

Q1) What is Selenium WebDriver

Selenium Web driver is set of class which used automate the web application.

Q2) Is Selenium Web Driver Automation Tool?

No Selenium web Driver is not an Automation tool, its a framework or set of class which is used to automate a web based application.

Q3) What kind of application best suited for Selenium and why it should be selected.

It is used to automate web based application and have browser and Platform compatibility. It supports multiple language such as java, C# and Python.

Q4) Can Selenium Web Services tested using Selenium?

No it is used to automate only web based application.

Q5) What is different locator supported in selenium.

Different Locators are:

- * ID
- * ClassName
- * Name
- * TagName
- * LinkText
- * Partial Link Text
- * XPATH
- * CSS

Q6) How Can we launch Different browser using python. For Chrome ?

```
driver=Webdriver.Chrome("PathofChromedriver")
```

For Firefox

```
driver=webdriver.Firefox("Path of Firefox Driver")
```

for IE

```
driver=webdriver.Ie("Path of IE Driver")
```

Q7) How Synchronization works in Selenium.

Synchronization is achieved using Implicit Wait and Explicit Wait.

Q8) What is Implicit Wait.

Implicit Wait is used to default waiting time.

Q9) What is explicit Wait.

Explicit Wait is used to halt the execution until condition is met or when time is elapsed.

Q10) How can enter the values in text box using python.

```
driver.find_element(By.ID,"Value").send_keys("Value")
```

Q11) How can we check if control enabled or not?

`driver.find_element(By.ID,"Value").is_enabled()`, this specific method will return if true or false.

Q12) How can we get text of a web element.

`driver.find_element(By.ID,"Value").text`, this specific method will return innertext of the control.

Q13) Is there any other way of reteriving text of web element.

```
driver.find_element(By.ID,"Value").get_attribute("InnerText")
```

Q14) What are the different ways of selecting the values in dropdown.

* `SelectByValue`: select the option based on option value.

* `selectByVisibleText`: selects the option based on visible text. * `selectByIndex`: selects the option base on Index

Q15) what are the different Navigation command in selenium
it is used to refresh

`driver.refresh()`

it is used to navigate back

`driver.back()`

it is used to move forward

`driver.forward()`

Q16) What is the difference b/w findelement and findelements
FindElement returns first matching element.

FindElements returns more than one elements.

Q17) what is the difference b/w Driver.Close and Diver.quit

close is used to close the entire browser

quit is used to close the working tab.

Q18) Can selenium handle the windows pop up?

No Selenium can't handle windows pop Up

Q19) Can selenium handle WebBased Pop Up.

Yes it can handle webbased pop up using “`driver.switch_to_alert()`”

Q20) How can capture screenshot in selenium.

`driver.get_screenshot_as_file(“filename”)` with this method we can take screen shot.

Q21) What are switch Class in selenium.

Switch class are used to switch between the different browser,frames and alert pop up.

Q22) What Action class in selenium ? Action class is user facing API to achieve complex user action events.

Q23) How can we perform drag drop in selenium

```
act=ActionChains(driver)
```

act.drag_and_drop(sourcelement,targetelement), with this method we can perform drag and drop

Q24) How can mouse hover on a control in selenium.

```
act=ActionChains(driver)
```

```
act.move_to_element(element)
```

with the above method we mouse hover on element, element argument is web element where we need to mouse hover.

Q25) Can we move the mouse control to specific coordinate in selenium

yes we can, selenium provides Action class in which we can make use of the act.move_by_offset() method to move the mouse control to specific coordinates.

Q26) Write a method to read data from webtable for specific row and column index using selenium.

```
def GetData(rowIndex,colIndex):
```

```
    tablerows=driver.find_element_by_id("IdValue").find_elements_by_tag_name("tr")
```

```
    tableDefinitons=tablerows[rowIndex+1].find_elements_by_tag_name("td")
```

```
    return tableDefinitons[colIndex].get_attribute("innerText")
```

Q27) What are Python unittest default methods

```
#Will execute before the execution of each test method
```

```
def setUp(self):
```

```
    pass
```

will execute once before it executes any test methods

```
def setUpClass(cls):
```

```
    pass
```

#will execute after the execution of each test method

```
def tearDown(self):
```

```
    pass
```

will execute after it executes all the test methods.

```
def tearDownClass(cls):
```

```
    pass
```

Q28) Can Image or Captcha be automated using selenium

No captcha or image can't be automated.

Q29) How will install selenium in python

We will use PIP command to install selenium

Pip install selenium Version_Nos

Q30) What is difference between list and tuple.

list is mutable while tuple is not mutable.

Q31) What is pass in python.

pass means no operation to be done

Q32) How can we convert string to int in python.

using int("23") to convert to int.

str(23) to convert to string

Q33) How will you handle exception in python

try,except and finally key word is used to handle the exception

try:

#code

except: #catches the exception

finally: #executes the block whether any exception is raised or not.

Q34) What are the different data types supported in python

integer,string,float and complex are the supported data types in python

Q35) What is MRO in python

Method Resolution order is used in inheritance concepts

where class is inheriting multiple class and in all the parent class, same method is defined. Child class is confused which method to call during run time. So the MRO helps to resolve the issue.

Q36) How will identify when web element doesn't have any of the unique locator.

we can make use of the get_attribute method to get web element from list of web elements.

for control in controls:

if control.get_attribute("attributename")== "attributeValue":

#found the matching control

Q37) What is difference b/w link text and partial link text.

LinkText will return element which exact match of given text.

partial link text will return which include the given text.

Q38) How to execute java script in selenium with python

driver.execute_script(script)

above method will execute the java script.

Q39) What are the different way entering value TextBox

* driver.find_element_by_id("val").send_keys("value to be entered")

* other is using Action Class

```
act=ActionChains(driver)
```

```
act.send_keys_to_element(control,"Valuetobeentered")
```

Q40) What is the use of Xpath

xpath is used to find the web element in webpage.

Q41) What is the difference between single and double slash in xpath

single slash is start selection from the document node.

double slash is start selection from anywhere in the document.

Q42) What are the different exception in selenium

* webdriver exception

* noalertPresent Exception

* nosuchwindow Exception

*nosuchelement exception

*timeoutexception

Q43) How will perform double click on web element

```
act=ActionChains(driver)
```

```
act.double_click(control)
```

Above method will be used to double click on the control.

Q44) What is difference between getWindowHandle() and getWindowHandles()

getWindowHandle()- it is used to get address of the the current browser where the control is and return type is string.

getWindowHandles()- it is used to get the address of the all open browser and its return type is set<string>

Q45) How will you handle multiple windows in selenium

we can use “switch_to_window” method to switch between the multiple windows. where in argument we would be sending the address of the window in the form of string.

Q46) How will you verify if you have switched between the correct window.

we can verify the window title to see if the we have switched to correct window or not.

```
driver.switch_to_window(“windowname”)
```

```
if driver.title==”ttitleName”:
```

```
#switched to correct window
```

Q47) What is framework?

Framework defines a set of rules or best practices which we can follow in systematic way to achieve the desired results.

Q48) What are the different automation Frameworks

- * Data Driven Framework
- * Key Driven Framework
- * Hybrid Framework

Q49) How to login into any site if it showing authentication pop up for username and password

To achieve this type of issue we can pass username and pwd with URL

```
http://username:password@url
```

Q50) How to fetch the current URL in selenium

driver.current_url returns the current url of the browser

Q51) How do you convert a number into a string?

>use str() inbuilt function in python

Q52) What is slicing?

> Slicing is used to fetch a specific range of items from sequence types like list, tuple, Strings etc

Q53) what is the o/p of teh following code?

```
i = 6
```

```
while (i==6):
```

```
print("Value of i is"+i)
```

> 'value of 1 is 6' and program will not stop

Q54) what is the o/p of len("")

> 0

Q57) Which of the following functions does return a list of elements

1. find_element_by_xpath
2. find_element_by_id
3. find_elements_by_name
4. find_element_by_partial_link_text
5. find_element_by_link_text

>3

Q58)which of the following is not a valid webdriver class

1. webdriver.ActionChains
2. webdriver.Firefox
3. webdriver.HtmlUnit
4. webdriver.Chrome

> 1

1. What is Automation Testing?

Automation testing is the process of testing the software using an automation tool to find the defects. In this process, executing the test scripts and generating the results are performed automatically by automation tools. Some most popular tools to do automation testing are HP QTP/UFT, [Selenium WebDriver](#), etc.,

2. What are the benefits of Automation Testing?

- Saves time and money. Automation testing is faster in execution.
- Reusability of code. Create one time and execute multiple times with less or no maintenance.
- Easy reporting. It generates automatic reports after test execution.
- Easy for compatibility testing. It enables parallel execution in the combination of different OS and browser environments.
- Low-cost maintenance. It is cheaper compared to manual testing in a long run.
- Automated testing is more reliable.
- Automated testing is more powerful and versatile.
- It is mostly used for regression testing. Supports execution of repeated test cases.
- Minimal manual intervention. Test scripts can be run unattended.
- Maximum coverage. It helps to increase the test coverage.

3. What type of tests have you automated?

Our main focus is to automate test cases to do Regression testing, Smoke testing, and Sanity testing. Sometimes based on the project and the test time estimation, we do focus on End to End testing.

4. How many test cases you have automated per day?

It depends on Test case scenario complexity and length. I did automate 2-5 test scenarios per day when the complexity is limited. Sometimes just 1 or fewer test scenarios in a day when the complexity is high.

5. What is a Framework?

A framework defines a set of rules or best practices which we can follow in a systematic way to achieve the desired results. There are different types of automation frameworks and the most common ones are:

- Data Driven Testing Framework
- Keyword Driven Testing Framework
- Hybrid Testing Framework

6. Have you created any Framework?

If you are a beginner: No, I didn't get a chance to create a framework. I have used the framework which is already available.

If you are an experienced tester: Yes, I have created a framework. Or I have involved in the creation of the framework.

7. Can you explain the Framework which you have used in your Selenium Project?

Here we have clearly explained each component of Framework.

8. Why do you prefer Selenium Automation Tool?

- Free and open source
- Have large user base and helping communities
- Cross browser compatibility

- Platform compatibility
- Multiple programming languages support

9. What is Selenium?

Selenium is an open source (free) automated testing suite to test web applications. It supports different platforms and browsers. It has gained a lot of popularity in terms of web-based automated testing and giving a great competition to the famous commercial tool HP QTP (Quick Test Professional) AKA HP UFT (Unified Functional Testing).

Selenium is a set of different software tools. Each tool has a different approach in supporting web based automation testing.

It has four components namely,

- i Selenium IDE (Integrated Development Environment)
- ii Selenium RC (Remote Control) – selenium 1
- iii Selenium WebDriver – selenium 2 & 3
- iv Selenium Grid

10. What is Selenium IDE?

Selenium IDE (Integrated Development Environment) is a Firefox plugin. It is the simplest framework in the Selenium Suite. It allows us to record and playback the scripts. Even though we can create scripts using Selenium IDE, we need to use Selenium RC or Selenium WebDriver to write more advanced and robust test cases.

11. What is Selenese?

Selenese is the language which is used to write test scripts in Selenium IDE.

12. Which is the only browser that supports Selenium IDE to be used?

Firefox

13. What is Selenium RC?

Selenium RC AKA Selenium 1. Selenium RC was the main Selenium project for a long time before the WebDriver merge brought up Selenium 2. Selenium 1 is still actively supported (in maintenance mode). It relies on JavaScript for automation. It supports Java, Javascript, Ruby, PHP, Python, Perl and C#. It supports almost every browser out there.

14. What is Selenium WebDriver?

Selenium WebDriver AKA Selenium 2 is a browser automation framework that accepts commands and sends them to a browser. It is implemented through a browser-specific driver. It controls the browser by directly communicating with it. Selenium WebDriver supports Java, C#, PHP, Python, Perl, Ruby.

15. What is Selenium Grid?

Selenium Grid is a tool used together with Selenium RC to run tests on different machines against different browsers in parallel. That is, running multiple tests at the same time against different machines running different browsers and operating systems.

In simple words, it is used to distribute your test execution on multiple platforms and environments concurrently.

16. When do you use Selenium Grid?

Selenium Grid can be used to execute same or different test scripts on multiple platforms and browsers concurrently so as to achieve distributed test execution

17. What are the advantages of Selenium Grid?

It allows running test cases in parallel thereby saving test execution

time.

It allows multi-browser testing

It allows us to execute test cases on multi-platform

18. What is a hub in Selenium Grid?

A hub is a server or a central point that controls the test executions on different machines.

19. What is a node in Selenium Grid?

Node is the machine which is attached to the hub. There can be multiple nodes in Selenium Grid.

20. What are the types of WebDriver APIs available in Selenium?

- Firefox Driver
- Gecko Driver
- InternetExplorer Driver
- Chrome Driver
- HTMLUNIT Driver
- Opera Driver
- Safari Driver
- Android Driver
- iPhone Driver
- EventFiringWebDriver

21. Which WebDriver implementation claims to be the fastest?

The fastest implementation of WebDriver is the HTMLUnitDriver. It is because the HTMLUnitDriver does not execute tests in the browser.

22. What are the Programming Languages supported by Selenium WebDriver?

- Java
- C#
- Python
- Ruby
- Perl
- PHP

23. What are the Operating Systems supported by Selenium WebDriver?

- Windows
- Linux
- Apple

24. What are the Open-source Frameworks supported by Selenium WebDriver?

- JUnit
- TestNG
- CUCUMBER
- JBHEAVE

25. What are the Locators available in Selenium? **Different types of locators are:**

- 1 ID –
- 2 ClassName –
- 3 Name –
- 4 TagName –
- 5 LinkText –
- 6 PartialLinkText –
- 7 XPath –
- 8 CSS Selector –

26. What is a XPath?

XPath is used to locate the elements. Using XPath, we could navigate through elements and attributes in an XML document to locate web

elements such as textbox, button, checkbox, Image etc., in a web page.

27. What is the difference between “/” and “//”

Single Slash “/” – Single slash is used to create XPath with absolute path i.e. the XPath would be created to start selection from the document node/start node.

Double Slash “//” – 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.

28. What is the difference between Absolute Path and Relative Path?

Absolute XPath starts from the root node and ends with desired descendant element's node. It starts with top HTML node and ends with input node. It starts with a single forward slash(/) as shown below.

```
/html/body/div[3]/div[1]/form/table/tbody/tr[1]/td/input
```

Relative XPath starts from any node in between the HTML page to the current element's node(last node of the element). It starts with a single forward slash(//) as shown below.

```
//input[@id='email']
```

29. What is the difference between Assert and Verify in Selenium?

Assert: In simple words, if the assert condition is true then the program control will execute the next test step but if the condition is false, the execution will stop and further test step will not be executed.

Verify: In simple words, there won't be any halt in the test execution

even though the verify condition is true or false.
For detailed post check the below link.

30. What are Soft Assert and Hard Assert in Selenium?

Soft Assert: Soft Assert collects errors during *@Test* Soft Assert does not throw an exception when an assert fails and would continue with the next step after the assert statement

Hard Assert: Hard Assert throws an `AssertException` immediately when an assert statement fails and test suite continues with next *@Test*

31. What are the verification points available in Selenium?

In Selenium IDE, we use Selenese Verify and Assert Commands as Verification points

In Selenium WebDriver, there is no built-in features for verification points. It totally depends on our coding style. some of the Verification points are

To check for page title

To check for certain text

To check for certain element (text box, button, drop down, etc.)

32. How to launch a browser using Selenium WebDriver?

WebDriver is an Interface. We create Object of a WebDriver Interface.

<2.53 – no geckodriver

3.x – geckodriver for FF

To launch Firefox Driver:

```
WebDriver driver = new FirefoxDriver();
```

To launch Chrome Driver:

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

To launch Internet Explorer Driver:

```
WebDriver driver = new InternetExplorerDriver();
```

33. Is the FirefoxDriver a Class or an Interface?

FirefoxDriver is a Java class, and it implements the WebDriver interface.

34. What is the super interface of WebDriver?

SearchContext.

35. Explain the line of code Webdriver driver = new

FirefoxDriver(); ?

‘WebDriver’ is an interface and we are creating an object reference of type WebDriver instantiating an object of FirefoxDriver class.

36. We do create a reference variable ‘driver’ of type WebDriver

```
WebDriver driver = new FirefoxDriver();
```

instead of creating

```
FirefoxDriver driver = new FirefoxDriver();
```

```
1 WebDriver driver = new FirefoxDriver();
```

```
2
```

```
3 instead of creating
```

```
4
```

```
5 FirefoxDriver driver = new FirefoxDriver();
```

What is the purpose of doing this way?

If we create a reference variable driver of type WebDriver then we could use the same driver variable to work with any browser of our choice such as IEDriver, SafariDriver etc.,

```
//FirefoxDriver driver = new FirefoxDriver();
```

```
ChromeDriver driver = new ChromeDriver();
```

```
driver.get("http://www.google.com");
```

```
WebDriver driver = new FirefoxDriver();
```

37. What are the different exceptions you have faced in Selenium

WebDriver?

- WebDriverException
- TimeoutException
- NoAlertPresentException
- NoSuchWindowException
- NoSuchElementException
- StaleElementReferenceException
- IllegalStateException

38. How To Login Into Any Site If It Is Showing Any Authentication Pop-Up For Username And Password?

To do this we pass username and password with the URL

`http://username:password@url`

e.g. <http://admin:admin123@xyz.com>

39. What are the types of waits available in Selenium WebDriver?

In Selenium we could see three types of waits such as Implicit Waits, Explicit Waits and Fluent Waits.

- Implicit Waits –
- Explicit Waits –
- Fluent Waits –
- PageLoadTimeOut
- Thread.sleep() – static wait

40. What is Implicit Wait In Selenium WebDriver?

Implicit waits tell to the WebDriver to wait for a certain amount of time before it throws an exception. Once we set the time, WebDriver will wait for the element based on the time we set before

it throws an exception. The default setting is 0 (zero). We need to set some wait time to make WebDriver to wait for the required time.

41. What is WebDriver Wait In Selenium WebDriver?

WebDriverWait is applied on a certain element with defined expected condition and time. This wait is only applied to the specified element. This wait can also throw an exception when an element is not found.

42. What is Fluent Wait In Selenium WebDriver?

FluentWait can define the maximum amount of time to wait for a specific condition and frequency with which to check the condition before throwing an “ElementNotVisibleException” exception.

43. How to input text in the text box using Selenium WebDriver?

By using sendKeys() method

```
WebDriver driver = new FirefoxDriver();  
driver.get("https://www.gmail.com");  
driver.findElement(By.xpath("xpath")).sendKeys("test");
```

44. How to input text in the text box without calling the sendKeys()?

// To initialize js object

```
JavascriptExecutor JS = (JavascriptExecutor)driver;
```

// To enter username

```
JS.executeScript("document.getElementById('User').value=test.com"
);
```

45. How to clear the text in the text box using Selenium WebDriver?

By using clear() method

```
WebDriver driver = new FirefoxDriver();
driver.get("https://www.gmail.com");
driver.findElement(By.xpath("xpath_of_element1")).sendKeys("Soft
ware Testing Material Website");
driver.findElement(By.xpath("xpath_of_element1")).clear();
```

46. How to get a text of a web element?

By using getText() method

47. How to get an attribute value using Selenium WebDriver?

By using getAttribute(value);

48. How to click on a hyperlink using Selenium WebDriver?

We use click() method in Selenium to click on the hyperlink

```
driver.findElement(By.linkText("Software Testing Material
Website")).click();
```

```
1 driver.findElement(By.linkText("Software Testing Material Website")).click();
```

49. How to submit a form using Selenium WebDriver?

We use "submit" method on element to submit a form

```
driver.findElement(By.id("form_1")).submit();
```

1

Alternatively, you can use click method on the element which does form submission

50. How to press ENTER key on text box In Selenium WebDriver?

To press ENTER key using Selenium WebDriver, We need to use Selenium Enum Keys with its constant ENTER.

```
driver.findElement(By.xpath("xpath")).sendKeys(Keys.ENTER);
```

51. How to pause a test execution for 5 seconds at a specific point?

By using `java.lang.Thread.sleep(long milliseconds)` method we could pause the execution for a specific time. To pause 5 seconds, we need to pass parameter as 5000 (5 seconds)

```
Thread.sleep(5000)
```

52. Is Selenium Server needed to run Selenium WebDriver Scripts?

When we are distributing our Selenium WebDriver scripts to execute using Selenium Grid, we need to use Selenium Server.

53. What happens if I run this command.

```
driver.get("www.softwaretestingmaterial.com") ;
```

An exception is thrown. We need to pass HTTP protocol within `driver.get()` method.

```
driver.get("http://www.softwaretestingmaterial.com");
```

1

```
driver.get("http://www.softwaretestingmaterial.com");
```

54. What is the alternative to driver.get() method to open an URL using Selenium WebDriver?

Alternative method to driver.get("url") method is driver.navigate.to("url")

55. What is the difference between driver.get() and driver.navigate.to("url")?

driver.get(): To open an URL and it will wait till the whole page gets loaded

driver.navigate.get(): To navigate to an URL and It will not wait till the whole page gets loaded

56. Can I navigate back and forth in a browser in Selenium WebDriver?

We use Navigate interface to do navigate back and forth in a browser. It has methods to move back, forward as well as to refresh a page.

driver.navigate().forward(); – to navigate to the next web page with reference to the browser's history

driver.navigate().back(); – takes back to the previous webpage with reference to the browser's history

driver.navigate().refresh(); – to refresh the current web page thereby reloading all the web elements

driver.navigate().to("url"); – to launch a new web browser window and navigate to the specified URL

57. What are the different types of navigation commands?

Refer above question (Can I navigate back and forth in a browser)

58. How to fetch the current page URL in Selenium?

To fetch the current page URL, we use `getCurrentUrl()`
`driver.getCurrentUrl();`

59. How can we maximize browser window in Selenium?

To maximize browser window in selenium we use `maximize()` method. This method maximizes the current window if it is not already maximized

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

60. How to delete cookies in Selenium?

To delete cookies we use `deleteAllCookies()` method
`driver.manage().deleteAllCookies();`

61. What are the ways to refresh a browser using Selenium

WebDriver?

There are multiple ways to refresh a page in selenium

- Using `driver.navigate().refresh()` command as mentioned in the question 45
- Using `driver.get("URL")` on the current URL or using `driver.getCurrentUrl()`
- Using `driver.navigate().to("URL")` on the current URL or `driver.navigate().to(driver.getCurrentUrl());`
- Using `sendKeys(Keys.F5)` on any textbox on the webpage

62. What is the difference between `driver.getWindowHandle()` and `driver.getWindowHandles()` in Selenium WebDriver?

`driver.getWindowHandle()` – It returns a handle of the current page (a unique identifier)

`driver.getWindowHandles()` – It returns a set of handles of the all the pages available.

63. What is the difference between `driver.close()` and `driver.quit()` methods?

Purpose of these two methods (`driver.close` and `driver.quit`) is almost same. Both allow us to close a browser but still, there is a difference.

`driver.close()`: To close current WebDriver instance

`driver.quit()`: To close all the opened WebDriver instances

64. What is the difference between `driver.findElement()` and `driver.findElements()` commands?

The difference between `driver.findElement()` and `driver.findElements()` commands is-

- `findElement()` returns a single `WebElement` (found first) based on

the locator passed as parameter. Whereas findElements() returns a list of WebElements, all satisfying the locator value passed.

- Syntax of findElement()-
- WebElement textbox =
driver.findElement(By.id("textBoxLocator"));
- Syntax of findElements()-
- List <WebElement> elements =
element.findElements(By.id("value"));
- Another difference between the two is- if no element is found then findElement() throws NoSuchElementException whereas findElements() returns a list of 0 elements.

```
List<WebElement> list = driver.findElements(By.tagName("a"));
Sop(list.size()); ==40
```

65. How to find whether an element is displayed on the web page?

WebDriver facilitates the user with the following methods to check the visibility of the web elements. These web elements can be buttons, drop boxes, checkboxes, radio buttons, labels etc.

1 isDisplayed()

2 boolean elePresent =

```
driver.findElement(By.xpath("xpath")).isDisplayed();
```

```
3 4 boolean elePresent = driver.findElement(By.xpath("xpath")).isDisplayed();
```

1 isSelected()

2 boolean eleSelected=

```
driver.findElement(By.xpath("xpath")).isSelected();
```

```
3 4 boolean eleSelected= driver.findElement(By.xpath("xpath")).isSelected();
```

1 isEnabled()

2 boolean eleEnabled=

```
driver.findElement(By.xpath("xpath")).isEnabled();
```

```
3 4 boolean eleEnabled= driver.findElement(By.xpath("xpath")).isEnabled();
```

66. How to select a value in a dropdown?

By using Select class

```
WebElement mySelectElement =  
driver.findElement(By.name("dropdown"));
```

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

```
dropdown.selectByVisibleText(Text);  
dropdown.selectByIndex(Index);  
dropdown.selectByValue(Value);
```

.

67. How to capture Screenshot in Selenium WebDriver?

By using TakesScreenshot Interface

In Selenium 3, we may face few issues while capturing Screenshots. To overcome we use aShot utility. Click on below links to see posts related to the normal way of capturing a screenshot and capturing a screenshot using aShot utility.

68. How to mouse hover on a web element using WebDriver?

By using Actions class

```
WebElement ele = driver.findElement(By.xpath("xpath"));  
//Create object 'action' of an Actions class  
Actions action = new Actions(driver);  
//Mouseover on an element
```

```
action.moveToElement(ele).build().perform();
```

—

69. How can we handle web based pop-up?

To handle alerts popups we need to do switch to the alert window and call Selenium WebDriver Alert API methods.

—

70. How can we handle windows based pop up?

Selenium doesn't support windows based applications. It is an automation testing tool which supports only web application testing. We could handle windows based popups in Selenium using some third party tools such as AutoIT, SIKULI, Robot class etc.

71. How to handle hidden elements in Selenium WebDriver?

It is one of the most important selenium interview questions. We can handle hidden elements by using javaScript executor
(JavascriptExecutor(driver)).executeScript("document.getElementsByTagName('div').length");

72. How can you find Broken Links in a page using Selenium WebDriver?

—

73. How to find more than one web element in the list?

```
// To store the list
List <WebElement> eleList =
driver.findElements(By.xpath("xpath"));
// To fetch the size of the list
int listSize = eleList.size();
//for loop
for (int i=0; i<listSize; i++)
{
    // Clicking on each link
    links.get(i).click();
    // Navigating back to the previous page that stores the links
    driver.navigate().back();
}
```

74. How to read a JavaScript variable in Selenium WebDriver?

By using JavascriptExecutor

// To initialize the JS object.

```
JavascriptExecutor JS = (JavascriptExecutor) webdriver;
```

// To get the site title.

```
String title = (String)JS.executeScript("return document.title");
System.out.println("Title of the webpage : " + title);
```

75. How do you read test data from excels?

Test data can efficiently be read from excel using JXL or POI API. POI API has many advantages than JXL.

76. Is it possible to automate the captcha using Selenium?

No, It's not possible to automate captcha and bar code reader.

77. List some scenarios which we cannot automate using Selenium WebDriver?

1. Bitmap comparison Is not possible using Selenium WebDriver
2. Automating Captcha is not possible using Selenium WebDriver
3. We can not read bar code using Selenium WebDriver
4. windows OS based pop ups
5. third party calendars/element
6. Image
7. Word/PDF

78. What is Object Repository in Selenium WebDriver?

Object Repository is used to store element locator values in a centralized location instead of hard coding them within the scripts. We do create a property file (.properties) to store all the element locators and these property files act as an object repository in Selenium WebDriver.

80. How to Upload a file in Selenium WebDriver?

There are two cases which are majorly used to upload a file in Selenium WebDriver such as using SendKeys Method and using AutoIT Script.

–

```
Browser Button – type =“file”  
SendKeys (c:\\test\\naveen.jpg);
```

81. How to Download a file in Selenium WebDriver?

By using AutoIT script, we could download a file in Selenium WebDriver.

83. How to switch between frames in Selenium?

By using the following code, we could switch between frames.
`driver.switchTo().frame();`

84. How to connect a Database in selenium?

As we all know Selenium WebDriver is a tool to automate User Interface. We could only interact with Browser using Selenium WebDriver.

We use JDBC Driver to connect the Database in Selenium (While using Java Programming Language).

85. How To Resize Browser Window Using Selenium WebDriver?

To resize the browser window to particular dimensions, we use ‘Dimension’ class to resize the browser window.

```
//Create object of Dimensions class  
Dimension d = new Dimension(480,620);
```



```
//Resize the current window to the given dimension  
driver.manage().window().setSize(d);
```

86. How To Scroll Web Page Down Or UP Using Selenium WebDriver?

JavaScript scrollBy() method scrolls the document by the specified number of pixels.

87. How To Perform Right Click Action (Context Click) In Selenium WebDriver?

We use Actions class in Selenium WebDriver to do Right-Click (Context Click) action.

```
action.contextClick(driver.findElement(By.xpsjht())).build().perform()  
;
```

88. How To Perform Double Click Action In Selenium WebDriver?

We use Actions class to do Double click action in selenium.

89. How To Perform Drag And Drop Action in Selenium

WebDriver?

We use Actions class to do Drag And Drop Action

90. How To Highlight Element Using Selenium WebDriver?

By using JavascriptExecutor interface, we could highlight the specified element