What is Automation?

1.Converting Manual test case into automation test script by using some automation tool is called as Automation.

Why we go for Automation?

1.To maintain the Accuracy in Repetitive Task

2.To reduce the time for testing the Repetitive Task

3.To reduce the Human effort

When we go for Automation?

1.Once the Application is Stable after the First Release

What are the two types of Application?

1.Stable Application

2.Unstable Application.

What are all the Test Cases we are unable to Automate?

1.Otp and Captcha related test cases

2.MP3 and MP4 related test cases

3.Printed Related test cases

4.Bar code and Gaming related test cases

Is 100 % automation is Possible? - Yes (expect the Above scenarios')

1.What is Selenium?

1.Selenium is a free and open-source web application automation tool

2.Functional Testing application Tool.

2.From where we can download the selenium for free?

https://www.selenium.dev/downloads/

3.What Is the Stable version of Selenium?

3.141.59

4.What are the Flavors of Selenium?

1.Selenium Core,

2. Selenium IDE (Record & Play),

3.Selenium RC (Remote Control),

4.Selenium Web driver,

5.Selendroid (Selenium Grid),

6.Appium,

7.Winium.

5.What are the Browser's supported by the Selenium Tool?

1.Google Chrome,

2. Internet Explorer,

3.Microsoft Edge,

4.Safari,

5.Opera,

6.Mozilla Firefox etc...

6.What are all the Languages which will be getting supported by Selenium?

1.Java,

2.C#,

3.Ruby

4.Pearl,

5.Python,

6.PHP,

7.Dart

8. Java Script,

9.Objective C,

10.Node JS,

11.Haskell,

12.R,

13.TCL,

14.Exilit,

7.Using Selenium what type of Application can we Automate?

Only web Applications

8.Important Java topics required for Selenium

1.Class and Objects.

2.Static and Non-Static Methods.

3.Conditional Statements like if and if else

4.Looping Statements like for, for each, while, do while.

5.Oops Concept like Inheritance, Polymorphism, Encapsulation, Abstraction.

6.Collection Framework like List, Set and Maps.

7.Exception Handling.

8.Methods of the String Class.

9.File Handling.

9.Required Software's?

1.JDK (Java Development Kit)

2.Eclipse, Browser (chrome)

3.Selenium Jar Files

4.Driver Executables (chrome driver .exe)

5.Any Web Application.

10.Is Selenium is an Open Source? How?

Yes, it's is an open-source tool, it means we can across the source code of selenium software itself, (we can download customize as per our requirement)

11.What are all the OS (Operating System) supported by Selenium?

1.Windows

2.Linux

3.Mac

4.other Unix flavor's like Android, IOS, Ubuntu…etc.

12.What are all the OS (Operating System) not supported by Selenium?

Unix Operating System is not supported by Selenium.

13.Can we do performance testing using Selenium?

No, but we can integrate selenium with the JMeter.

14.What type of test cases we can automate?

Regression Testing

15.Do we automate Integration Testing?

Yes, Different type of the Test cases which is the part of the regression.

16.Do we automate the Negative test cases?

Yes

17.Which test Cases are automated First?

Smoke Test Cases (Sanity, Dry Run (Automation)), Build Verification Testing (BVT), Skim (UAT).

18.Is 100 % automation is possible? If no why?

No, because we don't have technology to automate the following features or it may.

1.3D Games

2.Verification of Audio and Video files

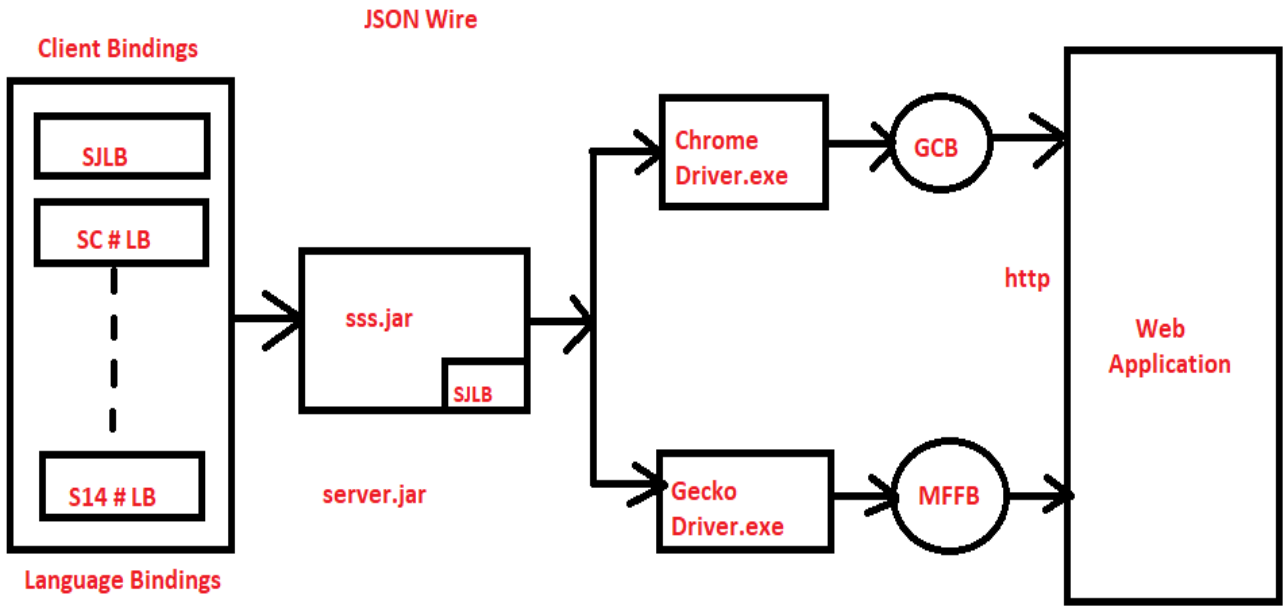
3.Capturing the Attendance using the Access cards & Biometric Scanners, Entering the product details using the barcode scanner, payment through credit card swiping.

4.Captcha (Completely Automated Public Turing test to tell Computers & Humans Apart).

19.Which is browser & OS supported by the Selenium IDE?

Firefox and Windows.

Selenium Architecture



1.Selenium supports 14 different coding languages like Java, Python, Ruby etc. They are called as Language Binding or Client Binding.

2.Three Client Library communicates with the Selenium stand Alone. Then the server will perform the Action on the browser with the help of Driver Executables.Inorder to perform the Action it uses JSON Wire Protocol (Java Script Object Notation).

3.Selenium server internally contains Selenium Java Client binding also, hence while installing Selenium we use only the Server.jar file and Driver Executables.

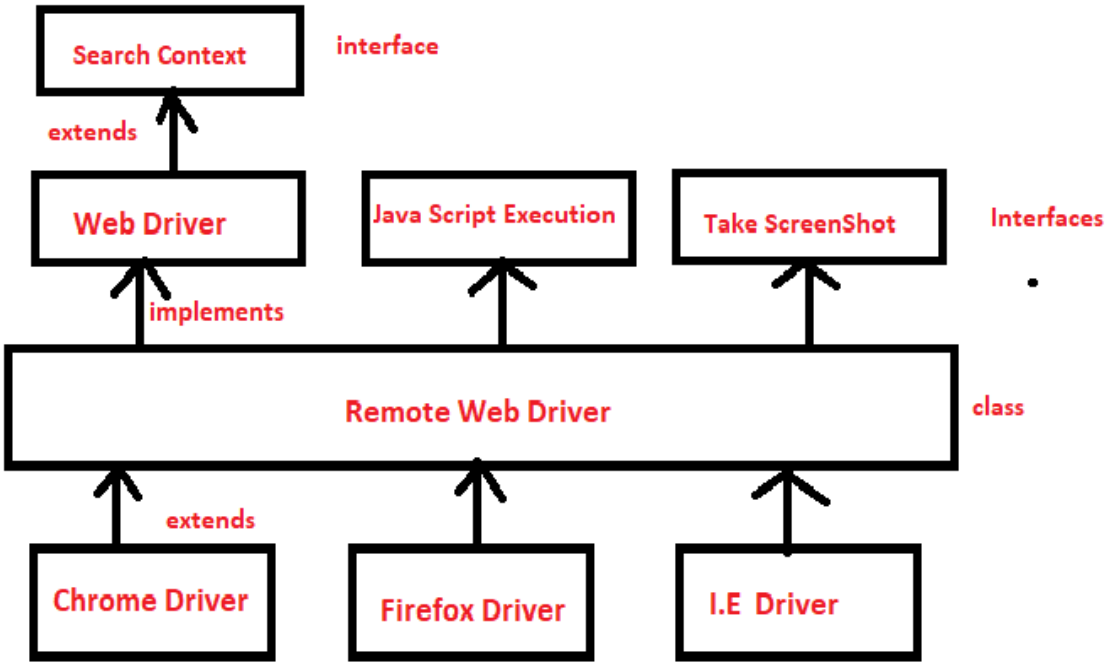
Architecture of Selenium Web Driver

What is Selenium Web driver? (or)

Why we are using Web driver?

1.Selenium Web Driver is a tool is used to automate the web applications for testing.

2.It supports many browsers such as Chrome, Firefox, Safari and Internet Explorer.



1.The Super most Interface is Search Context, which is Extended by the Web driver interface.

2.Web driver interface is implemented in Remote Web driver class.

3.Remote Web driver class implements other interfaces also such as Java Script Executor, Take Screenshot etc.

4.All the browsers specified as classes such as Chrome Driver, Firefox Driver, Internet Explorer Driver etc. extends to Remote Web driver Class.

How to Install Selenium?

1.Before installing selenium (Eclipse tool) check whether JDK has been installed or Not

If not Download the JDK through - https://www.oracle.com/java/technologies/downloads/

2.Then install the Automation Tool (Eclipse)

https://www.eclipse.org/downloads

Note: Apply Filter – Eclipse will display methods of Object class also we have press control button after any variable. To hide it go to the Window tab in eclipse →preference→java →appearances →type filters. Click on the Add button & type java. lang. object then Click Ok & click on apply and close.

20.How Selenium performs action on the Browser?

By Calling the native methods of the Browser.

21.Which protocol is used by selenium tool to interact / communicate with the Browser?

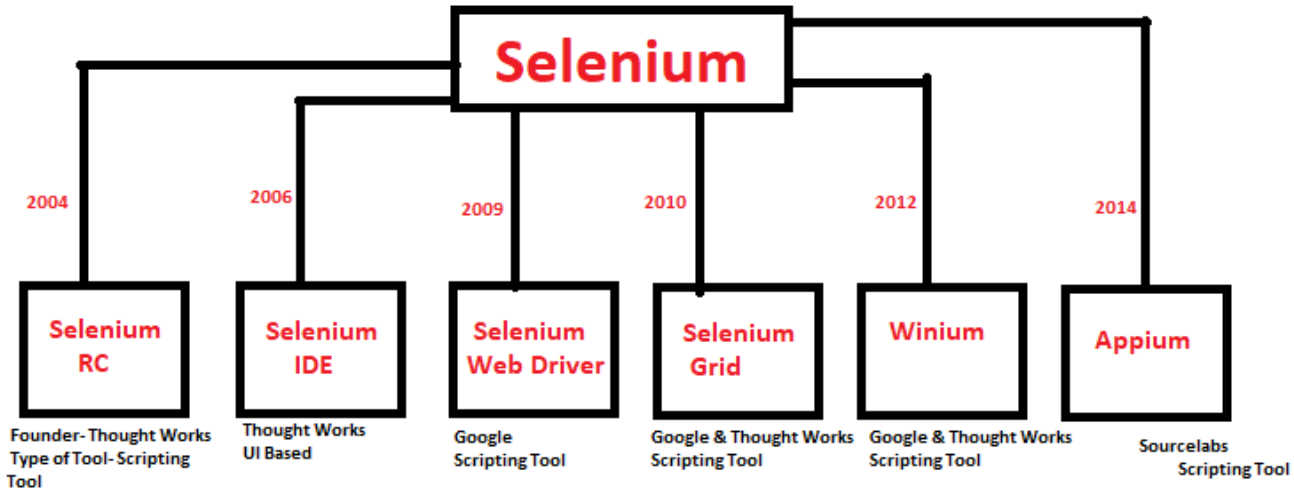
JSON Wire Protocol (Java Script Object Notation).

History of Selenium

22.Types of Automation tool?

1)QTP – Quick Test Professional – Paid - UI Based Automation Tool

2)Selenium – Open Source – Script based Automation tool.



1.Selenium RC: (Remote Control)

1.It is the first ever automation tool connected under the Selenium Community, it is a script-based tool to automate the Web Application.

2.It supports any language like Java, Python, Ruby, Pearl, etc.…

3.It supports any Operating system like Linux, Windows, Mac, etc.…

4.It supports any browsers like Chrome, Firefox, Edge, Opera, Safari etc.…

Limitations & Disadvantages

1.A separate RC server compulsory has installed it takes huge memory due to this RC will lagger the performance.

2.It can't support secured web application

3.Maintenance of Browser version is difficult

4.It can't support Desktop Application

2.Selenium IDE:

1.It is one of the Automation tools under selenium community and it is an open-source UI based automation tool.

Advantages

1.Automation script creation is very easy

2.non-technical persons can also perform automation

3.Batch Execution is very easy

Disadvantages

1.Can't able to automate the complex scenario's

2.There is no reusability of Common Scenario's

3.Supprots only Chrome and Firefox browser's

4.Data Driven Testing is not possible.

3.Selenium GRID:

1.Selenium Grid is an automation tool which supports Remote execution

2.Through Selenium Grid we can run Web driver script in multiple sub machines

3.Such kind of Execution is considered as Hub node Execution.

4.Winium

1.It is an automation tool or Windows Desktop Application and it is a script-based automation tool

2.Nowdays Winium is not in use with the Web driver automation because web driver is compatable with other 3rd party desktop automation tools to handle desktop scenarios.

3.This 3rd party tools are open source and much easier.

4.Some 3rd party tools for desktop applications are

a] AUTO IT

b] SIKULI

5.Appium

1.It is open-source automation tool for the Mobile Applications

2.Its support's both android and IOS type of Applications.

6.Selenium WebDriver

1.It is a most advanced highly efficient automation tool for Web Applications

2.It is invented by Google on 2009

3.This tool will support any programming language and any OS

4. This tool will support any browser

5.It supports secured & non secured both the Applications (http & https)

6.It supports other 3rd party Desktop Automation tool which Web driver is compatible with it.

7.No need of Separate RC server to run.

8.Data driven testing and the Framework creation is possible.

Disadvantages

1.It will not support Desktop Automation tool

2.This drawback is already overcome by providing compatibility other 3rd party tools.

Methods of Search Context:

1.find Element (By arg [ ]) = Web element

2.find Elements (By arg [ ]) = List <Web Element>

Take Screenshot Methods:

1.getScreenshotAs ()**.**

Web driver Methods: (No of Web driver Methods – 11)

1. Close ()
2.get ()
3.getCurrentUrl ()
4.getPagesource( )
5.getTitle ()
6.getWindowHandle( )
7.getWindowHandles ()
8. manage ()
9.navigate ()
10.quit()
11.switchto ()

Java Script Executor Method

1.ExecuteAsyncScript ()

2.executescript ()

Web Element Methods: (No of Web driver Methods – 14)

1)Clear ()
2) Click ()
3)Get Attribute ()
4) Get CSS Value ()
5)get location ()
6) Get Rect ()→x
7)get Size ()
8) get Tag name ()
9)get Text ()
10) is Displayed ()
11)is Enabled ()
12) is Selected ()
13)Send keys ()
14) submit ()

Installation of Eclipse

1.Create a new Java Project called Automation & create two new folders called Jar and Driver

2.Go to the Website (selenium.dev →Click on Downloads→Then click on the Stable Version→It will download the Jar file).

3.Scrool down until find Browser→ Click on the Chrome Driver → Download the Chromedriver32 bit (check the browser version and download)

4.Extract the driver file and copy and paste it in the Driver folder created in the Eclipse & same do the jar file as well

5.Then do right on the Jar folder and add the new Build path to the jar folder.

Demo Program

```
package qsp;

import org.openqa.selenium.chrome.ChromeDriver;

public class Demo {

    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");

        ChromeDriver driver = new ChromeDriver();
        driver.get("https://www.google.com/");
        driver.close();
    }

}
```

What is Upcasting? Why it is used in Selenium?

1.Converting Sub class object into supertype is called as upcasting

Example: - WebDriver driver = new ChromeDriver ();

In selenium we use upcasting so that we can execute same script on any browser.

Explain WebDriver driver = new ChromeDriver ();

Webdriver is an Interface.

Driver is a reference Variable.

= is an assignment operator.

New is a keyword to create an object

ChromeDriver () is a constructor to initialize the Object

; is the statement delimiter or used to end the Statement.

How do you enter url without using get method?

Using navigate ().to () method

driver. Navigate().to("http://www.google.com");

What is the difference between get () & navigate ()?

By using get () we can enter the url only, whereas by using navigate () we can enter the url, click on back, forward & refresh functions.

What is the difference between get () & to ()?

There is no difference between get and to method. To method is internally calls get method, hence there is no difference between to & get method. Both are used to enter the url.

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class NavigateClass {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
    }
    public static void main(String[] args) throws InterruptedException {

        WebDriver driver = new ChromeDriver();
        driver.navigate().to("https://www.google.com/");
        Thread.sleep(3000);
        driver.get("https://www.google.com/");
        Thread.sleep(3000);
        driver.navigate().refresh();
        Thread.sleep(3000);
        driver.close();
}

}
```


How to close the browser without using close method?

By using quit ().

Ex: driver. quit ()

What is the difference between close () & quit ()?

Close () closed the Current Browser whereas quit () closes all the browsers.

Write a script to Open Google.com & verify that title is google and also verify that it is redirected to google.co.in ?

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Demo1 {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
    }
        public static void main(String[] args) {
            WebDriver driver = new ChromeDriver();
            driver.get("https://www.google.com/");
            String title = driver.getTitle();
            if (title.equals("Google")) {
                System.out.println("Pass : Title is Google ");
            }
            else {
                System.out.println("Fail : Title is Not Google ");
            }
            String url = driver.getCurrentUrl();
            if (url.contains("google.co.in")) {
                System.out.println("Pass : Url has co.in");
            }
            else {
                System.out.println("Fail : Url doesn't have co.in");
            }
        }

}
```


Write a script to delete all the cookies present in the browser and maximize the browser?

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class CookiesAndMaximize {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
    }
        public static void main(String[] args) {
            WebDriver driver = new ChromeDriver();
            driver.manage().deleteAllCookies();
            driver.manage().window().maximize();
            driver.close();
    }

}
```

Write a script to print html source code (script) of the Webpage?

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Htmlcode {
static {
    System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.google.com/");
        String htmlcode = driver.getPageSource();
        System.out.println(htmlcode);
        driver.close();

    }

}
```

Web Element

1.Anything present on the Webpage is called WebElement.Webelements are created using HTML Language (Hyper Text Markup Language)

2.Each HTML Elements contains three things

"Tags", "Attribute" & "Text".

Example: HTML Code of a Login Page

Login

In the Above Html code,

div is the Tag,

id is the attribute name

d1 is the Attribute Value

Login is the Visible Text

In order to see the HTML code of the required element right click on that element& select the option 'Inspect', if Right click is disabled then

Ctrl+Shift+i , or press F12,it will display the developer's tool bar.

In order to inspect another element, click on inspect button & then click on required element.

In selenium, before performing any action such as clicking, typing, selecting etc. we should find the elements by using Locators.

Locators

1.Locators are used to find the Elements. In selenium there are 8 types of locators & all of them are static method present in 'By' class.

2.By is an Abstract Class

Syntax: - By. tagname ()

1.tagName ()

2.id ()

3.name ()

4.className ()

5.Linktext ()

6.Partial LinkText ()

7.Css Selector ()

8.Xpath ()

1.Absolute Xpath

2.Relative Xpath

What is the Return type of find Element?

Web element

What is the argument accepted by find Element ()?

By type

If specified locator is not matching with any of the Elements, then what find element does?

It throws No such Element Exception

If the specified locator is matching with the Multiple Elements what find Element does?

It will take the first available element & it returns the address of First matching Element.

Write a program to Click google link elements by using Tagname, id, Name, Class Name?

Link text & Partial Link Text

1.Both Link text and Partial Link Text locators are can be used to find the link. If we try on any other type of element, we get No such element exception

Ex: -driver. findelement (By. link text("Google")). Click ();

If the text of the Link is changing, Partially link text.

Ex:- HTML Code of a Link is

<a…>Inbox (25)

Selenium code

driver. findelement (By. PartialLinkText("Inbox")). Click ();

Limitations of Partial Link Text

1.Elements should be a link

Ex:Inbox (25) //Cannot used

2.Text should be partially Changing

Ex:30 //we cannot use partial link text

CSS Selector (Cascading Style Sheet)

1.Css selector is one of the locators in Selenium

Syntax: - tag [Attribute Name = 'Attribute Value ']

Login

Ex: -div[id='d1']

Limitations of CSS Selector

We can check Css expression in the browser by using the following steps

1.Inspect any element

2.Press CTRL+F

3.Type the above Expression

1 of 1 →One Matching Element

1 of 3 → Multiple Matching element

0 of 0 → No matching Element

Note: If we do any syntax mistake while writing the CSS expression ,then we get Invalid Selector Exception

Xpath

1.It is the path of the Element in the HTML Tree is called as Xpath. While writing Xpath it starts with "." Which also represents Current webpage on HTML Document using "." Is not mandatory.

2.To navigate from Parent element to child element we use single "/"

Xpath Expression: - . /html / body / a (or) /html / body / a

Xpath in Selenium: - driver.findElement(By.*xpath*("./html/body/a")).click();The Above XPath are called as Absolute Xpath.

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Xpath {
static {
    System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver = new ChromeDriver();
        driver.get("file:///D:/Desktop/Yahoo.html");
        driver.findElement(By.xpath("./html/body/a")).click();
        }
}
```

Types of XPath

1.Absolute Xpath

2.Relative Xpath

1.Xpath by Attribute

2.Xpath by text () function

3.Xpath by contains () function

4.Traversing in Xpath

5.Independent-Dependent Xpath

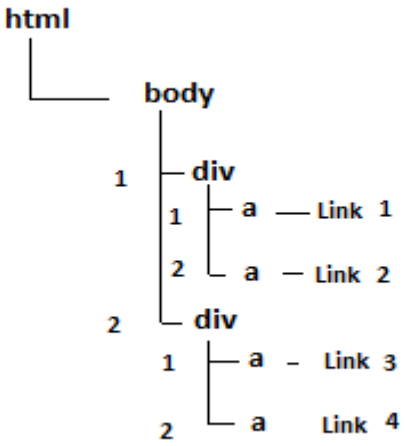
6.Xpath by Group Index

1.Absolute Xpath

Starting from the Html Tag to the required or desired element is called as Absolute Xpath.

Sample HTML Code

```
<html>
<body>
<div>
<a href=https://www.swiggy.com>Swiggy</a><br>
<a href=https://www.zomato.com>Zomato</a><br>
</div>
<a href=https://www.flipkart.com>Flipkart</a><br>
<a href=https://www.amazon.com>Amazon</a><br>
</div>
</body>
</html>
```



Link1 → /html/body/div[1]/a[1]

Link3 → /html/body/div[2]/a [1]

Link2 → /html/body/div[1]/a[2]

Link4 → /html/body/div[2]/a[2]

Link 1,2,3,4 → /html/body/div/a

Link 1&3 → /html/body/div/a[1]

Link 2 & 4 → /html/body/div/a[2]

Link 1&2 → /html/body/div[1]/a

Link 3&4 → /html/body/div[2]/a

Question Scenario?

1.Open the browser

2.Enter the Url

3.Click on the Zomato webpage

4.Maximize the browser

5.Close the Browser

```java
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class AbsoluteXpath {
static {
    System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver=new ChromeDriver();
        driver.get("file:///D:/Desktop/Xpath.html");
        driver.findElement(By.xpath("/html/body/div[1]/a[2]")).click();
        driver.manage().window().maximize();
        driver.close();

    }

}
```

Xpath by Index

1.In Xpath, we can use Index which starts from 1,if there is another element under the same parent with the same tag then the index become2 and so on.

Relative Xpath

/ → child

// → descendent

Link 1 → //div[1]/a[1]

Link 2 → //div[1]/a[2]

Link 3 → //div[2]/a[1]

Link 4 → //div[2]/a[2]

Link 1&2 → //div[1]/a

Link 2&4 → //div/a[2] or //a[2]

Link 1,2,3,4 → //a

Link 3&4 → //div[2]/a

In relative Xpath, if starts with "// " which represents descendent(Child, Grandchild, Great grandchild…etc.)

Assignment?

1.Create the table ?--> //table//a

2.Write a Xpath for an Image ?

What is the Difference between //a & //table //a ?

// a matches with all the links which are present in any webpage

//table//a matches all the link inside the table.

XPath with Attribute

We can also use the Attribute of the element while writing the XPath.

The syntax is //tag [@Attribute Name='Attribute Value]

Syntax ==>1) //tag[@An='AV']

➔ //tag[@An1='AV1']

➔ //tag[@An2='AV2']

Examples: - 1) //input[@type='text']

2) //input[@id='twotabsearchtextbox']

XPath with Text Function[text()]

In Xpath we can also use text (visible text)

Syntax is: //tag[text()='textvalue']

Html code: //div[text()='Login ']

Xpath: //div[text()='Login ']

Xpath with Contains

If text value is partially changing, we can use Contains () function on the Xpath because in this type, version or value changes as an when new version releases.

Usually if you see any number element in visible text it usually dynamic in nature, it is likely to change in future

Syntax: //tag[contains(text(),'value')]

Example:

→//a[contains(text(),'Inbox')]

→//a[contains(text(),'New meeting')]

How do you identify the element without using Partial Link Text?

By using Xpath with the Contains function

Example: html code:- <a>Inbox(25)

Xpath://a[contains(text(),'Inbox')]

Note:

(For Links) We can use contains function for attribute also the syntax is

Syntax://tag [contains(@Attribute Name,'Attribute Value')]

Example://img[contains(@src,'logo')]

Xpath with Traversing

Navigating from one element to another element is called as traversing. There are two types of Traversing.

1.Forward Traversing

Navigating from Parent element to child element is called as Forward Traversing.

2.Backward Traversing

Navigating from the Child element to parent Element is called Backward Traversing

HTML Code:

```
<table border="1">
<tr>
<td>java</td>
<td>100</td>
</tr>
<tr>
<td>Selenium</td>
<td>200</td>
</tr>
</table>
```

Example for forward traversing {From html code to Selenium}

Absolute Xpath →/html/body/table/tbody/tr[2]/td[1]

Relative Xpath → //td[.='Selenium']

Example for backward traversing {From Selenium code to HTML}

//td[.='Selenium']/../../../..

(/.. will move the cursor up to one level same as cmd.cd.:)

Independent Dependent Xpath

If the value is changing completely then we handle it using Independent Dependent Xpath

Here we use the Xpath using backward & forward traversing element

We always start from Independent element and ends with Dependent

Example: -1} //td[.='Selenium']/../td[2]

2} //td[.='java']/../td[2]

Steps to Construct html tree and write the Xpath

1.Identify the Independent and Dependent element

2.Inspect Independent Element & place the mouse pointer on the source of the Independent Element.

3.Note down the Html Code of the Independent Element

4.Move the Mouse pointer in upward direction in step by step till it highlights both independent and dependent element, till it will common parent

5.Add the path Common parent above the code independent Element

6.Use down arrow key to navigate from common parent to dependent Element

7.Add its path below the Common Parent.

Xpath By Group Index

When writing Xpath expression we can write the expression within the brace & we can specify the index outside the brace is called as Group Index.

Syntax:- (//tagname)[Index-Position]

Example:- (//td)[4]

(//a)[3]

Assignment

1.Write an Xpath to identify the Price of 1st T-shirt present in Myntra.com?

2. Write an Xpath to identify the image of HRK T-shirt present in Myntra.co/m?

3. Write an Xpath to identify the Price of Women Khushal dress present in Myntra.com?

4.Write an Xpath to identify the image of Bonkids T-shirt in Kids section present in Myntra.com?

5.Write an Xpath to identify Oneplus7t mobile phone present in amazon.in?

Question

1.Write a script to remove the text function present in email text box of Opensourcebilling.org?

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Removetext {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
    }

    public static void main(String[] args) {

        WebDriver driver = new ChromeDriver();
        driver.get("http://demo.opensourcebilling.org/");
        driver.findElement(By.id("email")).clear();
    }

}
```

2.Write a script to print height and width of email text box present in Facebook?

```
import org.openqa.selenium.By;

public class PrintHeightandWidth {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
    }

    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.facebook.com/");
        WebElement Emailtextbox = driver.findElement(By.id("email"));
        int Height = Emailtextbox.getSize().getHeight();
        int Width  = Emailtextbox.getSize().getWidth();
        System.out.println("Height" +Height);
        System.out.println("Width" +Width);
        driver.close();
    }

}
```

3.Write a script to check whether the Facebook logo is displayed or Not in the Facebook login page?

```
import org.openqa.selenium.By;

public class LogoisDisplayed {

static {
    System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
}

    public static void main(String[] args) {
        WebDriver driver =new ChromeDriver();
        driver.get("https://www.facebook.com/");
        boolean Image = driver.findElement(By.xpath("//img[contains(@class,'fb')]")).isDisplayed();
        if(Image==true)
        {
            System.out.println("Logo is Displayed and so Pass");
                }
        else
        {
            System.out.println("Logo is Not displayed and so Fail");
        }
        driver.close();
    }
}
```

3.Write a script to check whether the Facebook login button is displayed or Not in the Facebook login page?

```
public class LoginButtonisDisplayed
{static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}

    public static void main(String[] args) throws InterruptedException {
         WebDriver driver =new ChromeDriver();
         driver.get("https://www.facebook.com/");
         Thread.sleep(3000);
         boolean Loginbutton = driver.findElement(By.name("login")).isEnabled();
         if(Loginbutton==true)
         {
             System.out.println("Login Button is Enabled and so True");
         }
         else
         {
             System.out.println("Login Button is not Enabled and so Fail");
         }
         driver.close();

    }

}
```

Write a script to check whether checkbox is selected or not for Actitime Application?

```
public class CheckboxSelect {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        boolean checkbox = driver.findElement(By.id("keepLoggedInCheckBox")).isSelected();
        if(checkbox==true)
        {
            System.out.println("Checkbox is Selected");
        }
        else
        {
            System.out.println("Checkbox is not Selected");
        }
        driver.close();
    }
}
```

Write a script to login to the Actitime Application

1.Open the Browser

2.Enter the URL→ https://demo.actitime.com/

3.Enter the Username and Password→ Username: admin, Password: manager

4.Click on the Login Button

5.Close the Browser

```
public class ActiimeLogin {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
public static void main(String[] args) throws InterruptedException {
    WebDriver driver = new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    driver.findElement(By.id("username")).sendKeys("admin");
    Thread.sleep(3000);
    driver.findElement(By.name("pwd")).sendKeys("manager");
    Thread.sleep(3000);
    driver.findElement(By.xpath("//div[text()='Login ']")).click();
    Thread.sleep(10000);
    driver.close();
    }
}
```

Write a script check whether width of the username and password text box are equal or not?

```
public class DimensionWidth {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
public static void main(String[] args) {
    WebDriver driver =new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    int UserNameTextboxWidth = driver.findElement(By.id("username")).getSize().getWidth();
    int PasswordTextboxWidth = driver.findElement(By.name("pwd")).getSize().getWidth();
    if(UserNameTextboxWidth==PasswordTextboxWidth)
    {
        System.out.println("Width of both are equal");
    }
    else
    {
        System.out.println("Width of both are not equal");
    }
    driver.close();
    }
}
```

Write a script check whether width of the username and password text box are aligned properly or not?

```
public class Alignment {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        int L1 = driver.findElement(By.id("username")).getLocation().getX();
        int L2 = driver.findElement(By.name("pwd")).getLocation().getX();
        if(L1==L2)
        {
            System.out.println("Both are equal and Aligned");
        }
        else
        {
            System.out.println("Both are not equal and Not Aligned");
        }
        driver.close();
    }
}
```

Write a script to click on the submit present in open-source billing application without using Click method?

1.Example: -

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Withoutclick {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver=new ChromeDriver();
        driver.get("http://demo.opensourcebilling.org/");
        driver.findElement(By.id("user_login_btn")).submit();
        driver.close();

    }

}
```

Note: Submit method will work on when the type=" Submit "attribute is present.

2.Example: -

```
package qsp;

import org.openqa.selenium.By;

public class Facebook {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.facebook.com/login/");
        driver.findElement(By.id("email")).sendKeys("abcd@efgh.com");
        driver.findElement(By.id("pass")).sendKeys("12345");
        driver.findElement(By.id("loginbutton")).submit();
    }

}
```

Write a script to copy the text in email text box and paste it into Password textbox for Opensource billing Application?

```
package qsp;
import org.openqa.selenium.By;
public class Copypaste {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
        }
public static void main(String[] args) throws InterruptedException {
         WebDriver driver = new ChromeDriver();
         driver.get("http://demo.opensourcebilling.org/");
         Thread.sleep(3000);
         driver.findElement(By.id("email")).sendKeys(Keys.CONTROL+"ac");
         Thread.sleep(3000);
         driver.findElement(By.id("password")).sendKeys(Keys.CONTROL+"a");
         Thread.sleep(3000);
         driver.findElement(By.id("password")).sendKeys(Keys.BACK_SPACE);
         Thread.sleep(3000);
         driver.findElement(By.id("password")).sendKeys(Keys.CONTROL+"v");
         Thread.sleep(3000);
         driver.close();
    }
}
```

Write the script to check the Hidden password which is present in Open-source billing application?

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class PrintPasswordTextbox {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
    WebDriver   driver = new ChromeDriver();
    driver.get("http://demo.opensourcebilling.org/");
    String Passwordtext = driver.findElement(By.id("password")).getAttribute("value");
    System.out.println(Passwordtext);
    driver.close();
    }

}
```

What are the frequently used locators in Selenium?

1.id,2.name,3. linktext,4. XPath

Write a script to print text of link “Forget your Password” in demo.actitime.com?

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class PrinttheText {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("https://demo.actitime.com/login.do");
    String forgettext = driver.findElement(By.id("toPasswordRecoveryPageLink")).getText();
    System.out.println(forgettext);
    driver.close();
    }
}
```

Write a script to verify that the email text box present in Facebook login page after giving the value intentionally check whether it is empty or not?

```
public class VerifytheLogintext {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver=new ChromeDriver();
    driver.get("https://www.facebook.com/login/");
    driver.findElement(By.id("email")).sendKeys("selenium");
    String gettext = driver.findElement(By.id("email")).getAttribute("value");
    if(gettext.isEmpty())
        {
        System.out.println("Textbox is Empty");
        }
    else
        {
        System.out.println("Textbox is Not empty");
        }
    driver.close();
    }
}
```

Write a script to remove the value present in text box of username without using clear() in open source billing application?

```java
package qsp;

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 Deletetext {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver= new ChromeDriver();
        driver.get("http://demo.opensourcebilling.org/");
        WebElement Emailtext = driver.findElement(By.id("email"));
        Emailtext.sendKeys(Keys.CONTROL+"a");
        Emailtext.sendKeys(Keys.DELETE);
    }
}
```

Handling Multiple Elements

In order to handle multiple elements we will use findelements method. Return type of the find elements is “List of Web elements”. List should be imported from java.util package.

If locators are matching with multiple element, it will return the address of all the matching element.

If locators are not matching with any one of the element, then the find element method will return the empty list

Sample HTML Code

```html
<html>
<body>
<a id="d1"name="n1"class="c1"href="https://www.qspiders.com/">Qspiders</a><br>
<a id="d2"name="n2"class="c2"href="https://www.jspiders.com/">Jspiders</a><br>
</body>
</html>
```

Write a script to check the no of links present in the Html code along with that print any particular link?

```
package qsp;

import java.util.List;

public class HandlingmultipleElements {
static {
System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver=new ChromeDriver();
    driver.get("file:///D:/Desktop/QSP.html");
    List<WebElement> Alllinks = driver.findElements(By.xpath("//a"));
    int count = Alllinks.size();//size () is used to get the no of links
    System.out.println(count);
    WebElement link = Alllinks.get(1);//get () is used to specify the element based on index value
    String text = link.getText();
    System.out.println(text);
    driver.close();
}
}
```

Write a script to count how many no of links present in amazon.com and print it on the console?

```
public class Printalllinks {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("https://www.amazon.com/");
    List<WebElement> Alllinks = driver.findElements(By.xpath("//a"));
    int count = Alllinks.size();//no of links by size()
    System.out.println(count);//no will be printed in the console
    for(int i=0;i<count;i++) {//each and every link will be monitored
    WebElement links = Alllinks.get(i);//get () is used to specify the element based on index value
    String text = links.getText();//text value will be printed
    System.out.println(text);
}
    driver.close();
}
}
```

Write a script to print all the links present in amazon.com but the input has to be given from the Scanner Class?

```
public class Printalllinks {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
System.out.println("https://www.flipkart.com");
Scanner sc = new Scanner(System.in);//it is used to take input from the user
String url = sc.nextLine();
WebDriver driver= new ChromeDriver();
driver.get("https://www.amazon.com/");
List<WebElement> Alllinks = driver.findElements(By.xpath("//a"));
int count = Alllinks.size();//no of links by size()
System.out.println(count);//no will be printed in the console
for(int i=0;i<count;i++) {//each and every link will be monitored
WebElement links = Alllinks.get(i);//get () is used to specify the element based on index value
String text = links.getText();//text value will be printed
System.out.println(text);
}
driver.close();
}
}
```

Auto Suggestions

Write a program to print all the auto suggestions in the google.com after typing java in search text box and count the number of auto suggestions?

Or

Automate the following scenario

1.Open the browser

2.Go to the www.google.com

3.Type 'java' in the search box textbox

4.Find all the auto suggestions and print count of Auto suggestions

5.Print the text of Auto suggestion

6.Select the first auto suggestion

```
public class Autosuggestions {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
        }
public static void main(String[] args) throws InterruptedException {
//Open the browser
    WebDriver driver= new ChromeDriver();
//Enter the Url or open the Webpage
    driver.get("https://www.google.com/");
//Type 'java' in the search text box
    driver.findElement(By.name("q")).sendKeys("java");
    Thread.sleep(3000);
//Find all the autosuggestion and print count of auto suggestions
    List<WebElement> Autosugg = driver.findElements(By.xpath("//span[contains(text(),'java')]"));
    int count = Autosugg.size();
//Count of Auto suggestions
    System.out.println(count);
    for(int i=0;i<count;i++) {
//Print all the text of the Auto suggestions in the Console
    String text = Autosugg.get(i).getText();
    System.out.println(text);
    }
//Select the First Auto suggestion
    Autosugg.get(0).click();
    driver.close();
    }
}
```

Difference Between Find Element and Find Element's

FIND ELEMENT	FIND ELEMENT'S
Return type of the Find Element is Web Element	Return type of the Find Element's is List<Web Elements>
If the locator is not matching it will throw No such Element Exception	Returns the Empty List
If the locator is matching with multiple elements, it will return the first matching element	It returns all the matching elements
Used to handle the single element	Used to handle the multiple elements

Write a script to print all the autosuggestion by using for each loop after typing 'selenium' and select the last autosuggestion?

```
package qsp;

import java.util.List;

public class Autosuggselenium {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver= new ChromeDriver();
        driver.get("https://www.google.com/");
        driver.findElement(By.name("q")).sendKeys("selenium");
        List<WebElement> Autosugg = driver.findElements(By.xpath("//span[contains(text(),'selenium')]"));
        int count = Autosugg.size();
        System.out.println(count);
        for(WebElement Sugg:Autosugg) {
        String text = Sugg.getText();
        System.out.println(text);
}
        Autosugg.get(count-1).click();
        driver.close();
    }
}
```

Automate the Following Scenario?

1.Open chrome Browser

2.Go to www.flipkart.com

3.Type iPhone in the search text box

4.Find all the auto suggestions and print the count of suggestions

5.Print the text of auto suggestions (Note you have to use for each loop)

6.Select the last auto suggestion

Go to snapdeal.com and type 't-shirt' in from text box then print all the Auto suggestions and select and select the Black T-shirts in it.

Synchronization

The process of matching the selenium speed with the application is called as Synchronization.

Most of the time selenium is faster than the application because of the reason, we may not get the expected result or we get exception such as **'No Such element Exception'**

Example: using thread. Sleep method

Write a script for login and logout the actitime.com

```java
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Synchronization {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver=new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        driver.findElement(By.id("username")).sendKeys("admin");
        driver.findElement(By.name("pwd")).sendKeys("manager");
        driver.findElement(By.xpath("//div[text()='Login ']")).click();
        Thread.sleep(3000);
        driver.findElement(By.id("logoutLink")).click();
        driver.close();
    }
}
```

Implicit Wait

In selenium there are different ways to synchronize the script.One of the frequently used option is implicit wait

Syntax→ driver.manage().timeouts().implicitlywait(10,Timeunit.Seconds);

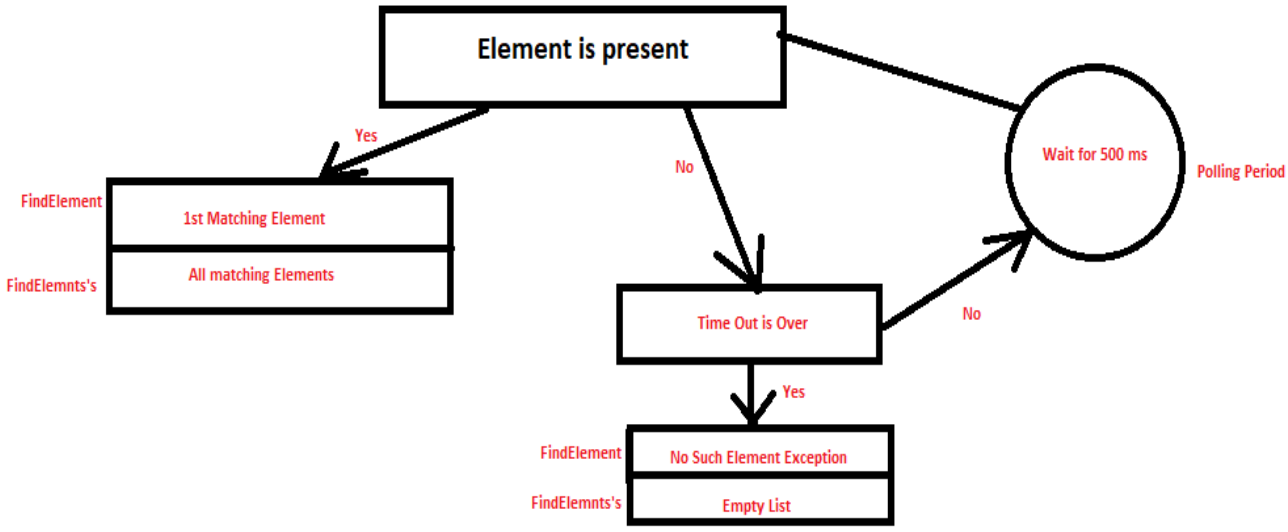
It takes two arguments,1st one is long(duration) second one is Time unit

Time unit can be Days, Hours, Minutes, Seconds, Milliseconds, Microseconds, Nanoseconds…

The specified duration is only used by Find element& Find Elements Statement

Means implicit wait will work only for the find element and find elements function not for other functions

Default value of implicit wait is zero Seconds



When the control comes to any find element or find elements statement, it will check whether the element is present or not.

If the element is present the find element method returns the 1st matching element whereas find element's method returns all the matching Elements.

If the specified element is not present then it will check for the Timeout

If time is over the find element method will throw the No such element exception whereas find element's method returns the Empty List

If the timeout is not over it waits for 500ms(1/2sec) which is called as Polling Period.

Then it will continue to check whether the element is present or not.

```java
package qsp;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Synchronization {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver=new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    driver.manage().timeouts().implicitlyWait(1,TimeUnit.DAYS);
    driver.findElement(By.id("username")).sendKeys("admin");
    driver.findElement(By.name("pwd")).sendKeys("manager");
    driver.findElement(By.xpath("//div[text()='Login ']")).click();
    driver.findElement(By.id("logoutLink")).click();
    driver.close();
    }
}
```

Explicit Wait

In order to handle the synchronization of any method we can use Explicit Wait.

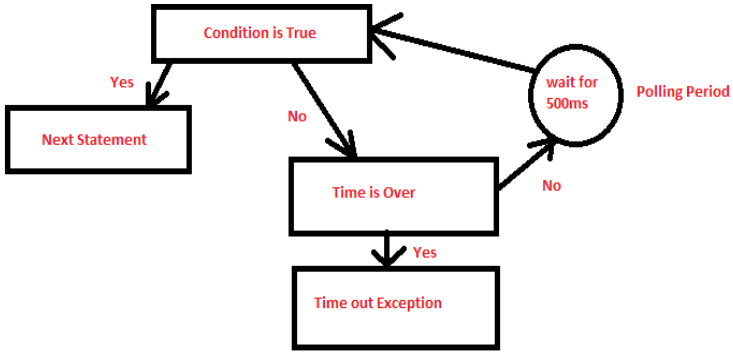
Web driver wait itself is called Explicit Wait; we have to specify the waiting condition explicitly.

Syntax

Webdriver wait = new Webdriverwait(driver,Duration.ofseconds(20));

Wait.until(Expectedconditions.titleas("actiTime-EnterTime-Track"));

Flow Diagram



When the control comes to wait.until statement it will check the specific condition

If the condition is true it will go to the Next statement, if the condition is false it will check for the timeout.

If the time is over, it will throw the TimeoutException, else it will wait for 500ms & it will continue to check the condition.

Note: In the above flow diagram title is(“actiTime-EnterTime-Track”) conditions

```
package qsp;

import java.time.Duration;

public class Explicitwait {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver=new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        driver.findElement(By.id("username")).sendKeys("admin");
        driver.findElement(By.name("pwd")).sendKeys("manager");
        driver.findElement(By.xpath("//div[text()='Login ']")).click();
        WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(30));
        wait.until(ExpectedConditions.titleContains("Enter"));
        String title = driver.getTitle();
        System.out.println(title);
        driver.findElement(By.id("logoutLink")).click();
        driver.close();
        }
    }
```

Can we handle the synchronization of Find element method using Explicit Wait? Yes.

```
package qsp;

import java.time.Duration;

public class Explicitwait1 {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        driver.findElement(By.id("username")).sendKeys("admin");
        driver.findElement(By.name("pwd")).sendKeys("manager");
        driver.findElement(By.xpath("//div[text()='Login ']")).click();
        WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(20));
        wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("logoutLink")));
        driver.findElement(By.id("logoutLink")).click();
        driver.close();

    }

}
```

Difference Between Implicit wait and Explicit Wait

Implicit Wait	Explicit Wait
We do not specify the waiting condition explicitly	We should specify the waiting condition explicitly
We can handle synchronization of all Find element and Find element's	We can handle synchronization of any method but only one at a time
After the time duration we get "No Such Element "Exception	After the time duration we get "Time out Exception"
Time duration can be days, hours, Minutes, Seconds, etc..	Time duration will be only in seconds

Custom wait

Handling the synchronization of the automation script by writing own code is called as Custom wait.

Handling List box (Drop Down or Combo Box)

List Box is created using the Select Tag.

There are two types of List Box

1.Single Select List Box (Drop down or Combo box)

2.MultiSelect List Box

Content of the List Box is created using Option Tag

To handle the list box, we used Select Class of Selenium. It should be imported from the following Package.import.org.openqa. Selenium.Support.ui.Select

Select class has parameterized constructor it takes an argument of the type Web element (i.e., address of the List Box). In order to select the required option present in the List Box we can use any one of the following methods of Select Class.

1.SelectbyVisibleText(str)→Takes String as an Argument

2.SelectbyIndex(int)→Takes Integer as an Argument

3.SelectbyValue (Str)→ Takes String as an Argument

If the specified option is duplicate then it will select first matching option(in dropdown list) and if the specified option is not present(text,value,index),we get No Such Element exception.

Select Class can also be used to handle multiselect list box. In select class we can also have the following of methods. This can be used on Multiselect List Box

1.deselectByVisibleText (Str)

2. deselectByIndex (int)

3.deselectbyValue (Str)

4.deselectAll ()

5.isMultiple ()

6.getAllSelectedOption ()

7.getFirstSelectedOption ()

8.getOptions ()

9.getWrappedElement ()

If the specified option is duplicate in multiselect List box, it selects all the matching option.

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.Select;

public class HandlingListbox {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.facebook.com/");
        driver.findElement(By.linkText("Create New Account")).click();
        Thread.sleep(3000);
        WebElement Monthlist = driver.findElement(By.id("month"));
        Select sele = new Select(Monthlist);
        sele.selectByIndex(1);
        Thread.sleep(3000);
        sele.selectByValue("1");
        Thread.sleep(3000);
        sele.selectByVisibleText("Dec");

    }

}
```

Write a script to select your DOB in Facebook.com after clicking on Create New Account?

Ex:-21-sep-1997

```java
package qsp;

import java.util.concurrent.TimeUnit;

public class DOBinFB {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.facebook.com/");
        driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);
        driver.findElement(By.linkText("Create New Account")).click();
        WebElement Day = driver.findElement(By.id("day"));
        Select d = new Select(Day);
        d.selectByIndex(20);
        WebElement Month = driver.findElement(By.id("month"));
        Select m = new Select(Month);
        m.selectByVisibleText("Sep");
        WebElement Year = driver.findElement(By.id("year"));
        Select y = new Select(Year);
        y.selectByValue("1997");
    }
}
```

Handling Multiselect List

HTML code to create Multiselect List Box

```html
MTR:<br>
<select id="mtr" multiple>
<option value="i">idly</option>
<option value="v">vada</option>
<option value="d">Dosa</option>
<option value="p">poori</option>
<option value="k">kharabath</option>
<option value="c">chowchowbath</option>
<option value="a">akkirotti</option>
<option value="l">laddu</option>
<option value="u">upma</option>
</select><br>
<br>
<br>
<select id="slv"multiple>
<option value="b"selected>biriyani</option>
<option value="bb">Bisibalabath</option>
<option value="p"selected>Pulaw</option>
<option value="Pa">Paddu</option>
<option value="Pr">Parota</option>
<option value="Vb">Vegbiriyani</option>
<option value="MD">Masala Dosa</option>
<option value="R">Ragidosa</option>
</select><br>
```

```java
package qsp;

import org.openqa.selenium.By;

public class MultipleListBox {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe") ;
    }
    public static void main(String[] args) throws InterruptedException {
         WebDriver driver = new ChromeDriver();
         driver.get("file:///D:/Desktop/MTRLISTBOX.html");
         WebElement mtr = driver.findElement(By.id("mtr"));
         Select s = new Select(mtr);
         s.selectByValue("i");
         s.selectByIndex(1);
         s.selectByVisibleText("Dosa");
         Thread.sleep(3000);
         s.deselectByIndex(1);
         s.deselectByValue("i");
         s.deselectByVisibleText("Dosa");
         System.out.println(s.isMultiple());
    }
}
```

Write a script to print the selected text in slv text box in the html page?

```java
package qsp;

import org.openqa.selenium.By;

public class Printfirstselectedoption {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("file:///D:/Desktop/MTRLISTBOX.html");
    WebElement slv = driver.findElement(By.id("slv"));
    Select s = new Select(slv);
    String Firstoption = s.getFirstSelectedOption().getText();
    System.out.println(Firstoption);
    driver.close();
    }
}
```

Write a script to print the count of all the selected options present in slv text box?

```
package qsp;

import java.util.List;

public class Printallselectedoption {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/MTRLISTBOX.html");
    WebElement slv = driver.findElement(By.id("slv"));
    Select s = new Select(slv);
    List<WebElement> option = s.getAllSelectedOptions();
    int count = option.size();      //size() is used to get the no.of links
    System.out.println(count);

    for(int i=0;i<count;i++) {      //check each and every element present in the list
    String text = option.get(i).getText();
    System.out.println(text );
    }
    driver.close();
    }
}
```

Write a script to print all the options present in MTR LIST BOX.

```
package qsp;

import java.util.List;

public class AllMtrList {
static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/MTRLISTBOX.html");
    WebElement mtr = driver.findElement(By.id("mtr"));
    Select s = new Select(mtr);
    List<WebElement> alloption = s.getOptions();
    int count = alloption.size();
    System.out.println(count);
    for(int i=0;i<count;i++) {
    System.out.println(alloption.get(i).getText());
    }
driver.close();
    }
}
```

Write a script to print all the options present in SLV LIST BOX but using different for loop

```
package qsp;

import java.util.List;

public class ALLSlvListbox {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver   driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/MTRLISTBOX.html");
    WebElement slv = driver.findElement(By.id("slv"));
    Select s = new Select(slv);
    List<WebElement> Alloptions = s.getOptions();
    int count = Alloptions.size();
    System.out.println(count);
    for(WebElement a:Alloptions){
    System.out.println(a.getText());
    }
    driver.close();
    }
}
```

Hash set

Hash set class extends Abstract Set class which implements Set interface.

1.Contains Unique Elements

2.Allows the Null Value

3.Class is not Synchronized

4.Best approach to search options

5.Insertion order is not Maintained

Write a script to print all the options present in MTR LIST BOX without duplicates in sorted order?

```
package qsp;

import java.util.HashSet;

public class WithoutDuplicate {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    HashSet<String> hs = new HashSet<String>();
    WebDriver driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/MTRLISTBOX.html");
    WebElement mtr = driver.findElement(By.id("mtr"));
    Select s = new Select(mtr);
    List<WebElement> Alloptions = s.getOptions();
    int count = Alloptions.size();
    System.out.println(count);
    for(int i=0;i<count;i++);
    {
        Object text = Alloptions.get(count).getText();
        hs.add((String) text);
    }
    for(String text:hs)
    {
        System.out.println(text);
    }
    driver.close();
    }
}
```

Handling Pop ups.

In selenium, depending on popup we write different types of code to perform action on the popup.

Popup are generally categorized as follows.

1.Java Script or alert popup or Confirmation Popup

2.Hidden division or Calendar Popup

3.File Upload Popup

4.File Download Popup

5.Print Popup

6.Child Window Popup

7.Notification Popup

8.Authentication Popup

1.Java Script or Alert Popup (Confirmation Popup)

Characteristics:

1.We cannot inspect this Popup

2.We cannot move this popup

3.This Popup will be having Ok button(alert) or it contains OK r Cancel button (confirmation)

4.It will be present below the address bar in the middle section

```
package qsp;

import org.openqa.selenium.Alert;

public class Alertpopup {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("https://demo.automationtesting.in/Alerts.html");
    driver.findElement(By.xpath("//button[@class='btn btn-danger']")).click();
    Alert a = driver.switchTo().alert();
    String text = a.getText();
    a.accept();
    System.out.println(text);
    driver.close();
    }
}
```

Solutions:

To handle alert or java script popup we can use **driver.switchto().alert()** statement. After switching to alert popup, we can use

1.accept()→to click on OK Button

2.dismiss()→to click on Cancel Button

3.getText()→to get the text which is present in Popup

4.Sendkeys()→to type the text on the popup

All the above code works on Confirmation popup also, if the popup is alert then there will not be any difference between the accept and Dismiss(Both will click on Ok Button)

Hidden Division or Calendar popup

Characteristics:

1.We can inspect this Popup

2.We cannot able to move this popup

Solution:

We can handle hidden division popup by using driver.findelement().method

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class HiddenPopup {
static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
        }
public static void main(String[] args) throws InterruptedException {
        WebDriver driver= new ChromeDriver();
        driver.get("https://www.flipkart.com/");
        Thread.sleep(3000);
        driver.findElement(By.xpath("(//button)[2]")).click();
    }
}
```

Calendar Pop-up

1.Open the browser

2.EnterUrl→https://www.careinsurance.com/rhicl/proposalcp/renew/index-care

3.Enter the policy Number-12345

4.Clock on DOB (Req Date)

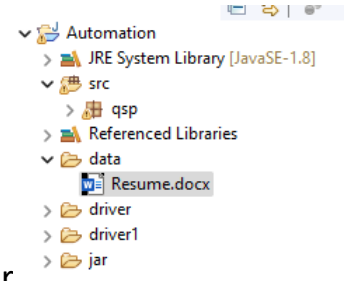
5.Specify the Contact number and Click on Renew Button.

```java
package qsp;

import java.util.concurrent.TimeUnit;

public class CalenderPopup {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) {
        WebDriver driver= new ChromeDriver();
        driver.get("https://www.careinsurance.com/rhicl/proposalcp/renew/index-care");
        driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);
        driver.findElement(By.id("policynumber")).sendKeys("12345");
        driver.findElement(By.id("dob")).click();
        WebElement yr = driver.findElement(By.className("ui-datepicker-year"));
                Select s = new Select(yr);
                s.selectByVisibleText("2022");
        WebElement Mnth = driver.findElement(By.className("ui-datepicker-month"));
                Select s1 = new Select(Mnth);
                s1.selectByVisibleText("Feb");
        driver.findElement(By.linkText("12")).click();
        driver.findElement(By.id("alternative_number")).sendKeys("9876543210");
        driver.findElement(By.id("renew_policy_submit")).click();
    }
}
```

Note: In the most of the cases calendar popup will be hidden division popup which will be handled using the find element statement.

File Upload Popup

HTML CODE→ <input type="file"id="cv"/>



New Folder

```java
public class Fileupload {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver= new ChromeDriver();
        driver.get("file:///D:/Desktop/FU.html");
        Thread.sleep(3000);
        File f = new File("./data/Resume.docx");
        String Absolutepath = f.getAbsolutePath();
        driver.findElement(By.id("cv")).sendKeys(Absolutepath);
    }
}
```

Robot Class

Robot class is Java class present in Java.awt.package(awt=Abstract Window Toolkit)

Robot class works similar to the Sendkeys, whenever I want to perform keyboard operation in windows, we go for Robot Class. We commonly use two methods in the robot class

1.Keypress

2.Key-Release

```
package qsp;

import java.awt.AWTException;
import java.awt.RenderingHints.Key;
import java.awt.Robot;
import java.awt.event.KeyEvent;
import java.io.IOException;

public class DemoRobotclass {

    public static void main(String[] args) throws AWTException, IOException {
        Runtime.getRuntime().exec("notepad");
        Robot r = new Robot();
        r.keyPress(KeyEvent.VK_SHIFT);
        r.keyPress(KeyEvent.VK_Q);
        r.keyRelease(KeyEvent.VK_SHIFT);
        r.keyPress(KeyEvent.VK_S);
        r.keyPress(KeyEvent.VK_P);
        }
}
```

Notification Popup

Characteristics:

1.We can’t able to inspect this Pop-up

2.We can’t able to move this pop-up

3.It will have two buttons allow and block

4.It is displayed below the address bar in the beginning similar to alert pop-up

Solution

To handle this popup we change the setting of the pop-up, so that notification pop-up will not be displayed.

To change the settings of the browser we use add arguments method of Chrome options Class

Note

Add arguments is an example for method overloading(it takes string or list of string as an argument).

For every browser we have respective options class.

So→Chrome options, Firefox Options, Internet Explorer options, etc.

In order to open the browser with modified setting we use parameterized constructor in respective browser class.

newChromedriver()→will open the browser in default setting

newChromeDriverOptions()→will open the browser with modified settings.

This above statement is the example for Constructor Overloading

```
package qsp;

import org.openqa.selenium.WebDriver;

public class Notificationpopup {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    ChromeOptions option = new ChromeOptions();
    option.addArguments("disable-notifications");
    WebDriver driver= new ChromeDriver(option);
    driver.get("https://www.yatra.com/");
    }
}
```

Authentication Popup

Characteristics

1.We can't move this pop-up

2.We cannot inspect this Pop-up

3.This pop-up will be having the username and Password textbox along with the sign in and cancel button

4.This Pop-up will be displayed just below the address bar middle section of the browser similar to alert pop-up

Solution

To handle the authentication pop-up we send the username and password along with the url inside the get method

url- https://the-internet.herokuapp.com/basic_auth

Username-admin

Password-admin

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Authenticationpopup {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
        }
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("https://admin:admin@the-internet.herokuapp.com/basic_auth");
    }
}
```

Child Window Pop-up or Child Browser Pop-up

Characteristics

We can move this popup & inspect this popup

This Popup will have maximize, minimize & close button along with the address bar

Solution

To handle child window popup we use getwindowhandle() & switchto().windows statement

Note

Address of the browser present on the desktop is called window handle (section id)

In order to retrieve it we use getwindowhandle ().

```
package qsp;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Childwindowpopup {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.google.com/");
        String wh = driver.getWindowHandle();
        System.out.println(wh);
        driver.close();
    }
}
```

URL→https://nxtgenaiacademy.com/multiplewindows/

```
package qsp;

import java.util.Set;

public class Childwindowpopup1 {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("https://nxtgenaiacademy.com/multiplewindows/");
    driver.findElement(By.name("newbrowserwindow123")).click();
    Set<String> allwindow = driver.getWindowHandles();
    int count = allwindow.size();
    System.out.println(count);
    driver.quit();
    }
}
```

Difference Between GetwindowHandle & GetwindowHandles

GetwindowHandle	GetwindowHandle's
It will return the address of the current browser	It will return the address of all the browser's
Return type is String	Return type is Set<String>

Write a script to print the title or window Handle's of all the browsers open by https://nxtgenaiacademy.com/multiplewindows/ ?

```
package qsp;

import java.util.Set;

public class Childwindowpop2 {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("https://nxtgenaiacademy.com/multiplewindows/");
    driver.findElement(By.name("newbrowserwindow123")).click();
    driver.findElement(By.name("newbrowserwindow123")).click();
    Set<String> allwh = driver.getWindowHandles();
    for(String Win:allwh)
    {
        driver.switchTo().window(Win);
        String title = driver.getTitle();
        System.out.println(title);
    }
    driver.quit();
    }
}
```

Write a script to print the address or window Handles of all the browsers open by https://nxtgenaiacademy.com/multiplewindows/ ?

```
package qsp;

import java.util.Set;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Childwindowaddress {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://nxtgenaiacademy.com/multiplewindows/");
        driver.findElement(By.name("newbrowserwindow123")).click();
        driver.findElement(By.name("newbrowserwindow123")).click();
        Set<String> allwh = driver.getWindowHandles();
        int count = allwh.size();
        System.out.println(count);
        for(String wh:allwh)
        {
            System.out.println(wh);
        }
        driver.quit();
    }

}
```

How do you close all the browser's without using Quit() ?

By using driver.Close() inside the for each loop

```
package qsp;

import java.util.Set;

public class CloseBrowserwithoutquit {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
public static void main(String[] args) throws InterruptedException {
    WebDriver driver = new ChromeDriver();
    driver.get("https://nxtgenaiacademy.com/multiplewindows/");
    driver.findElement(By.name("newbrowserwindow123")).click();
    driver.findElement(By.name("newbrowserwindow123")).click();
    Thread.sleep(3000);
    Set<String> allwh = driver.getWindowHandles();
    for(String cs:allwh)
    {
    driver.switchTo().window(cs);
    driver.close();
        }
    }
}
```

How do you close all the child browser's without using Quit() but not the Parent browser has to be closed?

```
package qsp;

import java.util.Set;

public class CloseBrowserwithoutquit {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
public static void main(String[] args) throws InterruptedException {
    WebDriver driver = new ChromeDriver();
    driver.get("https://nxtgenaiacademy.com/multiplewindows/");
    String Pwh = driver.getWindowHandle();
    driver.findElement(By.name("newbrowserwindow123")).click();
    driver.findElement(By.name("newbrowserwindow123")).click();
    Thread.sleep(3000);
    Set<String> allwh = driver.getWindowHandles();
    for(String cs:allwh)
    {
    driver.switchTo().window(cs);
    if(cs.equals(Pwh)) {
            }
    else {
    driver.close();
            }
        }
    }
}
```

How do you close all the one particular child browser without using Quit() but not the Parent browser has to be closed?

Summary of the Pop-up

Pop-up's	Solution
Alert Pop-up or Java Script Popup	driver.switchto().alert→Accept,dismiss,gettext()
Hidden Division or Calendar Pop-up	By using find element() method
File Upload Pop-up	Browser button.sendkeys(absolute path)
File Download Pop-up	By using Robot Class(for chrome no need to handle)
Print Pop-up	By using Robot Class (for chrome we use findelement)
Authentication Pop-Up	By sending Username and Password along with the URL inside the get() method
Notification Pop-up	By using add arguments() method of the browser options class(chrome options)
ChildWindow or Childbrowser Pop-up	By using Getwindowhandles() and driver.switchto().window() statement

Handling Tabs

Tab is also treated as new window in selenium

Here we are going to handle the tab in the same way as we handle child browser or child window browser pop-up

Write a script to count the no.of.tabs present in actiTime application after clicking the actitime Inc.link?

```
package qsp;

import java.util.Set;

public class HandlingTab {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("https://demo.actitime.com/");
        driver.findElement(By.linkText("actiTIME Inc.")).click();
        Set<String> alltb = driver.getWindowHandles();
        int count = alltb.size();
        System.out.println(count);
        driver.quit();
    }

}
```

How do you close the Current tab? **By using driver.close().**

How do you close the all the tabs? **By using driver.quit().**

Write a script to close all the Tabs without using quit ()?

```
package qsp;

import java.util.Set;

public class Closealltabswithoutquit {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    driver.findElement(By.linkText("actiTIME Inc.")).click();
    Set<String> alltb = driver.getWindowHandles();
    for(String tb:alltb)
        {
            driver.switchTo().window(tb);
            driver.close();
        }
    }
}
```

Iterator

→Iterator is an object that can be used to loop through the collections, like Array List and Hash Set. It is called as “Iterator” because “iterating” is the technical term of looping.

→It should import from java.util package

Note: Trying to remove items using a for or a for each loop would not work correctly because the collection is changing size at the same time that the code is trying to loop.

Write a script to close all the Tabs without using quit () and for each loop?

```
package qsp;

import java.util.Iterator;

public class Withoutforloop {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    driver.findElement(By.linkText("actiTIME Inc.")).click();
    Set<String> Allwh = driver.getWindowHandles();
    int count = Allwh.size();
    System.out.println(count);
    Iterator<String> it = Allwh.iterator();
    String parent = it.next();
    String child = it.next();
    driver.switchTo().window(parent);
    driver.close();
    driver.switchTo().window(child);
    driver.close();
    }
}
```

Write a script to automate the following scenario?

1.Go to the chrome and search for Wiprojobs

2.Capture all the links present in the search page & print the URL of all the links present in search page.

```
package qsp;

import java.util.List;

public class Wiprojoblink {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("https://www.google.com/");
    driver.switchTo().activeElement().sendKeys("Wiprojobs"+Keys.ENTER);
    List<WebElement> alllink = driver.findElements(By.xpath("//a"));
    int count = alllink.size();
    System.out.println(count);
    for(WebElement text:alllink)
    {
        System.out.println(text);
    }
    driver.close();
    }
}
```

Write a script to automate the following scenario

1.Open the browser

2.Enter the url(https://demo.actitime.com/)

3.Login to the Application (Un: Admin, PW: manager)

4.Click on about your ACTi time present in help dropdown

5.Click on read service agreement link present in the popup

6.Print all the header present in license agreement tab

7.Close the agreement tab

8.Close the main browser.

```java
public class Handlingtabs2 {
static{
System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver = new ChromeDriver();
    driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);
    driver.get("https://demo.actitime.com/");
    driver.findElement(By.id("username")).sendKeys("admin");
    driver.findElement(By.name("pwd")).sendKeys("manager");
    driver.findElement(By.xpath("//div[text()='Login ']")).click();
    driver.findElement(By.xpath("(//div[@class='menu_icon'])[4]")).click();
    driver.findElement(By.linkText("About your actiTIME")).click();
    driver.findElement(By.linkText("Read Service Agreement")).click();
    Set<String> Allwh = driver.getWindowHandles();
    Iterator<String> it = Allwh.iterator();
    String ptab = it.next();
    String ctab = it.next();
    driver.switchTo().window(ctab);
    List<WebElement> Allheadings = driver.findElements(By.xpath("//ul"));
    for(int i=0;i<Allheadings.size();i++)
    {
        String HeadingText = Allheadings.get(i).getText();
        System.out.println(HeadingText);
    }
    driver.close();
    driver.switchTo().window(ptab);
    driver.close();
    }
}
```

Handing Mouse Actions

By using mouse, we can perform the following actions

1.Handling Dropdown menu (By using Mouse Hover)

2.Context Click (Right Click)

3.Drag and Drop

4.Double Click

How to Handle Dropdown Menu? (Or handling mouse hover)?

Mouse hover means your mouse pointer to a particular position

→ Drop down menu is an element on which if we move the mouse pointer it will display the list of options. To handle the drop-down menu, we use “movetoElement ()” method of Actions Class.

→ movetoElement () is nothing but mousehover.

→ In selenium” Action is an interface & Actions is a Class”

→ ” Actions” class is present inside the ‘Interactions Package’. “Actions” class is mainly used for mouse hover actions.

→ Actions Class has a parameterized constructor, where it takes web driver as an argument whenever we call any method of actions class, we have to use ‘perform method’ at the end of the statement.

Note: All the methods of Actions Class are Overloaded Method.

Write a script to open the ACTi-time link in new Window?

```
package qsp;

import java.awt.AWTException;

public class Mousehover {
static {
System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) throws AWTException, InterruptedException {
    WebDriver driver = new ChromeDriver();
    driver.get("https://demo.actitime.com/");
    WebElement link = driver.findElement(By.linkText("actiTIME Inc."));
    Actions a = new Actions(driver);
    a.contextClick(link).perform();
    Thread.sleep(5000);
    Robot r = new Robot();
    r.keyPress(KeyEvent.VK_W);
    Thread.sleep(5000);
    driver.quit();
    }
}
```

How do you perform double click in Selenium?

We can use actions class to double click on the Element

Example: - a.doubleclick(webelement).perform();

Write a script to perform drag and drop?

```
package qsp;

import org.openqa.selenium.By;

public class DragandDrop {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
    public static void main(String[] args) throws InterruptedException {
    WebDriver   driver = new ChromeDriver();
    driver.get("http://www.dhtmlgoodies.com/scripts/drag-drop-custom/demo-drag-drop-3.html");
    WebElement from = driver.findElement(By.id("box3"));
    WebElement to = driver.findElement(By.id("box106"));
    Thread.sleep(3000);
    Actions a = new Actions(driver);
    a.dragAndDrop(from, to).perform();
    }
}
```

Automate the following scenario

1.Go to the V-tiger Application

2.Mouse Hover to resources tab & click on customers in the drop down menu

3.Double click on Read full story button and check whether Hacker Earth Page is Displayed or Not.

```
package qsp;

import java.util.concurrent.TimeUnit;

public class Mousehover1 {
static {
System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.manage().window().maximize();
    driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);
    driver.get("https://www.vtiger.com/");
    WebElement resource = driver.findElement(By.linkText("Resources"));
    Actions a = new Actions(driver);
    a.moveToElement(resource).perform();
    driver.findElement(By.linkText("Customers")).click();
    WebElement readstory = driver.findElement(By.xpath("//a[text()='READ FULL STORY']"));
    a.doubleClick(readstory).perform();
    }
}
```

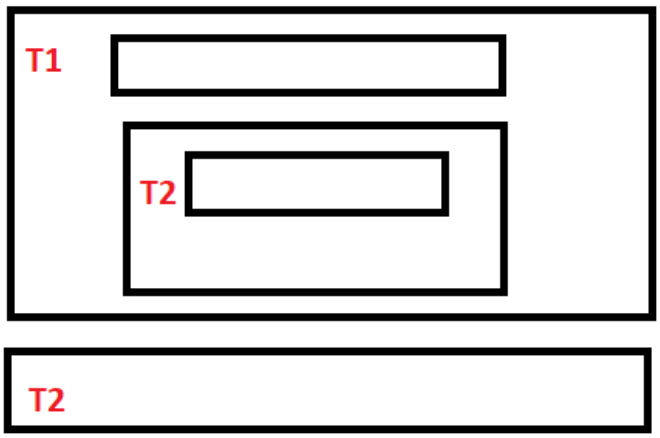
Handling Frames

→Webpage inside another webpage is called as Embedded Webpage. Developers creates embedded webpage using iframe tag.

→While automating the element, if it is present inside the frame we should transfer the driver control into the frame using switchto().frame() statement.

HTML Code for frames:

```
T1: <input id="d1"type="text"/><br>
<iframe id="f1"src="Handling.html"></iframe>
T2: <input id="d2"type="text""/><br>
```



Write a script to send the values to T2 and T1?

```
package qsp;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Handlingframes {
static {
    System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/Frames.html");
    driver.switchTo().frame(0);
    driver.findElement(By.id("d2")).sendKeys("qsp");
    driver.switchTo().parentFrame();
    driver.findElement(By.id("d1")).sendKeys("jsp");
    }
}
```

Write a script to switching the frames from one to another?

```
package qsp;

import org.openqa.selenium.By;

public class Handlingframes1 {
static {
System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
}
public static void main(String[] args) {
    WebDriver driver= new ChromeDriver();
    driver.get("file:///D:/Desktop/Frames.html");
    driver.findElement(By.id("d1")).sendKeys("Deepak");
    driver.switchTo().frame(0);
    driver.findElement(By.id("d1")).sendKeys("Veeresh");
    driver.switchTo().defaultContent();
    driver.findElement(By.id("d2")).sendKeys("Nidhi");
    driver.switchTo().frame("f1");
    driver.findElement(By.id("d2")).sendKeys("Nithish");
    driver.switchTo().parentFrame();
    driver.findElement(By.id("d1")).sendKeys("threeothers");
    WebElement frame = driver.findElement(By.xpath("//iframe"));
    driver.switchTo().frame(frame);
    driver.findElement(By.id("d2")).sendKeys("you");
    }
}
```

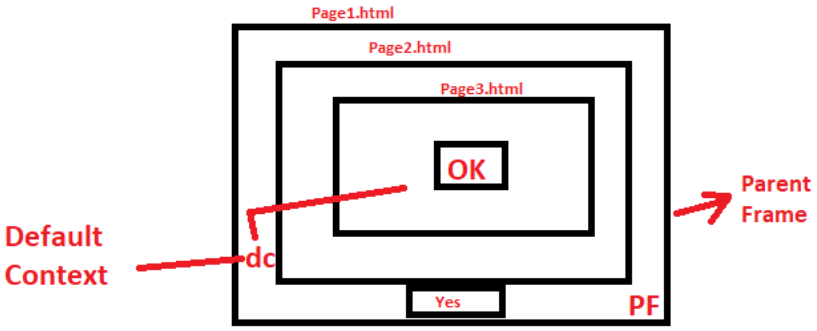
Note: -In the above example frame method is overloaded. It takes only one argument of any of the following 3 types

1.int (index of the frame starts from Zero)

2.String (id of the frame)

3.Webelement (address of the frame)

Nested Frame



1) driver.switchto.frame(0);

driver.switchto.frame(0);

driver.findelement.click()----→Ok button

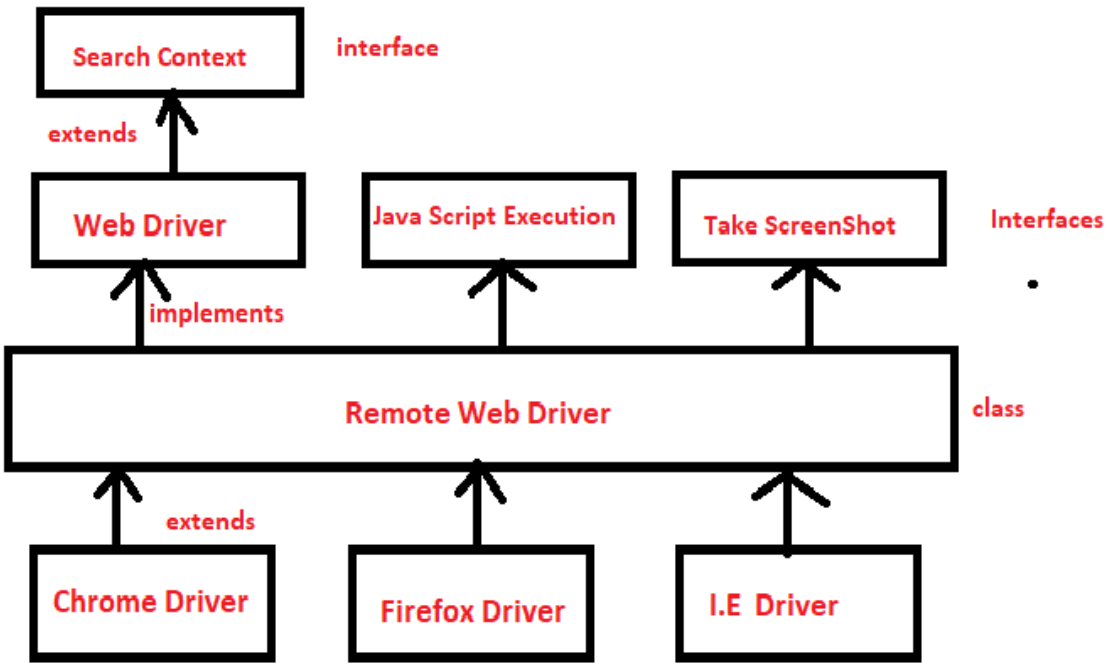
2) driver.switchto().parentframe();

driver.switchto().pf();

driver.findelement.click();-----→Yes Button

Handling Disabled Elements and Scroll Bar

The executescript() method is declared in javascript executor interface which is implemented in Remote webdriver class. Since we already upcasted browser specific classes to webdriver interface, this (execute script()) method will be hidden, I order to access this method we should downcast it(remotewebdriver) or typecast the object to java script executor.



In order to validate the java script in browser inspect the element and click on the console tab present in the developer tool bar and type the script

```
"document.getElementById('d2').value=\"manager\"→ for texting tn Hidden box

document.getElementById('d3').click();→ for clicking on Element
```

HTML CODE for HANDLING DISABLED ELEMENTS

```
UN:<input type="text"id="d1"/><br>
PW:<input type="text"id="d2"disabled/><br>
<input type="button"id="d3"value="Login"/>
```

```
package qsp;

import org.openqa.selenium.By;

public class HandlingdisabledElements {
    static {
        System.setProperty("webdriver.chrome.driver","./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        driver.get("file:///D:/Desktop/Disable.html");
        driver.findElement(By.id("d1")).sendKeys("admin");
        JavascriptExecutor js = (JavascriptExecutor)driver;
        js.executeScript("document.getElementById('d2').value=\"manager\"");
    }
}
```

Write a script to scroll 2000px vertically in BBC.com

```
package qsp;

import org.openqa.selenium.JavascriptExecutor;

public class Scrollingvertically {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver   driver  = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.bbc.com/");
        JavascriptExecutor js = (JavascriptExecutor)driver;
        js.executeScript("window.scrollBy(0,2000)");
    }
}
```

Write a script to scroll to the Particular Element in BBC.com

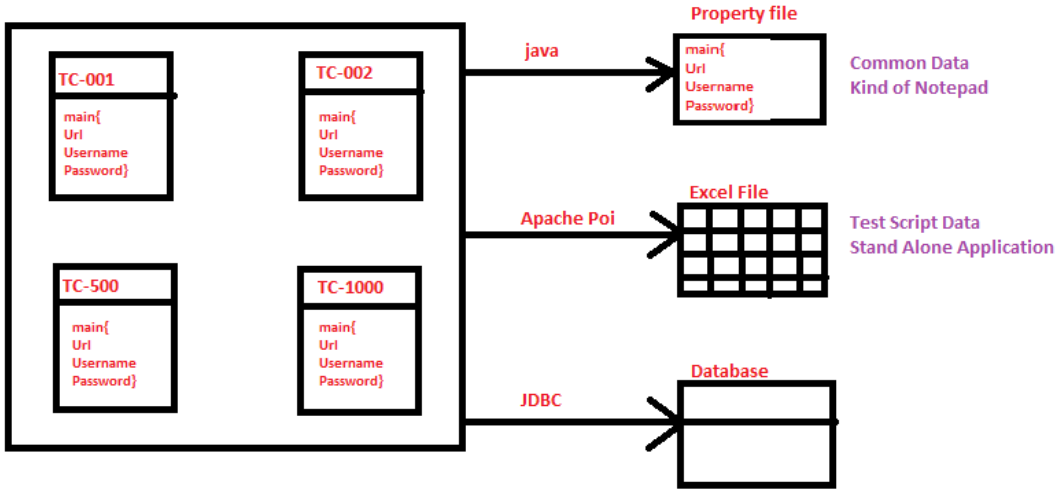
```
package qsp;

import org.openqa.selenium.By;

public class ClassBBCScroll {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) {
        WebDriver driver= new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.bbc.com/");
        int y = driver.findElement(By.xpath("//a[text()=' CEO Secrets ']")).getLocation().getY();
        JavascriptExecutor j = (JavascriptExecutor)driver;
        j.executeScript("window.scrollBy(0,"+y+")");
    }
}
```

Write a script to scroll to the Bottom of Webpage and get back to the top of the Webpage in BBC.com

```
package qsp;

import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class BBCScroll {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) throws InterruptedException {
        WebDriver driver= new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.bbc.com/");
        JavascriptExecutor j = (JavascriptExecutor)driver;
        j.executeScript("window.scrollTo(0,document.body.scrollHeight)");
        Thread.sleep(3000);
        j.executeScript("window.scrollTo(0,0)");
    }
}
```

Data Driven Testing

Testing the application with multiple test data which is kept in external resource file like excel, property file and database etc. is called as data driven testing.




```java
package qsp;

import java.util.ArrayList;

public class Datadriven {

    public static void main(String[] args) {
        ArrayList<String> first = new ArrayList<String>();
        first.add("https://demo.actitime.com");
        first.add("admin");
        //if we try to add email in between UN and PW tommorow
        first.add("admin@actitime.com");
        first.add("manager");
        //we get email instead of Password
        System.out.println(first.get(2));
    }
}
```

```java
package qsp;

import java.util.HashMap;

public class Datadriven1 {

public static void main(String[] args) {
    HashMap<String, String> hs = new HashMap<String, String>();
    hs.put("username", "admin");
    hs.put("email", "admin@actitime.com");
    hs.put("password", "manager");
    System.out.println(hs.get("username"));
    System.out.println(hs.get("email"));
    }
}
```

Note: -

1)If we use index, we will not retrieve the same data if any data will add or deleted tomorrow.

2)We get the same output even if new data is added in between tomorrow if we use key value pair (of maps). Same concept is used to store in the property file.

3)As per the rule of automation, test data should not be hardcoded within the test script because modification and maintenance of test data is difficult whenever we want to run the script with different set of data.

4)Instead of Hardcoding, get the data from external resource file and run the test script. This is called as Data driven Testing (external resource might be property file (. property) or Excel(.xlsx) or Database.

Handling Property File

In property file data will be stored in the form of key value pair.

Key and value should be separated by single space

By default, all the data present in property file is a string

Copy this code in notepad and save it as any file name with ".property" extension.

```
package qsp;

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.Properties;

public class Propertyfile {

    public static void main(String[] args) throws IOException {
        FileInputStream fis = new FileInputStream("./data/Commondata.property.txt");
        Properties p = new Properties();
        p.load(fis);
        String url = p.getProperty("Url");
        String un = p.getProperty("Username");
        String pw = p.getProperty("Password");
        System.out.println(url);
        System.out.println(un);
        System.out.println(pw);
    }
}
```

Write a script to login to Actitime.com using Property file

```
package qsp;

import java.io.FileInputStream;

public class Loginbyproperty {
    static {
        System.setProperty("webdriver.chrome.driver", "./driver1/chromedriver.exe");
    }
    public static void main(String[] args) throws IOException {
        FileInputStream fis = new FileInputStream("./data/Commondata.property.txt");
        Properties p = new Properties();
        p.load(fis);
        String url = p.getProperty("Url");
        String un = p.getProperty("Username");
        String pw = p.getProperty("Password");
        WebDriver driver = new ChromeDriver();
        driver.get(url);
        driver.findElement(By.id("username")).sendKeys(un);
        driver.findElement(By.name("pwd")).sendKeys(pw);
        driver.findElement(By.xpath("//div[text()='Login ']")).click();
    }
}
```

Advantages of property File:

It is very light weight when compare to external resource file

It is faster in execution

In order to read the data from the property file we should get the java object of the physical file by using fileinputstream class.

Then by taking the help of property file, load the file and get the value by using the key {by using getproperty ()}

Disadvantages of Property File:

We cannot store more data in the property file because we need to remember the key.

Handling the Excel File

In order to read the data from the excel file we need to use Apache Poi plugin or third-party tool

By using apache poi we can read and write from all the Microsoft documents like .xlsx,.xls,.ppt,.docx,.outlook,etc…

Apache poi is a free and open-source library tool like selenium.

Installation of Apache-Poi

1.Go to google and search for apache poi

2.Click on the first link and navigate to the apache poi community

3.click on poi-bin-5.22 zip file under binary distribution

4.Click on the first link or url and download the zip file

5.Unzip the folder and copy all the jars present in the each and every folder and paste it inside the jar folder of eclipse

6.Add to build path.

Note: -Make sure that all the jars available in lib folder.ooxmi-lib should be also added to the jar folder in eclipse.

Write a script to read the data from the excel file?

```
package qsp;

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.ss.usermodel.WorkbookFactory;

public class Handlingexcelfile {

    public static void main(String[] args) throws IOException {
        //get the java representative objects of the physical file
        FileInputStream fis = new FileInputStream("./data/Testscript.xlsx");
        //Create the workbook or open the Excel in read mode
        Workbook wb = WorkbookFactory.create(fis);
        //get the control of the sheet,row and cell then read the data
        String data = wb.getSheet("Sheet1").getRow(0).getCell(1).toString();
        //print the data to the console
        System.out.println(data);
    }
}
```

Write a script to write the data from the excel file?

```
package qsp;

import java.io.FileInputStream;

public class Writethedatainexcel {

    public static void main(String[] args) throws EncryptedDocumentException, IOException {

        FileInputStream fis = new FileInputStream("./data/Testscript.xlsx");
        Workbook wb = WorkbookFactory.create(fis);
        wb.getSheet("Sheet1").createRow(0).createCell(0).setCellValue("Software Testing");
        FileOutputStream fos = new FileOutputStream("./data/Testscript.xlsx");
        wb.write(fos);
        wb.close();
    }
}
```

Write a script to read the multiple data from excel?

```
package qsp;

import java.io.FileInputStream;

public class Multipledatafromexcel {

    public static void main(String[] args) throws EncryptedDocumentException, IOException {
        FileInputStream fis = new FileInputStream("./data/Testscript.xlsx");
        Workbook wb = WorkbookFactory.create(fis);
        int rowcount = wb.getSheet("Sheet1").getLastRowNum();
        for(int i=1;i<=rowcount;i++)
        {
            String tx1 = wb.getSheet("Sheet1").getRow(i).getCell(0).getStringCellValue();
            String tx2 = wb.getSheet("Sheet1").getRow(i).getCell(1).getStringCellValue();
            System.out.println(tx1+","+tx2);
        }
    }
}
```

Advantages of Data Driven Testing

Reusability of common data and test script data

Modification of Data in excel and external resource file is easier

Maintenance of Data in Excel file ore external file is easier

Test data can be created explicitly before the test execution

We can test the application with the huge volume of data.

Generic Library

→Generic library is one of the components of the automation framework which contains common classes and common methods which can be reusable for all the test script and any project as well.

→Generic classes contain reusable methods which is created by framework developers

→All the generic classes will be present in separate classes and Package name should be com.projectname.generics

Note: - Whenever we create any class or method we should follow the coding standard as shown below

We should not make every method static because it will take more time to execute so make it as non static.

Advantages of Generic Library: -

Common classes which can be used in any test script and any project

Reusability of codes

Code optimization

Test script development is faster

No need to put the effort to re-write the same program

No need to remember the syntax for every option

It can be easily shared to all the team members via Github.

Unit Testing Framework Tool

Junit→ java---------- 1 Test Ng(1,2)

Nunit→.net----------2

Pydev→Python

Rspec→ruby

TestNG (Test Next Generation)

Test NG is a unit testing framework tool which is mainly used for batch execution. (It is a third-party open-source tool).

Basically, testing is used by developers to perform unit testing and it is also used in Selenium to perform black box testing.

Test Ng is a plugin for eclipse and it is inspired by Junit and Nunit with some additional frames.

Advantages or Additional features of TestNg

Batch Execution(Run multiple test cases)

Group Execution(To run only smoke test cases or integration test cases)

Parallel Execution (Execute in Chrome, Firefox, MS Edge. In same time)

Generates the Reports automatically(HTML Report)

Listeners Features

Additional Annotations

Run only the failed test script

Installation of TestNG

Go to Eclipse and Click on help and select market place

Type Test Ng in search text box and click on Search button (Enter)

Click on Install button present under Test Ng for Eclipse

Click on confirm and select the radio button as I Accept License

Click on Finish (Click on Install anyway and restart now button)

- Another way of installing Test Ng

Eclipse→help→install new software→work with→type

https://testng.org/testng-eclipse-update-site

After installation in order to check or verify TestNg tool is installed or not,go to the eclipse→window→Showview→Others→expand java folder→Search for Test Ng

- After Testng Installation to the eclipse add Testng library for all the projects in eclipse(required projects)
- Right click on the project go the build path and Select add libraries
- Select Testng & click on next and Click on Finish

Note: Whenever we using TestNg or TestNg classes we should not use the following thing

- No main method (Do not use the main method)
- No system.out.println();
- Do not use any default package

In case of Automation, unit testing framework tool like Test Ng will be used to achieve batch execution without any manual interruption. And it will generate HTML Report and also provide screenshot for the failed test script.In order to achieve batch execution every test script automation is executed using Testng Annotation.TestNg also handle Framework components during the batch execution and it is the main controller of the framework.

```
package qsp;

import org.testng.Reporter;
import org.testng.annotations.Test;

public class DemoTNG {

@Test

public void TestDemo() {
    Reporter.log("Welcome");
    Reporter.log("Welcome to Tiffion Box", false);
    Reporter.log("Welcome to TestNg", true);
    System.out.println("Hi");
}
}
```

Note: -

- Whenever we execute the above class it will automatically generate the execution result(report) in HTML format

- In order to see it do the following procedure

  Refresh the Java Project(F5) which will display the test-output folder and right click on the emailable icon.report.html file. Go to open with and Select Web browser.
- In order to open the report in Excel format right click on the report and select export to Microsoft Excel and then click on Import and Ok.

System.out.println("Hi")→Print only on the console
Reporter.log("Welcome to Test Ng" , true)→Print on console & report
Reporter.log("Welcome to Tiffin Box", false)→Print only on the Report
Reporter.log("Welcome")→Print only on the report