51. Update with latest Selenium Version - Selenium Manager (Beta)

System.set property step is optional.

this is an intermediate step which will help us to invoke the actual Chrome browser.

So by default, we don't have a permission to invoke Chrome browser and we are first invoking Chrome Driver third party plugin, which is eventually responsible to open our actual browser. So this is the step we are writing in our code. But what Selenium guys came up is this code they were in build handling in their jars itself in the

form of a new package called Selenium Manager.

If you add system.set Property in the code selenium manager feature will off automatically. If you not mention in the code then selenium manager will on and it takes from the selenium manager jars.

52. Handling Static dropdowns with Select webdriver API ?

<https://rahulshettyacademy.com/dropdownsPractise/>

Important UI components and see how to automate them using selenium Webdriver.

Selenium Automation

1. Handling Static Dropdowns
2. Handling Dynamic Dropdowns
3. Handling Checkboxes
4. Handling Radio buttons
5. Handling Text buttons
6. Handling Alerts-java popups
7. Selenium webdriver form methods

So if any dropdown have a tagname with select, then there is one special class in selenium to handle select dropdowns. Mostly “select” tagname dropdowns are nothing but static .

how would you know whether dropdown is static or not?

Just see if the tag name is select, then definitely that will be static, so options will not change.

Select dropdown = new select(staticdropdown);

I create an object(dropdown) for this class(select)

Select() – class argument

Select class, have all these methods. To call these methods, you need to create object of that class. Using object only you will be able to call the methods of select class.

If you are dropdown is multi select option, then you can go with all selected.

But do remember that all these methods will come to handy only when you are dropdown is having select tag.

If you don't have selected tag like this, then this options methods will not support.

these are the three different ways for you to select any option from the static dropdown text index and value.

WebDriver driver = **new** ChromeDriver();

driver.get("https://rahulshettyacademy.com/dropdownsPractise/");

WebElement staticDropdown = driver.findElement(By.*id*("ctl00\_mainContent\_DropDownListCurrency"));

Select dropdown = **new** Select(staticDropdown);

dropdown.selectByIndex(3);

System.***out***.println(dropdown.getFirstSelectedOption().getText());

dropdown.selectByVisibleText("AED");

System.***out***.println(dropdown.getFirstSelectedOption().getText());

dropdown.selectByValue("INR");

System.***out***.println(dropdown.getFirstSelectedOption().getText());

53. Updated lecture on latest Dropdown looping UI ( EX:Passengers field)

So when you have ID, you can blindly use this because ID is the unique element.

And it's more stable than XPath or CSS.

this while loop only stop when condition here becomes false and then it comes out of the loop and proceed with next steps.

working with while loop, you should initialize first(i=1) and then you have to compare (i<5)and put condition to increment(i++) it every time.

can we use for loop to repeat same test instead of using while loop?

The only difference between for and while loop is instead of writing that in pieces, you write everything in one single line. ( i=1;i<5;i++)

WebDriver driver = **new** ChromeDriver();

driver.get("https://rahulshettyacademy.com/dropdownsPractise/");

driver.findElement(By.*id*("divpaxinfo")).click();

Thread.*sleep*(2000L);

**int** i=1;

**while**(i<5)

{

driver.findElement(By.*id*("hrefIncAdt")).click(); //4 times

i++;

}

System.***out***.println(driver.findElement(By.*id*("divpaxinfo")).getText());

// while loop or for loop

**for**(i=1;i<4;i++)

{

driver.findElement(By.*id*("hrefIncAdt")).click();

}

driver.findElement(By.*id*("btnclosepaxoption")).click();

Assert.assertEquals(driver.findElement(By.id("divpaxinfo")).getText(), "5 Adult");

System.***out***.println(driver.findElement(By.*id*("divpaxinfo")).getText());

55. Handle Dynamic dropdowns with Webdriver API

56. Parent-Child relationship locator to Identify the objects Uniquely

options are loaded, so these are loaded based upon user actions.

That's why we call it as dynamic dropdowns.

System.setProperty("webdriver.chrome.driver", "C://work//chromedriver.exe");

WebDriver driver =new ChromeDriver();

driver.get("http://spicejet.com"); //URL in the browser

//  //a[@value='MAA']  - Xpath for chennai

//  //a[@value='BLR']

driver.findElement(By.id("ctl00\_mainContent\_ddl\_originStation1\_CTXT")).click();

driver.findElement(By.xpath("//a[@value='BLR']")).click();

Thread.sleep(2000);

We are giving[2] bcze in “from” and “to” both the field Chennai name is present that why we given as [2]

//driver.findElement(By.xpath("//a[@value='MAA'][2]")).click();

If the client don’t want to add index no in the while writing the code we use below this parent to chils traverse.

Parent xpath id entire source window got selected “CTNR” then add the child xpath

driver.findElement(By.xpath("//div[@id=”glsctl00\_mainContent\_ddl\_originStation1\_CTNR”] //a[@value='BLR']”)).click();

driver.findElement(By.xpath("//div[@id='glsctl00\_mainContent\_ddl\_destinationStation1\_CTNR'] //a[@value='MAA']")).click();

//select the current date in the Depart date field

driver.findElement(By.cssSelector(".ui-state-default.ui-state-highlight.ui-state-active")).click();

58. Handling AutoSuggestive dropdowns using Selenium

Ind based upon our input we will get suggestions but this is something that option comes based upon your input. That why we call it as autosuggestive dropdowns

select class to handle the dropdown, which already have options fixed.

So if your goal is to click India, then you will scan and write a condition that whenever there is a match with India, then go and click it.

So you should write one generic xpath, CSS, which should extract all the items present

in the list.

import java.util.List;

import org.openqa.selenium.By;

import org.openqa.selenium.Keys;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class AutoSuggestive {

public static void main(String[] args) throws InterruptedException {

// TODO Auto-generated method stub

System.setProperty("webdriver.chrome.driver", "C://chromedriver.exe");

WebDriver driver =new ChromeDriver();

driver.get("https://rahulshettyacademy.com/dropdownsPractise/");

driver.findElement(By.id("autosuggest")).sendKeys("ind");

Thread.sleep(3000);

//the written type of findelements is the list of webelement

List<WebElement> options =driver.findElements(By.cssSelector("li[class='ui-menu-item'] a"));

for(WebElement option :options)

{

if(option.getText().equalsIgnoreCase("India"))

{

option.click();

break;

}

}

}

}

60. Handling Checkbox and getting the size of them with Selenium

WebDriver driver = **new** ChromeDriver();

driver.get("https://rahulshettyacademy.com/dropdownsPractise/");

//\* - star means any id which is matches with the SeniorCitizenDiscount

System.***out***.println(driver.findElement(By.*cssSelector*("input[id\*='SeniorCitizenDiscount']")).isSelected()); // first we will get false

driver.findElement(By.*cssSelector*("input[id\*='SeniorCitizenDiscount']")).click();

System.***out***.println(driver.findElement(By.*cssSelector*("input[id\*='SeniorCitizenDiscount']")).isSelected()); // now we will get true bcze it is checked

//How do you know the number of checkboxes present in the page

driver.findElements(By.*cssSelector*("input[type='checkbox']")).size();

61. Importance of Assertions in Automation testing and how to use them.

TestNG is one Testing framework, which we will use when we are building automation framework using selenium, we will use this library to build robust utilities with selenium framework.

Asseritons which will help us to validate if the result, whatever selenium return is expected or not. If not, it will fail. Asseritons concept is brought by TestNG.

If you want to use assertion in your project first we need to download the TestNG jars in MVN repo.

Then we need to add those jars in our project

Project name>right click>properties>java build path>libraries>click on add external jars>add open>okay – this jar is added to this project. Now we are able to apply assertions.

Assertions - it will help us to validate your results, which are being

written by Selenium with your expect result.

If you start with Assert dot you will get all the methods

assert equal is one more method which actually compair with two arguments.

One argument is what you are. Selenium script is writaning and one argument is what actually you are expecting.

 if they are equal, then test will pass.They are not equal.Test will fail.

3 asserts are important – assertequal,asserttrue,assertfalse

Assert.*assertFalse*(driver.findElement(By.*cssSelector*("input[id\*='SeniorCitizenDiscount']")).isSelected());

Assert.*assertTrue*(driver.findElement(By.*cssSelector*("input[id\*='SeniorCitizenDiscount']")).isSelected());

Assert.*assertEquals*(driver.findElement(By.*id*("divpaxinfo")).getText(),"6 Adult");

62. Assignment

1.Check the first  Checkbox and verify if it is successfully checked and Uncheck it again to verify if it is successfully Unchecked

1. WebDriver driver = new ChromeDriver();
2. driver.get("https://rahulshettyacademy.com/AutomationPractice/"); driver.findElement(By.id("checkBoxOption1")).click();
3. System.out.println(driver.findElement(By.id("checkBoxOption1")).isSelected()); driver.findElement(By.id("checkBoxOption1")).click();
4. System.out.println(driver.findElement(By.id("checkBoxOption1")).isSelected());

2.How to get the Count of number of check boxes present in the page

System.***out***.println(driver.findElements(By.*cssSelector*("input[type='checkbox']")).size());

63. Handling Calendar UI in travel websites using Selenium

if you want to turn class name to CSS, you need to put dot before the class name so that it will be CSS.

If you want turn ID to CSS then we have to give # before your ID

….there should not be any spaces between the classname. Whenever there are spaces, you have to remove that and provide dot.

Below is the current date path ( this is in dynamic dropdown in class eclipse )

driver.findElement(By.*cssSelector*(".ui-state-default.ui-state-highlight.ui-state-hover")).click();

64. Validating if UI Elements are disabled or enabled with Attributes

we need to understand what is the attribute(style) which is getting changed in the middle when something is getting disabled. ( select the one way – 0.5 round trip – 1 you can see the diff in the html)

And we will pull that attribute using that attribute method and see the name of that attribute, how it is before disabled and after disabled It should change right, based upon your disable or enable.

When something got disabled or some color got changed,developer has to set some attribute to that HTML to make it look like disabled.

Contains(“1”) in java method where it will scan entire string and it will check if the character(1)what you provided is inside this or not

//click on round trip then only the return date field should be enable

///Driver.findElement(By.id(“ctl00\_mainContent\_rbtnl\_Trip\_1”).click();

//entire return type date field window name element

//if its enabled it return true if it disabled it retruns false ( .isEnabled) (.isDisabled) – sometime it is work some times it won’t work

System.out.println(driver.findElement(By.name("ctl00$mainContent$view\_date2")).isEnabled());

//select the one way – 0.5 round trip – 1 you can see the diff in the html

System.out.println(driver.findElement(By.id("Div1")).getAttribute("style"));

driver.findElement(By.id("ctl00\_mainContent\_rbtnl\_Trip\_1")).click();

System.out.println(driver.findElement(By.id("Div1")).getAttribute("style"));

if(driver.findElement(By.id("Div1")).getAttribute("style").contains("1"))

{

System.out.println("its enabled");

Assert.assertTrue(true);

}

else

{

Assert.assertTrue(false);

}

65.Assignment 2

[ProtoCommerce (rahulshettyacademy.com)](https://rahulshettyacademy.com/angularpractice/)

WebDriver driver = **new** ChromeDriver();

driver.get("https://rahulshettyacademy.com/angularpractice/");

driver.findElement(By.*name*("name")).sendKeys("Hari Sankar");

driver.findElement(By.*name*("email")).sendKeys("harisankar@gamil.com");

driver.findElement(By.*id*("exampleInputPassword1")).sendKeys("john777");

driver.findElement(By.*id*("exampleCheck1")).click();

WebElement staticDropdown = driver.findElement(By.*id*("exampleFormControlSelect1"));

Select dropdown = **new** Select(staticDropdown);

driver.findElement(By.*id*("exampleFormControlSelect1")).click();

dropdown.selectByIndex(0);

System.***out***.println(dropdown.getFirstSelectedOption().getText());

driver.findElement(By.*id*("inlineRadio1")).click();

driver.findElement(By.*name*("bday")).sendKeys("07/23/2010");

driver.findElement(By.*className*("btn-success")).click();

System.***out***.println(driver.findElement(By.*className*("alert-dismissible")).getText());

66. End to End Automation using all UI Elements with selenium

public class e2e {

public static void main(String[] args) throws InterruptedException {

// TODO Auto-generated method stub

System.setProperty("webdriver.chrome.driver", "C://work//chromedriver.exe");

WebDriver driver =new ChromeDriver();

driver.get("http://spicejet.com"); //URL in the browser

driver.findElement(By.id("ctl00\_mainContent\_rbtnl\_Trip\_0")).click();

driver.findElement(By.id("ctl00\_mainContent\_ddl\_originStation1\_CTXT")).click();

driver.findElement(By.xpath("//a[@value='DEL']")).click();

Thread.sleep(2000);

driver.findElement(By.xpath("//div[@id='glsctl00\_mainContent\_ddl\_destinationStation1\_CTNR'] //a[@value='MAA']")).click();

driver.findElement(By.cssSelector(".ui-state-default.ui-state-highlight.ui-state-active")).click();

if(driver.findElement(By.id("Div1")).getAttribute("style").contains("0.5"))

{

System.out.println("its disabled");

Assert.assertTrue(true);

}

else

{

Assert.assertTrue(false);

}

driver.findElement(By.cssSelector("input[id\*='SeniorCitizenDiscount']")).click();

driver.findElement(By.id("divpaxinfo")).click(); // passengers field

   Thread.sleep(2000L);

for(int i=1;i<5;i++)

{

driver.findElement(By.id("hrefIncAdt")).click(); // + icon in adults

}

driver.findElement(By.id("btnclosepaxoption")).click();

Assert.assertEquals(driver.findElement(By.id("divpaxinfo")).getText(), "5 Adult");

System.out.println(driver.findElement(By.id("divpaxinfo")).getText());

// driver.findElement(By.cssSelector("#ctl00\_mainContent\_btn\_FindFlights")).click();

driver.findElement(By.cssSelector("input[value='Search']")).click();

// driver.findElement(By.xpath("//input[@value='Search']")).click();

// driver.findElement(By.name("ctl00$mainContent$btn\_FindFlights")).click();

}

}

68. Handling Java Alerts using Selenium Webdriver

String text = "Hari";

WebDriver driver =**new** ChromeDriver();

driver.get("https://rahulshettyacademy.com/AutomationPractice/");

driver.findElement(By.*id*("name")).sendKeys(text);

driver.findElement(By.*id*("alertbtn")).click();

System.***out***.println(driver.switchTo().alert().getText());

driver.switchTo().alert().accept();

driver.findElement(By.*id*("confirmbtn")).click();

System.***out***.println(driver.switchTo().alert().getText());

driver.switchTo().alert().dismiss();