



# Selenium Easy

For Complete Web Automation Testing Tutorials using Selenium Webdriver



Home >> Selenium Tutorials >> Working with Checkbox using Resuable Methods

## Working with Checkbox using Resuable Methods



Working with checkbox is very simple using webdriver. It is same as click operation. But it is always recommended to check if that is already in selected mode or deselected. Because, if it is already selected and when you click on the same element it will get deselected. We will look into different ways to perform select and de-select operations on checkboxes. We will also use reusable methods to perform the operation in multiple tests.

The below is the simple command to do that:

```
WebElement checkBoxElement=driver.findElement(By.id("persist_box"));
checkBoxElement.click();
```

I will try to explain you both select and de-select with the sample code:

```

package com.pack.methods;
import java.util.List;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.annotations.Test;

public class CheckBoxExample {

    private WebDriver driver;
    private String basePageURL;

    @Test
    public void testCaseToCheck() {
        driver = new FirefoxDriver();
        driver.get(basePageURL);
        WebElement checkBoxElement=driver.findElement(By.id("persist_box"));
        //Wait for the checkbox element to be visible
        new WebDriverWait(driver, 10).until(ExpectedConditions.visibilityOf(checkBoxElement));
        Select_The_Checkbox(checkBoxElement);
    }

    @Test
    public void testCaseToUnCheck() {
        driver.navigate().to(basePageURL);
        WebElement checkBoxElement=driver.findElement(By.id("persist_box"));
        new WebDriverWait(driver, 10).until(ExpectedConditions.visibilityOf(checkBoxElement));
        DeSelect_The_Checkbox(checkBoxElement);
    }

    @Test
    public void testCaseToCheckDesired(){
        driver.navigate().to("someother page");
        WebElement element = driver.findElement(By.cssSelector(".display"));
        Select_The_CheckBox_from_List(element, "soccer");
    }
}

```

In the above program, we have used reusable methods to select, deselect and select a particular value from multiple checkboxes. It is always better to have reusable methods, so that we can reuse the same methods in multiple tests instead of writing the same code in multiple tests to perform the same operation.

```

Select_The_Checkbox(checkBoxElement);
DeSelect_The_Checkbox(checkBoxElement);
Select_The_CheckBox_from_List(element, "soccer");

```

Below are the reusable methods that are used in the above example code. You can write these methods in a separate class called generics or utils class where you can have all the reusable methods like below

*Below method is used to Select a Checkbox, if it is not selected already*

```

public void Select_The_Checkbox(WebElement element) {
    try {
        if (element.isSelected()) {
            System.out.println("Checkbox: " + element + "is already selected");
        } else {
            // Select the checkbox
            element.click();
        }
    } catch (Exception e) {
        System.out.println("Unable to select the checkbox: " + element);
    }
}

```

*Below method is used to De-select a Checkbox, if it is selected already*

```

public void DeSelect_The_Checkbox(WebElement element) {
    try {
        if (element.isSelected()) {
            //De-select the checkbox
            element.click();
        } else {
            System.out.println("Checkbox: "+element+"is already deselected");
        }
    } catch (Exception e) {
        System.out.println("Unable to deselect checkbox: "+element);
    }
}

```

*Below method is used to select the checkbox with the specified value from multiple checkboxes.*

```

public void Select_The_CheckBox_from_List(WebElement element, String valueToSelect) {
    List<WebElement> allOptions = element.findElements(By.tagName("input"));
    for (WebElement option : allOptions) {
        System.out.println("Option value "+option.getText());
        if (valueToSelect.equals(option.getText())) {
            option.click();
            break;
        }
    }
}

```

◀ Assertions in TestNG    Working with Select examples ▶

#### Tags:

Checkboxes  
Reusables




#### Selenium Tutorials:

Selenium Tutorials

-----

Electronics

Search product... GO

	<b>Lenovo K3 Note</b> Rs. 9999
	<b>Moto E (2nd Gen) 3G</b> Rs. 5999
	<b>Moto G (3rd Generation)</b>

[See more products on Flipkart.](#)

amazon.in

Branded bags  
starting from

**Rs. 499**

[Shop now](#)

Limited Period Offer



**SpiceJet's ₹749\***

**DIWALI SALE IS HERE.**

International sector fares are  
all-inclusive and start from **₹3999\***

**Sale Period:**  
27th-29th October 2015

**Travel Period:**  
1st February- 29th October 2016

[Book Now](#)



**SpiceJet**  
Red. Hot. Spicy.

\*T&C apply.

## 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
- › StaleElementReference Exceptions 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 AutoIT 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)
- › Find Broken / Invalid Images on a Page
- › Find out broken links on website using selenium webdriver and HTTP Client
- › How to verify tooltip text with selenium webdriver using java

## Automation Framework

- › Framework Structure
- › Object Repository
- › Test Data
- › Test Scripts

## Recent Post

---

How to verify tooltip text with selenium webdriver using java

---

Configure Log4j.xml file in java

---

Upload file using AutoIT

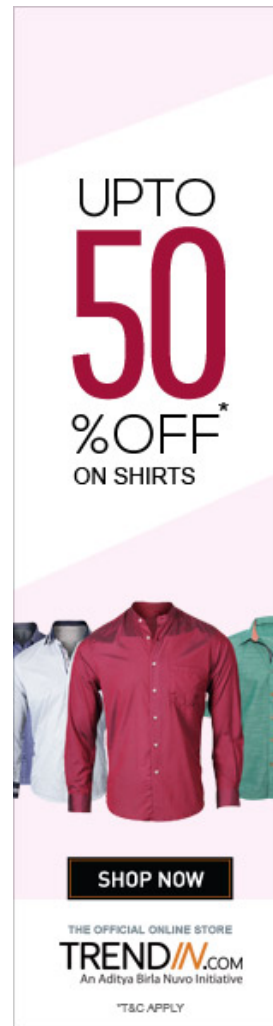
Keyword Driven Framework Example

Read data from Properties file using Java Selenium

- › 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) Excel(4) Exceptions(4) File Download(3)  
Firefox Profile(2) FrameworkFrameworks(5) iFrames(1)  
Images(1) Interview Questions(1) JavascriptExecutor(3) JUnit  
(2) Keyword Driven(1) Listeners(3) Log4j(4) log4j.xml(1)  
Manual Testing(1) Maven(5) Mouse Hover(2)  
Page Object Model(2) Page Scroll(1) Parallel Test(3)  
properties file(1) Re-Run Failed Tests(2) Reusables(2)  
Screenshot(3) Select(3) Selenium(2) Testing Types(1)  
testNG(2) ToolTip(1) Upload(5) Waits(3) Webdriver(9)  
Webdriver Examples(6) Webdriver methods(2) Windows(2)  
Xpath(3) Xpath Examples(2)



## 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

Copyright © 2015 - All Rights Reserved : Selenium Easy

We welcome your questions and requests