## Driver Initialization

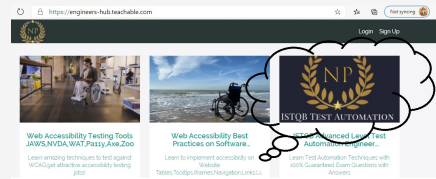| | |
|---|---|
| Firefox (*Below v.47*)[Refer Part2-Page58] | System.setProperty("webdriver.gecko.driver", "PathTo\\geckodriver.exe"); |
| Download from https://github.com/mozilla/geckodriver/releases | WebDriver driver = new FirefoxDriver(); |
| Chrome[Refer Part2-Page59] | System.setProperty("WebDriver.chrome.driver", System.getProperty("user.dir")+"//BrowserDrivers//chromedriver.exe"); |
| Download from https://sites.google.com/a/chromium.org/chromedriver/downloads | WebDriver driver = new ChromeDriver(); |
| Internet Explorer[Refer Part2-Page59] | System.setProperty("WebDriver.ie.driver", "PathTo//IEDriverServer.exe"); |
| Download from The Internet Explorer Driver Server at http://docs.seleniumhq.org/download/ | WebDriver driver = new InternetExplorerDriver(); |
| Safari Driver | WebDriver driver = new SafariDriver(); |

## Selenium Locators

| | |
|---|---|
| Locating by ID [Refer Part2-Page37] | driver.findElement(By.id("q")).sendKeys("Selenium 3"); |
| Locating by Name[Refer Part2-Page37] | driver.findElement(By.name("q")).sendKeys("Selenium 3"); |
| Locating by Xpath[Refer Part2-Page37/38] | driver.findElement(By.xpath("//input[@id='q']")).sendKeys("Selenium 3"); |

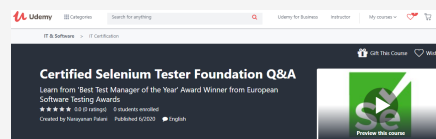## Selenium Locators (cont)

| | |
|---|---|
| Locating Hyperlinks by Link Text [Refer Part2-Page38/39] | driver.FindElement(By.LinkText("edit this page")).Click(); |
| Locating by DOM[Refer Part2-Page39] | dom =document.getElementById('signinForm') |
| Locating by CSS [Refer Part2-Page40-43] | driver.FindElement(By.CssSelector("#rightbar > .menu > li:nth-of-type(2) > h4")); |
| Locating by ClassName[Refer Part2-Page37] | driver.findElement(By.className("profile-header")); |
| Locating by TagName[Refer Part2-Page37] | driver.findElement(By.tagName("select")).Click(); |
| Locating by LinkText[Refer Part2-Page37-40] | driver.findElement(By.linkText("NextPage")).click(); |
| Locating by PartialLinkText[Refer Part2-Page37-40] | driver.findElement(By.partialLinkText("NextP")).click(); |

## ISTQB Advanced Test Automation Engineering



https://engineers-hub.teachable.com/p/istqb-advanced-level-test-automation-engineer-professional-training-with-q-a

## Selenium Q&A



https://www.udemy.com/course/certified-selenium-tester-foundation/

By **Narayanan Palani** (Narayanan Palani)
cheatography.com/narayanan-palani/

Published 15th November, 2016.
Last updated 23rd October, 2020.
Page 1 of 5.

# Cheatography

## Selenium WebDriver Cheat Sheet Cheat Sheet
by Narayanan Palani (Narayanan Palani) via cheatography.com/32112/cs/9838/

## Find element VS Find elements

| | |
|---|---|
| driver.fi ndE lem ent() | When no match has found(0) throws NoSuchElementException when 1 match found returns a WebElement instance when 2+ matches found returns only the first matching web element |
| driver.fi ndE lem ents() | when no macth has found (0) returns an empty list when 1 match found returns a list with one WebElement when 2+ matches found returns a list with all matching WebElements |

## Handle alerts: (Web-based alert pop-ups)

| | |
|---|---|
| driver.switchTO().alert.getText() | to retrieve the alert message |
| driver.switchTO().alert.accept() | to accept the alert box |
| driver.switchTO().alert.dismiss() | to cancel the alert box |
| driver.switchTO().alert.sendKeys("Text") | to send data to the alert box |

## Page Loading Strategy

| | |
|---|---|
| normal | ChromeOptions chromeOptions = new ChromeOptions(); chromeOptions.setPageLoadStrategy(PageLoadStrategy.NORMAL); WebDriver driver = new ChromeDriver(chromeOptions); |
| eager: When setting to eager, Selenium WebDriver waits until DOMContentLoaded event fire is returned. | ChromeOptions chromeOptions = new ChromeOptions(); chromeOptions.setPageLoadStrategy(PageLoadStrategy.EAGER); WebDriver driver = new ChromeDriver(chromeOptions); |

## Page Loading Strategy (cont)

| | |
|---|---|
| none: When set to none Selenium WebDriver only waits until the initial page is downloaded. | ChromeOptions chromeOptions = new ChromeOptions(); chromeOptions.setPageLoadStrategy(PageLoadStrategy.NONE); WebDriver driver = new ChromeDriver(chromeOptions); |

The document.readyState property of a document describes the loading state of the current document. By default, WebDriver will hold off on responding to a driver.get() (or) driver.navigate().to() call until the document ready state is complete

By default, when Selenium WebDriver loads a page, it follows the normal pageLoadStrategy.

## Cookies

| | |
|---|---|
| addCoo kie(arg) | driver.manage().addCookie(new Cookie("foo", "bar")); |
| getCoo kies() | driver.manage().getCookies(); // to get all cookies |
| getCoo kie Named() | driver.manage().getCookieNamed("foo"); |
| delete Cookie Named() | driver.manage().deleteCookieNamed("foo"); |
| delete Cookie() | Cookie cookie1 = new Cookie("test2", "cookie2"); driver.manage().addCookie(cookie1); driver.manage().deleteCookie(cookie1); // deleting cookie object |
| delete AllCoo kies() | driver.manage().deleteAllCookies(); // deletes all cookies |

## Take screenshot

| | |
|---|---|
| Capture the screenshot and store | getScreenshotAs |

getScreenshotAs — used to Capture the screenshot and store it in the specified location. This method throws WebDriverException. copy() method from the File Handler class is used to store the screenshot in a destination folder.
Code Example:
TakesScreenshot screenShot =(TakesScreenshot)driver; FileHandler.copy(screenShot.getScreenshotAs(OutputType.FILE), new File("path/to/destination/folder/screenshot.png"));

By **Narayanan Palani** (Narayanan Palani)
cheatography.com/narayanan-palani/

Published 15th November, 2016.
Last updated 23rd October, 2020.
Page 2 of 5.

# Cheatography

## Selenium WebDriver Cheat Sheet Cheat Sheet
by Narayanan Palani (Narayanan Palani) via cheatography.com/32112/cs/9838/

## Execute Javascript

| | |
|---|---|
| Executes an asynchronous piece of JavaScript | executeAsyncScript() |
| Executes JavaScript | executeScript() |

Code Example:
if (driver instanceof JavascriptExecutor) {
((JavascriptExecutor)driver).executeScript("alert('hello world');");
}

## Annotations

| | |
|---|---|
| TestNG | @BeforeSuite @AfterSuite @BeforeTest @AfterTest @BeforeGroups @AfterGroups @BeforeClass @After-Class @BeforeMethod @AfterMethod |
| JUnit | @After @AfterClass @Before @BeforeClass @Ignore @Test |

## Selenium Exceptions

| | |
|---|---|
| 'Server Exception : sessionId should not be null[Refer Part2-Page160] | Replace Chrome Driver with Latest Version |
| ERROR: No sessionId provided[Refer Part2-Page160] | Use Jenkins |
| UnreachableBrowserException[Refer Part2-Page160] | Registry Key Changes Recommended |
| Illegal state exception using Selenium 3[Refer Part2-Page161] | Get latest Firefox Gecko Driver |

## Keyboard and Mouse events

| | |
|---|---|
| keyboard events | keyDown(); keyUp(); sendKeys() |

## Keyboard and Mouse events (cont)

| | |
|---|---|
| Mouse events | clickAndHold(); contextClick()// — performs the mouse right-click action; doubleClick(); dragAndDrop(source,target); dragAndDropBy(source,xOffset,yOffset); moveByOffset(xOffset,yOffset); moveByElement(); release() |

Code Example:
Actions builder = new Actions(driver);
Action actions = builder
.moveToElement("login-textbox")
.click()
.keyDown("login-textbox", Keys.SHIFT)
.sendKeys("login-textbox", "hello")
.keyUp("login-textbox", Keys.SHIFT)
.doubleClick("login-textbox")
.contextClick()
.build();

actions.perform() ;

## Element validation

| | |
|---|---|
| **isEnab-led()** | determines if an element is enabled or not, returns a boolean. |
| **isSele-cted()** | determines if an element is selected or not, returns a boolean. |
| **isDisp-layed()** | determines if an element is displayed or not, returns a boolean. |

## Handle multiple windows and tabs

| | |
|---|---|
| getWindowHandle() | used to retrieve the handle of the current page (a unique identifier) |
| getWindowHandles() | used to retrieve a set of handles of the all the pages available |
| driver.switchTo().window-("windowName/handle") | switch to a window |
| driver.close() | closes the current browser window |
| driver.quit(); | This method Closes all windows opened by the WebDriver. |

# Cheatography

## Selenium WebDriver Cheat Sheet Cheat Sheet
by Narayanan Palani (Narayanan Palani) via cheatography.com/32112/cs/9838/

## Switch Frames

| | |
|---|---|
| Mentioning the frame index number, the Driver will switch to that specific frame | **driver.switchTo.frame(int frameNumber)** |
| Mentioning the frame element or ID, the Driver will switch to that specific frame | **driver.switchTo.frame(string frameNameOrID)** |
| Mentioning the frame web element, the Driver will switch to that specific frame | **driver.switchTo.frame(WebElement frameElement)** |
| Switching back to the main window | **driver.switchTo().defaultContent()** |

## Materials

| | |
|---|---|
| Selenium Certification | https://isqi.org/en/54-certified-selenium-tester-foundation.html |
| Selenium Certification Q&A | https://engineers-hub.teachable.com/ |
| Part1-Software Automation Testing Secrets Revealed | https://www.amazon.com/Software-Automation-Testing-Secrets-Revealed/dp/1-729432506/ |
| Part2-Software Automation Testing Secrets Revealed | https://www.amazon.com/gp/product/17-29432379/ |
| Part3-Software Automation Testing Secrets Revealed | https://www.amazon.com/Advanced-Selenium-Web-Accessibility-Testing/dp/19-49449432/ |
| Udemy Course-Selenium | https://engineers-hub.teachable.com/ |
| Udemy Course-ISTQB Test Automation Engineering | https://engineers-hub.teachable.com/ |

## Materials (cont)

| | |
|---|---|
| Useful Commands (Credits: Madhan K) | https://medium.com/@madhankumaravelu93/selenium-cheat-sheet-a-comprehensive-list-of-selenium-commands-fa4c5c9d11ab |
| Useful Commands (Credits: Bijan Patel) | https://qascript.com/selenium-webdriver-commands-cheat-sheet/ |

## Get methods in Selenium

| | |
|---|---|
| **getTitle()** | used to retrieve the current title of the webpage |
| **getCurrentUrl()** | used to retrieve the current URL of the webpage |
| **getPageSource()** | used to retrieve the current page source of the webpage |
| **getText()** | used to retrieve the text of the specified web element |
| **getAttribute()** | used to retrieve the value specified in the attribute |

## Navigation

| | |
|---|---|
| Refresh Page | **driver.navigate().refresh()** |
| Navigate forwards in the browser history | **driver.navigate().forward()** |
| Navigate backward in the browser history | **driver.navigate().back()** |

## Desired Capabilities-Chrome

| |
|---|
| DesiredCapabilities caps = new DesiredCapabilities(); caps.setCapability("browserName", "chrome"); caps.setCapability("browserVersion", "80.0""); caps.setCapability("platformName", "win10"); WebDriver driver = new ChromeDriver(caps); // Pass the capabilities as an argument to the driver object |

# Cheatography

## Selenium WebDriver Cheat Sheet Cheat Sheet
by Narayanan Palani (Narayanan Palani) via cheatography.com/32112/cs/9838/

## Desired Capabilities-Firefox

DesiredCapabilities caps = new DesiredCapabilities(); caps.setCapability("browserName", "firefox"); caps.setCapability("browserVersion", "81.0""); caps.setCapability("platformName", "win10");
WebDriver driver = new FirefoxDriver(caps); // Pass the capabilities as an argument to the driver object

## Practice Scripts

| | |
|---|---|
| Launch Webpage[Refer Part2-Page60] | driver.get("www.webdriverinselenium.com"); |
| Click Button[Refer Part2-Page60] | driver.findElement(By.id("submit")).click(); |
| Store Text[Refer Part2-Page60] | String txtDropdown = driver.findElement(By.ID("select")).getText(); |
| Enter Text[Refer Part2-Page61] | driver.findElement(By.xpath("//input[@name='FirstName034']")).sendKeys("NarayananPalani"); |
| Handle Alert[Refer Part2-Page62] | Alert Alertpopup = driver.switchTo().alert(); |
| (Mouse)Click[Refer Part2-Page62] | driver.findElement(By.xpath("//input[@value='OBJECT NAME']")).click(); |
| Compare Text[Refer Part2-Page64] | Assert.assertTrue(chkbox12.isSelected()); |
| Disable a Field[Refer Part2-Page64] | driver.getElementsByName('<ObjectID>')[0].setAttribute('disabled', ''); |
| Enable a Field[Refer Part2-Page64] | driver.getElementsByName('<ObjectID>')[0].removeAttribute('disabled'; |
| Screenshot[Refer Part2-Page64] | File snapshot = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE); |

## Practice Scripts (cont)

| | |
|---|---|
| | FileUtils.copyFile(snapshot, new File("C:\\screenshot.jpg")); |
| Print the Title of the Page[Refer Part2-Page68] | String pagetitle = driver.getTitle(); |
| | System.out.print(pagetitle); |
| Implicit Wait[Refer Part2-Page69] | driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS); |
| Explicit Wait[Refer Part2-Page70] | WebDriverWait wait = new WebDriverWait(driver, 20); |
| | wait.until(ExpectedConditions.textToBePresentInElementLocated(By.xpath("//div[@id='<Object ID>']"), "Enter Your Personal Details")); |
| Fluent Wait[Refer Part2-Page70] | Wait wait = new FluentWait(driver) |
| | .withTimeout(20, SECONDS) |
| Sleep[Refer Part2-Page70] | Thread.Sleep(10); |