**1.Basic Method in webDriver Interface :**

* **Get(java.lang.string.url)a :** Load a new web page in the current browser window.
* **findElement():**  find the element in the webpage
* **manage() :** Gets the option interface.(like maximize,minimize etc.)
* **getCurrentUrl() :** get a string representing the current URL that the browser is looking at. (Return string value)
* **getTitle() :** The Title of the current page.(Return string value)
* **getPageSource() :** Get the source of the last loaded page.(return string)
* **navigate() :** An abstraction allowing the driver to access the browser’s history and to navigate to a given URL.
* **quit() :** quites this driver, closing every associated window.
* **close() :** close the current window, quitting the browser if it’s the last window currently open.

**2.Some More Basic methods in WebDriver Interface :**

* **getWindowHandle() :**  Return an opaque handle to this window that uniquely identifies it within this driver instance.
* **getWindowHandles() :** Return a set of window handles which can be used to iterate over all open windows of this WebDriver instance by passing them to switchTo().WebDriver.Options.window()
* **switchTo() :** send future commands to a different frame or window.
* **findElement(By by):** find the first WebElement using the given method.
* **findElements(By by) :** find all elements within the current page using the given mechanism.

**3.Working with WebElements in Selenium WebDriver**

* **sendKeys(java.lang.CharSequence…KeysToSend):** use this method to simulate typing into an element, which may set its value.
* **clear() :** if this element is a text entry element, this will clear the value.
* **click() :** click this elements.
* **getAttribte(java.lang.String name) :** get the value of the given attribute of the element.
* **getCssValue(java.lang.String propertyName):** get the value of a given css property.
* **getLocation() :** where on the page is the top left-hand corner of the rendered element?
* **getSize() :** what is the width and height of the rendered element?
* **getTagName() :** get the tag name of this element.
* **getText() :** get the visible text.
* **isDisplayed():** is this element displayed or not? This method avoids the problem of having to parse an element’s “style” attribute.
* **isEnabled():** is the element currently enabled or not? This will generally return true for everything but disabled input elements.
* **isSelected() :** determine whether or not this element is selected or not.

**Globalsqa,seleniumeasy,demo.seleniumeasy.com,**

1. **Handleframe-** In frame we use to switchTo() method. Frame ko switch krne ke liye. Frame is nothing it is a seprate HTML documents.
2. **->Alert-** it have some method like

**Alert object=driver.**switchTo().alert();

Note- we can also call the alert method using create alert class object.

Accept(); accept the alert box

Dismiss(); dismiss the alert box

getText(); get the text of alert box

sendKey(); send text in alert box

1. **Multiple Winidow Handle-** two method hote hai or eska use multiple window ko handle krne ke liye kiya jata hai
2. **getWindowHandle()-return the handle for the current window**
3. **getWindowHangles()- it return set becs set has not duplicate value.**

**X-PATH-** these type id is dynamic its change when we refresh the page.

Dynamic Id-

id=test\_123,

id=test\_556,

id=test\_test\_526\_test(used start-with method)

id=1253\_test\_t

**Xpath- //tagname[contains(@id,’test\_’)]**

id=test\_123,

**Xpath- //tagname[starts-with(@id,’test\_’)] (**start-with is a method)

Example is the id Is – test\_test\_526\_test

id=1253\_test\_t

Xpath= //tagname[ends-with(@id,’\_test\_t’)]

**How to create on Xpath :-** it is specially for link for custom xpath

All the link tag is represent is <a> tag in HTML.

***Xpath= //a[contains(text(),’Text Write here’)]***

Here text() is a function don’t use @ before the text

IF MORE THAN ONE ELEMENTS HAS SAME XPATH :-

It is done by using indexing

-first put xpath in **parenthesis ()[put element** index **here]**

**(xpath)[index of element]**

**Example**- (//button[@class="multiSelectExpand"])[4]

**Selenium by Code Studio**

What is Selenium:-

* it is **open source** Automation test suite for **web based application** across **different browsers** (Chrome,Firefox,Opera etc.) and different platforms (Windows, Linux, mac os)
* It supports multiple programming language: Java, C#, Python, ruby etc.

**Component of Selenium:-**

* Selenium suite

1. selenium IDE
2. selenium RC(remote control)
3. selenium WebDriver
4. Selenium grid

Selenium IDE :-

* Selenium IDE stands for selenium integrated development environment
* It is simple and easy to use
* No prior programming knowledge is required
* It is used as prototyping tool

Selenium WebDriver:-

* It is a component of selenium suite
* It is a java interface, implemented by browser classes
* It is an API(Application Program Interface), mediator between browser and client libraries
* It has various commands to interact with web elements

What is selenium grid:-

It allows test cases to run parallelly in

* Different machines
* Different platforms
* Different browsers

Advantages of selenium WebDriver:-

* It is an open source tool
* It supports multiple browsers (Firefox, Chrome, Opera etc.)
* It supports multiple platforms(windows, Linux, Mac os etc)
* It supports multiple language (java, python, ruby, c# etc.)

Disadvantages of selenium WebDriver:-

* It supports only web based application and doesn’t support window based application
* It does not provide facility for data driven testing
* It does not have reporting feature

**Third party tools/library to overcome disadvantages :-**

Window operation – Autolt

Data driven testing – Apache POI

Reporting – Extent Report

Question :-What are the testing type supported by Selenium?

Answer:- Regression Testing and functional testing both

Steps how to download and install selenium Simple JAVA Project:-

* Download selenium in zip format
* Unzip krne ke baad project ki properties m java build path select krke libraries m jake add external Jars pr click kre or selenium jar ko add kre

Steps how to download and install selenium MAVEN Project:-

* Simply add selenium dependency POM.xml file

Interviews:-

Q. how to launch browser in selenium web deriver?

A. we have two statement to launch the web browser

a. System.setProperty(“Key”,”Value”);

key browser to browser vary krega or Value m path denge browser class ki exe file ka

for Chrome : System.setProperty(“webdriver.chrome.driver”,”Path of webderiver.exe”);

b. WebDriver driver = new BrowserClass();

for Chrome : Webdriver deriver = new ChromeDriver();

Q. Web driver commands to open web page ?

|  |  |
| --- | --- |
| Get() method | Navigate() method |
| driver.get(“URL”); | Driver.navigate.to(“URL”); |
| Webdriver will wait until the page has fully loaded before returning the control to test or script. | It is only responsible for redirecting the page and then returning immediately. |
| It can’t track the history of the browser. | It tracks the browser history and can perform back and forth in the browser. Page refresh option is also available in navigate() method.  driver.navigate().refresh(); driver.navigate().forward();  driver.navigate().back(); |

Q. Webdriver command to capture title & URL of the web page?

Title : driver.getTitle();

URL : driver.getCurrentUrl();

Session 3

Q. what is Locators ?

A. It is used to identify web element on web page

Locating Strategies:-

1. Basic Locating strategies- ( By ID, By Name, By Class Name, By Tage Name, By Link Text, By Partial Link Text)
2. Customized locating strategies - (By CSS & By XPath)

WebDriver methods to find web elements :-

1. findElement() = - it return single web element
2. findElements() = it return list of elements

=>if element not found on webpage findElement() return Exception and findElements() return List of zero element.

Basic Locators =>

1. Id
2. Name
3. Class
4. Tag Name
5. Link Text
6. Partial Link Text

Session 5

Q. What is XPath ?

* + XPath stands for XML Path
  + It is a query language to locate nodes in XML document
  + It gives complete address of web element.

Q. How to get XPath Web Element?

* 1. Generate automatically using browser plugin

1. Chrome Browser- SelectorsHub
2. FireFox Browser – Firepath, Chropath

* 2. Write Xpath manually

Different ways & methods for writing XPath

* Absolute Xpath
* Relative xpath
* Single attribute
* Multiple attribute
* AND
* OR
* contains()
* starts\_with()
* text()
* position()
* Last()

Absolute XPath :-

* Begins from the **root** of the element to be located. (root note se element tk complete path lega ye)
* Starts with the **/** symbol
* Example –

Webpage- [**https://www.saucedemo.com/**](https://www.saucedemo.com/)

Web element – **username**

/html[1]/body[1]/div[1]/div[1]/div[1]/ div[2]/ div[1]/ div[1]/ div[1]/ form[1]/ div[1]/ input[1]/

Relative Xpath

* A relative XPath begins from the element to be located and not from the root.
* Starts with the **//** symbol
* Syntax :-

**XPath = //tagname[@attribute name=’value’]**

Xpath – Using single attribute

//<html tag>[@attribute\_name=’attribute\_value’]

Xpath – Using multiple attribute

//<html tag>[@attribute\_name1=’value1’][@attribute\_name2=’value2’]

**Note:-** “\*” after double slash is to match any tag with the desired text.

**Example:** //\*[@attribute=’value’]

XPath- using AND expression :

//html tag[@attribute1=’value1’ **and** @attribute2=’value2’]

XPath- using OR expression :

//html tag[@attribute1=’value1’ **or** @attribute2=’value2’]

//**\***[@attribute1=’value1’ **or** @attribute2=’value2’]

XPath- using contains() :

Contains() method helps in locating the UI element(Web Element) using partial text.

(kuch value change hoti rhti hai like user name, id , etc

Syntax :- **//<html tag>[contains(@attribute\_name,’attribute value’)]**

**//\*[ contains(@attribute\_name,’attribute value’)]**

**//tagname[contains(@id,’test\_’)]**

id=test\_123,

id=test\_556,

id=test\_test\_526\_test(used start-with method)

id=1253\_test\_t

Xpath- starts-with() : when initial text is same and ending text is dynamically change.

Syntax : //html tag[starts-with(@attribute,’attribute value’)]

**//tagname[starts-with(@id,’test\_’)] (**start-with is a method)

Example is the id Is – test\_test\_526\_test

Xpath- ends-with():- when ending text is same and initial text is dynamically change.

id=1253\_test\_t

Xpath= //tagname[ends-with(@id,’\_test\_t’)]

Xpath- using text() method

// tagname[text()=’text displayedon webpage’]

**How to create on Xpath :-** it is specially for link for custom xpath

All the link tag is represent is <a> tag in HTML.

***Xpath= //a[contains(text(),’Text Write here’)]***

Here text() is a function don’t use @ before the text

IF MORE THAN ONE ELEMENTS HAS SAME XPATH :-

It is done by using indexing

-first put xpath in **parenthesis ()[put element** index **here]**

**(xpath)[index of element]**

**Example**- (//button[@class="multiSelectExpand"])[4]

Xpath- Using position() method :

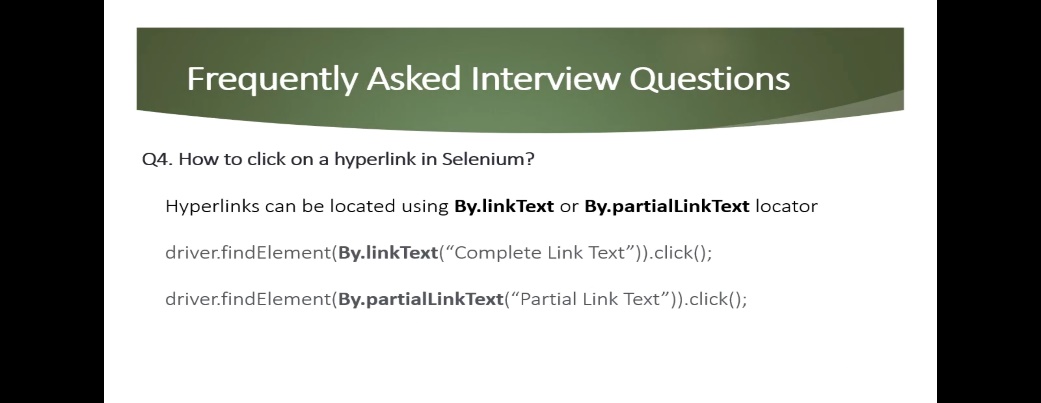
Eska use jb krte hai jb parent node ke ander multiple child node hote hai or hume kisi particular position ke child node ko access krna hai to hum position method use krte hai

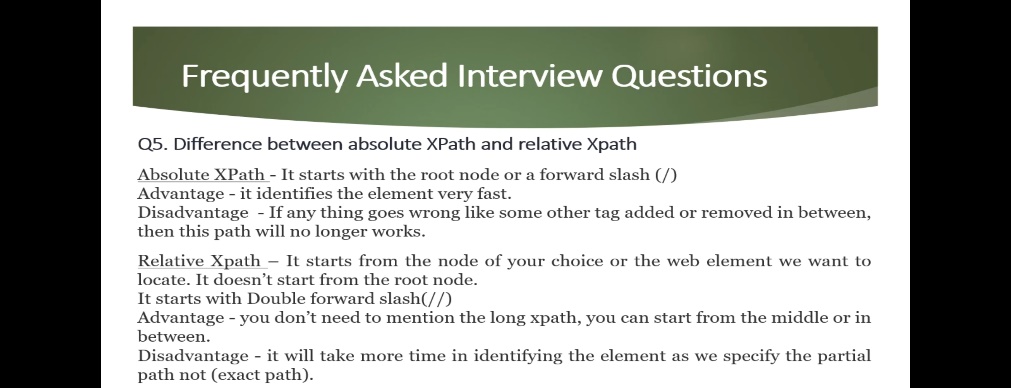
**//tag name[@attribute-type=’attribute-value’][position()=value]**

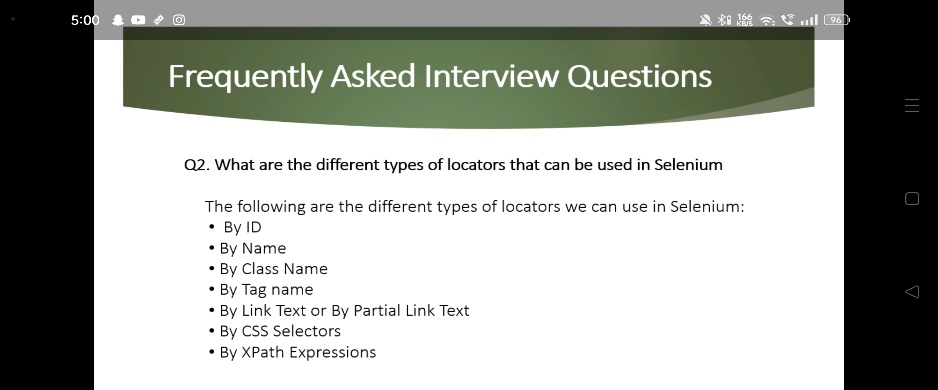
Xpath- using last() method :

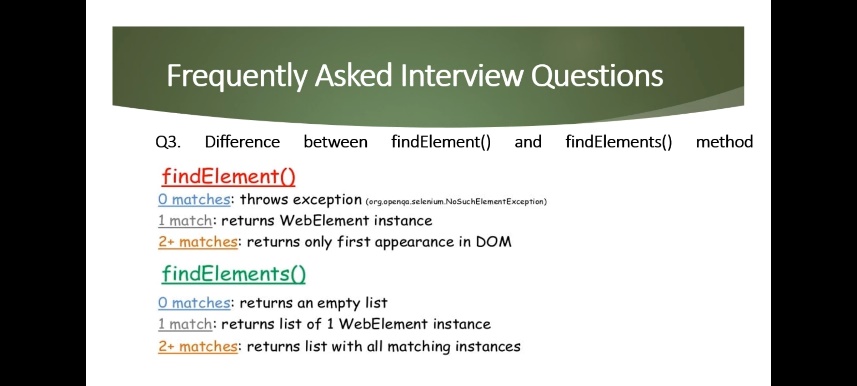
Jb last child node access krna ho tb use krte hai

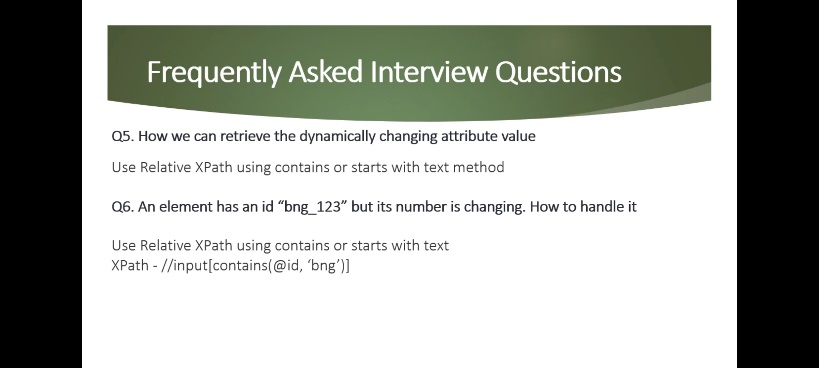
**//tag name[@attribute-type=’attribute-value’][last()]**

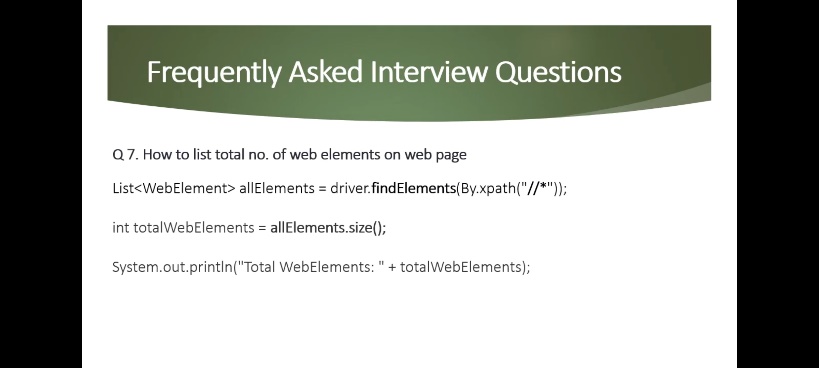












Session- 6

**Handling of Dropdown:**

Select class is used to interact with Dropdown web elements

* Commonly used command for dropdown interaction.

|  |  |
| --- | --- |
| ACTION | WEB DRIVER METHOD |
| To select/deselect value from dropdown | selectByVisibleText()  deselectByVisibleText()  selectByValue() // deselectByValue()  selectByIndex() // deselectByIndex()  deselectAll() |
| To check if the dropdown element **allows multiple selections** at a time | isMultiple() |
| To get options from dropdown | getOptions() |

// Handling dropdown using select class

**package** SeleniumDemo;

**import** java.util.List;

**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** Test5 {

**public** **static** **void** main(String[] args) {

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\Ankur.3.Kumar\\Downloads\\chromedriver.exe");

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

driver.get("https://way2automation.com/way2auto\_jquery/index.php");

WebElement element=driver.findElement(By.*name*("country"));

//select by value

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

//select by visible text

dropdown.selectByVisibleText("Austria");

//select by index

dropdown.selectByIndex(4);

//select by value

//dropdown.selectByValue("Belarus");

// dropdown multiple hai ya nhi

**if**(dropdown.isMultiple())

{

System.***out***.println("drop down is multiple");

}

**else** {

System.***out***.println("dropdown is not multiple");

}

// Print all dropdown element =

List<WebElement> alldropdown=dropdown.getOptions();

**for**(WebElement e1:alldropdown)

{

System.***out***.println(e1.getText());

}

}

}

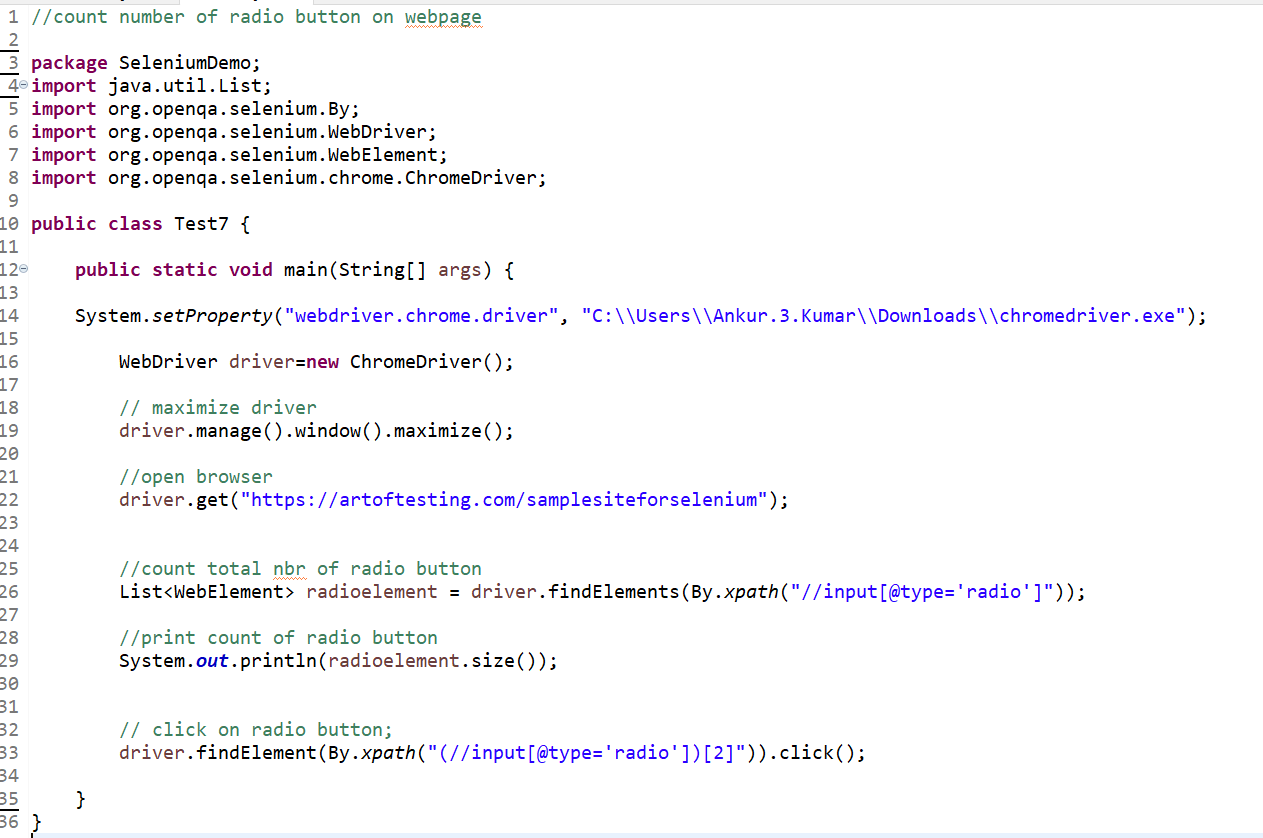
WHAT ARE BROKEN LINKS:-(Sessoin 6)

* Broken links are links or URLs that are not reachable. They may be down or not functioning due to some error
* The user entered an improper/misspell URL
* The destination web page is down, moved, or no longer exists.
* With activated firewall settings, also the browser cannot access the destination web page at times.

If the link value is 200 or under 400 it is valid and if greater than 400

COUNT NO. OF RADIO BUTTON ON A WEB PAGE :

* Find **common locator** for all radio buttons
* Capture **list** of radio button using **FindElements()** methods
* Get count of elements in a **list** using **size(**) method to obtain total no. of radio button on webpage.



HOW TO CLICK ON IMAGE LINKS IN SELEMNIUM WEBDRIVER :

* Image link are the **links** in web pages represented by an image which when clicked **navigates to a different window or page.**
* In this case, we can use **By.cssSelector or By.xpath** to locate the image link.

****

HOW TO PERFORM MOUSE OPERATION :

* In selenium webdriver, **Action class** is used for mouse operation
* Right click= Right click performe by contextClick(webelement).perform();
* Double click= Double click performe by doubleClick(webelement).perform();
* Drag drop= dragAndDrop(sourceElement, targetelement).perform();
* Mouse Hover = moveToElement(WebElement).perform();

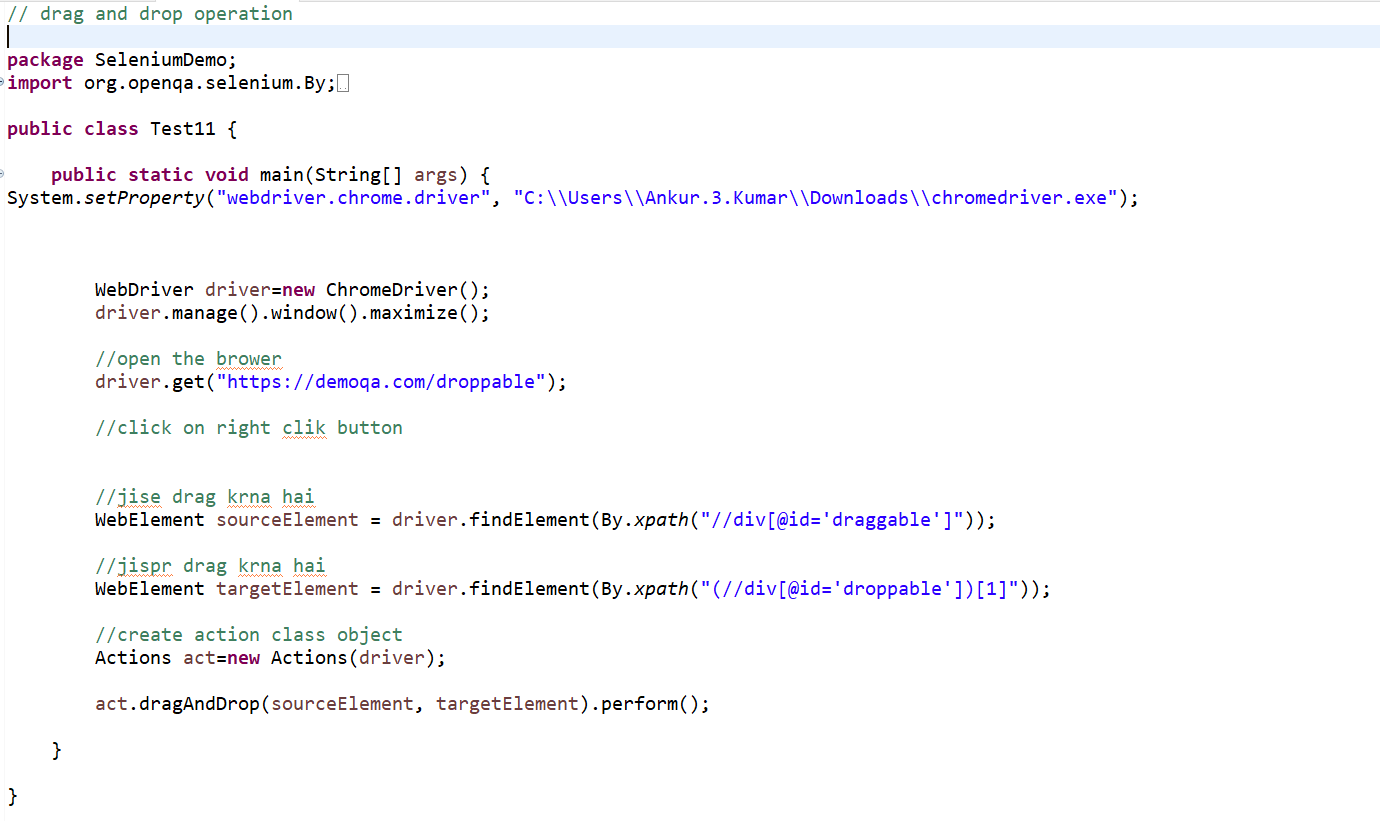
Right click -



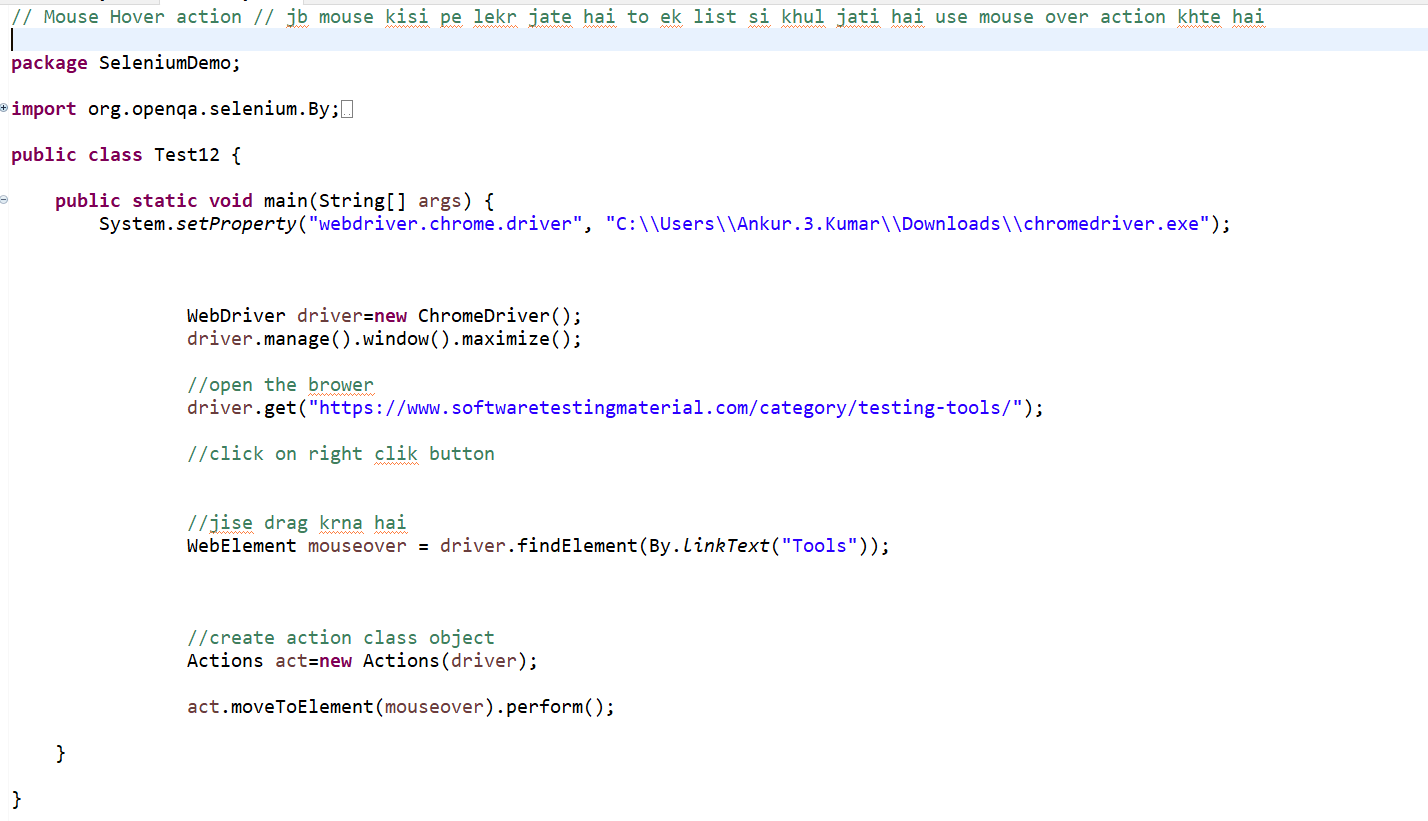
Double click -



Drag and Drop –



Mouse Hover Action –

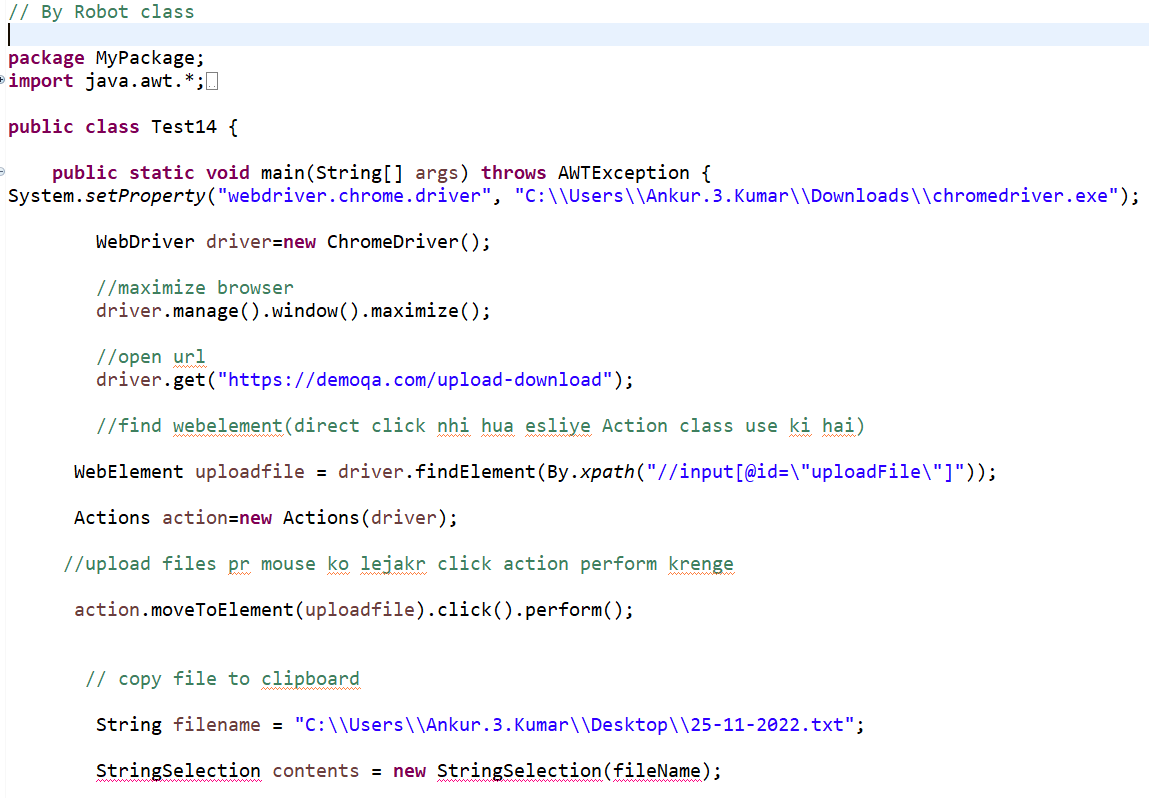


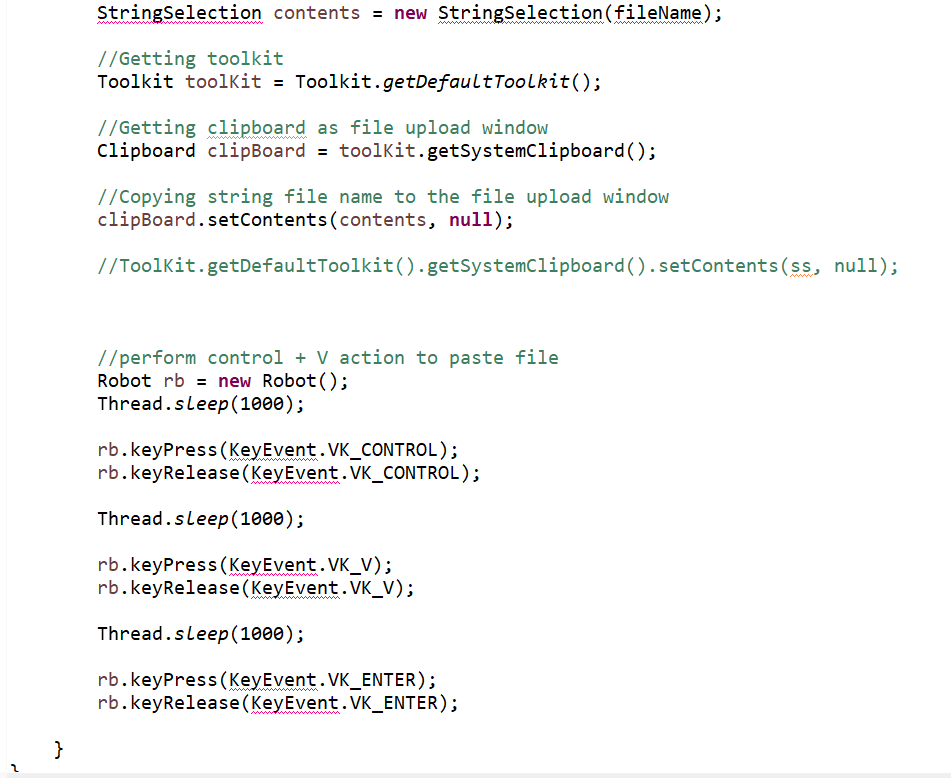
**Session 12**

How to Upload files : there is two method to upload files

1. SendKeys() method : -( jis webelemet ke xpath m ‘**type=file**’ key given hai usme sendKeys() method ka use kr skte hai or file upload kr skte hai.
2. Robot Class: - if “type=file” is not present in HTML property of webElement then we are using Robot class.





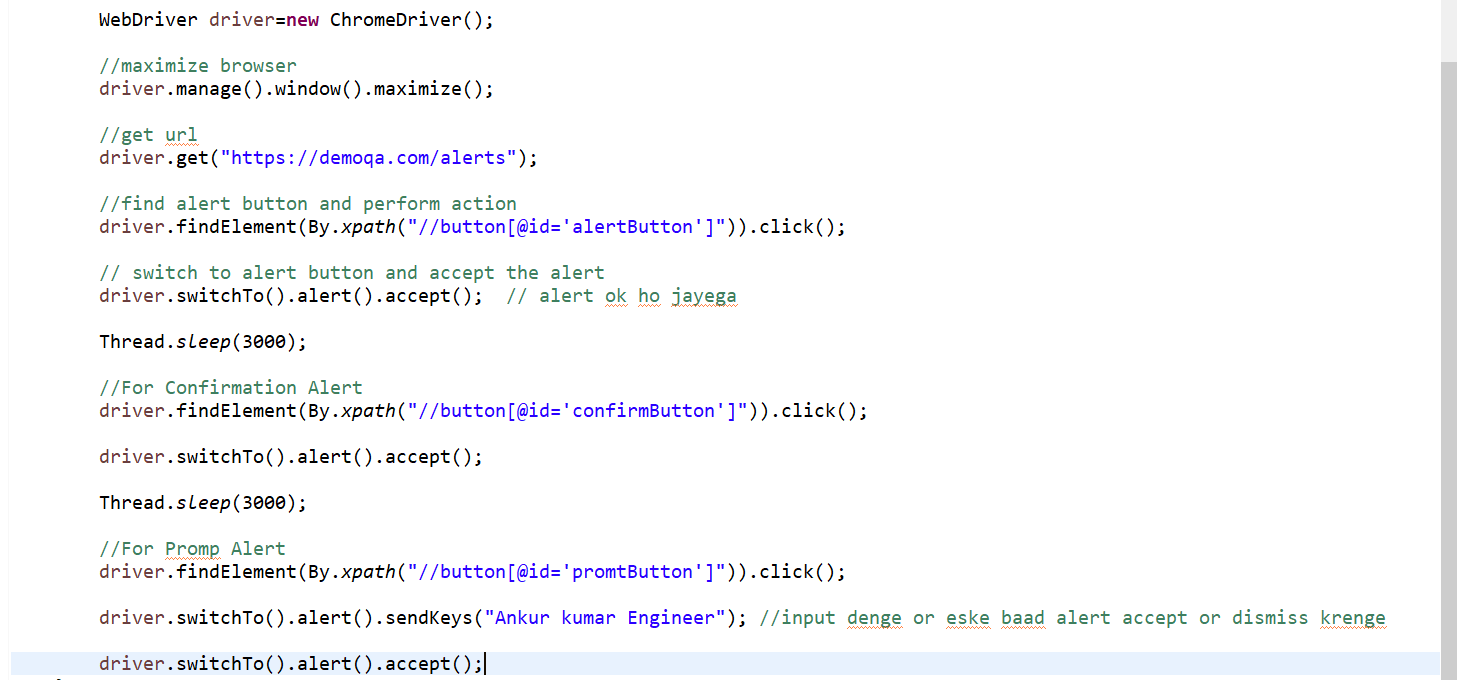


What is Alert in Selenium :- An Alert in selenium is a **small message box** which **appears on screen**

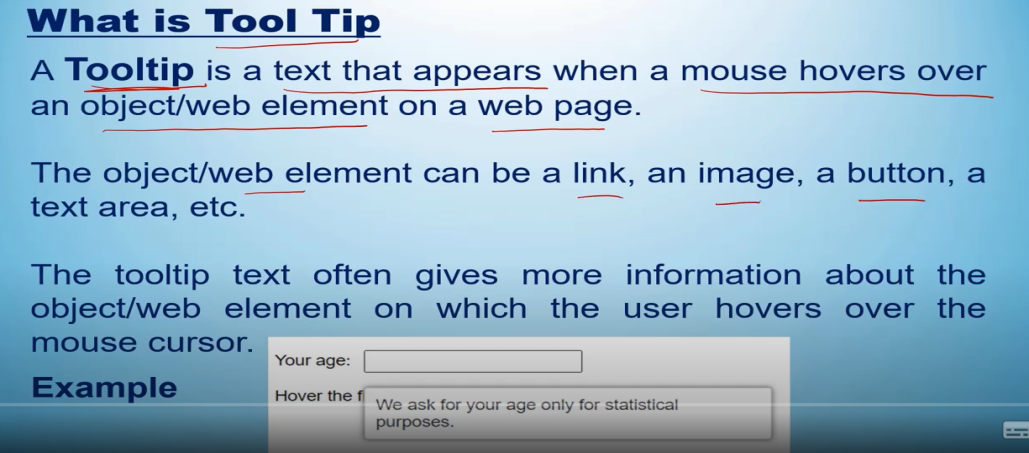
To give the **user some information or notification.**

Type of Alerts :-

1. **Simple alert** – display information or warning
2. **Confirmation alert**- asks permission to do some type of operation
3. **Prompt Alert**- Pompt alert asks some input from the user.



How to handle ToolTit :-



//Handle ToolTip

**package** SeleniumDemo;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

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

**public** **class** Test16 {

**public** **static** **void** main(String[] args) {

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\Ankur.3.Kumar\\Downloads\\chromedriver.exe");

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

//maximizing browser

driver.manage().window().maximize();

//url open

driver.get("https://testautomationpractice.blogspot.com/");

//find element, get attribute tooltip text ek string m le lega

String ExpectedToolTip = "We ask for your age only for statistical purposes.";

String OrignalToolTip = driver.findElement(By.*xpath*("//input[@id='age']")).getAttribute("title");

// verify tooltip

**if**(ExpectedToolTip.equals(OrignalToolTip))

{

System.***out***.println("TestCase is Pass");

}

**else** {

System.***out***.println("TestCase is fail");

}

}

}

Session – 16

HOW TO CAPTURE A SCREENSHOT :-

* To capture screenshot, we will use method “getScreenShotsAs()” of special interface called “TakeScreenshot”

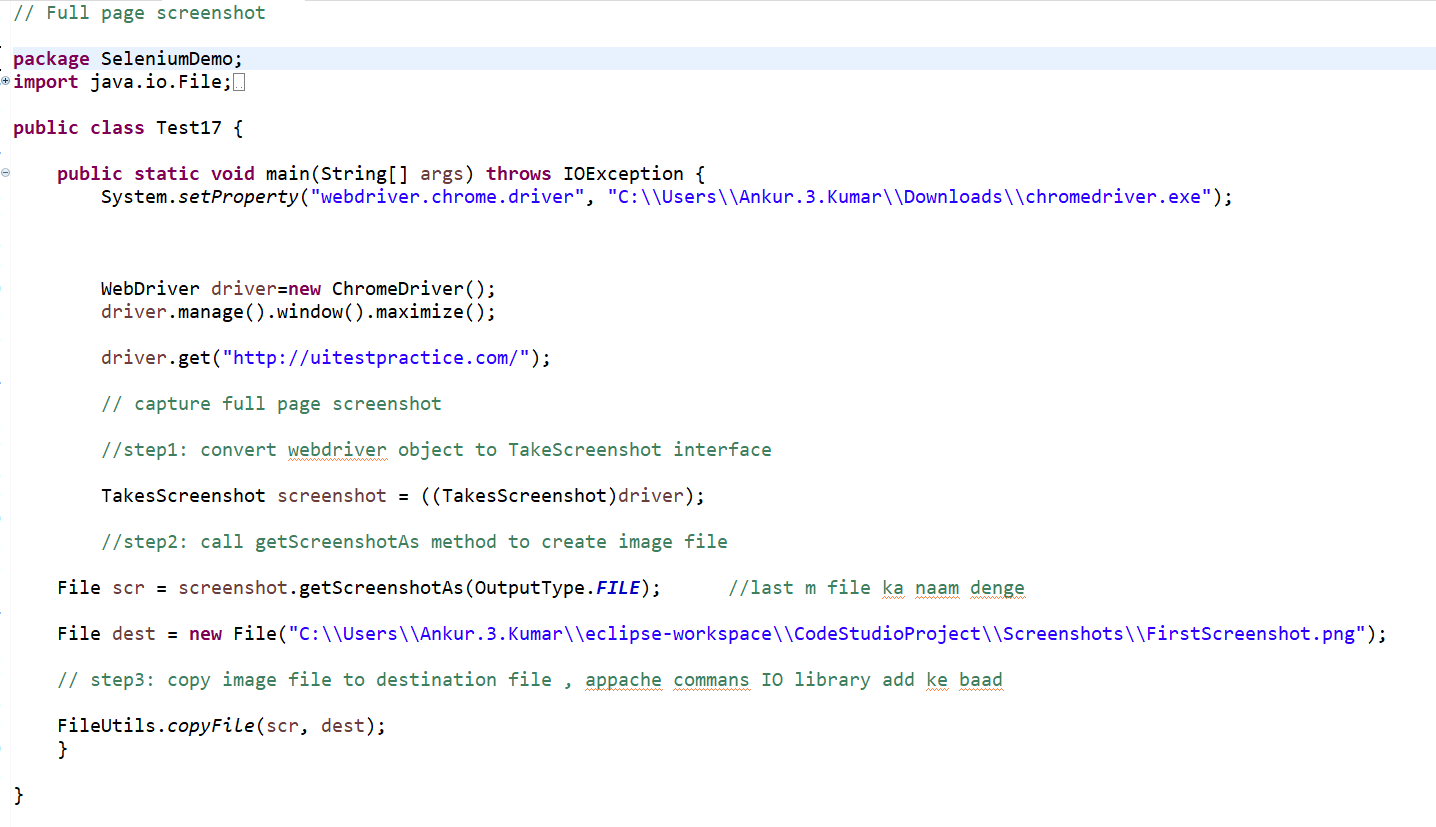
1. Convert web driver object to **TakeScreenshot object**

TakesScreenshot scrShot = ((TakesScreenshot)**webdriver**);

1. Call **getScreenshotAs** method to create image file

File srcFile = **scrShot.getScreenshootAs**(OutputType.FILE);

1. Copy file to desired location by using library **“Apache Commons IO- FileUntils”**



Session-17

HOW TO OPEN NEW TAB OR WINDOW:-

In **selenium version 4** , you can open new window & tab using **“newWindow”** command.

Example :-

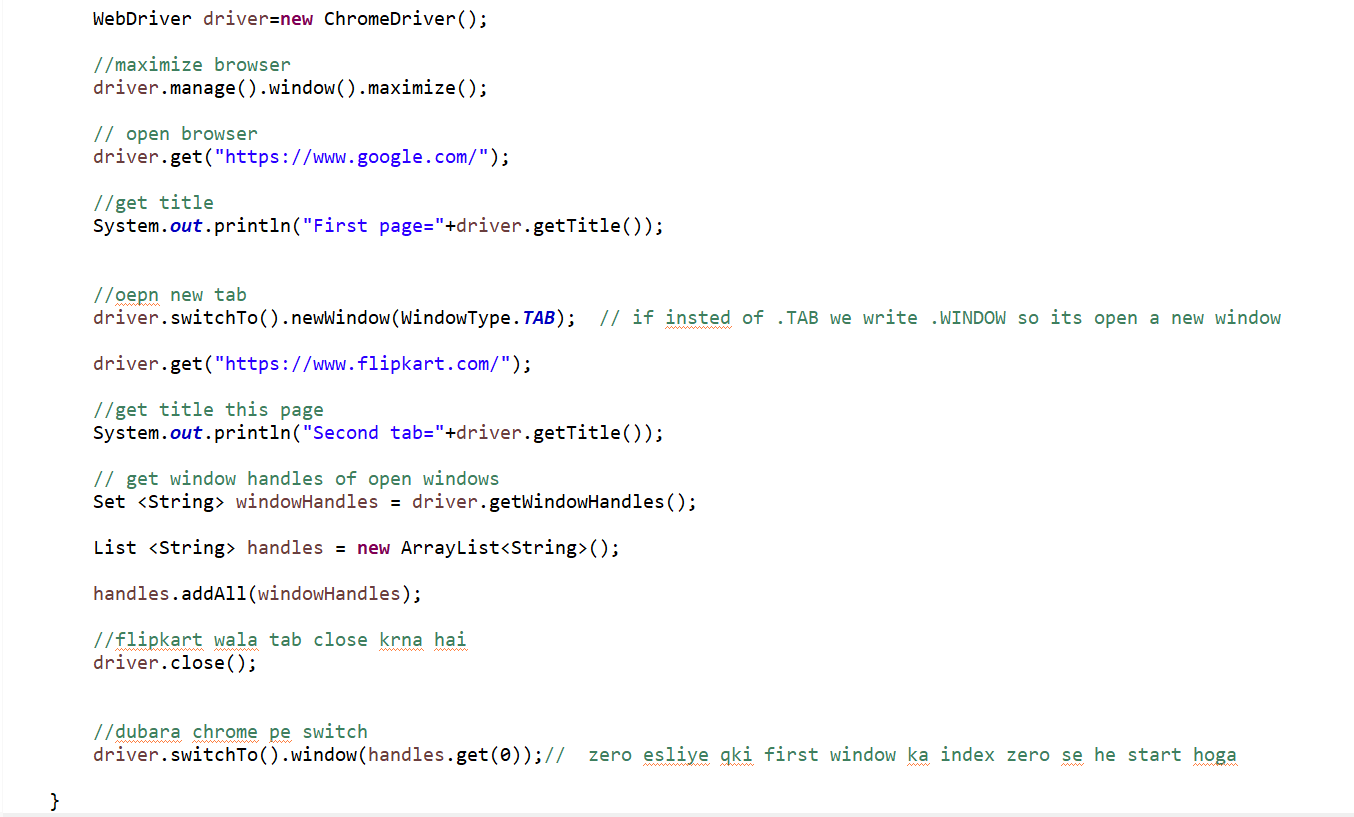
* WebDriver driver = new ChromeDriver();

Open new window-

* driver.switchTo().**newWindow**(WindowType.**WINDOW**);

Open new Tab-

* driver.switchTo().**newWindow**(WindowType.**TAB**);



Session- 18

SYNCHRONISATION ISSUE :

* Process of **matching test automation tool speed** with **speed of application under test** is **synchronization.**
* Whenever **webdriver** tries to **perform operation on element not laded in application under test,** in such case webdriver will **throw exception “**NoSuchElementException” or “ElementNotVisibleException”. This happens **due to synchronization issue.**

1. Thread.sleep(ms)
2. Implicit wait

**Thread.sleep(ms):** ye unconditionally hota hai jitney sec esme lenge utne ke liye program pause ho jayega or ye java ka method hai selenium ka nhi hai.

**Implicit wait :** it is a global wait and it’s not webelement specific once we create and applicable for all the webelement.

**Disadvantage**- agr jo time dete hai use jayada time lgta hai element find hone m to element find ni hoga

Syntax:

driver.manage().TimeOuts().implicitlyWait(10,Timeunit.SECONDS); //**before selenium 4**

driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10)); //**after selenium 4**

**Explicit wait :** explicit wait is based on wabelement specific and esme condition bhi de skte hai.

Syntax : explicit wait for dropdown webelement

WebDriverWait **wait** = new WebDriverWait(driver,10) //**before selenium 4**

WebdriverWait **wait =** new WebDriverWait(driver,Duration.ofSeconds(10)); //**after selenium 4**

Wait.until(ExpectedConditions.visibilityOfElementLocated(By.id(“**dropdown**”)));





Session-19

What is JavaScriptExecutor:-

JavaScriptExecutor is an **interface** that helps to execute **JavaScript** through selenium Webdriver.

Syntax:-

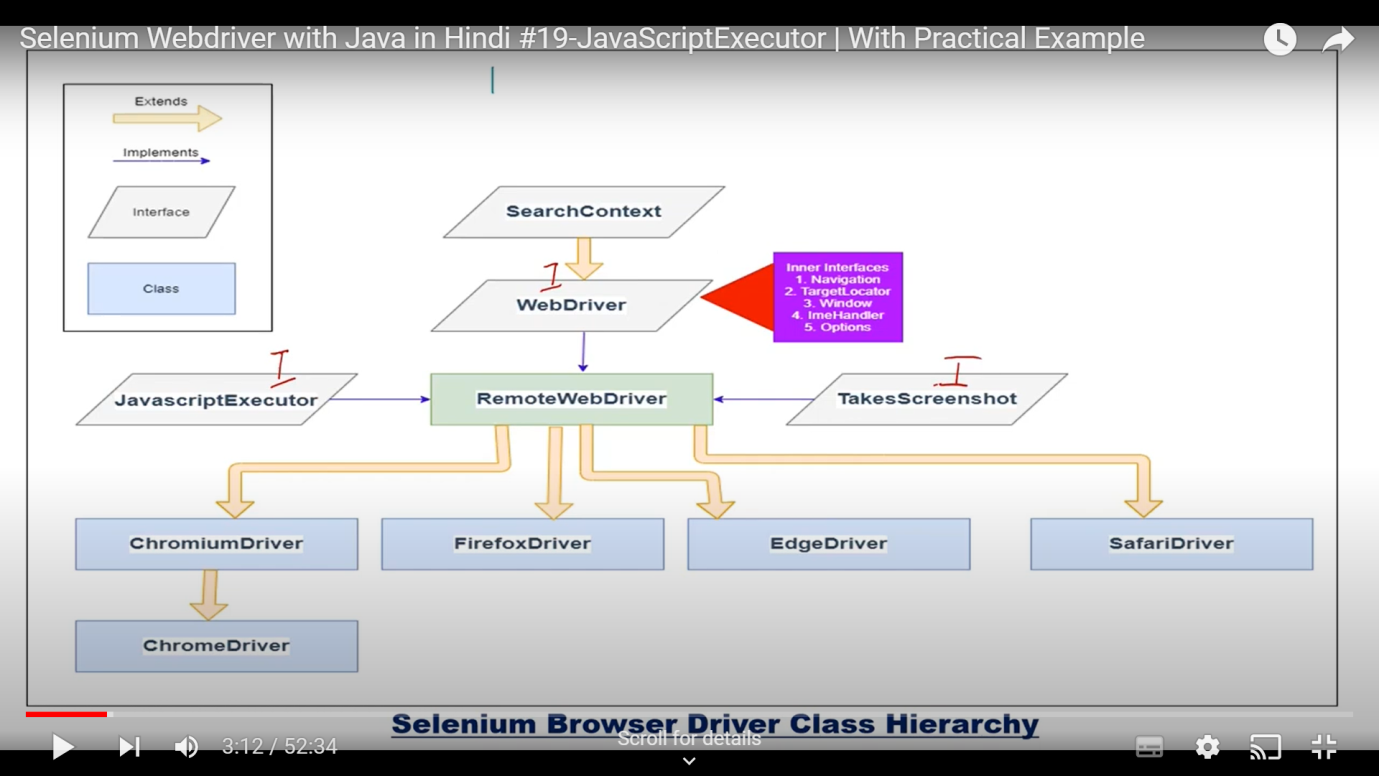
JavascriptExecutor js = (JavascriptExecutor)driver;

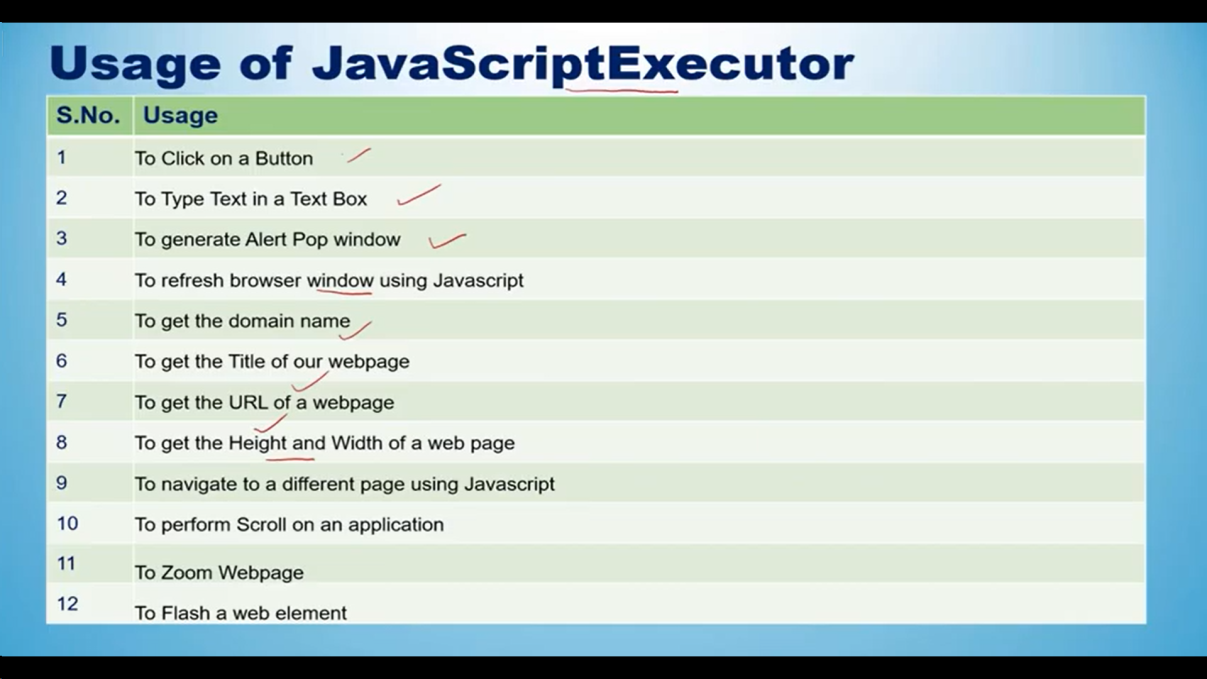
Js.executeScript(Script,Arguments);

Script = this the javascript that needs to execute.

Arguments = It is the arguments to the script. It’s optional.

Returns = could be anything from Boolean, Long, String, List, WebElement, or null.





Session – 20

Frame & Iframe :-

**Frame**  is a HTML tag that is used to **divide the web page into various frames/windows.** Each window can load a separate HTML document. In HTML, **<frame> tag** denotes a frame, and all the frames are present in a **<frameset>** tag.

**Iframe** as **<iframe>**  is also a tag used in HTML but it specifies an inline frame which means it is used to embed some **other document** **or content from other sources/external sources within the current HTML document.**

**How to switch Iframe :**

1. By id or by name
2. By webelement
3. BY index

Sytax:-

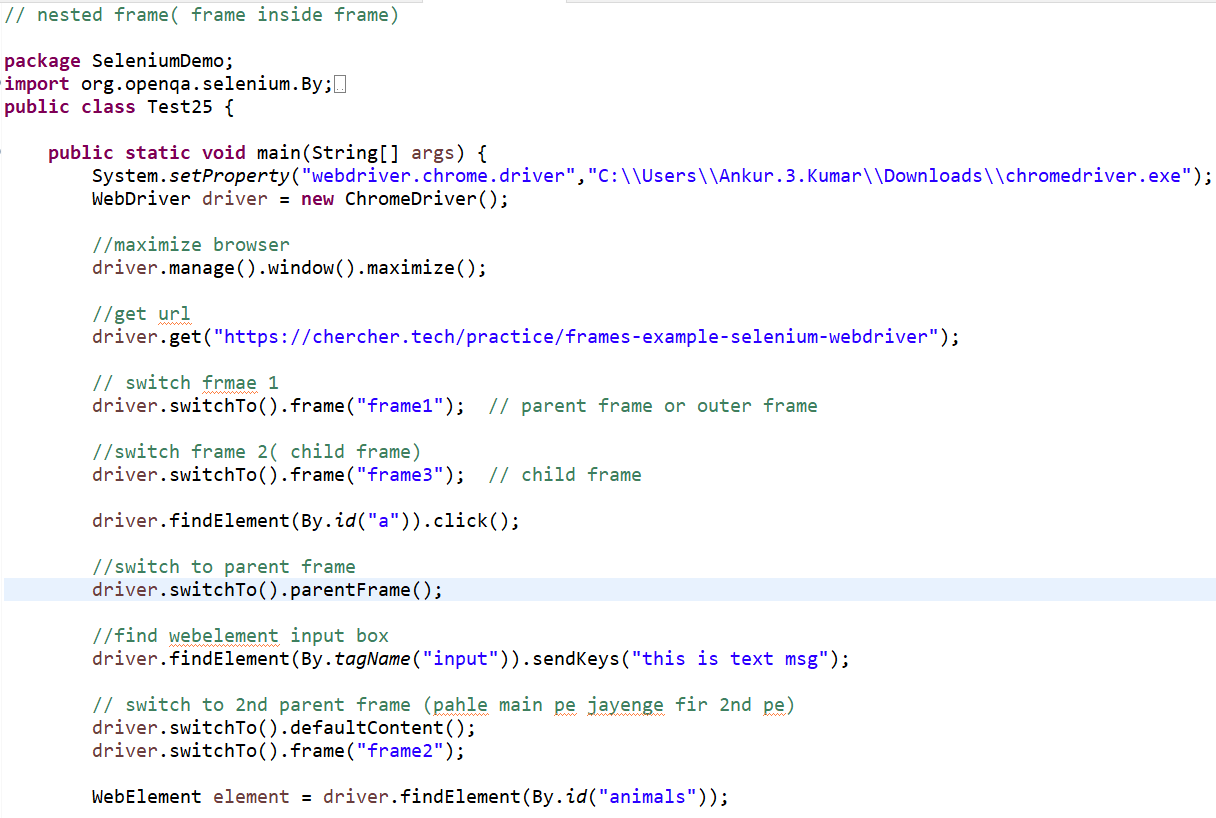
Driver.switchTo().frame()

**Note.** First iframe se dusre iframe directly switch nhi kr skte wapis main pe switch krna hoga uske baad kisi dusre iframe pe switch kr skte hai

1. switchTo.frame(int frameNumber)
2. switchTo.frame(string frameName)
3. switchTo.frame(WebElement frameElement)
4. switchTo().defaultContent()
5. switchTo().parentFrame()









session -21

How to scroll Webpage :- To scroll selenium, you can use JavaScriptExecutor interface that helps to execute JavaScript methods through Selenium Webdriver.

Syntax :

JavascriptExecutor js = (JavascriptEdxecutor) driver;

js.executeScript(Script,Arguments);

1. To scroll down the web page by pixel

executeScript(“window.scrollBy(x-pixels,y-pizels)”);

js.executeScript(“window.scrollBy(0,500)”); // scroll vertically down by 1000 pixels

1. To Scroll down the web page by the visibility of the element

js.executorScript(“arguments[0].scrollIntoView();”,Element);

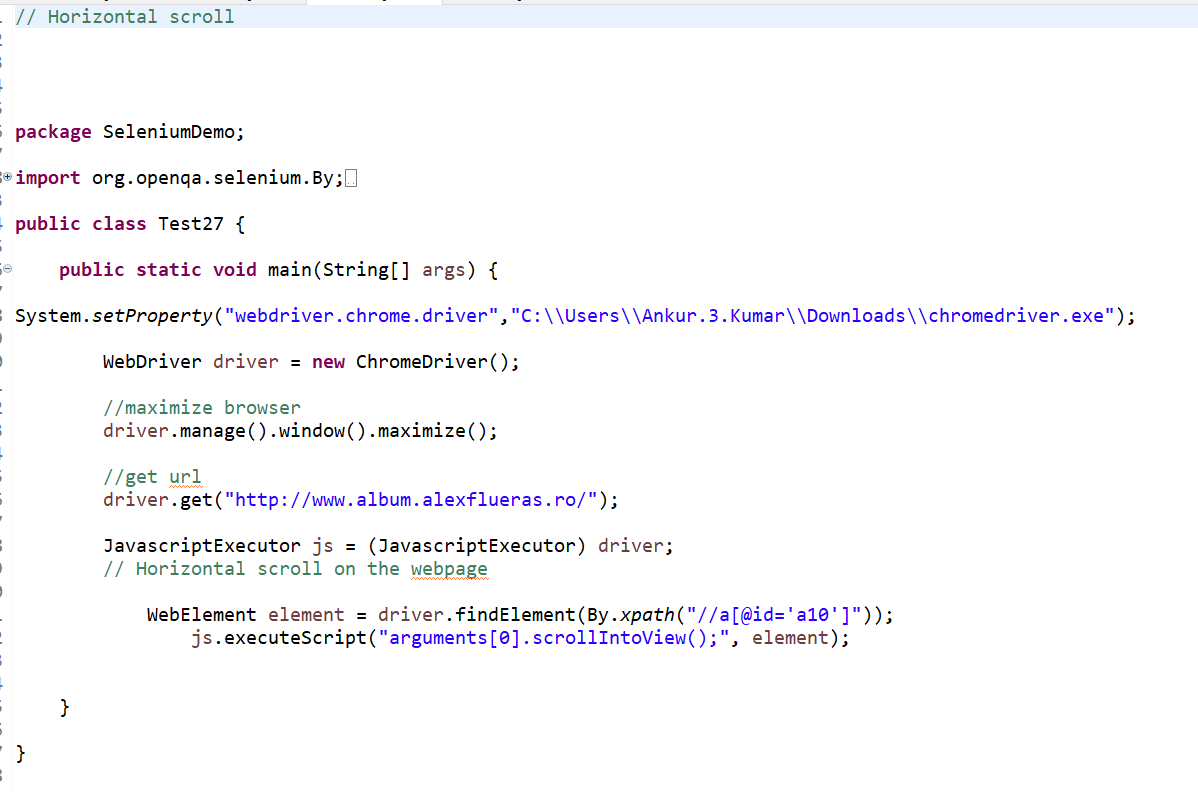
1. To scroll down the web page at the bottom of the page

js.executeScript(“window.scrollTo(0, document.body.scrollHeight)”);

1. Horizontal scroll on the web page

js.executeScript(“arguments[0].scrollIntoView();”,Element);





What is WebDriverManager :-

**Selenium WebDriver :-** is an automation tool widely popular and is useful to run tests against **multiple browsers**  like Google Chrome browser, fireox browser, internet Explorer, etc. we need browser drivers to launch any of these above browser.

WebDriverManager :- **in selenium, is a class that allows users to automate the handling of driver executables like chromedriver.exe, geckodriver.exe etc required by selenium Webdriver API.**

* automate the management of WebDriver binaries.
* Downloads the appropriate driver binaries, if not already present, into the local cache.
* Downloads the latest version of the browser binary, unless otherwise specified.
* Eliminates the need to store driver binaries locally. We also need not maintain various versions of the binary driver files for different browsers.

**System.setProperty(); instead of this statement we write the WebDriverManager class**

**WebDriverManager.chromedriver.setup();**

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

WebDriverManager.chromedriver().setup();

WebDriverManager.firefoxdriver().setup();

WebDriverManager.iedriver().setup();

WebDriverManager.edgedriver().setup();

WebDriverManager.operadriver().setup();

**package WebDriverManagerSeleniumDemo;**

**import org.openqa.selenium.WebDriver;**

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

**import io.github.bonigarcia.wdm.WebDriverManager;**

**public class WebdriverManagerDemo {**

**public static void main(String[] args) {**

**//launch the browser using WebDriverManager**

**WebDriverManager.chromedriver().setup();**

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

**driver.get("https://google.com/");**

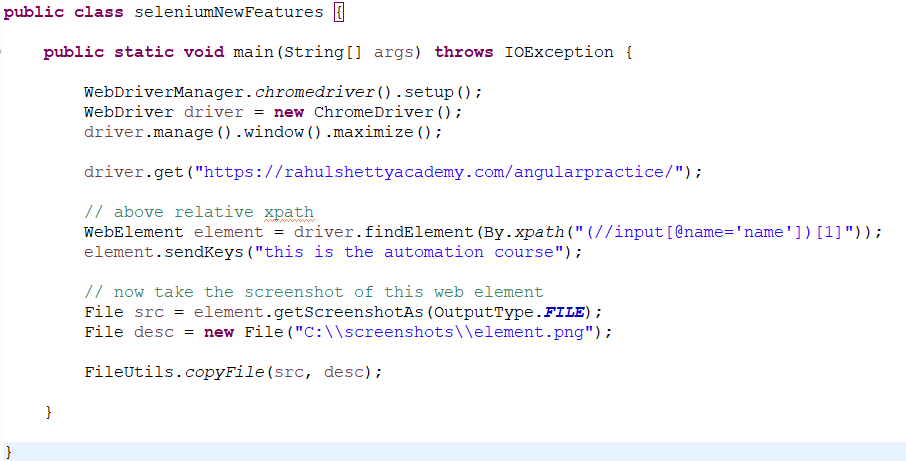
**}**

**}**

**How to Handle Window Authentication Popup :- jo URL open hone se pahle user name or password mangte hai or enko inspect nhi kr skte web element ke jaise qki ye window base application hoti hai**

**28 to 29D tk date picker example hai project m**

**Take single element screenshot –**



Get hight and width of webelement –

