*driver*);

}

We get the height of the Body element from the DOM (Document Object Model) and we use the scrollTo() method to scroll to the maximum height of the page.

**Scroll to the Top of the page:**

**public** **static** **void** scrollToUp(WebDriver driver) {

*jse* = (JavascriptExecutor) driver;

*jse*.executeScript("window.scrollTo(document.body.scrollHeight, 0)");

}

@Test(priority=1)

**public** **void** scrollingToBottomofAPage() **throws** Exception {

ScrollWebPage.*scrollToBottom*(*driver*);

ScreenShot.*screenShotTC*(*driver*);

System.***out***.println("Moved to bottom of the screen");

*driver*.manage().timeouts().implicitlyWait(20, TimeUnit.***SECONDS***);

ScrollWebPage.*scrollToUp*(*driver*);

ScreenShot.*screenShotTC*(*driver*);

System.***out***.println("Moved to Top of the screen");

}

**Scroll until the element is found:**

**public** **static** **void** scrollToship(WebDriver driver, WebElement element) {

*jse* = (JavascriptExecutor) driver;

*jse*.executeScript("arguments[0].scrollIntoView();", element);

}

//\*[@id="staticGalleryComponent-687622393"]/a

//\*[@id="staticGalleryComponent-1232500634"]/a

**public** **static** List<WebElement> rhapsodyofseas(WebDriver driver) {

List<WebElement> listofcruise =driver.findElements(By.*xpath*("//\*[@class=\"static\_gallery\_\_item\"]/a"));

**return** listofcruise;

}

**public** **void** scrollingtoElement() **throws** Exception{

*listofships* = Home\_Page.*rhapsodyofseas*(*driver*);

**for**(WebElement a : *listofships*)

{

*shref* = a.getAttribute("href");

**if**(*shref*.equals("https://www.royalcaribbean.com/cruise-ships/rhapsody-of-the-seas"))

{

ScrollWebPage.*scrollToship*(*driver*,a);

a.click();

ScreenShot.*screenShotTC*(*driver*);

System.***out***.println("Clicked on \"Rhapsody of the Seas Cruise\"");

**break**;

}

}

}

**Scroll by Pixel:**

executeScript("window.scrollBy(x-pixels,y-pixels)");

* x-pixels is the number at x-axis, it scroll by (moves) to the left if number is negative and it move to the right if number is positive.
* y-pixels is the number at y-axis, it scroll by (moves) to the down if number is positive and it move to the up if number is in negative.

First we need to **locate the element and find its location**. Then from location, we can **get the X and Y coordinates**, which we can pass in scrollBy method.

**public** **static** **void** scrollToElePixel(WebDriver driver, **int** x, **int** y) {

*jse* = (JavascriptExecutor) driver;

*jse*.executeScript("window.scrollBy(" + x + ", " + y + ");");

}

@Test(priority=3)

**public** **void** scrollingbyPixel() **throws** Exception {

*element* = Home\_Page.*platedPerf*(*driver*);

point = *element*.getLocation();

x\_coordinate = point.getX();

y\_coordinate = point.getY();

System.***out***.println("x\_coordinate of Element is : "+ x\_coordinate);

System.***out***.println("y\_coordinate of Element is : "+ y\_coordinate);

ScrollWebPage.*scrollToElePixel*(*driver*,x\_coordinate, y\_coordinate);

ScreenShot.*screenShotTC*(*driver*);

System.***out***.println("Moved to a specific pixel of the screen");

*element*.click();

ScreenShot.*screenShotTC*(*driver*);

System.***out***.println("clicked the element after scroll");

System.***out***.println("The current page is : " + *driver*.getCurrentUrl());

}

**Both ways of code are same:**

Code for Scroll down:

jse.executeScript("window.scrollBy(0,500)", "");

OR,

jse.executeScript("scroll(0, 250);");

Code for Scroll up:

jse.executeScript("window.scrollBy(0,-500)", "");

OR,

jse.executeScript("scroll(0, -250);");

**Scroll to the right and left of the page:**

**public** **static** **void** scrollToEleright(WebDriver driver) {

*jse* = (JavascriptExecutor) driver;

*jse*.executeScript("window.scrollBy(2000,950)");

}

**public** **static** **void** scrollToEleleft(WebDriver driver) {

*jse* = (JavascriptExecutor) driver;

*jse*.executeScript("window.scrollBy(-2000,950)");

}

@Test(priority=4)

**public** **void** scrollingbyRight() **throws** Exception {

*element* = Home\_Page.*eleright*(*driver*);

point = *element*.getLocation();

x\_coordinate = point.getX();

y\_coordinate = point.getY();

System.***out***.println("x\_coordinate of Element is : "+ x\_coordinate);

System.***out***.println("y\_coordinate of Element is : "+ y\_coordinate);

ScrollWebPage.*scrollToEleright*(*driver*);

ScreenShot.*screenShotTC*(*driver*);

}

@Test(priority=5)

**public** **void** scrollingbyLeft() **throws** Exception {

ScrollWebPage.*scrollToEleleft*(*driver*);

ScreenShot.*screenShotTC*(*driver*);

}

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">

<suite name=*"Functional Testing"* parallel=*"none"*>

<test name=*"cruise-chromeTest"*>

<parameter name=*"browser"* value=*"chrome"*/>

<classes>

<class name=*"scrollJS.TestCaseScrollby"*/>

</classes>

</test> <!-- Test -->

<test name=*"cruise-firefoxTest"*>

<parameter name=*"browser"* value=*"firefox"*/>

<classes>

<class name=*"scrollJS.TestCaseScrollby"*/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

**Reference links:**

JavaScriptExecutor in Selenium WebDriver with Examples

<https://www.softwaretestingmaterial.com/javascriptexecutor-selenium-webdriver/>

<https://www.w3resource.com/javascript-exercises/javascript-dom-exercise-9.php> Javascript

Test script uses Actions class to Scroll Up/Down a page:

<https://www.edureka.co/community/51052/how-to-scroll-up-down-a-page-using-actions-class-in-selenium>

<https://www.seleniumeasy.com/selenium-tutorials/scrolling-web-page-with-selenium-webdriver-using-java>

<https://selenium99.com/scroll-up-and-down-selenium-webdriver/>

<https://www.360logica.com/blog/multiple-ways-to-scroll-a-page-using-selenium-webdriver/>

<https://www.softwaretestingmaterial.com/scroll-web-page-using-selenium-webdriver/>

<https://www.edureka.co/community/757/how-to-scroll-the-page-or-down-selenium-webdriver-using-java>

<https://www.guru99.com/scroll-up-down-selenium-webdriver.html>

<https://www.edureka.co/community/51225/scroll-page-using-coordinates-webelement-selenium-webdriver>

Sample websites for scroll:

<https://www.softwaretestingmaterial.com/>

https://www.royalcaribbean.com/cruise-ships