Selenium FAQs

1. **What is Stale Element Exception? How to handle it?**
   - A WebElement is a reference to an element in the DOM.
   - A StaleElementException is thrown when the element you were interacting is destroyed and then recreated. Most complex web pages these days will move things about on the fly as the user interacts with it and this requires elements in the DOM to be destroyed and recreated.
   - When this happens the reference to the element in the DOM that you previously had becomes stale and you are no longer able to use this reference to interact with the element in the DOM. When this happens you will need to refresh your reference, or in real world terms find the element again
   - We can use Actions class to click on specific element where this exception is coming up

2. **What are different Locators that are used by WebDriver?**
   - ID
   - Name
   - Xpath
   - Tag Name
   - CSS Selector
   - Partial Link Text
   - LinkText
   - Class Name

3. **What are different Xpath functions that you have used in your Project?**
   - Relative Path
   - Contains
   - By Attribute name and value
   - Parent to Child and Vice Versa relationship
   - Following-sibling
   - Preceeding-sibling
   - And

4. **What will happen in background when execute new FirefoxDriver() ?**
   - Firefox binary will be triggered and FF browser will open with default options.
   - FirefoxDriver object is created

5. **What is the below statement means and Why?**

--- WebDriver driver = new FirefoxDriver();

- WebDriver is an interface which contain several abstract methods such as get(...), findElamentBy(...) etc.
- We simply create reference of web Driver and we can assign objects (Firefox driver, CromeDriver, IEDriver, Andriod driver etc) to it.

6. **How do you handle inner Frames and Adjacent Frames?**

- SwitchTo frame1, SwitchTo frame2 (inner frame) work on the element and switchto default content
- Use SwitchTo frame to move the control inside frame.

7. **How to click on an element which is not visible using selenium WebDriver?**

- We can use JavascriptExecutor to click

8. **Difference between verify and assert?**

- If you want your test execution not to abort when a validation fails then we use Verify otherwise we use Assert.

9. **What is the use of @FindByannotation?**

- @FindBy is used to identify element in the Page Factory approach

10. **What are the common exceptions you see in Selenium? How are those handled?**

- **NoSuchElementException** :FindBy method can't find the element.
- **StaleElementReferenceException** : This tells that element is no longer appearing on the DOM page.
- **TimeoutException**: This tells that the execution is failed because the command did not complete in enough time.
- **ElementNotVisibleException**: Thrown to indicate that although an element is present on the DOM, it is not visible, and so is not able to be interacted with
- **ElementNotSelectableException**: Thrown to indicate that may be the element is disabled, and so is not able to select.

11. **Do you use Thread.sleep?**

- Rarely

12. **What are different wait mechanisms that can be used in scripting?**

- driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

- WebDriverWait
- Thread.sleep(8000);
- FluentWait

13. What are different pop-ups that you have handle in your projects?

- JavaScript Pop Ups --- Alert alert = driver.switchTo().alert();
- Browser Pop Ups --- Browser Profiles, Robot tool Kit, AutoIT
- Native OS Pop Ups--- Browser Profiles, Robot tool Kit, AutoIT

14. What are multiple ways of refreshing the current browser?

- Driver.navigate().refresh();
- SendKeys – Keys.F5
- Driver.get(driver.getCurrentURL)
- Driver.navigate().to(driver.getCurrentURL)

15. How do you handle HTTP Proxy Authentication pop ups in browser?

- Form authentications URL - http://UserName:Password@Example.com

16. How do you handle Ajax dropdowns?

- With help of Selenium Sync commands like ImplicitWait, WebDriverWait or FluentWait

17. What is the difference between findelement() and findelements()?

findelement()

- findELement will find the first matching element.
- findELement will throw NoSuchElementException if no matching element found
- public WebElementfindElement(By by)

findelements()

- findELements will all the matching elements.
- findELements – will return an empty list if no matching elements found and no exception will be thrown
- public java.util.List<WebElement>findElements(By by)

18. What are the limitations of web driver?

- It does not support and non-web-based applications; it only supports web based applications.
- It cannot readily support new browsers or new firefox versions
- It is open source tool so in case of any technical issues you need to rely on the selenium community forums to get your issue resolved.
- No inbuilt reporting capability so you need plugins like JUnit and TestNG for test reports.

19. What is the default port for Selenium Grid?

- 4444

20. What is Exit/Done Criteria?

- Test can be executed multiple times and does not fail (unless defect has been found). In a nutshell, the automated test case is not fragile.
- Test is verifying/validating expected results
- Code reviewed and all comments are addressed
- Updates test case execution status in QC/Rally successfully
- The script is not dependent on login credentials/etc as it should be smart enough to randomly select users from the DB. The point here is to try and emulate real-life scenarios.
- NO STATIC INPUT. We want to make each test unique each time it executes as this emulates a real life scenario (e.g. different users interacting with the system). There are some instances where static data/input is required/needed. Use good judment in this area. Normally static input should be used for setup operations (e.g. Test/Dev – environment specifics in your testing.xml file)
- Code committed in SVN

21. Difference between Absolute path & Relative path?

- Absolute path will start with root path (/) and Relative path will from current path (//)
- Absolute xPath: /html/body/div[3]/div[2]/div[2]/div[2]/div[2]/div[2]/div[2]/div/div[4]/div[1]/div/div[@id='main'] /div[@id='Blog1']/div[1]/div[1]/div/div[1]/div/h3/a
- Relative xPath : //h3/a[text()='Working on New Window']

22. What are desiredcapabilities?

- Desired Capabilities help to set properties for the Web Driver. A typical use case would be to set the path for the Firefox Driver if your local installation doesn't correspond to the default settings.
- https://code.google.com/p/selenium/wiki/DesiredCapabilities

23. Difference between Web driver listener and TestNG Listener?

- TestNG and Web driver Listener have different interfaces to implement and call them. They bothmodify respective behavior. You can use Listeners in Annotation. Below 2 URL gives the detailed list of listener and their interfaces.
- http://testng.org/doc/documentation-main.html#testng-listeners

- http://selenium.googlecode.com/git/docs/api/java/org/openqa/selenium/support/events/AbstractWebDriverEventListener.html

24. Describe your framework.

- Explain the framework

24. Which is the best way to locate an element?

- Finding elements by ID is usually going to be the fastest option, because at its root, it eventually calls down to document.getElementById(), which is optimized by many browsers.
- Finding elements by XPath is useful for finding elements using very complex selectors, and is the most flexible selection strategy, but it has the potential to be very slow, particularly in IE. In IE 6, 7, or 8, finding by XPath can be an order of magnitude slower than doing the same in Firefox. IE provides no native XPath-over-HTML solution, so the project must use a JavaScript XPath implementation, and the JavaScript engine in legacy versions of IE really is that much slower.

- If you have a need to find an element using a complex selector, I usually recommend using CSS Selectors, if possible. It's not quite as flexible as XPath, but will cover many of the same cases, without exhibiting the extreme performance penalty on IE that XPath can.

25. What is the difference between "GET" and "NAVIGATE" to open a web page in selenium web driver?

- navigate().to() and get() do exactly the same thing. One's just a lot easier to type than the other!
- The navigate interface also has the ability to move backwards and forwards in your browser's history:

26. Please tell me the difference b/w implicitly Wait and Explicit wait.

- **Implicit Wait** sets internally a timeout that will be used for all consecutive Web Element searches. It will try lookup the element again and again for the specified amount of time before throwing a NoSuchElementException if the element could not have been found. It does only this and can't be forced into anything else - it waits for elements to show up.

- **Explicit Wait** or just Wait is a one-timer used by you for a particular search. It is more extendible in the means that you can set it up to wait for any condition you might like. Usually, you can use some of the prebuilt Expected Conditions to wait for elements to become clickable, visible, invisible, etc., or just write your own condition that suits your needs.

27. How we can retrieve the dynamically changing Ids?

- When we login Facebook the login label's id changes dynamically thus resulting in failure.
- We have a hierarchy of locators and Facebook Is dynamic in nature, so we are not able to use "id" for identification for that we have remaining 7 locator's for that : 2. xpath ().. 3. name..4. css.. 5. link text.. 6. partiallinktext...7.tag name. So u can use any one for identifying it. Most probably u can use "xpath" or "css-locator" and if there r tag then link text or partial-link text. it depend on u . But we never use id's in Ajax application because it’s not possible.

28. What is the difference between driver.Close() and driver.Quit () method?

- driver.close() - It is used to close the browser or page currently which is having the focus.
- driver.quit() - It is used to shut down the web driver instance or destroy the web driver instance (Close all the windows)

29. How to scroll web element?(not browser)

```
FirefoxProfile profile=new FirefoxProfile();
profile.setEnableNativeEvents(true);
WebDriver driver=new FirefoxDriver(profile);
driver.navigate("http://jqueryui.com/draggable/");
Thread.sleep(6000L);
WebElement element=driver.findElement(By.xpath("//div[@id='draggable']"));
Actions actn=new Actions(driver);
actn.dragAndDropBy(element, 50, 50).build().perform();
}
```

30. What is the basic use of Firefox profiles and how can we use them using selenium?

- A profile in Firefox is a collection of bookmarks, browser settings, extensions, passwords, and history; in short, all of your personal settings.
- We use them to change user agent, changing default download directory, changing versions etc.
- http://code.google.com/p/selenium/wiki/FirefoxDriver

31. Customize the name of file going to be downloaded?

- You have to download AUTO IT.exe file and has to be install and later you have create .au3 file (in this file you have to specify the commands in VB script like your file name, where have to save, it will be easy may be 3 or 4 steps) using AUTOIT...then right click the .au3 file you have to compile ....after that you will get the .exe file with the name of .au3 file ..In eclipse you will give the code like this
- <----processbuilddder .exe="" .start="" au3="" file="" of="" path="" processbuilder="" ps="new" the="">

32. Difference between flex and flash application.

- In flash there is no code just based on creativity(design) we will complete the work(time consuming process) whereas flex contain some small functions which is integrated with mxml, PHP..(no tool is there to develop flex we want to use the properties of css and style sheet)

33. What is Error Collector in TestNG? What is its use?

- This class allows the collection of errors during the process of retrieving thetest data for the test method parameters
- http://testngdatabind.sourceforge.net/apidocs/net/sf/testng/databinding/core/error/ErrorCollector.html

34. How to run tests in multiple browser parallel?

- selenium grid

35. How to prepare Customized html Report using TestNG in hybrid framework.

- Junit: with the help of ANT.
- TestNG: using inbuilt default.html to get the HTML report. Also XST reports from ANT,Selenium, TestNG combination.
- Using our own customized reports using XSL jar for converting XML content to HTML.

36. How to refresh a page without using context click?

- Using sendKeys.Keys method
- Using navigate.refresh() method
- Using navigate.refresh() method
- Using get() method
- Using sendKeys() method

1.UsingsendKeys.Keys method

```
driver.get("https://accounts.google.com/SignUp");
driver.findElement(By.id("firstname-placeholder")).sendKeys(Keys.F5);
```

2.Usingnavigate.refresh() method

```
driver.get("http://ruchi-myseleniumblog.blogspot.in/2013/12/100-selenium-interview-questions.html");
driver.navigate().refresh();
```

3.Using navigate.to() method

```
driver.get("http://ruchi-myseleniumblog.blogspot.in/2014/01/selenium-hybrid-framework-using.html");
```

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

4.Using get() method
driver.get("http://ruchi-myseleniumblog.blogspot.in/2013/12/basic-core-java-interview-questions.html");
driver.get(driver.getCurrentUrl());

5.UsingsendKeys() method
driver.get("https://accounts.google.com/SignUp");
driver.findElement(By.id("firstname-placeholder")).sendKeys("\uE035");

37. How to find broken images in a page using Selenium Web driver.

- Get xpath and then using tag name; get all the links in the page
- Use HttpURLConnector class and sent method GET
- Get the response code for each link and verify if it is 404/500

38. How to disable cookies in browser.

- Using deleteAllVisibleCookies() in selenium

39. How to change user agent in Firefox by selenium web driver.

FirefoxProfile profile = new FirefoxProfile();
profile.setPreference("general.useragent.override", "some UA string");
Web Driver driver = new FirefoxDriver(profile);

40. What is the MOST challenging test problem in my career in Automation?

In my career
Changing XPATHS' between testing server and production server-by keeping generic xpath
Keep separate property files for production and UAT
Automating flash apps

41. How to handle network latency using selenium?

- Using driver.manage.pageLoadingtime for network latency

42. How does u handle dynamic elements without using xpath (with example?)

- By using classname or css.

43. How to work with dynamic web table?

- You can get the total number of tags within a tag by giving the xpath of the element by using this function -List ele = driver.findElements(By.xpath("Xpath of the table"));
- Now you can use a for each loop to loop through each of the tags in the above list and then read each value by using getText() method.

44. What is object repository?

It is collection of object names their properties, attributes and their values. It may be excel, XML, property file or text file

45. How to handle alerts and confirmation boxes. Confirmation boxes and Alerts are handled in same way in selenium.

```
var alert = driver.switchTo().alert();
alert.dismiss();  //Click Cancel or Close window operation
alert.accept();   //Click OK
```

Handle Confirmation boxes via JavaScript,

```
driver.executeScript("window.confirm = function(message){return true;};");
```

46. How to mouse hover on an element?

```
Actions action = new Actions(webdriver);
WebElement we = webdriver.findElement(By.xpath("html/body/div[13]/ul/li[4]/a"));
action.moveToElement(we).moveToElement(webdriver.findElement(By.xpath("/expression-
here"))).click().build().perform();
```

47. How to switch between the windows?

```
private void handlingMultipleWindows(String windowTitle) {
        Set windows = driver.getWindowHandles();
for (String window : windows) {
driver.switchTo().window(window);
          if (driver.getTitle().contains(windowTitle)) {   return;   }     }     }
```

48. How to switch between frames?

WebDriver's driver.switchTo().frame() method takes one of the three possible arguments:

A number.

Select a frame by its (zero-based) index. That is, if a page has three frames, the first frame would be at index "0", the second at index "1" and the third at index "2". Once the frame has been selected, all subsequent calls on the WebDriver interface are made to that frame.

A name or ID.

Select a frame by its name or ID. Frames located by matching name attributes are always given precedence over those matched by ID.

A previously found WebElement.
Select a frame using its previously located WebElement.
Get the frame by it's id/name or locate it by driver.findElement() and you'll be good.

49. How to take the screen shots in seelnium2.0?

```
//store screenshots
public static void captureScreenShot(String filePath) {
  File scrFile = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
try {
FileUtils.copyFile(scrFile, new File(filePath));
  } catch (IOException e) {
   // TODO Auto-generated catch block
e.printStackTrace();
}
}
```

50. What is the default time for selenium Ide and webdriver?

Default timeout in selenium ide is 30 seconds.
For web driver go to below URL:
http://assertselenium.com/2013/01/29/webdriver-wait-commands/

51. Write down scenarios which we can't automate?

Barcode Reader, Captcha etc.

52. Differences between jxl and ApachePOI.

- jxl does not support XLSX files
- jxl exerts less load on memory as compared to ApachePOI
- jxl doesn't support rich text formatting while ApachePOI does.
- jxl has not been maintained properly while ApachePOI is more up to date.
- Sample code on Apache POI is easily available as compare to jxl.

53. Does Selenium support https protocols?

- Yes

54. "I want to find the location of ""b"" in the below code, how can I find out without using xpath, name, id, csslocator, index.

a
b
c

- driver.findElement(By.xpath("//*[contains(text(),'b')]")).click(); or

- //div/button[contains(text(),'b']

55. Name 5 different exceptions you had in selenium web driver and mention what instance you got it and how do you resolve it?

- WebDriverException
- NoAlertPresentException
- NoSuchWindowException
- NoSuchElementException
- TimeoutException

1. WebDriverException

WebDriver Exception comes when we try to perform any action on the non-existing driver.

```
WebDriver driver = new InternetExplorerDriver();
driver.get("http://google.com");
driver.close(); driver.quit();
```

2. NoAlertPresentException

When we try to perform an action i.e., either accept() or dismiss() which is not required at a required place; gives us this exception.

```
try{
driver.switchTo().alert().accept();
}
catch (NoAlertPresentException E){
E.printStackTrace();
}
```

3. NoSuchWindowException

When we try to switch to an window which is not present gives us this exception:

```
WebDriver driver = new InternetExplorerDriver();
driver.get("http://google.com");
driver.switchTo().window("Yup_Fail");
driver.close();
```

In the above snippet, line 3 throws us an exception, as we are trying to switch to an window that is not present.

4. NoSuchFrameException

Similar to Window exception, Frame exception mainly comes during switching between the frames.

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

```
driver.get("http://google.com");
driver.switchTo().frame("F_fail");
driver.close();
```

In the above snippet, line 3 throws us an exception, as we are trying to switch to an
frame that is not present.

5. NoSuchElementException

This exception is thrown when we WebDriver doesn’t find the web-element in the DOM.

```
WebDriver driver = new InternetExplorerDriver();
driver.get("http://google.com");
driver.findElement(By.name("fake")).click();
```

6. TimeoutException

Thrown when a command does not complete in enough time.

All the above exceptions were handled using try catch exceptions.

56. How do you manage the code versions in your project?

- Using SVN or other versioning tools

57. What are the test types supported by Selenium?

- Selenium supports UI and functional testing. As well it can support performance testing for reasonable load using selenium grid.

58. In what all case we have to go for “JavaScript executor”.

- Consider FB main page after you login. When u scrolls down, the updates get loaded. Tohandle this activity, there is no selenium command. So you can go for javascript to setthe scroll down value like driver.executeScript("window.scrollBy(0,200)", "");

59. What is the difference between data driven and Keyword driven framework?

- Data Driven Framework: Basically this type of Framework we read data from excel/some external sheet.
- Keyword Driven Framework: In the main script keyword drives which workflow/ testcases to execute in generic term

60. What is the difference between Text area and Text field component?

- **Text Area**: in generally like comment area where you are provide like 200 characters etc.
- **Text filed**: where you type predefined text like country name etc.

61. How you will configure TestNG with selenium?

- *help-> new software -> add url*

- http://beust.com/eclipse

62. How to count total no of hyperlinks in a page and click on the 5 th link if exist else display an error message 5th link is not exist.

- List<Webelemet>alllinks=driver,finElementsbytagname("a");
  syso(alllinks.size());

63. There is an airline website from where you can choose source to destination place (e.g Bangalore to Delhi) which gives you 10 different flight details like

A001N
A002N
A003N
A004N
A00234M
A00123M
…

From the above list include the flights which are ends with “M “character and exclude the flights which are ends with “N” character.

- Use xpath functions :- ***endWith()*** to find element ending with N

64. What is the difference between Web driver wait and fluent wait when we need to choose web driver wait and fluent wait?

- Fluent wait / driver wait belongs to explicit wait where it used to wait for Specific element
- Fluent wait: suppress some expectation when element not found, Polling of web element we can control which web driver wait doesn’t provide

68. Types of locator in selenium and which locator you used frequently?

- ID, Name, Xpath and CSS Selector

69. How to click on an element which is not visible using selenium WebDriver?

- We can use JavascriptExecutor to click

70. What is the use of @FindBy annotation?

- @FindBy is used to identify element in the Page Factory approach