[Unable to enter the text in a text box](https://sqa.stackexchange.com/questions/15928/unable-to-enter-the-text-in-a-text-box)

[Ask Question](https://sqa.stackexchange.com/questions/ask)

1

2

I am trying to enter the text in a textbox, but I am getting an error as follows

org.openqa.selenium.InvalidElementStateException: Element must not be hidden, disabled or read-only

Anybody can help me regarding this issue to be fixed.

You may try with javascriptexecutor- Check below code-

driver.switchTo().frame("WallItemItemDetail\_ifr");

// select the editor nad do the input

WebElement TextBox = driver.findElement(By.id("tinymce"));

String input= "editor done";

TextBox.getAttribute("name");

JavascriptExecutor jst= (JavascriptExecutor) driver;

jst.executeScript("arguments[1].value = arguments[0]; ", input, TextBox);

// switch back to the default window

driver.switchTo().defaultContent();

[Selenium - sendKeys() not inserting value into textbox](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox)

[Ask Question](https://sqa.stackexchange.com/questions/ask)

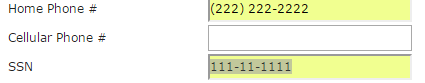
4

3

I am not able to validate an international number text box in Selenium.

I have tried earlier by name, class and other locators but it is not inserting the value into the textbox.

Below is the text box and the corresponding HTML:



The code I am using is:

driver.findElement(By.name("phone")).sendKeys("(222)222-2222");

driver.findElement(By.id("ssn")).sendKeys("555-55-5555");

[selenium](https://sqa.stackexchange.com/questions/tagged/selenium) [webdriver](https://sqa.stackexchange.com/questions/tagged/webdriver" \o "show questions tagged 'webdriver')

[share](https://sqa.stackexchange.com/q/18415)[improve this question](https://sqa.stackexchange.com/posts/18415/edit)

[edited May 9 '16 at 11:28](https://sqa.stackexchange.com/posts/18415/revisions)

[Kate Paulk](https://sqa.stackexchange.com/users/422/kate-paulk)**♦**

**24.9k**64085

asked May 3 '16 at 7:25

[Rajat Sharma](https://sqa.stackexchange.com/users/18106/rajat-sharma)

**21**112

* 1

What exactly you want to do with these Text boxes? by using any locators you have to enter text in to the textbox or you have to validate these textboxes? – [Bharat Mane](https://sqa.stackexchange.com/users/17411/bharat-mane) [May 3 '16 at 7:34](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox#comment21017_18415)

* if you see in above image home phone has a format of international number ,now when i try to insert any number using any locator and using sendkeys method it is not accepting the input. – [Rajat Sharma](https://sqa.stackexchange.com/users/18106/rajat-sharma" \o "21 reputation) [May 3 '16 at 11:38](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox#comment21026_18415)
* What are you using to drive Selenium, and can you post your code? – [Cronax](https://sqa.stackexchange.com/users/11431/cronax" \o "318 reputation) [May 3 '16 at 13:45](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox#comment21028_18415)
* driver.findElement(By.name("phone")).sendKeys("(222)222-2222"); driver.findElement(By.id("ssn")).sendKeys("555-55-5555"); – [Rajat Sharma](https://sqa.stackexchange.com/users/18106/rajat-sharma) [May 5 '16 at 6:55](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox#comment21059_18415)
* So is it working or not? what's your problem with this? What you expect from us? – [Bharat Mane](https://sqa.stackexchange.com/users/17411/bharat-mane) [May 7 '16 at 6:02](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox#comment21116_18415)

[show **5** more comments](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox)

3 Answers

[active](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox?answertab=active#tab-top)[oldest](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox?answertab=oldest#tab-top)[votes](https://sqa.stackexchange.com/questions/18415/selenium-sendkeys-not-inserting-value-into-textbox?answertab=votes#tab-top)

6

If sendkeys() methods are not working then use following two ways to input text:

1. Before sendkeys() use click() method to click inside textfield i.e:
2. driver.findElement(By.name("phone")).click();
3. driver.findElement(By.name("phone")).sendKeys("(222)222-2222");
4. driver.findElement(By.id("ssn")).click();

driver.findElement(By.id("ssn")).sendKeys("555-55-5555");

1. Second way is to use javascript to input text in texfields like this:
2. WebElement wb = driver.findElement(By.name("phone"));
3. JavascriptExecutor jse = (JavascriptExecutor)driver;
4. jse.executeScript("arguments[0].value='(222)222-2222';", wb);

jse.executeScript("document.getElementById('ssn').value='555-55-5555';");

Please let me know if that works for you. Webdriver provided the javscript implementation library so that we could use javascript to perform actions on browser which are not possible with with webdriver native API. And in web automation testing services we encountered many such issues where we have to go beyond webdriver native API and have to use javscript methods.

# [How to type some text in hidden field in Selenium WebDriver using Java](https://stackoverflow.com/questions/11858366/how-to-type-some-text-in-hidden-field-in-selenium-webdriver-using-java)

[Ask Question](https://stackoverflow.com/questions/ask)

13

4

I am using WebDriver with Java for test automation. I have the following HTML code for input field which is hidden:

<input type="hidden" value="" name="body" id=":6b">

How to type something in hidden field in Selenium2 (WebDriver)? I have written code as:

driver.findElement(By.name("body")).sendKeys("test body");

But it was shown the following error: org.openqa.selenium.ElementNotVisibleException: Element is not currently visible and so may not be interacted with Command duration or timeout: 30.04 seconds

Can anybody please help me to write/type some text in hidden field?

## 

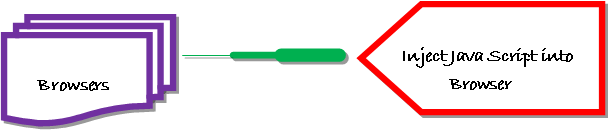
## 2 Answers

[active](https://stackoverflow.com/questions/11858366/how-to-type-some-text-in-hidden-field-in-selenium-webdriver-using-java?answertab=active#tab-top)[oldest](https://stackoverflow.com/questions/11858366/how-to-type-some-text-in-hidden-field-in-selenium-webdriver-using-java?answertab=oldest#tab-top)[votes](https://stackoverflow.com/questions/11858366/how-to-type-some-text-in-hidden-field-in-selenium-webdriver-using-java?answertab=votes#tab-top)

20

First of all you have to change the value of type attribute as text from hidden. The following code using javascript would work for that:

jse.executeScript("document.getElementsByName('body')[0].setAttribute('type', 'text');");

****

Now, you are able to type on that text by using WebDriver. So, the overall code for typing with WebDriver using Java and Javascript as follows:

WebDriver driver = new FirefoxDriver();

JavascriptExecutor jse = (JavascriptExecutor)driver;

jse.executeScript("document.getElementsByName('body')[0].setAttribute('type', 'text');");

driver.findElement(By.xpath("//input[@name='body']")).clear();

driver.findElement(By.xpath("//input[@name='body']")).sendKeys("Ripon: body text");

# JAVASCRIPTEXECUTOR POWERS UP SELENIUM TO HANDLE SCENARIOS

[Avinash Mishra](http://www.inviul.com/author/avinash/" \o "Posts by Avinash Mishra)  March 24, 2018  [Selenium WebDriver](http://www.inviul.com/selenium-tutorials/selenium-webdriver/" \o "View all posts in Selenium WebDriver)  [2 Comments](http://www.inviul.com/javascriptexecutor-selenium/#comments)

Selenium and JavaScript have the long-term relationship since the Selenium evolved. The earlier version of Selenium (i.e., Selenium 1) was fully based on JavaScript and Selenese. Even in advanced Selenium which we are currently using (Selenium WebDriver) does not have a solid solution to handle some of the operations like element is not clickable, not visible, element highlighter, etc. Therefore, Selenium WebDriver implemented a separate interface (JavascriptExecutor) to handle out of the box scenarios through JavaScript.

## ****SELENIUM DEALS ON OUT OF THE BOX SCENARIOS WITH JAVASCRIPTEXECUTOR****

In our previous tutorials, we have discussed some of the out of the box scenarios which WebDriver implements with the help of JavascriptExecutor by using JavaScript commands. It is highly recommended that you must learn these tutorials. Here is the list for you:

* [How to scroll the web page in Selenium using JavaScript?](http://www.inviul.com/scrolling-webpage/)
* [How to handle click operation in Selenium using JavaScript?](http://www.inviul.com/click-using-javascriptexecutor/)
* [How to highlight elements in Selenium using JavaScript?](http://www.inviul.com/highlight-elements-selenium/)
* [How to handle checkbox and radio button in Selenium using JavaScript?](http://www.inviul.com/checkbox-radio-button-selenium/)
* [How to send texts and retrieve text in Selenium using JavaScript?](http://www.inviul.com/send-texts-get-texts-selenium/)
* [How to click on (X, Y) coordinates in Selenium using JavaScript?](http://www.inviul.com/click-coordinates-using-javascriptexecutor/)

I am 100% sure that [these tutorials will help you somewhere](https://seleniumhq.github.io/selenium/docs/api/java/org/openqa/selenium/JavascriptExecutor.html) in your Selenium Automation development journey.

### **WHAT IS JAVASCRIPT?**

JavaScript is one among the group of OOP (Object Oriented Programming Language). It is basically used to perform interaction with the HTML and the elements in DOM. It signifies that all the browsers have an implementation to understand and send the response to the JavaScript commands. You can even disable the JavaScript from your browser’s settings

### **ADVANTAGES OF JAVASCRIPT**

* It changes the content of the HTML through innerHTML
* JavaScript changes the attributes of the HTML
* It also changes the CSS Style
* It can entirely change, hide and show the HTML elements

Image Credit: Medium

### **WHAT IS JAVASCRIPTEXECUTOR?**

JavascriptExecutor is an interface in Selenium WebDriver, which is basically used to inject the JavaScript code at runtime for the purpose of functional testing. It performs the actions using two methods, which are discussed as follows:

#### **1. EXECUTESCRIPT() METHOD**

This method used to pass complicated arguments in the body of the random methods. It executes the command by considering the context of the window or the frames.

The return type of executeScript() method is listed as follows:

* Long
* List
* WebElement
* String
* Boolean

Most of the time we use executeScript()  method for our scenarios. Here is the syntax for it:



|  |  |
| --- | --- |
| 1  2  3 | JavascriptExecutor js = (JavascriptExecutor)driver;    Js.executeScript(script, arguments); |

#### **2. EXECUTEASYNCSCRIPT() METHOD**

It is more powerful than the executeScript() method. It is also best for performance improvement of your Selenium Automation test. The browser executes everything which is written inside the script, not by the server, as it sends call back to the server-side testing. Whereas, executeScript() method blocks the execution.

Here is the syntax for executeAsyncScript() method:



|  |  |
| --- | --- |
| 1  2  3 | JavascriptExecutor js = (JavascriptExecutor)driver;    Js.executeAsyncScript(script, arguments); |

## Why do we need JavaScriptExecutor?

In Selenium Webdriver, locators like XPath, CSS, etc. are used to identify and perform operations on a web page.

In case, these locators do not work you can use JavaScriptExecutor. You can use JavaScriptExecutor to perform an desired operation on a web element.

Selenium supports javaScriptExecutor. There is no need for an extra plugin or add-on. You just need to import (**org.openqa.selenium.JavascriptExecutor**) in the script as to use JavaScriptExecutor

**What are the advantages of using a JavaScript executor in Selenium?**

[Answer](https://www.quora.com/What-are-the-advantages-of-using-a-JavaScript-executor-in-Selenium)

[Follow· 11](https://www.quora.com/What-are-the-advantages-of-using-a-JavaScript-executor-in-Selenium)

[Request](https://www.quora.com/What-are-the-advantages-of-using-a-JavaScript-executor-in-Selenium)

7 Answers

[Alex Siminiuc](https://www.quora.com/profile/Alex-Siminiuc-2), enjoys writing about Selenium test automation and labradors

[Updated Oct 9, 2017](https://www.quora.com/What-are-the-advantages-of-using-a-JavaScript-executor-in-Selenium/answer/Alex-Siminiuc-2)

The main advantage of executing Javascript code in Selenium tests is this:

**When you cannot do something through the Selenium WebDriver classes and methods, you have another way of doing it.**

What does this mean?

I have been in many situations where interacting with elements through Javascript was the only possible solution.

For example.

1. **click() method does not click the button.**

You may face some situations in Selenium where click operations do not go well. There could be many reasons; some of the reasons are listed below:

* On –click method triggers JavaScript code
* A clickable element is disabled

In this scenario, [exceptions like, IllegalStateException, ElementNotVisibleException, StaleElementReferenceException and NoSuchElementException](http://www.inviul.com/exceptions-in-selenium-webdriver/) comes.

The click() method of the WebElement interface was executed and it did not throw any error or exception.

Still, the button was not clicked.

This was not a problem of synchonization, the button was available and with the proper state when clicking was done.

Doing the same thing through Javascript worked well.

1. JavascriptExecutor jsExecutor = (JavascriptExecutor) driver;
3. String query = "document.getElementById('elementId').click()";
5. jsExecutor.executeScript(query);

or

1. JavascriptExecutor jsExecutor = (JavascriptExecutor) driver;
3. String query = "document.querySelector(\"class='search\_button']\").click()";
5. jsExecutor.executeScript(query);

**2. typing a keyword in the textbox does not work**

Same thing as above.

The code tried typing a keyword in a textbox using the sendKeys() method and nothing happened.

Doing the same thing through Javascript just worked.

1. JavascriptExecutor jsExecutor = (JavascriptExecutor) driver;
3. String query = "document.getElementById('searchbox').value = 'java'";
5. jsExecutor.executeScript(query);

**3. selecting all checkboxes from a list selected only the visible ones**

This is the most recent case where Javascript helped me.

I was working on some code to check all checkboxes embedded in a listbox.

Only 5 checkboxes out of 10 were displayed at a time.

To see the other ones, you would have to scroll the listbox.

I thought that the code was working correctly to discover that it was only checking the visible checkboxes.

The not-visible ones were not checked.

I could not find a good way using the WebElement interface.

So, again, I used Javascript.

…………………………………………………………………………….

Having the ability to execute Javascript in the Selenium tests is very important.

Sometimes,**the Javascript code is the only thing that works.**

**You can do everything that the WebElement interface does and more with Javascript.**

[See on this link](https://seleniumjava.com/2016/09/05/how-to-execute-javascript-code-in-selenium-webdriver/) how to do many other things using Javascript:

* get an element’s text
* find an element
* change the color of an element
* zoom in the page
* scroll to an element
* wait until the page is loaded

If it is one thing that I would like to know more of for my test automation work, that is Javascript.

# Is it possible to interact with hidden elements using WebDriver?

[Answer](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Follow· 6](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Request](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

4 Answers

[Vinoth Kumar Mahendira](https://www.quora.com/profile/Vinoth-Kumar-Mahendira), works at HCL Technologies

[Answered Jan 28, 2017](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver/answer/Vinoth-Kumar-Mahendira)

Ofcourse.. all the locator identification method works on hidden element   
Says ById, ByXpath...  
Even you find the visibility of element by isDisplayed() method..  
But if you try to click or sendKeys.. it throws ElementNotVisibleException... since driver engine cannot support direct interaction of Invisible element

Stop stop... don't conclude that selenium incapable to handle hidden element.

We have another option, salient feature of Selenium is JavaScriptExecutor..  
Try this  
JavascriptExecutor js = (JavascriptExecutor)driver;  
js.executeScript("arguments[0].click();", element);  
Even you can send keys using js method value, arguments[0].value='typed'; so on

Sounds pretty good right..

But don't expect Js will have replace method for all interaction like drag and drop,.. :-p

6.7k Views

Your feedback is private.

Is this answer still relevant and up to date?

[Upvote](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Share](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[RecommendedAll](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Vaibhav Gupta](https://www.quora.com/profile/Vaibhav-Gupta-908), Founder and Author at HelpingTesters

[Answered Jan 30, 2017](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver/answer/Vaibhav-Gupta-908)

There is this thing about hidden elements. The web-element is actually not hidden, it just doesn’t appear on the UI. The elements are hidden on the UI with some sorcery with CSS (pun intended).

If for an element:

* display:none (then the element would be hidden)
* display:block (element is visible)

So for the former case, you can interact with hidden elements in either of 2 ways.

1.**Using javascript executor**

1. JavascriptExecutor js = (JavascriptExecutor) driver;
2. js.executeScript("javascript to perform action on webelement");

2. **Using Css selector**to target element with certain CSS property (in case of hidden, display=none)

Thanks,  
Vaibhav(Founder [Helping Testers | Software Testing Training Courses Online](http://www.helpingtesters.com/) )

7k Views · [View Upvoters](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

Your feedback is private.

Is this answer still relevant and up to date?

[Upvote· 1](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Share](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[RecommendedAll](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Sumeet Lall](https://www.quora.com/profile/Sumeet-Lall-1), works at Cognizant Technology Solutions

[Answered Feb 4, 2017](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver/answer/Sumeet-Lall-1)

Hi,

There are few questions coming in my mind.

1. what do you mean by Hidden elements? Are you talking about some fields which are not visible on UI level?
2. why you want to interact with those elements if those are not at all visible?

I guess from an Automation testers point of view you will have to automate the manual process. Manual testing work is generally done on Visible elements on UI level.

If that hidden element has some business function related it will definitely appear on UI level , may be after some checkbox or dropdown value gets selected. You need to understand the business flow and automate according to that.

If you are able to see any element on UI level & able to test that, you can definitely work on them using Automation tool. You may need to use various tools.

Selenium Webdriver has many inbuilt functions to interact with web elements you can use those functions as per the suitability.

thanks

4.2k Views · [View Upvoters](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

Your feedback is private.

Is this answer still relevant and up to date?

[Upvote· 2](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Share](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[RecommendedAll](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver)

[Shadab Ansari](https://www.quora.com/profile/Shadab-Ansari-15), Founder of TeachMeSelenium.com

[Answered Apr 25, 2018](https://www.quora.com/Is-it-possible-to-interact-with-hidden-elements-using-WebDriver/answer/Shadab-Ansari-15)

**Technically, Yes.**

You can use WebDriver’s JavascriptExecutor to execute a piece of Javascript that can interact with any element in the DOM, visible or hidden.

If you are using Java, following would be the steps -

* Cast driver to JavascriptExecutor

1. JavascriptExecutor js = (JavascriptExecutor)driver;

* Write the plain old JavaScript to interact with that element

1. String strJS = “document.findElementsByName('myname')[0].click()”

* Pass the JavaScript to executeScript() method

1. js.executeScript(strJS)

This was just one of the usage of JavascriptExecutor. You can also pass the target element to executeScript() method.

If you are interested in complete tutorial around JavascriptExecutor, you can check the following tutorials at TeachMeSelenium.com- [Free Selenium tutorials in Java and Python | Selenium Jobs](http://teachmeselenium.com/) -

* [Java-Selenium | Executing JavaScript through JavascriptExecutor](http://www.teachmeselenium.com/2014/06/15/java-selenium-interacting-with-the-browser-executing-javascript-through-javascriptexecutor/)
* [Python-Selenium | Executing JavaScript through JavascriptExecutor](http://www.teachmeselenium.com/2018/04/17/python-selenium-interacting-with-the-browser-executing-javascript-through-javascriptexecutor/)

Now here is an important question.

**Why do you need your WebDriver script to interact with the hidden elements?**

In Test Automation, we simulate real-user actions on the Application Under-test. Which is the scenario where you think the user can interact with the hidden element? Think about it before designing your tests.

# <http://learn-automation.com/solve-elementnotvisibleexception-in-selenium-webdriver/>

# How to Solve ElementNotVisibleException in Selenium Webdriver

Hello, Guys, Welcome back to [**Selenium**](http://learn-automation.com/selenium-introduction-and-automation-introduction/)tutorial, today we will see how to handle element not visible  exception in Webdriver.

I faced this exception number of times, I struggled a lot while searching solution, and finally, I got so many solutions to solve this exception.

Before jumping to ElementNotVisibleException you should check below two posts that actually will help you to understand exception better.

[**How to handle Exception in Selenium**](http://learn-automation.com/handling-exception-in-selenium/)

[**What are different exception is available in Selenium.**](http://learn-automation.com/exceptions-in-selenium-webdriver/)

Full Exception document is available here-**https://selenium.googlecode.com/git/docs/api/py/common/selenium.common.exceptions.html**

Let us see what the reason behind this exception is:

## Reasons for ElementNotVisibleException in Selenium Webdriver

**Reason 1- Duplicated XPATH**

While writing xpath for your application, you might have taken xpath that is matching with more than 1 element, in this case, Selenium will throw Element, not the visible exception.

If you are new to Selenium and facing issues while writing XPath then please check below post which will help you to write xpath from basic to advance level.

[**How to write Dynamic Xpath in Selenium Webdriver**](http://learn-automation.com/how-to-write-dynamic-xpath-in-selenium/)

**Reason 2-**

If you are trying to access some particular element on Webpage that is not currently visible, in this case also you will get the Element, not visible exception.

ElementNotVisible Exception in Selenium

## Solutions for ElementNotVisibleException in Selenium Webdriver

**First Solution**: Try to write unique XPATH  that matches with a single element only.

**Second Solution:**

Use [Explicit wait](http://learn-automation.com/explicit-wait-in-selenium-webdriver/) feature of Selenium and wait till the element is not visible. Once it is visible then you can perform your operations.

Syntax for Explicit wait



|  |  |
| --- | --- |
| 1 | new WebDriverWait(driver, 30).until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//span[text()='index.html']"))); |

Explanation- In above example, Selenium will wait 30 seconds until the element is not visible and once it is visible, Selenium will perform an action and will move on to the next step.

You can also give a try using [JavaScript click that forcefully click on Element](http://learn-automation.com/click-in-selenium-webdriver-using-java-script/)

**Third solution:**

This actually works for me a number of times. I was struggling with the OK button in my application. Even I was writing unique xpath but still I was facing Element Not Visible exception.  I have tried below code that solved my issue.



|  |  |
| --- | --- |
| 1  2  3 | int ok\_size=driver.findElements(By.xpath("//button[text()='OK']")).size();    driver.findElements(By.xpath("//button[text()='OK']")).get(ok\_size-1).click(); |

#### **Explanation**

First, I have taken the size of the element then in next statement, I took the first element from the list and I clicked on OK Button.

I have seen so many scenarios where I solved my issue using above scenario.

I also faced lots of Hidden Web Elements in my web application and used below trick which helped me a lot.Check out below youtube video for the same

## Exceptions in Selenium WebDriver

**Before moving to Runtime Exception, you should be aware of Exception meaning.**

Exception general meaning a condition that is not common, in Exceptions in Selenium WebDriver occurs many times in different scenarios let’s have a look to all exception one by one.

Exceptions in Selenium WebDriver

## Exceptions in Selenium WebDriver

## NoSuchElementException

This exception occurs when WebDriver is unable to identify the elements during run time. Due to wrong selector or selector, which is, not exist.  
Example:-



|  |  |
| --- | --- |
| 1 | driver.findElement(By.id("invalidid")).sendKeys("Mukesh"); |

## ElementNotVisibleException

This Exception occurs when the element presence is in DOM, it is not visible.  
Example:-  
Hidden Elements, which has presence in DOM and it, is not visible. Visibility means the height and  
width should be greater than zero. Hidden Elements are defined in HTML using of type=”hidden”.



|  |  |
| --- | --- |
| 1 | driver.findElement(By.id("hiddenid")).sendKeys("Mukesh"); |

## NoSuchFrameException

This Exception occurs when the driver is switching to an invalid frame, which is not available.  
Example:-



|  |  |
| --- | --- |
| 1  2  3 | driver.switchTo().frame(invalidindex);  (or)  driver.switchTo().frame("frame\_z"); |

//frame\_z is the name of the invalid frame  
For frames indexing starts from Zero. Try to access the frame by providing invalid index.

## NoAlertPresentException

This Exception occurs when the driver is switching to an invalid Alert, which is not available.  
Example:-



|  |  |
| --- | --- |
| 1 | driver.switchTo().alert().accept(); |

//Execute this command on browser without invoking the alert.

## NoSuchWindowException

This Exception occurs when the driver is switching to an invalid Window, which is not available.  
Example:-



|  |  |
| --- | --- |
| 1 | driver.switchTo().window("invalidwindowname"); |

## WebDriverException

This Exception occurs when the driver is performing the action after immediately closing the browser.  
Example:-



|  |  |
| --- | --- |
| 1  2 | driver.close();  driver.findElement(By.id("username")).sendKeys("Mukesh"); |

## SessionNotFoundException

This Exception occurs when the driver is performing the action after immediately quitting the browser.  
Example:-



|  |  |
| --- | --- |
| 1  2 | driver.quit();  driver.findElement(By.id("username")).sendKeys("Mukesh"); |

## StaleElementReferenceException

This Exception occurs when the Element belongs to a different frame than the current one. The user has navigated away to another page.



|  |  |
| --- | --- |
| 1  2  3  4 | Example:-  WebElement element=driver.findElement(By.id("username"));// Element is available in parent window  driver.switchTo().window(Child\_Window);//Switch to Child Window  element.sendKeys("Mukesh");//perform the action on the element which is not visible in the child window |

Check out more exceptions here.

Thanks for visiting my blog. Please leave a comment if you have any issues in Selenium.

Keep in touch. Have a nice day

By [Mukesh Otwani](http://learn-automation.com/author/mukesh50/" \o "Posts by Mukesh Otwani)|March 16th, 2015|[Uncategorized](http://learn-automation.com/SeleniumWebdriverTutorial/uncategorized/)|[20 Comments](http://learn-automation.com/exceptions-in-selenium-webdriver/#comments)

|  |  |
| --- | --- |
| [Selenium Users](https://groups.google.com/forum/#!forum/selenium-users) › | |
|  | |
| 4 posts by 3 authors |  |

|  |
| --- |
|  |

# Using sendKeys in a readonly WebElement

|  |  |  |
| --- | --- | --- |
|  | **pablomarcelo** | 21/09/2011 |

**Other recipients:**pabl...@hotmail.com



I'm using WebDriver and I need to write some data in a readonly   
textbox, I'm using sendKeys but it is not working.   
  
Someone can explain me how to send data into a readonly textbox?   
  
Thanks





|  |  |  |
| --- | --- | --- |
|  | **Luke Inman-Semerau** | 21/09/2011 |

**Other recipients:**pabl...@hotmail.com



You'll have to use the JavaScript Executor:

(Assuming your previous code looked like - element.sendKeys("my text to put in the value"); )

((JavascriptExecutor)driver).executeScript("arguments[0].value=arguments[1]", element, "my text to put in the value");

By default WebDriver behaves like an end user... if you can't do it manually, then WebDriver won't do it through the normal api.