**Drop downs, Pop ups, multiple windows.**

Drop down list:

Methods to select drop down list

How to manage pop up alerts

* How can I get to the drop down list :

**We use a index command**

@Test

**public** **void** testUntitled2() **throws** Exception {

myD.get(baseUrl);

String xP = "//\*[@id='rooms']"; //xpath of rooms

//Method 1 : drop down list - Select by index method

WebElement myE = myD.findElement(By.*xpath*(xP)); //decalring a web element

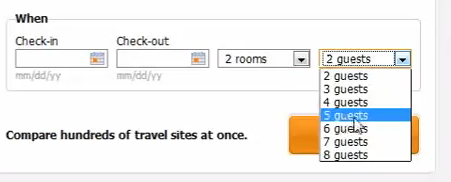
Select mylist = **new** Select(myE); // Pointing to a WE object which is of type Select

mylist.selectByIndex(2); //By index is poiting to order in the drop-down list

//method 2: using a send keys method

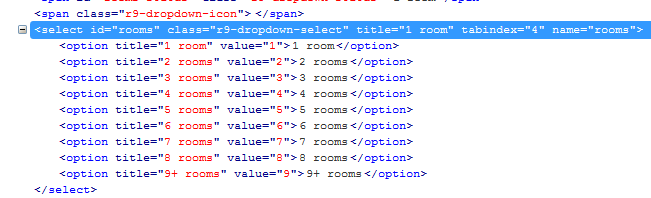
myD.findElement(By.*xpath*(xP)).sendKeys("2"); //When the mouse cursor is placed on the drop down and entered the number 2, it automatically selects the list2

If I want to select a specific information such as 2 rooms and 5 guests as in pic below

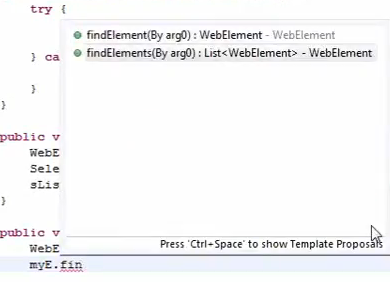


What do I do? I will create a method

The following is the xpath of drop down 2 room, if I expand select, it has the options and values. Such as 1 room has 1 value, 9+ etc. The tag here is option



So if I want to list all elements that are option as a child element under the parent select id rooms, how can I work with that? A webelement always returns an output. These get returned as an Array



If I find element on my web elements, it returns list either in 1 dimensional or multiple dimensional

@Test

**public** **void** testUntitled2() **throws** Exception {

myD.get(baseUrl);

String xP = "//\*[@id='rooms']"; //xpath of rooms

//Method 1 : drop down list - Select by index method

WebElement myE = myD.findElement(By.*xpath*(xP)); //decalring a web element

Select mylist = **new** Select(myE); // Pointing to a WE object which is of type Select

mylist.selectByIndex(2); //By index is poiting to order in the drop-down list

//method 2: using a send keys method

myD.findElement(By.*xpath*(xP)).sendKeys("2"); //When the mouse cursor is placed on the drop down and entered the number 2, it automatically selects the list2

list\_select\_value(xP, "2rooms");

}

//method3:

**public** **void** list\_select\_value(String xpath, String Text){

WebElement myV = myD.findElement(By.*xpath*(xpath));

java.util.List<WebElement> myL = myV.findElements(By.*tagName*("option")) ; //List is an array . when I use tag name, I can say the tag name of that elemenst are option

//The above line will list all the elements

System.*out*.println("size of items in drop down is" + myL.size());

**for**(**int** i=0; i<myL.size(); i++){

System.*out*.println("Value in the item" + i + "is" + myL.get(i).getText());

}

}

}

//it gives the items in the drop down

Output :

size of items in drop down is9

Value in the item0is1 room

Value in the item1is2 rooms

Value in the item2is3 rooms

Value in the item3is4 rooms

Value in the item4is5 rooms

Value in the item5is6 rooms

Value in the item6is7 rooms

Value in the item7is8 rooms

Value in the item8is9+ rooms

@Test

//3rd option :Select an item based on the actual value

list\_select\_value(xP, "9+ rooms");

}

//method3:

**public** **void** list\_select\_value(String xpath, String Text){

WebElement myV = myD.findElement(By.*xpath*(xpath));

java.util.List<WebElement> myL = myV.findElements(By.*tagName*("option")) ; //List is an array . when I use tag name, I can say the tag name of that elemenst are option

//The above line will list all the elements

System.*out*.println("size of items in drop down is" + myL.size());

**for**(**int** i=0; i<myL.size(); i++){

System.*out*.println("Value in the item" + i + "is" + myL.get(i).getText());

**if** (myL.get(i).getText().equalsIgnoreCase(Text)){

myL.get(i).click();

}

}

}

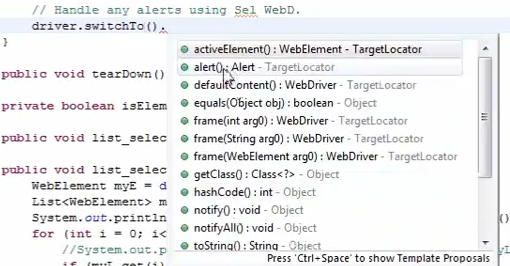
}

//it gives the items in the drop down

**How to Handle Alerts using the Selenium webdriver**

myD.switchTo() //swicths from the main browser to the pop up window

Switch to has more options such as below

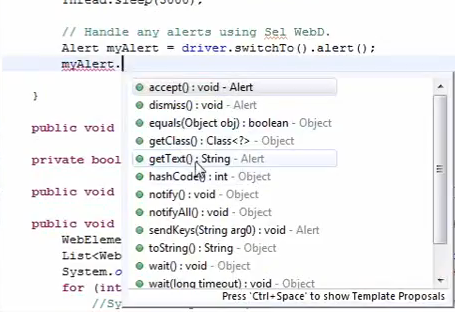


Switch to has different methods . Alert is a method, what it returns is of type alert

When I select switch to alert,, Im saying that lets switch to alert

myD.switchTo().alert()

* define my alert.



The most important ones in this is accept, dismiss and gettext

//Handle any alerts using Selenium Webdriver

Alert myAlert = myD.switchTo().alert(); //swicths to the pop up window. Alert is a method. it returns of type alert which is a class of an object

System.*out*.println(" ALert text is " + myAlert.getText()); //This should get the text from the box.

Thread.*sleep*(3000);

myAlert.Accept();

CTRL+ SHIFT + / - comments all

//Handle multiple browsers

//#set is very similar to arrays but slightly different.

// LIST ARRAY can have duplicate records

ArrayList myL = **new** ArrayList(); //List of type String, it creates an empty Array list and put it into myL

myL.add("Hello1");

myL.add("Hello2");

myL.add("Hello3");

myL.add("Hello3");

System.*out*.println ("myL size is" + myL.size()); //

//# Set Hashset - canNOT have duplicate records. It only adds unique values

Set<String> myS = **new** HashSet();

myS.add("Hello1");

myS.add("Hello2");

myS.add("Hello3");

myS.add("Hello3");

System.*out*.println ("myL size is" + myS.size()); //

}

Output :

myL size is4

myS size is3

//# Set Hashset - canNOT have duplicate records. It only adds unique values

//We use set becuase the windows handlers use set, becuase they want unique windows id to come.

Set<String> myS = **new** HashSet();

myS.add("Hello1");

myS.add("Hello2");

myS.add("Hello3");

myS.add("Hello3");

System.*out*.println ("myS size is" + myS.size()); //

//Iterator is lets go through one by one.Returns the iterator over the elements in this for different items and elements in the set. Into this myIT it is created a Array with all those values.

java.util.Iterator<String> myIt = myS.iterator();

/\* myIt.next(); //This gives a value of the next element of item in that

myIt.next(); // we can't use get method in hash code and we use iterator. It iterates i. We are starting with a value, go to the value and either incrementing/decrementing

myIt.next();\*/

**for** (**int** j=0; j<myS.size(); j++){

System.*out*.println("Set Item numnber" + j + "is" + myIt.next()); //This will print the value of the next iteration replacing the itreation.next each time

}

// The add method created only3. so if I say myIt.next it prints hello1

// If I say myIt.next again, it prints hello2

// If I again say myIt.next, it prints hello3

//If I put again, it says out of box or no value to put.

// So How do I then know where do I start and where do I end? I already know the size of it.

//if i

// If I list array number 3, it gives me 3.

// but for hashset I say iterate.next 1 time - it gives me first value, second iterate.next 2 value, and 3rf iterate 3rd value

// I can use it till the time I know i got to that point. How do I count? I will start with amother counter

}

O/P:

myL size is4

Item number0isHello1

Item number1isHello2

Item number2isHello3

Item number3isHello3

myS size is3

Set Item numnber0isHello1

Set Item numnber1isHello3

Set Item numnber2isHello2

Order is not retained for hash set.It is powerful asit doesn’t accept duplciates.

myD.getWindowHandle(); //This will returns all the handle for that specific browser and this point in time.

}

Windows Alerts:

@Test

**public** **void** testUntitled2() **throws** Exception {

myD.get(baseUrl);

//The following will do for 1 browser as it opens only 1. I will repeat the whole thing at the end where the windows pop up

Set<String> winId = myD.getWindowHandles(); //all that gets out of this gets converted to hashset and puts into set //If I want all the values, I will use iterartor.

java.util.Iterator<String> winIds = winId.iterator() ; //winid.iterator returns entire thing into winIds

// winIds.next(); //This will keep going to next one and getting printed

//If i want all the values

**for** (**int** i=0; i<winId.size(); i++){

System.*out*.println("Window ID for the browser#" + i + "is" + winIds.next());

}

String xP = "//\*[@id='rooms']";

list\_select\_value(xP, "9+ rooms");

Thread.*sleep*(2000);

//Handle any alerts

Alert myAlert = myD.switchTo().alert();

Thread.*sleep*(2000);

System.*out*.println("Alert text is " + myAlert.getText());

myAlert.accept();

Thread.*sleep*(2000);

//Windows handles when more than 1 browser is present. After we accepted the alert

winId = myD.getWindowHandles();

winIds = winId.iterator() ;

**for** (**int** i=0; i<winId.size(); i++){

System.*out*.println("Window ID for the browser#" + i + "is" + winIds.next());

}

myD.getWindowHandle();

// myD.getWindowHandle(); //This will get handles for all browsers which may have been open.

}

Output :

Window ID for the browser#0is{201cbf29-4bf1-4468-8928-45a80000c39d} //This is the unique identification for that browser

size of items in drop down is9

Value in the item0is1 room

Value in the item1is2 rooms

Value in the item2is3 rooms

Value in the item3is4 rooms

Value in the item4is5 rooms

Value in the item5is6 rooms

Value in the item6is7 rooms

Value in the item7is8 rooms

Value in the item8is9+ rooms

Alert text is We only support searching for up to eight rooms at once. Would you like to use our partner site Hotelplanner.com to search for multiple rooms?

Window ID for the browser#0is{201cbf29-4bf1-4468-8928-45a80000c39d}

Window ID for the browser#1is{cb9f6640-153a-4b46-90a3-0d9418011c83}