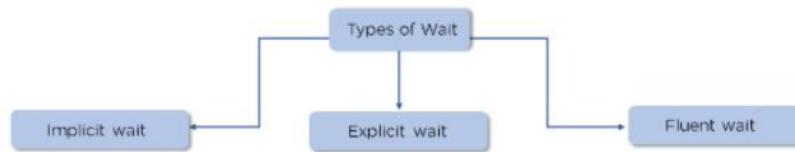


8. What are the types of waits supported by WebDriver?



Implicit wait - Implicit wait commands Selenium to wait for a certain amount of time before throwing a "No such element" exception.

```
driver.manage().timeouts().implicitlyWait(TimeOut, TimeUnit.SECONDS);
```

Explicit wait - Explicit wait is used to tell the Web Driver to wait for certain conditions before throwing an "ElementNotVisibleException" exception.

```
WebDriverWait wait = new WebDriverWait(WebDriver Reference, TimeOut);
```

Fluent wait - It is used to tell the web driver to wait for a condition, as well as the frequency with which we want to check the condition before throwing an "ElementNotVisibleException" exception.

```
Wait wait = new FluentWait(WebDriver reference).withTimeout(timeout, SECONDS).pollingEvery(timeout, SECONDS).ignoring(Exception.class);
```

9. Mention the types of navigation commands

`driver.navigate().to("https://www.ebay.in/");` - Navigates to the provided URL

`driver.navigate().refresh();` - This method refreshes the current page

`driver.navigate().forward();` - This method does the same operation as clicking on the Forward Button of any browser. It neither accepts nor returns anything.

`driver.navigate().back();` - This method does the same operation as clicking on the Back Button of any browser. It neither accepts nor returns anything.

12. How to type text in an input box using Selenium?

`sendKeys()` is the method used to type text in input boxes

Consider the following example -

`WebElement email = driver.findElement(By.id("email"));` - Finds the "email" text using the ID locator

`email.sendKeys("abcd.efgh@gmail.com");` - Enters text into the URL field

`WebElement password = driver.findElement(By.id("Password"));` - Finds the "password" text using the ID locator

`password.sendKeys("abcdefgh123");` - Enters text into the password field

13. How to click on a hyperlink in Selenium?

```
driver.findElement(By.linkText("Today's deals")).click();
```

The command finds the element using link text and then clicks on that element, where after the user would be redirected to the corresponding page.

```
driver.findElement(By.partialLinkText("Service")).click();
```

The above command finds the element based on the substring of the link provided in the parenthesis and thus `partialLinkText()` finds the web element.

Write a code snippet to perform right-click an element in WebDriver.

We will use **Action class** to generate user event like right-click an element in WebDriver.

```
Actions action = newActions(driver);  
WebElement element = driver.findElement(By.id("elementId"));  
action.contextClick(element).perform();
```

34) Write a code snippet to perform mouse hover in WebDriver.

```
Actions action = newActions(driver);  
WebElement element = driver.findElement(By.id("elementId"));  
action.moveToElement(element).perform();
```

14. How to scroll down a page using JavaScript?

scrollBy() method is used to scroll down the webpage

General syntax:

```
executeScript("window.scrollBy(x-pixels,y-pixels)");
```

First, create a JavaScript object

```
JavascriptExecutor js = (JavascriptExecutor) driver;
```

Launch the desired application

```
driver.get("https://www.amazon.com");
```

Scroll down to the desired location

```
js.executeScript("window.scrollBy(0,1000)");
```

The window is not scrolled vertically by 1000 pixels

15. How to assert the title of a webpage?

Get the title of the webpage and store in a variable

```
String actualTitle = driver.getTitle();
```

Type in the expected title

```
String expectedTitle = "abcdefgh";
```

Verify if both of them are equal

```
if(actualTitle.equalsIgnoreCase(expectedTitle))
```

```
System.out.println("Title Matched");
```

```
else
```

```
System.out.println("Title didn't match");
```

Alternatively,

```
Assert.assertEquals(actualTitle, expectedTitle);
```

16. How to mouse hover over a web element?

Actions class utility is used to hover over a web element in Selenium WebDriver

Instantiate Actions class.

```
Actions action = new Actions(driver);
```

In this scenario, we hover over search box of a website

```
actions.moveToElement(driver.findElement(By.id("id of the searchbox"))).perform();
```

17. How to retrieve CSS properties of an element?

getCssValue() method is used to retrieve [CSS properties](#) of any web element

General Syntax:

```
driver.findElement(By.id("id")).getCssValue("name of css attribute");
```

Example:

```
driver.findElement(By.id("email")).getCssValue("font-size");
```

Can Captcha be automated?

No, Selenium cannot automate Captcha.

20. How does Selenium handle Windows-based pop-ups?

Selenium was designed to handle web applications. Windows-based features are not natively supported by Selenium. However, third-party tools like AutoIT, Robot, etc can be integrated with Selenium to handle pop-ups and other Windows-based features.

21. How to take screenshots in WebDriver?

TakeScreenshot interface can be used to take screenshots in WebDriver.

getScreenshotAs() method can be used to save the screenshot

```
File scrFile = ((TakeScreenshot)driver).getScreenshotAs(outputType.FILE);
```

22. Is there a way to type in a textbox without using sendKeys()?

Yes! Text can be entered into a textbox using JavaScriptExecutor

```
JavascriptExecutor jse = (JavascriptExecutor) driver;
```

```
jse.executeScript("document.getElementById('email').value='abc.efg@xyz.com'");
```

23. How to select a value from a dropdown in Selenium WebDriver?

Select class in WebDriver is used for selecting and deselecting options in a dropdown.

The objects of Select type can be initialized by passing the dropdown webElement as a parameter to its constructor.

```
WebElement testDrop = driver.findElement(By.id("testingDropdown"));
```

```
Select dropdown = new Select(testDrop);
```

WebDriver offers three ways to select from a dropdown:

selectByIndex: Selection based on index starting from 0

```
dropdown.selectByIndex(5);
```

selectByValue: Selection based on value

```
dropdown.selectByValue("Books");
```

selectByVisibleText: Selection of option that displays text matching the given argument

```
dropdown.selectByVisibleText("The Alchemist");
```

24. What does the switchTo() command do?

switchTo() command is used to switch between windows, frames or pop-ups within the application. Every window instantiated by the WebDriver is given a unique alphanumeric value called "Window Handle".

Get the window handle of the window you wish to switch to

```
String handle= driver.getWindowHandle();
```

Switch to the desired window

```
driver.switchTo().window(handle);
```

Alternatively

```
for(String handle= driver.getWindowHandles())
```

```
{ driver.switchTo().window(handle); }
```

25. How to upload a file in Selenium WebDriver?

You can achieve this by using sendkeys() or Robot class method. Locate the text box and set the file path using sendkeys() and click on submit button

Locate the browse button

```
WebElement browse =driver.findElement(By.id("uploadfile"));
```

Pass the path of the file to be uploaded using sendKeys method

```
browse.sendKeys("D:\\SeleniumInterview\\UploadFile.txt");
```

26. How to set browser window size in Selenium?

The window size can be maximized, set or resized

To maximize the window

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

To set the window size

```
Dimension d = new Dimension(400,600);
```

```
driver.manage().window().setSize(d);
```

Alternatively,

The window size can be reset using JavaScriptExecutor

```
((JavascriptExecutor)driver).executeScript("window.resizeTo(1024, 768)");
```

27. When do we use findElement() and findElements()?

findElement() is used to access any single element on the web page. It returns the object of the first matching element of the specified locator.

General syntax:

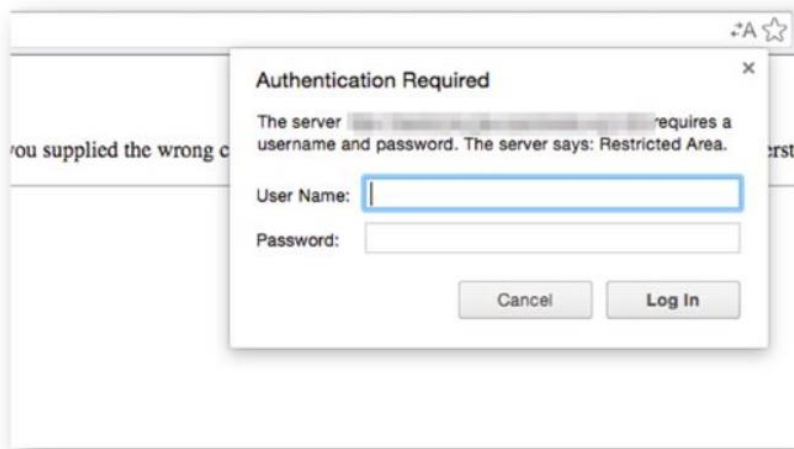
```
WebElement element = driver.findElement(By.id(example));
```

findElements() is used to find all the elements in the current web page matching the specified locator value. All the matching elements would be fetched and stored in the list of Web elements.

General syntax:

```
List <WebElement> elementList = driver.findElements(By.id(example));
```


29. How to login to any site if it is showing an Authentication Pop-Up for Username and Password?



To handle authentication pop-ups, verify its appearance and then handle them using an explicit wait command.

Use the explicit wait command

```
WebDriverWait wait = new WebDriverWait(driver, 10);
```

Alert class is used to verify the alert

```
Alert alert = wait.until(ExpectedConditions.alertIsPresent());
```

Once verified, provide the credentials

```
alert.authenticateUsing(new UserAndPassword(<username>, <password>));
```

30. What is the difference between single and double slash in Xpath?

Single slash is used to create Xpath with an absolute path i.e. the XPath would be created to start selection from the start node.

```
/html/body/div[2]/div[1]/div[1]/a
```

Double slash is used to create Xpath with relative path i.e. the XPath would be created to start selection from anywhere within the document

```
//div[class="qa-logo"]/a
```

31. How do you find broken links in Selenium WebDriver?

When we use `driver.get()` method to navigate to a URL, it will respond with a status of 200-OK

200 – OK denotes that the link is working and it has been obtained. If any other status is obtained, then it is an indication that the link is broken.

Some of the HTTP status codes are :

- 200 – valid Link
- 404 – Link Not Found
- 400 – Bad Request
- 401 – Unauthorized
- 500 – Internal error

As a starter, obtain the links from the web application, and then individually get their status.

Navigate to the interested webpage for e.g. www.amazon.com

Collect all the links from the webpage. All the links are associated with the Tag 'a'

```
List<WebElement> links = driver.findElements(By.tagName("a"));
```

Create a list of type `WebElement` to store all the Link elements in it.

```
for(int i=0; i<links.size(); i++) {  
  
    WebElement element = links.get(i);  
  
    String url=element.getAttribute("href");  
  
    verifyLink(url); }  

```

Now Create a Connection using URL object(i.e ., link)

```
URL link = new URL(urlLink);
```

Connect using Connect Method

```
HttpURLConnection httpConn =(HttpURLConnection)link.openConnection();
```

Use `getResponseCode ()` to get response code

```
if(httpConn.getResponseCode() != 200)
```

Through exception, if any error occurred

36) What are the different methods to refresh a web page in WebDriver?

There are multiple ways of refreshing a page in Webdriver.

1. Using `driver.navigate` command -

```
driver.navigate().refresh();
```

2. Using `driver.getCurrentUrl()` with `driver.get()` command -

```
driver.get(driver.getCurrentUrl());
```

3. Using `driver.getCurrentUrl()` with `driver.navigate()` command -

```
driver.navigate().to(driver.getCurrentUrl());
```

4. Pressing an F5 key on any textbox using the `sendKeys` command -

```
driver.findElement(By textboxLocator).sendKeys(Keys.F5);
```

5. Passing ascii value of the F5 key, i.e., "\uE035" using the sendKeys command -

```
driver.findElement(By textboxLocator).sendKeys("\uE035");
```

37) Write a code snippet to navigate back and forward in browser history?

Navigate back in browser history:

```
driver.navigate().back();
```

Navigate forward in browser history:

```
driver.navigate().forward();
```