



Resizing a web element using movebyoffset

Normally when ever we want to change the size of an element we do with the help of mouse manually. Now we will see to resize / change the size of an element using webdriver actions class with moveByOffset which moves the mouse position from its current position by the given offset.

Below are the simple steps that needs to be followed for the below example:-

Step 1: Open the URL

Step 2: Wait for the element that you want to resize. (Make sure if there are any frames then we need to shift to the frame and then perform operation).

Step 3: We will define a method using which we need to pass web element and the coordinates to the method.

We will now perform the below example on JQuery Website which has Resizable element.

```
package com.pack.jquery;

import org.openqa.selenium.By;
import org.openqa.selenium.NoSuchElementException;
import org.openqa.selenium.StaleElementReferenceException;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.annotations.Test;

public class ResizeExample {
    WebDriver driver;

    @Test
    public void testToResizeElement() {

        driver = new FirefoxDriver();
        driver.manage().window().maximize();
        driver.navigate().to("http://jqueryui.com/resizable/");
        WebDriverWait wait = new WebDriverWait(driver, 5);
        wait.until(ExpectedConditions.frameToBeAvailableAndSwitchToIt(By.cssSelector(".demo-frame")));
        WebElement resizableElement = driver.findElement(By.cssSelector(".ui-resizable-handle.ui-resizable-se"));
        resize(resizableElement, 50, 50);
    }

    public void resize(WebElement elementToResize, int xOffset, int yOffset) {
        try {
            if (elementToResize.isDisplayed()) {
                Actions action = new Actions(driver);
                action.clickAndHold(elementToResize).moveByOffset(xOffset, yOffset).release().build().perform();
            } else {
                System.out.println("Element was not displayed to drag");
            }
        } catch (StaleElementReferenceException e) {
            System.out.println("Element with " + elementToResize + "is not attached to the page document " + e.getStackTrace());
        } catch (NoSuchElementException e) {
            System.out.println("Element " + elementToResize + " was not found in DOM " + e.getStackTrace());
        } catch (Exception e) {
            System.out.println("Unable to resize" + elementToResize + " - " + e.getStackTrace());
        }
    }
}
```



Selenium Tutorials:

Selenium Tutorials

Selenium Tutorials

- › Introduction to selenium
- › Basic HTML concepts required for Selenium
- › Locators for Selenium
- › How to Run Webdriver in chrome browser?
- › How to run webdriver in IE browser?
- › Selenium webdriver using Eclipse
- › Synchronization in Selenium Webdriver
- › XPath tutorial for Selenium
- › CSS selectors for Selenium with example
- › Firefox Not Connected Exception in Selenium
- › Handle windows popups using Selenium Webdriver
- › Working with Frames in Selenium Webdriver
- › How to handle javascript alerts, confirmation and prompts?
- › Defining Firefox Profile preferences using Selenium Webdriver
- › Why Automation Testing is required?
- › Importance of Automation Testing
- › Mouse Hover Actions in Selenium Webdriver
- › StaleElementReferenceExceptions in Selenium Webdriver
- › Difference between Webdriver get() and Webdriver navigate()
- › Navigation Methods in Webdriver with Examples
- › Handling Cookies in Webdriver
- › How to addcookie with Selenium Webdriver
- › How to delete Cookies in Selenium Webdriver
- › WebDriver Methods
- › Automation Test Frameworks
- › Taking Screenshot using Webdriver
- › Webdriver SELECT Methods to work with Dropdowns
- › Webdriver Select with Multiple Attribute
- › Handling keyboard events and mouse hover events using Webdriver
- › Set browser width and height in Selenium Webdriver
- › Selenium Automation Framework Example
- › UI Map or Object Repository using Properties File
- › Test Data in automation framework
- › Reporting Structure in Selenium Automation Framework
- › Test Configuration File / Executable File
- › Uploading a file with Selenium Webdriver
- › Upload file using AutoIT
- › Uploading a file with sendKeys method
- › Read data from Properties file using Java Selenium
- › Scrolling web page with Selenium Webdriver using java
- › Keyword Driven Framework Example
- › Install Autot in Windows
- › Examples with xpath and Css (ID, Name, Text and Links)
- › Testing in multiple browsers using selenium and testng
- › Working with Checkbox using Resuable Methods
- › Working with Select examples
- › File upload using Robots
- › Download file using selenium webdriver
- › Perform operations on new window
- › Introduction to Page Object Model Framework
- › Mouse hover using javaScript Executor
- › Working with Multiple Checkboxes with Safecheck
- › Click element using JavaScriptExecutor
- › Drag and Drop using Webdriver Action Class
- › WebDriver Waits Examples

- › Simple Page Object Model example
- › Double Click on element using Webdriver
- › Right Click Context Menu example
- › Working with AutoComplete Text box
- › Working with Date Picker Calendar
- › Working with Ajax controls using Webdriver
- › Get Attribute values using Webdriver
- › Get Css values using Webdriver
- › Resizing a web element using movebyoffset
- › Validate Downloaded file after clicking on downloaded button/ link
- › How to verify entries in Exported CSV file
- › Element is not clickable at point SeleniumWebdriverException
- › Extract PDF text And Verify Text Present in PDF using WebDriver
- › Handling Authentication Window with WebDriver (In Firefox, Chrome and IE)

Recent Post

Handling Authentication Window with WebDriver (In Firefox, Chrome and IE)

Selenium Webdriver Tutorials

Locators for Selenium

Optional annotation in TestNG

Parameterization in TestNG using testng.xml

Automation Framework

- › Framework Structure
- › Object Repository
- › Test Data
- › Test Scripts
- › Reports
- › Configuration and Execution

Tags Cloud

Asserts (1) AutoComplete (1) Autolt (2) Automation (3) Build.xml (1)
 Checkboxes (2) Cookies (3) Css (2) Css Examples (2) DataProvider (1)
 Date Picker (1) Drag and Drop (1) Exceptions (4) File Download (3)
 Firefox Profile (2) Framework Frameworks (5) iFrames (1) Interview Questions
 (1) JavascriptExecutor (3) JUnit (2) Keyword Driven (1) Listeners (3)
 Manual Testing (1) Maven (5) Mouse Hover (2) Page Object Model (2)
 Page Scroll (1) Parallel Test (3) properties file (1)

Home

- › Home
- › Tutorials
- › About
- › Contact
- › Privacy Policy
- › Site Map

Tutorial Menu

- › Selenium Tutorials
- › TestNG Tutorial
- › JUnit Tutorial
- › JXL Tutorial
- › Apache POI
- › ANT Tutorial
- › Maven Tutorial

Selenium Tutorials

- › Basic Webdriver Tutorials
- › Advanced Webdriver Tutorials
- › Selenium Framework Examples
- › JavascriptExecutor Examples

Interview Questions

- › Testing Concepts
- › Quality Standards
- › General Questions
- › Roles & Responsibilities
- › Technology & Environment