

## Q1) What is Automation?

Making Testing process Automated using tools like Selenium or UFT or LoadRunner ect is called Automation testing.

## Q2) What are the advantages in Automation?

Faster than manual testing in execution.

code must be reusability and repeated.

Tests can be re-used on different versions of the software.

It saves time and money.

Automation scripts can be reused with no or minimal changes in the script.

## Q3) What is Selenium?

Selenium is a Web interface Automation framework, It automates browser components, When we are getting challenges in Manual testing like repeated tasks we will move for selenium. In manual test cases, we can convert into Automation Test Cases. It supports multiple languages like Java, C#, Ruby, Perl, and Python and also supports multiple browsers.

## Q4) Advantages of Selenium:

Selenium supports web-based and mobile Automation.

Selenium is an open source tool.

It will support Multiple Languages

Selenium supports cross-browser testing (chrome, safari, IE, firefox....).

Selenium supports third-party tools.

Selenium is a user-friendly tool.

Selenium supports different operating systems like Windows, MacOS, Linux, etc.

## Q5) What are the types of Selenium components?

It has generally 4 components. Each component had a different approach to automate. They are:

IDE

RC

WebDriver

Grid

## Q6) What is selenium IDE?

It supports record and playback.

It supports only user interface components in selenium (like the page, visible components).

Only component support base state application Automation.

It assists in exploratory testing.

## Q7) Advantages of Selenium IDE?

Recorded script convert into different components and languages.

Default object identification.

Can run test suites periodically.

Record and playback.

Create test cases.

Create test suites.

We can debug test cases.

We can edit test cases

## Q8) What are the disadvantage in IDE?

It Only supports Firefox.

It supports only page operations.

Limitations in Automation.

It does not support enhance test cases.

We can't store elements.

## Q9) What is Selenium RC?

RC is a server based API(Application Programming Interface) command to interact with interface browser.

Here server play key role to launch a browser to kill the browser. It is not a user interface140

## Q10) Advantages of selenium RC?

It supports multiple languages.

It supports different environment.

It supports multiple browser.

## Q11) What is WebDriver?

To overcome facing RC problems we move to WebDriver. It is a programming interface, to create and execute test cases.

Test engineer writes the Automation code, that code send to the browser driver components, the browser component executes the tester request, it sends its own request to the browser, the browser then executes the browser driver request.

Here using element locators we can create test cases and execute test cases.

## Q12) Advantages of WebDriver?

It supports multiple languages.

- It supports multiple browser.
- It supports third party tools.
- It support cross browser testing.
- It supports various environments.
- It supports web based automation and mobile base automation.

### Q13) DisAdvantages of WebDriver does?

- It not generate detailed test reports, it generates only a test summary report.
- It is not having any default repository.
- It doesn't support desktop applications

### Q14) What is Grid?

Selenium Grid is used to execute test cases in multiple browsers, multiple environment and multiple systems in parallel. Grid is not support to create test cases.

It supports parallel execution in a dual environment. Here we can execute webdriver test cases and also RC test cases.

### Q15) What is Locator?

Locators used to recognize an object in a web page.

### Q16) What are the types of Locator in WebDriver?

- id
- name
- classname
- css selector
- xpath: Xpath is unique locator in selenium to recognize an object based on tagname.
- LinkText: a link having some outer text ie., linktext.
- PartialLinktext: link text with unique characters.
- TagName: tagName generally we use for to identify a group of objects.

### Q17) Which Locator You Used? And Why?

Xpath is a unique locator builder in selenium. Based on html tag position xpath target object location.

XPaths is unique addresses for each and every web object. It can be used with selenium to perform operations on each object present in the webpage. In the web page, each web-object has a primary unique XPath.

### Q18) Types of Xpath?

- Absolute xpath

## Q19) What is Absolute Xpath and Syntax?

It is useful to target object using tagname position. Generally, we follow when object properties dynamically changing but object position static in the page.

*Syntax for absolute xpath:*

—>/html/body/child tagname.....(/tagname/tagname/tagname....)

Get Selenium Online Training

## Q20) What is Relative Xpath and any Syntax?

Relative xpath is useful to target object using Attribute names. We use relative xpath when object dynamically changing its position within a page.

*Syntax for Relative xpath:*

```
driver.findElement(by.xpath("./tagname[@property='property value']"))
```

```
driver.findElement(by.xpath("./*[@locator='locator value']"))
```

## Q21) How To Work With TextBox?

```
driver.findElement(by.locator("locator value")).sendKeys("Elugu Manojkumar");
```

## Q22) How to Work With Buttons?

```
driver.findElement(by.locator("locator value")).click();
```

## Q23) How to work With Radio Button?

```
driver.findElement(by.locator("locator value")).click();
```

## Q24) How to Work With Checkboxes?

```
driver.findElement(by.locator("locator value")).click();
```

## Q25) How to Work With DropDown?

```
Select obj=new Select(WebElement);
```

```
obj.SelectbyIndex();
```

```
obj.selectbyVisibleText("");
```

```
obj.SelectbyValue("");
```

## Q26) How to Create WebElement?

```
WebElement object=driver.findElement(by.locator("locator value"));
```

## Q27) How To Launch Application with FirefoxDriver?

```
WebDriver obj=new FireFoxDriver();
```

```
Obj.get("url")
```

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

## Q28) How to Launch Chrome Driver?

```
System.setProperty("webdriver.chrome.driver","chrome driver file location");
```

```
Drivder=new chromeDriver();
```

## Q29) How to Work with the beta version(gecko driver)?

```
System.setProperty("webdriver.gecko.driver", "folder path\\geckodriver.exe");
```

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

## Q30) How to Store Group of Elements into single Object?

```
List<WebElement> All_pageElements=driver.findElements(By.tagName("tagname value"));
```

## Q31) How to get cell value from Table?

```
WebElement table=driver.findElement(By.xpath("xpath value"));
```

```
List all_rows=table.findElements(By.tagName("tr"));
```

```
WebElement selected_row=all_rows.get(index);
```

```
int size=all_rows.size();
```

```
System.out.println(size);
```

```
List selected_coloum=selected_row.findElements(By.tagName("td"));
```

```
int coloum_size=selected_coloum.size();
```

```
System.out.println(coloum_size);
```

```
WebElement selected_cell=selected_coloum.get(number);
```

```
String cell_value=selected_cell.getText();
```

### **Q32) How to Get Page Title?**

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

### **Q33) How to Get url?**

```
String url=driver.getCurrentUrl();
```

### **Q34) How to get PageSource?**

```
String pagesource=driver.getPageSource();
```

### **Q35) How to get Attribute?**

```
String attribute=element.getAttribute("attribute name");
```

### **Q36) How to Element Colour?**

```
String colour=element.getCssValue("colour");
```

### **Q37) How to get text?**

```
String text=element.getText();
```

### **Q38) How to get TagName?**

using this method we can find out selected target html tagname

```
WebElement element=driver.FindElement(By.TagName("TagName value"));
```

```
String tag=element.getTagName();
```

```
system.out.println(tag);
```

### 37. How to Get AlertText?

```
//to switch to alert
```

```
Alert alrt=driver.switchTo().alert();
```

```
//to Get alert text
```

```
alrt.getText();
```

### Q39) How to send a value into Alert?

```
//to switch to alert
```

```
Alert alrt=driver.switchTo().alert();
```

```
//to send value into alert
```

```
alrt.sendKeys("Elugu Manojkumar");
```

## Get Selenium 100% Practical Training

### Q40) How to Element will be Displayed in a page or not?

isDisplayed is a boolean type return true or false, useful to verify

object displayed in a page (links, Edit boxes, button, Dropdown).

```
boolean b=driver.findElement(By.locator("locator value")).isDisplayed();
```

### Q41) How to Know Element will be Selected or not?

isSelected is a boolean type return true or false , useful to verify

checkboxes and radio button status check or uncheck

```
boolean b=driver.findElement(BY.locator("locator Value")).isSelected();
```

### Q42) How to Know Element is Enabled or Not?

isEnabled() is a boolean type it return true or false, useful to verify

any object enable and disable status..

```
boolean b=driver.findElement(BY.locator("locator Value")).isEnabled();
```

### Q43) What is ImplicitWait?

We define implicitly wait only once, at browser launch time

it apply synchronization and set timeout until browser window to open

(until dom(document object model) to load)

Syntax:–.

```
driver.manage().timeouts().implicitlyWait(100, TimeUnit.SECONDS);
```

### Q44) What are The timeout commands in Implicitwait?

Days

Hours

Minutes

Seconds

Microseconds

Nanoseconds

Milliseconds

### Q45) Advantages of ExplicitWait?

Ajax windows we can handle

Ajax controls we can handle

We can avoid no such element found an error

Can design advanced webdriver scripting

### Q46) How to store multiple windows into one object?

```
Set<String> multiplewindows=driver.getWindowHandles();
```

### Q47) How to switch one frame to another frame?

```
//Click Frames link
```



```
driver.findElement(By.xpath("xpath value")).click();
```

```
driver.switchTo().frame("packageFrame");
```

```
driver.findElement(By.linkText("Alert")).click();
```

```
driver.switchTo().defaultContent(); //Clear all switches before applying switch second time onwards
```

```
driver.switchTo().frame("classFrame");
```

```
//Click no frames link
```

```
driver.findElement(By.xpath("xpath value")).click();
```

## Q48) How to Take screenshot?

```
try {
```

```
//Timestamp using java
```

```
DateFormat df=new SimpleDateFormat("yyyy_MMM_dd hh_mm_ss");
```

```
Date d=new Date();
```

```
String time=df.format(d);
```

```
File src=((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
```

```
FileUtils.copyFile(src,new File("E:\\selenium_Workspace_Week20\\screens\\EluguManojkumar"+time+".png"));
```

```
} catch (Exception e)
```

```
{
```

```
System.out.println(e.getMessage());
```

```
}
```

## Q49) Why we are using junit framework in selenium:

To optimize test case and Scripts.

By default, Selenium doesn't have reporting options.

To compile methods in a class without creating an object to compile methods in a class we follow below annotations.

## Q50) What is annotation?

Annotation organize method execution within the class, In order to execute

methods in a class we follow annotations.

## Q51) What is @Test?

@Test:→

It enable method execution, Also enabled compile options to method

with in a class.

We define @Test annotation for method

syntax:→

@Test

public void methodname()

{

System.out.println("elugumanojkumar@gmail.com");

}

## Q52) What is Beforeclass?

Beforeclass annotation execute , before first @Test annotation is execute

with in class.

syntax:→

@BeforeClass

public void methodname()

```
{  
  
System.out.println("elugumanojkumar@gmail.com");  
  
//usually we declare ,Variable and object , Initiation under this method  
  
}
```

### Q53) what is @Afterclass?

AfterClass annotation execute, After last @Test annotation is executed

within class.

@AfterClass

public void methodname()

```
{  
  
System.poy.println("Elugu ManojKumar");  
  
}
```

### Q54) what is @Before?

Before the method is executed before each @Test method with in a class

All test methods needed a same precondition.

@Before

public void methodname()

```
{  
  
System.out.println("elugumanojkumar@gmail.com");  
  
}
```

### Q55) what is @After?

After annotation execute after each @Test method execution within a class

All test methods needed a same kind of post condition.

@After

```
public void methodname()
```

```
{
```

```
System.out.println("elugumanojkumar@gmail.com");
```

```
}
```

### Q56) what is @Ignore?

Ignore annotation skip method execution.

@Ignore

```
public void methodname()
```

```
{
```

```
System.out.println("elugumanojkumar@gmail.com");
```

```
}
```

### Q57) What is TestNG?

Testng is a test approach framework it derived from junit and

Nunit, Testng cover unit testing and functional testing it had

few advanced features compare to junit

### Q58) Advantages of TestNg?

- More annotation available to organize test cases.

- Support parameterization with Dataprovider.

- Excellent result generation compare to junit.

- Easy way to execute a test case from suite files.

- Can generate user define crystal reports.

Parameter can be define from suite file.

Without the main method also we can execute our script.

Testng is not default plugin for eclipse, need to download from marketplaces.

## Q59) What are the regular annotations in Testng and there purpose?

@BeforeSuite: The annotated method will be run before all tests in this suite have run.

@AfterSuite: The annotated method will be run after all tests in this suite have run.

@BeforeTest: The annotated method will be run before any test method belonging to the classes inside the <test> tag is run.

@AfterTest: The annotated method will be run after all the test methods belonging to the classes inside the <test> tag have run.

@BeforeClass: The annotated method will be run before the first test method in the current class is invoked.

@AfterClass: The annotated method will be run after all the test methods in the current class have been run.

@BeforeMethod: The annotated method will be run before each test method.

@AfterMethod: The annotated method will be run after each test method.

Get Selenium Training with Real time Live Projects

## Q60) How to excute multiples classes in on time?

```
<suite name="suite1">
```

```
</suite>
```

## Q61) How to prioritize test methods with in class?

```
public class testngclass
```

```
{
```

```
@Test(priority=0)
```

```
public void testmethod1()
```

```
{
```

```
System.out.println("elugumanojkumar@gmail.com")
```

```
}
```

```
}
```

### **Q62) How to provide description for testng method:**

```
@Test(description="This class done by Elugu Manojkumar")
```

```
public void y()
```

```
{
```

```
Reporter.log("method y is executed");
```

```
}
```

### **Q63) How to Skip method execution in Testng suite?**

```
@Test(enabled=false)
```

```
public void y()
```

```
{
```

```
Reporter.log("method y is executed");
```

```
}
```

```
Reporter.log("user report");—> Should define under test method only
```

### **Q64) How to Pass parameter from XML file?**

```
<suite name="Elugu" >
```

```
<test name="Manojkumar">
```

```
<classes>
```

```
<parameter name="browser" value="ff"></parameter>
```

```
<class name="packagename.classname"></class>
```

```
</classes>
```

```
</test>
```

```
</suite>
```

### **Q65) How to Execute Two or multiple test in suite with xml**

```
<suite name=" This is Elugu Manojkumar suite">
```

```
<test name="This is Elugu Manojkumar first test">
```

```
//Declare what classes to execute
```

```
</test>
```

```
<test name=" This is Elugu Manojkumar second test ">
```

```
//Declare what classes to execute
```

```
</test>
```

```
</suite>
```

### **Q66) How to retrieve Data from Dataprovider method?**

```
@Test(dataProvider = "dp")
```

```
public void f(String elugu,String Manojkumar)
```

```
{
```

```
//dp==> dp is a method name where we maintain original data
```

```
}
```

### **Q67) How to Apply assert in testng class?**

```
//Check Comparison expected to actual value
```

```
Assert.assertEquals(driver.getTitle(), "title name");
```

```
//Check boolean condition true or false
```

```
Assert.assertTrue(driver.findElement(By.locator("locator value")).isDisplayed());
```

## Q68) How to maintain data under dataprovider method?

```
@DataProvider
```

```
public Object[][] dp()
```

```
{
```

```
String data[][]=new String[1][1];
```

```
//First Row data
```

```
data[0][0]="elugumanojkumar@gmail.com";
```

```
data[0][1]="123456789";
```

```
}
```

## Q69) What is grid?

Selenium Grid is used to execute test cases in multiple browsers, multiple environment and multiple systems in parallel. Grid is not support to create test cases.

It supports parallel execution in dual environment. Here we can execute webdriver test cases and also RC test cases.

## Q70) What is Maven?

Maven is a management tool it build projects using project object model [POM]

## Q71) how to Add dependencies to pom.xml

```
<dependencies>
```

```
<dependency>
```

```
<groupId>org.seleniumhq.selenium</groupId>
```



```
<artifactId>selenium-java</artifactId>
```

```
<version>3.13.0</version>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.testng</groupId>
```

```
<artifactId>testng</artifactId>
```

```
<version>6.1.1</version>
```

```
<scope>test</scope>
```

```
</dependency>
```

```
</dependencies>
```

Selenium Questions:

## Q72).How to navigate to new window from current window?

driver.navigate().to(new window url) is a method used to navigate to the new window

## Q73).How to get and set the position of window in selenium?

driver.manage().window().getPosition() is used to get the window position. And  
driver.manage().window().setPosition(newPoint(200,400))

## Q74). How to navigate forward and backward from current window?

driver.navigate().back() used to move back a single entry in the browser's window.

driver.navigate().forward() used to move forward single entry in the browser's window

[Are You Interested in Selenium Course ? Click here](#)

## Q75). How to select all elements in the DOM using CSS selector?

This is a wildcard, which will select all elements in the DOM.

```
Ex:driver.findElement(By.cssSelector("*"));
```

### Q76). How to select only checked check boxes or radio buttons?

`driver.findElement(By.cssSelector("input[type=checkbox]:checked"))`. This will select only checked check box and radio button

### Q77). What is xpath?

xpath stands for XML path. It is used to find the element in the DOM. Success rate is high finding an element in the DOM using xpath

### Q78). How to find an element using xpath with 'and' operator?

`driver.findElement(By.xpath("//tagname[@attribute1='value1' and @attribute2='value2']"))` this Xpath will select the element with attribute1 with value1 and attribute2 with value2

### Q79). How to find Xpath using index?

`//tagname[number]`

Ex: `driver.findElement(By.xpath("//tr[2]/td[2]"))`

This will select second td element in second row(tr)

### Q80). How to use start-with() function?

`//input[starts-with(@id,"che")]`

This xpath will select the element with whose tagname is input, attribute is id and value starts with che

### Q81). How many methods present in the SelectElement class for drop-down and multi select dropdown?

Totally 11 methods present in the SelectElement class.

`(getOptions(),isMultiple(),selectByVisibleText(),deselectByVisibleText(),selectByValue(),`

`deselectByValue(),selectByIndex(),deselectByIndex(),getFirstSelectedOption(),`

`getAllSelectedOptions(),deselectAll())`

### Q82). What is Action class?

Advanced keyboard and mouse actions are performed by using the Action class.

Action class in org.openqa.selenium.interactions package

### **Q83). How to use moveToElement with and without parameter?**

moveToElement(WebElement element).This will move the mouse to specified webelement.

moveToElement(WebElement element,int Xoffset,int Yoffset).This will move mouse to the specified offset of the top-left corner of the specified element.

### **Q84). How to use click method with and without parameter?**

click(): This method clicks at the current mouse location

click(WebElement element):clicks the mouse on the specified element.

### **Q85). What is synchronization?**

matching the automation tool speed with application speed is synchronization.

### **Q86). What is implicit wait?**

If webdriver wait for defined amount of time until webdriver find the element in DOM

```
Driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS)
```

### **Q87). What is explicit wait?**

The webdriver wait until expected conditions found.

```
WebDriverWait wait=new WebDriverWait(driver,20);
```

```
Wait.until(ExpectedConditions.presenceOfAllElementsLocatedBy(By.className("ContactUS")))
```

### **Q88). How to handle multiple windows?**

When we have multiple window all we need to do is switching from one window to another.

```
ArrayList<String> newTab=new ArrayList<String>(driver.getWindowHandles());
```

```
Driver.switchTo().window(newTab.get(1));
```

### Q89). How to take screenshot in selenium?

TakesScreenshot interface method used to take the screenshot of the current page displayed in the driver instance.

Ex: File screenshot=(TakesScreenShot)driver).getScreenshotAs(OutputType.FILE);

### Q90). How to achieve PageObjectModel

Two methods are to achieve PageObjectModel

\*Normal Approach

\*using Page Factory and @Find By

### Q91). Advantages of POM?

\*Easy maintain

\*Readable format

### Q92). What is TestNG listeners and why it is used?

A listener is defined as an interface that modifies the default TestNG's behavior. As a name suggests Listeners "listen" to the event of defined in the selenium script and behave accordingly. TestNG is used in a selenium by implementing Listeners Interface. It allows customizing TestNG reports or logs. Widely used Listener is ITestListener

### Q93) how to identify tooltip in selenium webdriver?

Simple, with title tag and Action class

Write an xpath for an actual element

Get the attribute of title of that webelement

Ex: String tooltip = driver.findElement(By.xpath("//a[text()='Projects']")).getAttribute("title");

### Q94) How to execute Cucumber Tests in Groups using Cucumber Tags?

By specifying @RunWith(Cucumber.class) and @cucumberOptions annotations

code:

@RunWith(Cucumber.class)

```
@cucumberOptions(

feature="Feature name",

,glue={"stepdef","stepdef"},

tags={"@smokesuite","@Regrssion"}

)
```

Something we can do with TestNG groups attribute of @Test annotation as well.

Example:

```
Class GroupTestExample{

@Test(groups = { "functest", "checkintest" })

public void testPrintMessage() { }

@Test(groups = { "checkintest" })

public void testSalutationMessage() { }

@Test(groups = { "functest" })

public void testingExitMessage() { }

}
```

TestNG.xml:

```
<suite name = "Suite1">

<test name = "test1">

<groups>

<run>

<include name = "functest" />
```

```
</run>
```

```
</groups>
```

```
<classes>
```

```
<class name = "GroupTestExample" />
```

```
</classes>
```

```
</test>
```

```
</suite>
```

## Q95). How to run multiple test suites in selenium using testng?

TestNG is capable of running multiple TCs in one Testsuites as well as multiple test suites also. For that first we need to create testng.xml file inside our project structure, now inside xml give the below code (Test name and class name may differ as per your project)

```
<suite thread-count="1" verbose="1" name="Gmail Suite" annotations="JDK" parallel="tests">
```

```
<test name="test1">
```

```
<classes>
```

```
<class name="com.suite1.test1"/>
```

```
</classes>
```

```
</test>
```

```
<test name="test2">
```

```
<classes>
```

```
<class name="com.suite2.test2"/>
```

```
</classes>
```

```
</test>
```

```
</suite>
```

Now lets see how can we run multiple suite in TestNG

```
<suite name="TestNG Dadadriver suite" >
```

```
<!-- suite name="Suite Name" -->
```

```
<suite-files>
```

```
<suite-file path="./suiteA.xml" />  <!-- path of first suite path >
```

```
<suite-file path="./suiteB.xml" />  <!-- path of second suite path >
```

```
<suite-file path="./suiteC.xml" />  <!-- path of third suite path >
```

```
</suite-files>
```

```
</suite>
```

## Q96) How will you find running text in selenium?

```
getDriver().findElement(By.tagName("marquee")).click(); // click element
getDriver().findElement(By.tagName("marquee")).getText(); //read value
```

## Q97) What is varargs?

**Java has included a feature that simplifies the creation of methods that need to take a variable no. of arguments.**

A variable-length argument is specified by three periods(), In JDK 5

For ex:

```
public static void fun(int ... a)
```

```
{
```

```
// Body
```

```
}
```

## Q98) How to create a object repository in selenium?

We can make one test method run only after successful execution of dependent test method by using `dependsOnMethods` parameter inside `@Test` annotation in TestNG: `@Test(dependsOnMethods = { "preTests" })`

### Q99) How to create a object repository in selenium?

These are the types of object repositories created in Selenium WebDriver.

using Properties file

using XML file

### Q100) How will you handle windows components in selenium?

Using AutoIT, third party application which can be of two parts: AutoIT Editor, Element Identifier

And in the program we should use Runtime class. Likewise so many third party tools are available: Sikuli, Robot class, etc.

### Q101) How will you handle dynamic table in selenium?

Using Relative xpath

### Q102) How will you call TestNG inside Maven?

By adding plugin dependency, here is the code to be written inside pom.xml

### Q103) What is the jar file is used to read and write excel in selenium?

Org.apache.poi – POI is an API and the corresponding jar to be used in TestNG

### Q104) What is the difference between following:: and following-sibling:: in xpath?

Following:: -> it will give all the tags following by the mentioned tag

Following-sibling:: -> it will give only the same level of element tag followed by the mentioned tag

### Q105) What is the default method in Interface?

This is the new feature of Java8 , Interface which allows default accessed method body inside interface

Example: public interface InterfaceB {

default void defaultMethod(){



```
System.out.println("Interface B default method");
```

```
}
```

### **Q106) How will you handle if 2 interfaces having same signatred method?**

Using super keyword..

```
public class Impl implements InterfaceA, InterfaceB {
```

```
public void defaultMethod(){
```

```
// existing code here..
```

```
InterfaceA.super.defaultMethod();
```

```
}
```

```
}
```

### **Q107) What are the Seven Testing Principles?**

Testing shows the presence of defects

Exhaustive testing is impossible

Early testing

Defect clustering

Pesticide Paradox

Testing is context dependent

### **Q108) What are the difference between “/” and “//” in Xpath?**

Single Slash “/” : Create Xpath with absolute path it is used to Single slash

EX : the xpath would be created to start selection from the document node/start node.

Double Slash “//” : Create Xpath with relative path is used to Double slash

EX : the xpath would be created to start selection from anywhere within the document.

### **Q109) What is the Advantages of Test Automation Framework?**

These are the following Advantages of Automation framework:

- Reusability of code
- Easy Reporting
- Maximum coverage
- Recovery scenario
- Low-cost maintenance
- Minimal manual intervention

### **Q110) What are the different types of frame work?**

These are the different types of frameworks:

- Module Based Testing Framework
- Library Architecture Testing Framework
- Data Driven Testing Framework
- Keyword Driven Testing Framework
- Hybrid Testing Framework
- Behavior Driven Development Framework

Most Common Frameworks are:

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

### **Q111) What are the four components of selenium?**

There are four Components:

- 1)Selenium WebDriver
- 2)Selenium Grid
- 3)Selenium Remote Control (Selenium RC)
- 4)Selenium Integrated Development Environment (Selenium IDE)

### **Q112) How to type in a textbox using Selenium?**

User can be use sendKeys("String to be entered") to enter the string in the textbox.

Syntax:

```
WebElement username = drv.findElement(By.id("Email"));
```

```
// entering username
```

```
username.sendKeys("sth");
```

### **Q113) What is the Browser Support Selenium IDE?**

Firefox is the only Browser that Support Selenium IDE

### **Q114) What is a hub in Selenium Grid?**

Hub is a central point or a server, Hub controls the test execution on different machines.

### **Q115) How can you find if an element is displayed on the screen?**

The user follow the methods to check the visibility of the web elements.

These web elements can be buttons, labels, drop boxes, radio buttons, checkboxes etc.

isDisplayed()

isSelected()

isEnabled()

Syntax:

isDisplayed():

```
boolean buttonPresence = driver.findElement(By.id("gbqfba")).isDisplayed();
```

isSelected():

```
boolean buttonSelected = driver.findElement(By.id("gbqfba")).isSelected();
```

isEnabled():

```
boolean searchIconEnabled = driver.findElement(By.id("gbqfb")).isEnabled();
```

### Q116) Types of WebDriver APIs available in selenium?

- Firefox Driver
- Internet Explorer Driver
- Gecko Driver
- HTML unit Driver
- Chrome Driver
- Opera Driver
- Android Driver
- Safari Driver
- EventFiring WebDriver
- iPhone Driver

### Q117) What are the different type of OS Support in Selenium WebDriver?

- Linux
- Windows
- Apple

### Q118) What are the Different type of Programming languages Supported in Selenium Web Driver?

- Java
- Python
- Php
- Perl
- C#
- Ruby

### Q119) What are the Different type of Framework Supported in Selenium Web Driver?

- Junit
- TestNG

### Q120) Advantages of Page object model Framework?

- code reusability
- code maintainability
- object Repository
- Readability

### Q121) Different keyboard operation that can be performed in Selenium?

These are the following keyboard operation , Performed in selenium.

```
sendKeys("Sequence of character")
```

```
pressKey("non-text keys")
```

```
releaseKey("non-text keys")
```

Collapse

## Q122) What is TestNG listeners and why it is used?

Listener is defined as a interface that modifies the default TestNG's behavior. As a name suggests Listeners "listen" to the event of defined in the selenium script and behave accordingly. TestNG is used in a selenium by implementing Listeners Interface. It allows customizing TestNG reports or logs. Widely used Listener is ITestListener

## Q123) how to identify tooltip in selenium webdriver?

Simple, with title tag and Action class

Write an xpath for an actual element

Get the attribute of title of that webelement

```
Ex: String tooltip = driver.findElement(By.xpath("//a[text()='Projects']").getAttribute("title");
```

## Q124) How to execute Cucumber Tests in Groups using Cucumber Tags?

By specifying @RunWith(Cucumber.class) and @cucumberOptions annotations

code:

```
@RunWith(Cucumber.class)
```

```
@cucumberOptions(
```

```
feature="Feature name",
```

```
,glue={"stepdef","stepdef"},
```

```
tags={"@smokesuite","@Regrssion"}
)
```

Something we can do with TestNG groups attribute of `@Test` annotation as well.

Example:

```
Class GroupTestExample{

@Test(groups = { "functest", "checkintest" })

public void testPrintMessage() { }

@Test(groups = { "checkintest" })

public void testSalutationMessage() { }

@Test(groups = { "functest" })

public void testingExitMessage() { }

}
```

TestNG.xml:

```
<suite name = "Suite1">

<test name = "test1">

<groups>

<run>

<include name = "functest" />

</run>

</groups>

<classes>

<class name = "GroupTestExample" />

</classes>
```

```
</test>
```

```
</suite>
```

## Q125) how to run only failed test cases in testing?

By implementing IRetryAnalyzer interface.

Create one class which implements IRetryAnalyzer interface.

Then override retry(ITestResult result) method and inside write a condition which states when will the test should fail (Means, no. of time test should rerun)

Now add @Test(retryAnalyzer = Tests.RetryAnalyzer.class) inside actual class which has @Test annotation

## Q126) What is varargs?

Java has included a feature that simplifies the creation of methods that need to take a variable no. of arguments.

A variable-length argument is specified by three periods(), In JDK 5

For ex:

```
public static void fun(int ... a)
```

```
{
```

```
// Body
```

```
}
```

This syntax tells the compiler, that fun( ) can be called with zero or more arguments. As a result, here a is implicitly declared as an array of type int[].

## Q127) how to run multiple test suites in selenium using testng?

TestNG is capable of running multiple TCs in one Testsuites as well as multiple test suites also. For that first we need to create testng.xml file inside our project structure, now inside xml give the below code (Test name and class name may differ as per your project)

```
<suite thread-count="1" verbose="1" name="Gmail Suite" annotations="JDK" parallel="tests">
```

```
<test name="test1">
```

```
<classes>
```

```
<class name="com.suite1.test1"/>
```

```
</classes>
```

```
</test>
```

```
<test name="test2">
```

```
<classes>
```

```
<class name="com.suite2.test2"/>
```

```
</classes>
```

```
</test>
```

```
</suite>
```

Now lets see how can we run multiple suite in TestNG

```
<suite name="TestNG Dadadriver suite" >
```

```
<!-- suite name="Suite Name" -->
```

```
<suite-files>
```

```
<suite-file path="./suiteA.xml" /> <!-- path of first suite path >
```

```
<suite-file path="./suiteB.xml" /> <!-- path of second suite path >
```

```
<suite-file path="./suiteC.xml" /> <!-- path of third suite path >
```

```
</suite-files>
```

```
</suite>
```

## Q128) how to create a object repository in selenium?

An object repository is a common storage location for all objects. Selenium WebDriver does not offer an in-built object repository by default. However, object repositories can be built using the key-value pair approach wherein the key



refers to the name given to the object and value refers to the properties used to uniquely identify an object within the web page.

The following are the types of object repositories that can be created in Selenium WebDriver.

Object Repository using Properties file

Object Repository using XML file

### **Q129) how will you find running text in selenium?**

```
getDriver().findElement(By.tagName("marquee")).click(); // click element  
getDriver().findElement(By.tagName("marquee")).getText(); //read value
```

### **Q130) What is varargs?**

Java has included a feature that simplifies the creation of methods that need to take a variable no. of arguments.

A variable-length argument is specified by three periods(), In JDK 5

For ex:

```
public static void fun(int ... a)
```

```
{
```

```
// Body
```

```
}
```

### **Q131) How will you take screenshot in selenium?**

We need to type cast WebDriver instance to TakesScreenshot.

```
//Convert web driver object to TakeScreenshot
```

```
TakesScreenshot scrShot =((TakesScreenshot)webdriver);
```

```
//Call getScreenshotAs method to create image file
```

```
File SrcFile=scrShot.getScreenshotAs(OutputType.FILE);
```

```
//Move image file to new destination
```

```
File DestFile=new File(fileWithPath);
```

```
//Copy file at destination
```

```
FileUtils.copyFile(SrcFile, DestFile);
```

The same way we can add same thing inside `onTestFailure()` method inside Listener class also

### **Q132) How can we make one test method dependent on other using TestNG?**

We can make one test method run only after successful execution of dependent test method by using `dependsOnMethods` parameter inside `@Test` annotation in TestNG: `@Test(dependsOnMethods = { "preTests" })`

### **Q133) how to create a object repository in selenium?**

These are the types of object repositories created in Selenium WebDriver.

- using Properties file

- using XML file

### **Q134) How will you handle windows components in selenium?**

Using AutoIT, third party application which can be of two parts: AutoIT Editor, Element Identifier

And in the program we should use Runtime class. Likewise so many third party tools are available: Sikuli, Robot class, etc.

### **Q135) How will you handle dynamic table in selenium?**

Using Relative xpath

### **Q136) How will you call TestNG inside Maven?**

By adding plugin dependency, here is the code to be written inside pom.xml

### **Q137) What is the jar file is used to read and write excel in selenium?**

Org.apache.poi – POI is an API and the corresponding jar to be used in TestNG

### **Q138) How do I launch the browser using WebDriver?**

The syntax can be used to launch Browser:

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

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

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

### **Q139) What is the difference between following:: and following-sibling:: in xpath?**

Following:: -> it will give all the tags following by the mentioned tag

Following-sibling:: -> it will give only the same level of element tag followed by the mentioned tag

### **Q140) What is the default method in Interface?**

This is the new feature of Java8 , Interface which allows default accessed method body inside interface

Example: public interface InterfaceB {

```
default void defaultMethod(){
```

```
System.out.println("Interface B default method");
```

```
}
```

### **Q141) How will you handle if 2 interfaces having same signatured method?**

Using super keyword..

```
public class Impl implements InterfaceA, InterfaceB {
```

```
public void defaultMethod(){
```

```
// existing code here..
```

```
InterfaceA.super.defaultMethod();
```

```
}
```

```
}
```

### **Q142) What are the Seven Testing Principles?**

Testing shows the presence of defects

Exhaustive testing is impossible

Early testing

Defect clustering

Pesticide Paradox

Testing is context dependent

### **Q143) What are the difference between “/” and “//” in Xpath?**

Single Slash “/” : Create Xpath with absolute path it is used to Single slash

EX : the xpath would be created to start selection from the document node/start node.

Double Slash “//” : Create Xpath with relative path is used to Double slash

EX : the xpath would be created to start selection from anywhere within the document.

### **Q144) What is the Advantages of Test Automation Framework?**

These are the following Advantages of Automation framework:

- Reusability of code
- Easy Reporting
- Maximum coverage
- Recovery scenario
- Low-cost maintenance
- Minimal manual intervention

### **Q145) What are the different types of frame work?**

These are the different types of frameworks:

1. Module Based Testing Framework
2. Library Architecture Testing Framework
3. Data Driven Testing Framework
4. Keyword Driven Testing Framework
5. Hybrid Testing Framework
6. Behavior Driven Development Framework

Most Common Frameworks are:

1. Data Driven Testing Framework
2. Keyword Driven Testing Framework
3. Hybrid Testing Framework

### **Q146) What are the four components of selenium?**

There are four Components:

- Selenium WebDriver
- Selenium Grid
- Selenium Remote Control (Selenium RC)
- Selenium Integrated Development Environment (Selenium IDE)

### **Q147) How to type in a textbox using Selenium?**

User can be use sendKeys("String to be entered") to enter the string in the textbox.

Syntax:

```
WebElement username = drv.findElement(By.id("Email"));
```

```
// entering username
```

```
username.sendKeys("sth");
```

### **Q148) What is the Browser Support Selenium IDE?**

Firefox is the only Browser that Support Selenium IDE

### **Q149) What is a hub in Selenium Grid?**

Hub is a central point or a server, Hub controls the test execution on different machines.

### **Q150) How can you find if an element is displayed on the screen?**

The user follow the methods to check the visibility of the web elements.

These web elements can be buttons, labels, drop boxes, radio buttons, checkboxes etc.

- isDisplayed()
- isSelected()
- isEnabled()

Syntax:

isDisplayed():

```
boolean buttonPresence = driver.findElement(By.id("gbqfba")).isDisplayed();
```

isSelected():

```
boolean buttonSelected = driver.findElement(By.id("gbqfba")).isSelected();
```

isEnabled():

```
boolean searchIconEnabled = driver.findElement(By.id("gbqfb")).isEnabled();
```

## Selenium Modules For Better Job Opportunities

### **Q151)Types of WebDriver APIs available in selenium?**

- Firefox Driver
- Internet Explorer Driver
- Gecko Driver
- HTML unit Driver
- Chrome Driver
- Opera Driver
- Android Driver
- Safari Driver
- EventFiring WebDriver
- iPhone Driver

### **Q152) What are the Different type of OS Support in Selenium WebDriver?**

- Linux
- Windows
- Apple

### **Q153) What are the Different type of Programming languages Supported in Selenium Web Driver?**

- Java
- Python
- Php
- Perl
- C#

Ruby

### Q154) What are the Different type of Framework Supported in Selenium Web Driver?

Junit

TestNG

### Q155) Advantages of Page object model Framework?

code reusability

code maintainability

object Repository

Readability

### Q156) Different keyboard operation that can be performed in Selenium?

These are the following keyboard operation , Performed in selenium.

`.sendKeys("Sequence of character")`

`.pressKey("non-text keys")`

`.releaseKey("non-text keys")`

### Q157) Difference for get and navigate method in Selenium?

`get("google.com")`: start page for a script

`navigate.to("google.com")`: navigate to other pages during a script execution

### Q158) Difference between quit and close methods in Webdriver?

close method: close current browser window

quit method: close all open browser windows

### Q159) Difference between implicit and explicit wait?

Implicit: applies to all elements in the script

Explicit: applies to specific element

### Q160) Code to handle 3-child window?

```
Set<String> windows = driver.getWindowHandles();
```

```
iter = windows.iterator();
```

```
while(iter.hasNext()) // loop through all the windows
```

```
iter.next();
```

### **Q161) Different types of locators present in webdriver?**

xpath, css, id, classname, linkText, name

### **Q162) Write syntax for xpath and css if id and tag are given.**

xpath=//tagName[@id='value']

css=tagName[id=value]

### **Q163) How to use contains regex to xpath?**

//tagName[contains(@id,'partialtext')]

### **Q164) How to use regex to CSS?**

tagname[id\*='substring']

tagname[id^='startstring']

tagname[id\$='endstring']

### **Q165) What are the class available in selenium to handle drop downs?**

Select

### **Q166) What is the method to check if check box is selected?**

driver.findElement("identifier").isSelected(); // true or false

### **Q167) How to validate if element is hidden or visible on webpages?**

driver.findElement("identifier").isDisplayed(); // true or false

### **Q168) What is the use of Selenium Grid?**

Distribute tests across multiple machines or OS



**Q169) How to mouse over to an element on a webpage?**

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

```
Actions.moveToElement(locator).build().perform();
```

**Q170) Methods to handle Alert?**

```
driver.switchTo.alert().dismiss();
```

**Q171) How to get count of links in the page?**

```
driver.findElements (By.tagName("a")).size();
```

**Q172) How to navigate if we have navigated to child window correctly?**

```
driver.getTitle() // validate once a child window is opened
```

**Q173) Difference between relative and absolute xpath?**

Absolute: /html/body/div/a

Relative: //div/a

**Q174) Write xpath to handle parent from child window.**

```
//parent/child
```

**Q175) How to set driver path from script?**

```
System.setProperty("webdriver.chrome.driver", "path_to_chromedriver")
```

**Q176) List any two methods for explicit wait.**

```
visibilityOfElementLocated, invisibilityOfElementLocated
```

**Q177) How to take screenshots from webdriver?**

```
File src = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
```

```
FileUtils.copyFile(src, new File("path to new image file"))
```

**Q178) How to press Enter key from webdriver?**

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

### **Q179) What the various access modifiers?**

Public

Protected

Private

Default

### **Q180) What is the purpose of static methods and variables?**

To share among all the instances of a class

### **Q181) What do you mean by Checked Exceptions?**

Subclass of Exception

Code doesn't compile if not handled

### **Q182) Difference between Inner Class and Sub Class?**

Inner Class: Nested within another class

Sub Class: Inherits from another class (Base Class)

### **Q183) What is a singleton class?**

One instance

Control the object creation

### **Q184) What is Final Keyword in Java?**

Final class: cannot be inherited

Final variable: cannot be modified

Final method: cannot be overridden

### **Q185) What are Java Packages?**

Collection of classes and interfaces

Helps to modularize

Can be imported

## **Q186) Difference of Abstract class and Interface in Java?**

Abstract class:

- Can have non-abstract methods
- No need to implement all inherited methods
- Uses extends keyword
- Can extend only one abstract class

Interface:

- Provides absolute abstraction
- Must implement all inherited methods
- Uses implements keyword
- Can implement multiple interfaces

## **Q187) What is an immutable class?**

Content can't change

State can't change

Automatically synchronized

## **Q188) Why Strings in Java are called as Immutable?**

String object cannot be changed

## **Q189) What can you automate?**

Regression test suite

Smoke / Sanity test suite

Build deployment

Test data creation

Automating behind of the GUI like testing of APIs and methods.

### **Q190) When is Automation Testing useful?**

Regression testing

Non-functional testing

Complex calculations

Repeated executions

### **Q191) What is a framework?**

A framework is a set of the structure of the whole automation suite. It is also a guideline, which if followed can result in a structure which is easy to control and enhance.

### **Q192) What are the different types of an automation tool that you are aware of?**

Open source tool like Selenium, JMeter etc. Other Automation Tools

Paid tools like QTP, Load Runner, RFT, and Rational Robot.

### **Q193) Can you tell some good coding practices Do automation?**

Add appropriate comments.

Identify the methods and write it in a separate file.

Follow the language-specific coding conventions.

Maintain the test data of separate file.

Run your scripts regularly.

### **Q194) When do you prefer manual testing over the Automation testing?**

The project is short-term and writing scripts it will be time-consuming and costly when compared to manual testing.

Flexibility is required. Automated test cases is the programmed and run in a specific way of configurations.

Usability testing needs to be performed.

Application/module are newly developed is have no previous test cases.

Ad-hoc or exploratory testing needs to be the performed.

### **Q195) Explain key factors for SIT entry criteria in Testing?**

Requirement are signed off from Business

Test cases are approved and signed off by Business.

Test environment is ready for testing

Testers are having all the relevant access to front end and backend

When Unit testing is completed and results should be shared to Testing team.

Unit test results should be approved by Test lead.

Test cases uploaded and ready for execution in the test management tool.

### **Q196) Explain key factors for SIT exit criteria in testing?**

SIT should be completed without SEV1 defects.

There can 2 SEv2 defects in open status with low business priority.

There can 5 Sev3 defects in open status with low business priority.

There should be action Plan for all open defects.

Test completion report should be ready.

### **Q197) Explain the template for over all Exit report?**

Name of the project

Test Lead.

Summary:-

SIT: No of Test case passed,Failed,N/A. Defects Open,closed,Deferred.

Reason for defects deferred and NA scripts.

Regression:- No of Test case passed,Failed,N/A. Defects Open,closed,Deferred.

Reason for defects deferred and NA scripts.

UAT:- No of Test case passed,Failed,N/A. Defects Open,closed,Deferred.

Reason for defects deferred and NA scripts.

Test case execution status and count:-

List of Open defects:-

Defect summary:-

### **Q198) Explain HTTP 404 error?**

HTTP 404 error means client was able to communicate with server but server could not find what the response was.

### **Q199) Among 4 team members 2 members are on sick leave and you need to complete the testing today and there are 200 test cases with one team member with given time line you can completed only 80 test case . All other members in your project are occupied. How do you handle the situation and convince the client?**

First Tester should identify critical and high priority scripts. Get in touch with BA, Business and explain the situation to the project stakeholders. Get sign off from BA on the test case and should execute only the identified test case. By this process the risk is mitigated and the business is happy.

### **Q200) How to handle frames in an application using webdriver methods?**

```
driver.switchTo().frame([identifier])
```

Become an Selenium Expert with Certification in 25hours

### **Q201) What is the Difference between collection and collections?**

Answer: Collections is a class and collection is a framework.

### **Q202) What are the guidelines to be followed for POM class?**

Answer:

Execute the given test cases manually so that we get more clarity on the steps which are to be automated.

While executing the test cases note down the **page elements** present on the page & **actions to be performed** on that page with respect to test case steps.

After executing all the test cases develop respective 'Page' class with elements & methods.

### **Q203) What are the rules to develop POM-Class?**

Answer:

Number of POM classes should be same as number of webpages present in the application & POM class should be created under 'pages' package.

Name of the class should be same as respective webpage name & class name should end with 'Page' word. Ex:- LoginPage.

In every POM class, elements should be declared using '@FindBy' annotation.

Elements should be initialized using 'PageFactory.initElements()' inside Constructor.

Develop 'public' method for all the actions which are to be performed on the page.

### **Q204) How can we perform “Back” and “Forward” operations using selenium?**

Answer: use “driver.navigate().back()” command to move back and “driver.navigate().forward()” command to move forward.

### **Q205) Write a program to print binary format of a given number**

```
public class IntToBinary {  
    public static void main(String[] args) {  
        Scanner input=new Scanner(System.in);  
        System.out.println("Enter a number");  
        int number = input.nextInt();  
        Integer i=new Integer(number);  
        String binary = Integer.toBinaryString(i);  
        System.out.println(binary);  
    }  
}
```

### **Q206) Write a program to print each word in a new line from the given string “This is my java world”**

Answer:

```
public class PrintEachWordInNewLine  
{  
    public static void main(String[] args) {  
        String s1="This is my java world";  
        String output1 = s1.replace(" ",System.lineSeparator());  
        System.out.print(output1);  
    }  
}
```

### **Q207) Write a program to print the given below pattern**

1 2

1 2 3

1 2 3 4

1 2 3 4 5

Answer:

```
public class PrintTheGivenPattern {
    public static void main(String[] args) {
        for (int i=1; i<=5; i++)
        {
            for (int j=1; j<=i; j++)
            {
                System.out.print(j+" ");
            }
            System.out.println();
        }
    }
}
```

**Q208) Write a program to print any string in double quotes**

```
public class PrintWithDoubleQuotes {
    public static void main(String[] args) {
        System.out.println("\"Hello World\"");
    }
}
```

**Q209) Write a program to print pyramid pattern.**

```
public class PyramidPattern {
    public static void main(String[] args) {
        for(int i=0;i<5;i++)
        {
            for(int j=0;j<5-i;j++) //
            {
                System.out.print(" ");
            }
            for(int k=0;k<=i;k++)
            {
                System.out.print("$ ");
            }
            System.out.println();
        }
    }
}
```

**Q210) Write a java program to reverse the given string using StringBuffer**

```
public class ReverseStringUsingStringBuffer
{
    public static void main(String[] args)
```



```

{
StringBuffer sb= new StringBuffer("sudeer");
System.out.println(sb.reverse());
}
}

```

### Q211) Can you explain, how string is immutable?

Answer:

```

public class Sample
{
    public static void main(String[] args)
    {
        String s1= "Sudeer";
        s1= "Venkatesh";
        System.out.println(s1);
    }
}

```

In the above example, when you print "s1", it will print "Venkatesh". This means, if you try to modify the existing string, it will create a new string instead of modifying the existing string.

### Q212) How do you print the message only on the TestNG HTML reports in selenium?

Answer: Reporter.log("This Prints message only on the TestNG html report");

### Q213) How do you print the message on the TestNG HTML report and as well as in console in selenium?

Answer: Reporter.log("This Prints message on both TestNG html report and console", true);

### Q214) How to disable a test method in TestNG?

Answer :To disable the test method, we have to use enabled = false to the @Test annotation.

@Test(enabled = false)

### Q215) How to ignore/exclude a test method in TestNG?

Answer : To ignore/exclude the test method, we have to use enabled = false to the @Test annotation. @Test(enabled = false)

### Q216) How to Terminate the test method in TestNG?

Answer: @Test(timeout = 5000)

In the above case, it will terminate the test method in 5 seconds and the method will be marked as 'Failed'.

**Q217) Can you list the types of testing with example?**

Answer:

**1. Functional Testing.**

E.g.: System Testing, Black Box Testing, Acceptance Testing...Etc.

**2. Non-Functional Testing.**

E.g.: Performance Testing, Load Testing...Etc.

**Q218) What is sandwich approach?**

Answer: The combination of Top Down Incremental Integration Testing and Bottom Up Incremental Integration Testing is called as Sandwich Approach.

**Q219) What is Test Cycle?**

Answer: It's a duration given to test the application.

**Q220) What is Compatibility Testing.?**

Answer: Testing the application on different software and hardware platforms is called as compatibility testing.

Note: Testing on different browsers or different versions of the browsers and different operation systems (windows 10, windows 7, Linux...etc.)

Compatibility testing is not mandatory, it depends on the requirements.

**Q221) What is Regression Testing?**

Answer: Re-execution of same test cases on different build to ensure that the changes made in the application has not introduced the defect in the unchanged feature. E.g.: When you get a build for 1<sup>st</sup> time, execute the test cases which are already written. If new features added to the project, write new test cases and execute the new test cases + re-execute the old test cases. Here re-executing the old test cases is called regression testing.

We go for Automation whenever the regression testing is more.

**Q222) What is Baselineing or Bucketing?**

Answer: Baselineing is used to check whether the automation test cases are working as per the manual test cases or not.

**Q223) What is inconsistent defect?**

Answer: The defect which are occurring at irregular intervals of time are called as inconsistency defect.

**Q224) How do you confirm whether the particular web element is visible on the page or not?**

Answer: Using `isDisplayed()` method.

E.g.: `driver.findElement(By.name("name of element")).isDisplayed();`

**Q225) Can you explain the use of Selenium Grid?**

Answer: Selenium Grid can be used to execute same or different scripts on multiple browsers and platforms concurrently in order to achieve distributed test execution.

**Q226) Can you list some exceptions that you have seen in selenium?**

Answer:

`StaleElementReferenceException`

`NoSuchWindowException`

`TimeoutException`

`NoSuchElementException`

`NoAlertPresentException`

`IllegalStateException`

`ElementNotVisibleException`

`WebDriverException`

**Q227) List The types of WAITS available in selenium WebDriver**

Answer:

1. Implicit Waits
2. Explicit Waits
3. Fluent Waits
4. Custom Waits

**Q228) Can we delete cookies using Selenium.? If yes How?**

Answer: Yes, we can delete cookies using selenium. We use `deleteAllCookies()` method to delete cookies.

**Q229) What are the languages supported by Selenium?**

Answer: Java, C#, Ruby, Python, JavaScript (Node JS), Perl, PHP, Objective-C, JavaScript, Haskell, R, Dart, TCL.

### **Q230) What are the browsers supported by Selenium?**

Answer: Chrome, Firefox, IE, Edge, Opera, Safari, Android browser, I-PHONE, I-PAD, Opera Blink, Phantom JS, Html unit.

### **Q231) How do you find the element which is active or focused?**

Answer: Using 'driver.switchTo().activeElement()' statement.

### **Q232) How do you 'Resize' the browser using selenium.?**

Answer:

```
WebDriver driver=new ChromeDriver();
driver.manage().window().maximize();
Dimension d=new Dimension(300, 400);
driver.manage().window().setSize(d);
```

### **Q233) How do you move the browser using selenium?**

Answer:

```
WebDriver driver=new ChromeDriver();
driver.manage().window().maximize();
Point p=new Point(300, 50);
driver.manage().window().setPosition(p);
```

### **Q234) How do you find X & Y co-ordinates of an element? Write code for this?**

Answer: By using 'getLocation()' method.

To find x-value we use 'getX()' & for y-value 'getY()'. This method is present in 'WebElement' interface & method returns 'int' result.

E.g.:

```
int x=username.getLocation().getX();
```

```
int y=username.getLocation().getY();
```

### **Q235) What are the uses of JavascriptExecutor?**

Answer: Using JavascriptExecutor we can

1. Scroll the web page

2. We can enter the value to a textbox.
3. We can use to remove the value present in the textbox.
4. We can click on element.
5. We can use to perform the action even if the element is disabled.

### Q236) List the Differences between Iterator & ListIterator:

Answer:

Iterator	ListIterator
It is unidirectional.(Only Forward)	It is bidirectional. (Forward & Reverse)
It is used to retrieve the objects from collection. (List, Set & Queue)	It is used to retrieve the objects from List
We get iterator type object by 'iterator ()' method	We get list iterator object by 'listIterator()' method
It has methods i.e. hasNext (), next () & remove ()	It has methods i.e. hasNext (), next (), remove (), hasPrevious () & previous ().

### Q237) List the Differences between List & Set:

List	Set
List gives ordered output	Set gives un-ordered output
It is index based	It is non-index based
It accepts duplicates	No duplicates allowed
It receives one or more null values	It receives one or no null values
It supports for, for-each, Iterator & ListIterator	It supports for-each & Iterators only

### Q238) List the Differences between ArrayList & LinkedList:

Answer:

ArrayList	LinkedList
ArrayList is fast in retrieving objects because objects are sitting next to each other	LinkedList is slow in retrieving objects because objects are spreader out
Addition & deletion of objects is slow in ArrayList because number of swaps are more	Addition & deletion of objects is fast in LinkedList because number of swaps are less
ArrayList consumes more memory because it grows by half of its initial size	It consumes less memory compared to ArrayList because it grows by one
It does not have any additional methods	It has few additional methods like addFirst(), addLast(), getFirst(), getLast(), removeFirst(), removeLast()...etc

### Q239) List the Differences between ArrayList & Vector

Answer:

<b>ArrayList</b>	<b>Vector</b>
It is non-legacy class	It is legacy class
It consumes less memory compared to Vector because it grows by half of its initial size	It consumes more memory because it grows by double of its initial size
ArrayList class methods are non-synchronized	Vector class methods are synchronized

#### Q240) List the Differences between HashSet & LinkedHashSet

<b>HashSet</b>	<b>LinkedHashSet</b>
It gives un-ordered output	It gives ordered output
HashSet class methods are synchronized	LinkedHashSet class methods are not-synchronized

#### Q241) List the Differences between TreeSet & LinkedHashSet:

Answer:

<b>TreeSet</b>	<b>LinkedHashSet</b>
It gives sorted output.	It gives ordered output
It can receive only unique objects	It cannot have null value
It can receive any type of objects	It can have only one null value

#### Q242) For what purpose we use the methods of “Scanner class”?

Answer:

We use methods of “Scanner class” to read inputs from keyboard

It is available in java.util package

#### Q243) List Difference b/w final, finally and finalize?

Answer:

final:

final is used to apply restrictions on a class, method and a variable.

final classes cannot be inherited. final methods cannot be overridden and final variable values cannot be changed

final is keyword

finally:

finally block is used to place some important code.

finally block is always gets executed whether an exception is handled or not

finally is a block

finalize:

finalize is used to perform cleanup processing just before object is garbage collected

finalize is a method available in Object class.

## **Q244) Can you list some important points while overriding?**

Answer:

While overriding, method name, argument list and return type should match.

We can change the visibility in the sub-class but we can't reduce the visibility in the sub-class.

For overriding, inheritance is must.

We cannot override private, static members and constructors.

Overriding is applicable only for methods.

## **Q245) What kind of reporting mechanism have you included in your selenium project?**

Answer: We can have different types of reports like Extent report TestNG Surefire report etc.

## **Q246) The difference in the absolute path and relative path.**

Answer: (/) Kickoff and relative path to the path path (/) from the path to the full path

## **Q247) Some Test Gn.**

Answer:

@Test, @messages, @ auditors, @ frontSuite, @ AfterSuite, @FirstTest, @TestTest,

@ DataProvider, @ BeforeGroups, @ AfterGroups, @ BeforeClass, @ AfterClass,

@ FrontMethod, @ AfterMethod, @ factory

## **Q248) What are the desired skills?**

Web Driver attributes to the desired capabilities. If your local installation does not match the default settings, there is a common application suite to set the path of the Firefox driver.

## **Q249) Difference Between Web Driver Listeners and Testers Listeners.**

Answer: TestNG and WebDriver are various interfaces to implement and inherit them. They both change their behavior. You can use questions that are aware of. `FirefoxDriver driver = new FirefoxDriver ();` This URL provides a comprehensive list of listeners and their interfaces.

### **Q250) Which is the best way to find an element?**

Answer: The property elements are usually the fastest option by ID because at the root it will eventually be optimized by multiple browsers. Calling `getElementById ()`.

Property values are used by XPath for asset values using more complex selectors, and a more flexible choice strategy, but it is possible to be very slow in IE. In IE 6, 7, or 8, searches through XPath, can be slowly slower than Firefox. The IE provides the default XPath-over-HTML solution, so the program must use JavaScript enabled and the JavaScript engine in the legacy version of IE is really reluctant.

If you have an object using a complex selector, I usually recommend CSS selectors if possible. The CSS selector is not quite as flexible as XPath, but XPath can hide many of the same suits, without revealing the serious performance penalty in IE.

Become an Selenium Certified Expert in 25Hours

### **Q251) What is Selenium?**

Answer: Selenium is the robust test automation suite that is used for the automating web based applications. It supports multiple browsers, programming languages and platforms.

### **Q252) What are different forms of selenium?**

Answer: Selenium comes in four forms-

Selenium WebDriver – Selenium WebDriver is used to automate web applications using browser's native of methods.

Selenium IDE – A firefox plugin that works on the record and play back on principle.

Selenium RC – Selenium Remote Control(RC) is officially deprecated by selenium and it used to work of javascript to automate the web applications.

Selenium Grid – Allows selenium tests to run in the parallel across multiple machines.

### **Q253) What are some advantages of selenium?**

Answer: Following are the advantages of selenium-



Selenium is the open source and free to use without any licensing cost.

It supports multiple languages like Java, ruby, python etc.

It supports multi browser testing.

It has good amount of the resources and helping community over to the internet.

Using selenium IDE component, non-programmers can also write to automation scripts

Using the selenium grid component, distributed testing can be carried out on remote machines is possible.

### **Q254) What are some limitations of selenium?**

Answer: Following are the limitations of selenium-

We cannot test desktop on application using selenium.

We cannot test web services using to the selenium.

For creating robust scripts in the selenium webdriver, programming language knowledge are required.

We have to rely on external libraries and tools for the performing tasks like – logging(log4J), testing on framework- (testNG, JUnit), reading from external files(POI for excels) etc.

### **Q255) Which all browsers/drivers is supported by Selenium Webdriver?**

Answer: Some commonly used to browsers supported by selenium are-

Google Chrome – An ChromeDriver

Firefox – An FireFox Driver

Internet Explorer – An Internet Explorer Driver

Safari – An Safari Driver

HtmlUnit (Headless browser) – An HtmlUnitDriver

Android – An Selendroid/Appium

IOS – An ios-driver/Appium

**Q256) Can we test APIs or web services using to Selenium webdriver?**

Answer: No selenium webdriver uses through browser's native method to automate on web applications. Since web services are headless, so we cannot automate web services using to selenium webdriver.

**Q257) What are the testing type supported by Selenium WebDriver?**

Answer: Selenium webdriver can be used for the performing automated functional and the regression testing.

**Q258) What are various ways of locating an element in selenium?**

Answer:

The different locators in selenium are-

Id

XPath

cssSelector

className

tagName

name

linkText

partial LinkText

**Q259) What is an XPath?**

Answer: Xpath or XML path is the query language for selecting nodes from the XML documents. XPath is one of the locators supported by the selenium webdriver.

**Q260) What is an absolute XPath?**

Answer: An absolute XPath is a way of the locating an element using an XML expression beginning from the root node i.e. html node in case of the web pages. The main disadvantage of absolute xpath is that even with the slightest change in the UI or any element the whole absolute can XPath fails.

Example – html/body/div/div[2]/div/div/div/div[1]/div/input

**Q261) What is a relative XPath?**

Answer: A relative XPath is the way of locating an element using an XML expression beginning from the anywhere in the HTML document. There are different ways of creating on relative XPaths which are used for the creating robust XPaths (unaffected by changes in other UI elements).

Example – `//input[@id='username']`

**Q262) What is the difference between the single slash(/) and the double slash(//) in XPath?**

Answer: In XPath a single slash is used for the creating XPaths with a absolute paths beginning from root node.

Whereas double slash is used for the creating relative XPaths.

**Q263) How can we inspect the web element attributes in order to use them in the different locators?**

Answer: Using Firebug or developer tools we can inspect to the specific web elements.

Firebug is a plugin of firefox that provides various of development tools for the debugging applications. From automation perspective, firebug is used to specifically for inspecting web-elements in order to get use their attributes like id, class, name etc. in different locators.

**Q264) How can we locate an element by the only partially matching its attributes value in the Xpath?**

Answer: Using contains() method we can locate an element by the partially matching its attribute's value. This is particularly helpful in the scenarios like where the attributes have dynamic values with the certain constant part.

xPath expression = `//*[contains(@name,'user')]`

The above that statement will match the all the values of name attribute to containing the word 'user' in them.

**Q265) How can we locate elements using their an text in XPath?**

Answer:

Using the text() method –

xPathExpression = That `//*[text()='username']`

**Q266) How can we move to the parent of an element using XPath?**

Answer: Using '..' expression in XPath we can move to the parent of an element.

### **Q267) How can we move to the nth child element using XPath?**

Answer:

There are two ways of navigating to the nth element of using XPath-

Using square brackets with the index position-

Example – div[2] will find to the second div element.

Using a position()-

Example – div[position()=3] will find to the third div element.

### **Q268) What is the syntax of the finding elements by class using to CSS Selector?**

Answer: By .className we can select all the element belonging to the particular class e.g. '.red' will select all elements having to the class 'red'.

### **Q269) What is the syntax of finding elements by id using the CSS Selector?**

Answer: By #idValue we can select all the element belonging to an particular class e.g. '#userId' will select to the element having id – userId.

### **Q270) How can we select elements by their attribute value using CSS Selector?**

Answer: Using [attribute=value] we can select all the element of belonging to the particular class e.g. '[type=small]' will select to the element having attribute type of value 'small'.

### **Q271) How can we move to the nth child element using css selector?**

Answer: Using :nth-child(n) we can move to the nth an child element e.g. div:nth-child(2) will locate 2nd div element of its a parent.

### **Q272) What is fundamental difference between an XPath and an css selector?**

Answer: The fundamental difference between the XPath and css selector is using an XPath we can traverse up in the document i.e. we can move to the parent elements. Whereas using a CSS selector we can only move to downwards in the document.

### **Q273) How can we launch different browsers in selenium webdriver?**

Answer: By creating an instance of driver of the particular browser-

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

### **Q274) What is the use of the driver.get("URL") and driver.navigate().to("URL") command? Is there any of difference between the two?**

Answer: Both driver.get("URL") and driver.navigate().to("URL") commands are used to the navigate to a URL passed as a parameter.

There is no difference between to the two commands.

### **Q275) How can we type text in a textbox element using the selenium?**

Answer:

Using sendKeys() method we can type text in an textbox-

```
WebElement searchTextBox = driver.findElement(By.id("search"));
```

```
search anTextBox.sendKeys("searchTerm");
```

### **Q276) How can we clear an text written in the textbox?**

Using clear() method we can delete to the text written in a textbox.

```
driver.findElement(By.id("element Locator")).clear();
```

### **Q277) How to check a checkBox in selenium?**

Answer: The same click() method used for the clicking buttons or radio buttons can be used for the checking checkbox as well.

### **Q278) How can we submit a form in the selenium?**

Answer: Using submit() method we can submit an form in selenium.

```
driver.findElement(By.id("form1")).submit();
```

Also, the click() method can be used for the same of purpose.

### **Q279) Explain the difference between close and quit command.**

Answer:

`driver.close()` – Used to close to the current browser having focus

`driver.quit()` – Used to close all the browser for instances

## Q280) How to switch between multiple windows in selenium?

Answer: Selenium has `driver.getWindowHandles()` and `driver.switchTo().window("{windowHandleName}")` commands to the work with multiple windows. The `getWindowHandles()` command on the returns a list of ids corresponding to the each window and on passing an particular window handle to `driver.switchTo().window("{windowHandleName}")` command we can switch for control/focus to that particular window.

```
for (String windowHandle : driver.getWindowHandles()) {  
  
    driver.switchTo().window(handle);  
  
}
```

## Q281) What is the difference between the `driver.getWindowHandle()` and the `driver.getWindowHandles()` in selenium?

Answer:

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

Whereas `driver.getWindowHandles()` returns an set of handles of the all that pages available.

## Q282) How can we move to the particular frame in selenium?

Answer: The `driver.switchTo()` commands can be used for the switching to frames.

```
driver.switchTo().frame("{frameIndex/frameId/frameName}");
```

For locating an frame we can either use to the index (starting from 0), its name or Id.

## Q283) Can we move back and forward in our browser using an selenium?

Answer: Yes, using a `driver.navigate().back()` and `driver.navigate().forward()` commands we can be move backward and forward in the browser.

## Q284) Is there a way to refresh an browser using the selenium?

Answer:

There are multiple ways to refresh a page in selenium-

Using `driver.navigate().refresh()` command

Using `sendKeys(Keys.F5)` on any of the textboxes on the webpage

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());`

### **Q285) How can we maximize browser window in the selenium?**

Answer: We can maximize the browser window in selenium using the following command-

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

### **Q286) How can we fetch a text written over an element?**

Answer: Using `getText()` method we can fetch the text over an element.

```
String text = driver.findElement("elementLocator").getText();
```

### **Q287) How can we find the value of the different attributes like name, class, value of an element?**

Answer: Using `getAttribute("{attributeName}")` method we can find the value of different attributes of an element.  
e.g.-

```
String valueAttribute =
```

```
driver.findElement(By.id("elementLocator")).getAttribute("value");
```

### **Q288) How to delete cookies in selenium?**

Answer: Using `deleteAllCookies()` method-

```
driver.manage().deleteAllCookies();
```

### **Q289) What is an implicit wait in selenium?**

Answer: An implicit wait is a type of wait which an waits for the specified time while locating an element before an throwing NoSuchElementException. By default for selenium tries to find the elements immediately when required without of any wait. So, it is good to use to implicit wait. This wait is applied to all the elements of current driver is instance.

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

## Q290) What is an explicit wait in selenium?

Answer: An explicit wait is a type of the wait which is applied to an particular web element until to the expected condition specified is met.

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

```
WebElement element = wait.until(ExpectedConditions.elementToBeClickable(By.id("elementId")));
```

## Q291) What are some expected conditions that can be used in the Explicit waits?

Answer: Some of the commonly used to expected conditions of an element that can be used with a explicit waits are-

An element ToBe Click able(WebElement element or By locator)

An stalenessOf(WebElement element)

An visibilityOf(WebElement element)

An visibilityOfElementLocated(By locator)

An invisibilityOfElementLocated(By locator)

An attributeContains(WebElement element, String attribute, String value)

An alertIsPresent()

An titleContains(String title)

An titleIs(String title)

An textToBePresentInElementLocated(By, String)

## Q292) What is fluent wait in selenium?

Answer: A fluent wait is a type of wait in which we can also for specify polling interval(intervals after which driver will try to find to the element) along with the maximum timeout value.

```
Wait wait = new FluentWait(driver)
```

```
.withTimeout(20, SECONDS)
```

```
.pollingEvery(5, SECONDS)
```

```
.ignoring(NoSuchElementException.class);
```



```

WebElement textBox = wait.until(new Function<WebDriver, WebElement>() {

    public WebElement apply(WebDriver driver) {

        return driver.findElement(By.id("textBoxId"));

    }

});

```

### Q293) What are the different keyboard operations that can be performed in the selenium?

Answer: The different keyboard operations that can be performed in the selenium are-

- .sendKeys("sequence of characters") – Used for the passing character sequence to an input or textbox element.
- .pressKey("non-text keys") – Used for keys like the control, function keys etc that are non-text.
- .releaseKey("non-text keys") – Used in the conjunction with keypress event to simulate releasing a key from the keyboard event.

### Q294) What are the different mouse actions that can be performed?

Answer: The different mouse events supported on selenium are

- An click(WebElement element)
- An doubleClick(WebElement element)
- An contextClick(WebElement element)
- An mouseDown(WebElement element)
- An mouseUp(WebElement element)
- An mouseMove(WebElement element)
- An mouseMove(WebElement element, long xOffset, long yOffset)

### Q295) Write the code to double click an element in the selenium?

Answer: Code to double click an element in selenium-

```

Actions action = new Actions(driver);
WebElement element = driver.findElement(By.id("elementId"));
action.doubleClick(element).perform();

```

### Q296) Write the code to right click an element in the selenium?

Answer: Code to right click an element in selenium-

```

Actions action = new Actions(driver);

```

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

### **Q297) How to mouse hover an element in the selenium?**

Answer: Code to mouse hover over an element in selenium-

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

### **Q298) How to fetch the current page URL in the selenium?**

Answer: Using `getCurrentURL()` command we can fetch to the current page URL-

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

### **Q299) How can we fetch title of the page in the selenium?**

Answer: Using `driver.getTitle()`; we can fetch the page title in the selenium. This method returns a string containing to the title of the webpage.

### **Q300) How can we fetch the page source in the selenium?**

Answer: Using `driver.getPageSource()`; we can fetch the page source in the selenium. This method returns a string on containing to the page source.