



01/08

Selenium - Java Cheat Sheet (Part 2)



+





Handling Windows

Get the current window handle:

```
String mainWindowHandle = driver.getWindowHandles();
```

Get all window handles:

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

Switch to a window by handle:

```
String windowHandle = "window_handle";  
driver.switchTo().window(windowHandle);
```

Switch back to main window:

```
driver.switchTo().window(windowHandle);
```





Taking Screenshot

Take a screenshot of the whole page

```
File screenshotFile = ((TakeScreenshot)  
driver).getScreenshotAs(OutputType.FILE);
```

```
FileUtils.copyFile(screenshotFile, new  
File("screenshot.png"));
```

Take a screenshot of a specific element

```
File elementScreenshotFile =  
element.getScreenshotAs(OutputType.FILE);
```

```
FileUtils.copyFile(elementScreenshotFile, new  
File("element_screenshot.png"));
```





Use of JavascriptExecutor

Create a reference

```
JavascriptExecutor js = (JavascriptExecutor) driver;
```

To refresh the browser window

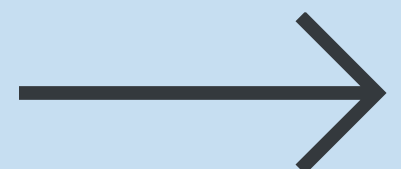
```
js.executeScript("location.reload()");
```

To send text

```
js.executeScript("document.getElementById('id ').value =  
'Alphabin';");
```

Generate Alert Pop Window

```
Js.executeScript("alert('hello world');");
```





Get InnerText of a Webpage

```
string text = js.executeScript("return  
document.documentElement.innerText;").toString();
```

Get Title of a WebPage

```
string text = js.executeScript("return  
document.title;").toString();
```

Click an Element

```
WebElement loginButton = driver.findElement(  
By.id("elementId"));
```

```
js.executeScript("arguments[0].click();", loginButton);
```



Wait Operations

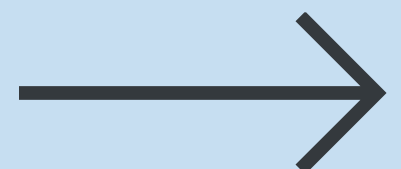
Implicit wait – global wait

```
driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(2));
```

Explicit wait – local wait

```
WebDriverWait wait = new WebDriverWait (driver,  
Duration.ofSeconds(Time in Seconds));
```

```
WebElement element =  
wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("id")));
```





Fluent wait – local wait. Is like Explicit with more options

```
Wait<WebDriver> fluentWait = new FluentWait<>(driver)
    .withTimeout(Duration.ofSeconds(Time in Seconds))
    .pollingEvery(Duration.ofMillis(Time in Milliseconds))
    .ignoring(ElementNotInteractableException.class);
```

```
WebElement element = wait.until(ExpectedConditions
    .visibilityOfElementLocated(By.id("id")));
```

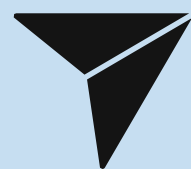
Java hard wait

```
Thread.sleep(Time in MilliSeconds);
```





Was This Helpful?



alphabin.co

