

Q #1) What is Automation Testing?

Automation testing or Test Automation is a process of automating the manual process to test the application/system under test. Automation testing involves the use of a separate testing tool which lets you create test scripts which can be executed repeatedly and doesn't require any manual intervention.

Q #2) What are the benefits of Automation Testing?

Benefits of Automation testing are:

1. Supports execution of repeated test cases
2. Aids in testing a large test matrix
3. Enables parallel execution
4. Encourages unattended execution
5. Improves accuracy thereby reducing human-generated errors
6. Saves time and money

Q #3) Why should Selenium be selected as a test tool?

Selenium

1. is a free and open source
2. have a large user base and helping communities
3. have cross Browser compatibility (Firefox, Chrome, Internet Explorer, Safari etc.)
4. have great platform compatibility (Windows, Mac OS, Linux etc.)
5. supports multiple programming languages (Java, C#, Ruby, Python, Pearl etc.)
6. has fresh and regular repository developments
7. supports distributed testing

Q #5) What are the testing types that can be supported by Selenium?

Selenium supports the following types of testing:

1. Functional Testing
2. Regression Testing

Q #6) What are the limitations of Selenium?

Following are the limitations of Selenium:

- Selenium supports testing of only web-based applications
- Mobile applications cannot be tested using Selenium
- Captcha and Barcode readers cannot be tested using Selenium
- Reports can only be generated using third-party tools like TestNG or JUnit.
- As Selenium is a free tool, thus there is no ready vendor support through the user can find numerous helping communities.
- The user is expected to possess prior programming language knowledge.

Q #7) What is the difference between Selenium IDE, Selenium RC, and WebDriver?

Feature	Selenium IDE	Selenium RC	WebDriver
Browser Compatibility	Selenium IDE comes as a Firefox plugin, thus it supports only Firefox	Selenium RC supports a varied range of versions of Mozilla Firefox, Google Chrome, Internet Explorer and Opera.	WebDriver supports a varied range of versions of Mozilla Firefox, Google Chrome, Internet Explorer and Opera.

Feature	Selenium IDE	Selenium RC	WebDriver
			Also supports HtmlUnitDriver which is a GUI less or headless browser.
Record and Playback	Selenium IDE supports record and playback feature	Selenium RC doesn't supports record and playback feature.	WebDriver doesn't support record and playback feature
Server Requirement	Selenium IDE doesn't require any server to be started before executing the test scripts	Selenium RC requires server to be started before executing the test scripts.	WebDriver doesn't require any server to be started before executing the test scripts
Architecture	Selenium IDE is a Javascript based framework	Selenium RC is a JavaScript based Framework.	WebDriver uses the browser's native compatibility to automation
Object Oriented	Selenium IDE is not an object oriented tool	Selenium RC is semi object oriented tool.	WebDriver is a purely object oriented tool
Dynamic Finders (for locating web elements on a webpage)	Selenium IDE doesn't support dynamic finders	Selenium RC doesn't support dynamic finders.	WebDriver supports dynamic finders
Handling Alerts, Navigations, Dropdowns	Selenium IDE doesn't explicitly provides aids to handle alerts, navigations, dropdowns	Selenium RC doesn't explicitly provides aids to handle alerts, navigations, dropdowns.	WebDriver offers a wide range of utilities and classes that helps in handling alerts, navigations, and dropdowns efficiently and effectively.
WAP (iPhone/Android) Testing	Selenium IDE doesn't support testing of iPhone/Android applications	Selenium RC doesn't support testing of iPhone/Android applications.	WebDriver is designed in a way to efficiently support testing of iPhone/Android applications. The tool comes with a large range of drivers for WAP based testing. For example, AndroidDriver, iPhoneDriver
Listener Support	Selenium IDE doesn't support listeners	Selenium RC doesn't support listeners.	WebDriver supports the implementation of Listeners
Speed	Selenium IDE is fast as it is plugged in with the web-browser that launches the test. Thus,	Selenium RC is slower than WebDriver as it doesn't communicates directly with the browser; rather it sends	WebDriver communicates directly with the web browsers. Thus making it much faster.

Feature	Selenium IDE	Selenium RC	WebDriver
	the IDE and browser communicates directly	selenese commands over to Selenium Core which in turn communicates with the browser.	

Q #8) When should I use Selenium IDE?

Selenium IDE is the simplest and easiest of all the tools within the Selenium Package. Its record and playback feature makes it exceptionally easy to learn with minimal acquaintances to any programming language. Selenium IDE is an ideal tool for a naïve user.

Q #9) What is Selenese?

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

Q #10) What are the different types of locators in Selenium?

The locator can be termed as an address that identifies a web element uniquely within the webpage. Thus, to identify web elements accurately and precisely we have different types of locators in Selenium:

- ID
- ClassName
- Name
- TagName
- LinkText
- PartialLinkText
- Xpath
- CSS Selector
- DOM

Q #11) What is the difference between assert and verify commands?

Assert: Assert command checks whether the given condition is true or false. Let's say we assert whether the given element is present on the web page or not. If the condition is true then the program control will execute the next test step but if the condition is false, the execution would stop and no further test would be executed.

Verify: Verify command also checks whether the given condition is true or false. Irrespective of the condition being true or false, the program execution doesn't halt i.e. any failure during verification would not stop the execution and all the test steps would be executed.

Q #12) What is an XPath?

XPath is used to locate a web element based on its XML path. XML stands for Extensible Markup Language and is used to store, organize and transport arbitrary data. It stores data in a key-value pair which is very much similar to HTML tags. Both being markup languages and since they fall under the same umbrella, XPath can be used to locate HTML elements.

The fundamental behind locating elements using XPath is the traversing between various elements across the entire page and thus enabling a user to find an element with the reference of another element.

Q #13) What is the difference between "/" and "/" in XPath?

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

Double Slash "/" – Double slash is used to create XPath with relative path i.e. the xpath would be created to start selection from anywhere within the document.

Q #14) What is Same origin policy and how it can be handled?

The problem of same origin policy disallows to access the DOM of a document from an origin that is different from the origin we are trying to access the document.

Origin is a sequential combination of scheme, host, and port of the URL. For example, for a URL `https://www.softwaretestinghelp.com/resources/`, the origin is a combination of `http`, `softwaretestinghelp.com`, `80` correspondingly.

Thus the Selenium Core (JavaScript Program) cannot access the elements from an origin that is different from where it was launched. For Example, if I have launched the JavaScript Program from `https://www.softwaretestinghelp.com`, then I would be able to access the pages within the same domain such as `https://www.softwaretestinghelp.com/resources` or

`https://www.softwaretestinghelp.com/istqb-free-updates/`. The other domains like `google.com`, `seleniumhq.org` would no more be accessible.

So, In order to handle the same origin policy, Selenium Remote Control was introduced.

Q #15) When should I use Selenium Grid?

Selenium Grid can be used to execute same or different test scripts on multiple platforms and browsers concurrently so as to achieve distributed test execution, testing under different environments and saving execution time remarkably.

Q #16) What do we mean by Selenium 1 and Selenium 2?

Selenium RC and WebDriver, in a combination, are popularly known as Selenium 2. Selenium RC alone is also referred to as Selenium 1.

Q #17) Which is the latest Selenium tool?

WebDriver

Q #18) How do I launch the browser using WebDriver?

The following syntax can be used to launch Browser:

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

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

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

Q #19) What are the different types of Drivers available in WebDriver?

The different drivers available in WebDriver are:

- FirefoxDriver
- InternetExplorerDriver
- ChromeDriver
- SafariDriver
- OperaDriver
- AndroidDriver
- IPHoneDriver
- HtmlUnitDriver

Q #20) What are the different types of waits available in WebDriver?

There are two types of waits available in WebDriver:

1. Implicit Wait
2. Explicit Wait

www.resolve6training.ca

Implicit Wait: Implicit waits are used to provide a default waiting time (say 30 seconds) between each consecutive test step/command across the entire test script. Thus, the subsequent test step would only execute when the 30 seconds have elapsed after executing the previous test step/command.

Explicit Wait: Explicit waits are used to halt the execution till the time a particular condition is met or the maximum time has elapsed. Unlike Implicit waits, explicit waits are applied for a particular instance only.

Q #21) How to type in a textbox using Selenium?

The user can 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");
```

Q #22) How can you find if an element is displayed on the screen?

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

1. `isDisplayed()`
2. `isSelected()`
3. `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();
```

Q #23) How can we get a text of a web element?

Get command is used to retrieve the inner text of the specified web element. The command doesn't require any parameter but returns a string value. It is also one of the extensively used commands for verification of messages, labels, errors etc displayed on the web pages.

Syntax:

```
String Text = driver.findElement(By.id("Text")).getText();
```

Q #24) How to select value in a dropdown?

The value in the dropdown can be selected using WebDriver's Select class.

Syntax:

selectByValue:

```
Select selectByValue = new Select(driver.findElement(By.id("SelectID_One")));  
selectByValue.selectByValue("greenvalue");
```

selectByVisibleText:

```
Select selectByVisibleText = new Select (driver.findElement(By.id("SelectID_Two")));  
selectByVisibleText.selectByVisibleText("Lime");
```

selectByIndex:

```
Select selectByIndex = new Select(driver.findElement(By.id("SelectID_Three")));
selectByIndex.selectByIndex(2);
```

Q #25) What are the different types of navigation commands?

Following are the navigation commands:

navigate().back() – The above command requires no parameters and takes back the user to the previous webpage in the web browser's history.

Sample code:

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

navigate().forward() – This command lets the user to navigate to the next web page with reference to the browser's history.

Sample code:

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

navigate().refresh() – This command lets the user to refresh the current web page there by reloading all the web elements.

Sample code:

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

navigate().to() – This command lets the user to launch a new web browser window and navigate to the specified URL.

Sample code:

```
driver.navigate().to("https://google.com");
```

Q #26) How to click on a hyper link using linkText?

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

The command finds the element using link text and then click on that element and thus the user would be re-directed to the corresponding page.

The above-mentioned link can also be accessed by using the following command.

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

The above command finds the element based on the substring of the link provided in the parenthesis and thus partialLinkText() finds the web element with the specified substring and then clicks on it.

Q #27) How to handle frame in WebDriver?

An inline frame acronym as iframe is used to insert another document within the current HTML document or simply a web page into a web page by enabling nesting.

Select iframe by id

```
driver.switchTo().frame("ID of the frame");
```

Locating iframe using tagName

```
driver.switchTo().frame(driver.findElements(By.tagName("iframe")).get(0));
```

Locating iframe using index**frame(index)**

```
driver.switchTo().frame(0);
```

frame(Name of Frame)

```
driver.switchTo().frame("name of the frame");
```

frame(WebElement element)**Select Parent Window**

```
driver.switchTo().defaultContent();
```

Q #28) When do we use findElement() and findElements()?

www.resolve6training.ca

findElement(): findElement() is used to find the first element in the current web page matching to the specified locator value. Take a note that only first matching element would be fetched.

Syntax:

```
WebElement element = driver.findElement(By.xpath("//div[@id='example']/ul/li"));
```

findElements(): findElements() is used to find all the elements in the current web page matching to the specified locator value. Take a note that all the matching elements would be fetched and stored in the list of WebElements.

Syntax:

```
List <WebElement> elementList = driver.findElements(By.xpath("//div[@id='example']/ul/li"));
```

Q #29) How to find more than one web element in the list?

At times, we may come across elements of the same type like multiple hyperlinks, images etc arranged in an ordered or unordered list. Thus, it makes absolute sense to deal with such elements by a single piece of code and this can be done using WebElement List.

Sample Code

```
// Storing the list
List <WebElement> elementList = driver.findElements(By.xpath("//div[@id='example']/ul/li"));
// Fetching the size of the list
int listSize = elementList.size();
for (int i=0; i<listSize; i++)
{
// Clicking on each service provider link
serviceProviderLinks.get(i).click();
// Navigating back to the previous page that stores link to service providers
driver.navigate().back();
}
}
```

Q #30) What is the difference between driver.close() and driver.quit command?

close(): WebDriver's close() method closes the web browser window that the user is currently working on or we can also say the window that is being currently accessed by the WebDriver. The command neither requires any parameter nor does it return any value.

quit(): Unlike close() method, quit() method closes down all the windows that the program has opened. Same as close() method, the command neither requires any parameter nor does it return any value.

Q #31) Can Selenium handle windows based pop up?

Selenium is an automation testing tool which supports only web application testing. Therefore, windows pop up cannot be handled using Selenium.

Q #32) How can we handle web-based pop-up?

WebDriver offers the users a very efficient way to handle these pop-ups using Alert interface.

There are the four methods that we would be using along with the Alert interface.

- void dismiss() – The dismiss() method clicks on the “Cancel” button as soon as the pop-up window appears.
- void accept() – The accept() method clicks on the “Ok” button as soon as the pop-up window appears.
- String getText() – The getText() method returns the text displayed on the alert box.
- void sendKeys(String stringToSend) – The sendKeys() method enters the specified string pattern into the alert box.

Syntax:

```
// accepting javascript alert
```

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

Q #33) How can we handle windows based pop up?

Selenium is an automation testing tool which supports only web application testing, that means, it doesn't support testing of windows based applications. However Selenium alone can't help the situation but along with some third-party intervention, this problem can be overcome. There are several third-party tools available for handling window based pop-ups along with the selenium like AutoIT, Robot class etc.

Q #34) How to assert the title of the web page?

```
//verify the title of the web page
assertTrue("The title of the window is incorrect.",driver.getTitle().equals("Title of the page"));
```

Q #35) How to mouse hover on a web element using WebDriver?

WebDriver offers a wide range of interaction utilities that the user can exploit to automate mouse and keyboard events. Action Interface is one such utility which simulates the single user interactions.

Thus, In the following scenario, we have used Action Interface to mouse hover on a drop down which then opens a list of options.

Sample Code:

```
// Instantiating Action Interface
Actions actions=new Actions(driver);
// hovering on the dropdown
actions.moveToElement(driver.findElement(By.id("id of the dropdown"))).perform();
// Clicking on one of the items in the list options
WebElement subLinkOption=driver.findElement(By.id("id of the sub link"));
subLinkOption.click();
```

Q #36) How to retrieve CSS properties of an element?

The values of the css properties can be retrieved using a get() method:

Syntax:

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

Q #37) How to capture screenshot in WebDriver?

```
import org.junit.After;
import org.junit.Before;
import org.junit.Test;
import java.io.File;
import java.io.IOException;
import org.apache.commons.io.FileUtils;
import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
```

```
public class CaptureScreenshot {
    WebDriver driver;
    @Before
    public void setUp() throws Exception {
www.resolve6training.ca
```

```

driver = new FirefoxDriver();
driver.get("https://google.com");
}
@After
public void tearDown() throws Exception {
driver.quit();
}

@Test
public void test() throws IOException {
// Code to capture the screenshot
File scrFile = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
// Code to copy the screenshot in the desired location
FileUtils.copyFile(scrFile, new File("C:\\CaptureScreenshot\\google.jpg"))
}
}

```

Q #38) What is Junit?

Junit is a unit testing framework introduced by Apache. Junit is based on Java.

Q #39) What are Junit annotations?

Following are the JUnit Annotations:

- **@Test:** Annotation lets the system know that the method annotated as @Test is a test method. There can be multiple test methods in a single test script.
- **@Before:** Method annotated as @Before lets the system know that this method shall be executed every time before each of the test methods.
- **@After:** Method annotated as @After lets the system know that this method shall be executed every time after each of the test method.
- **@BeforeClass:** Method annotated as @BeforeClass lets the system know that this method shall be executed once before any of the test methods.
- **@AfterClass:** Method annotated as @AfterClass lets the system know that this method shall be executed once after any of the test methods.
- **@Ignore:** Method annotated as @Ignore lets the system know that this method shall not be executed.

Q #40) What is TestNG and how is it better than Junit?

TestNG is an advanced framework designed in a way to leverage the benefits by both the developers and testers. With the commencement of the frameworks, JUnit gained enormous popularity across the Java applications, Java developers and Java testers with remarkably increasing the code quality. Despite being easy to use and straightforward, JUnit has its own limitations which give rise to the need of bringing TestNG into the picture. TestNG is an open source framework which is distributed under the Apache Software License and is readily available for download.

TestNG with WebDriver provides an efficient and effective test result format that can, in turn, be shared with the stakeholders to have a glimpse on the product's/application's health thereby eliminating the drawback of WebDriver's incapability to generate test reports. TestNG has an inbuilt exception handling mechanism which lets the program to run without terminating unexpectedly.

There are various advantages that make TestNG superior to JUnit. Some of them are:

- Added advance and easy annotations
- Execution patterns can set

www.resolve6training.ca

- Concurrent execution of test scripts
- Test case dependencies can be set

Q #41) How to set test case priority in TestNG?

Setting Priority in TestNG

Code Snippet

```
package TestNG;
import org.testng.annotations.*;
public class SettingPriority {
    @Test(priority=0)
    public void method1() {
    }
    @Test(priority=1)
    public void method2() {
    }
    @Test(priority=2)
    public void method3() {
    }
}
```

Test Execution Sequence:

1. Method1
2. Method2
3. Method3

Q #42) What is a framework?

The framework is a constructive blend of various guidelines, coding standards, concepts, processes, practices, project hierarchies, modularity, reporting mechanism, test data injections etc. to pillar automation testing.

Q #43) What are the advantages of the Automation framework?

The advantage of Test Automation framework

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

Q #44) What are the different types of frameworks?

Below are the different types of frameworks:

1. **Module Based Testing Framework:** The framework divides the entire “Application Under Test” into the number of logical and isolated modules. For each module, we create a separate and independent test script. Thus, when these test scripts have taken together builds a larger test script representing more than one module.
2. **Library Architecture Testing Framework:** The basic fundamental behind the framework is to determine the common steps and group them into functions under a library and call those functions in the test scripts whenever required.
3. **Data Driven Testing Framework:** Data Driven Testing Framework helps the user segregate the test script logic and the test data from each other. It lets the user store the test data into an external database. The data is conventionally stored in “Key-Value” pairs. Thus, the key can be used to access and populate the data within the test scripts.

4. **Keyword Driven Testing Framework:** The Keyword Driven testing framework is an extension to Data-driven Testing Framework in a sense that it not only segregates the test data from the scripts, it also keeps the certain set of code belonging to the test script into an external data file.
5. **Hybrid Testing Framework:** Hybrid Testing Framework is a combination of more than one above mentioned frameworks. The best thing about such a setup is that it leverages the benefits of all kinds of associated frameworks.
6. **Behavior Driven Development Framework:** Behavior Driven Development framework allows automation of functional validations in an easily readable and understandable format to Business Analysts, Developers, Testers, etc.

Q #45) How can I read test data from excels?

Test data can efficiently be read from excel using JXL or POI API. [See detailed tutorial here.](#)

Q #46) What is the difference between POI and jxl jar?

#	JXL jar	POI jar
1	JXL supports “.xls” format i.e. binary based format. JXL doesn't support Excel 2007 and “.xlsx” format i.e. XML based format	POI jar supports all of these formats
2	JXL API was last updated in the year 2009	POI is regularly updated and released
3	The JXL documentation is not as comprehensive as that of POI	POI has a well prepared and highly comprehensive documentation
4	JXL API doesn't support rich text formatting	POI API supports rich text formatting
5	JXL API is faster than POI API	POI API is slower than JXL API

Q #47) What is the difference between Selenium and QTP?

Feature	Selenium	Quick Test Professional (QTP)
Browser Compatibility	Selenium supports almost all the popular browsers like Firefox, Chrome, Safari, Internet Explorer, Opera etc	QTP supports Internet Explorer, Firefox and Chrome. QTP only supports Windows Operating System
Distribution	Selenium is distributed as an open source tool and is freely available	QTP is distributed as a licensed tool and is commercialized
Application under Test	Selenium supports testing of only web based applications	QTP supports testing of both the web based application and windows based application
Object Repository	Object Repository needs to be created as a separate entity	QTP automatically creates and maintains Object Repository

Feature	Selenium	Quick Test Professional (QTP)
Language Support	Selenium supports multiple programming languages like Java, C#, Ruby, Python, Perl etc	QTP supports only VB Script
Vendor Support	As Selenium is a free tool, user would not get the vendor's support in troubleshooting issues	Users can easily get the vendor's support in case of any issue

Q #48) Can WebDriver test Mobile applications?

WebDriver cannot test Mobile applications. WebDriver is a web-based testing tool, therefore applications on the mobile browsers can be tested.

Q #49) Can captcha be automated?

No, captcha and barcode reader cannot be automated.

Q #50) What is Object Repository? How can we create an Object Repository in Selenium?

Object Repository is a term used to refer to the collection of web elements belonging to Application Under Test (AUT) along with their locator values. Thus, whenever the element is required within the script, the locator value can be populated from the Object Repository. Object Repository is used to store locators in a centralized location instead of hardcoding them within the scripts.

In Selenium, objects can be stored in an excel sheet which can be populated inside the script whenever required.

Q #51)How will you find an element using Selenium?

In Selenium every object or control in a web page is referred as an elements, there are different ways to find an element in a web page they are

- ID
- Name
- Tag
- Attribute
- CSS
- Linktext
- PartialLink Text
- Xpath etc

Q #52) List out the test types that are supported by Selenium?

For web based application testing selenium can be used

The test types can be supported are

- a) Functional, Learn More about [Functional Testing](#).
- b) Regression

For post release validation with continuous integration automation tool could be used

- a) Jenkins
- b) Hudson
- c) Quick Build
- d) CruiseCont

Q #53) Explain what is assertion in Selenium and what are the types of assertion?

www.resolve6training.ca

Assertion is used as a verification point. It verifies that the state of the application conforms to what is expected. The types of assertion are “assert” , “verify” and “waitFor”.

Q #54) Mention what is the use of X-path?

X-Path is used to find the WebElement in web pages. It is also useful in identifying the dynamic elements.

Refer Complete Guide on [XPath](#)

Q #55) Explain the difference between single and double slash in X-path?

Single slash ‘/’

- Single slash (/) start selection from the document node
- It allows you to create ‘absolute’ path expressions

Double Slash ‘//’

- Double slash (//) start selection matching anywhere in the document
- It enables to create ‘relative’ path expressions

Q #56)List out the technical challenges with Selenium?

Technical challenges with Selenium are

- Selenium supports only web based applications
- It does not support the Bitmap comparison
- For any reporting related capabilities have to depend on third party tools
- No vendor support for tool compared to commercial tools like HP UFT
- As there is no object repository concept in Selenium, maintainability of objects becomes difficult

Q #57) What is the difference between type keys and type commands ?

TypeKeys() will trigger JavaScript event in most of the cases whereas .type() won't. Type key populates the value attribute using JavaScript whereas .typekeys() emulates like actual user typing

Q #58) What is the difference between verify and assert commands?

Assert: Assert allows to check whether an element is on the page or not. The test will stop on the step failed, if the asserted element is not available. In other words, the test will terminated at the point where check fails.

Verify: Verify command will check whether the element is on the page, if it is not then the test will carry on executing. In verification, all the commands are going to run guaranteed even if any of test fails.

Q #59) What is JUnit Annotations and what are different types of annotations which are useful ?

In JAVA a special form of syntactic meta-data can be added to Java source code, this is know as Annotations. Variables, parameters, packages, methods and classes are annotated some of the JUnit annotations which can be useful are

- Test
- Before
- After
- Ignore
- BeforeClass
- AfterClass
- RunWith

Q #60) While using click command can you use screen coordinate?

www.resolve6training.ca

To click on specific part of element, you would need to use clickAT command. ClickAt command accepts element locator and x, y co-ordinates as arguments- clickAt (locator, cordString)

Q #61) What are the advantages of Selenium?

- It supports C#, PHP, Java, Perl, Python
- It supports different OS like Windows, Linux and Mac OS
- It has got powerful methods to locate elements (Xpath, DOM , CSS)
- It has highly developer community supported by Google

Q #62) Why testers should opt for Selenium and not QTP?

Selenium is more popular than QTP as

- Selenium is an open source whereas QTP is a commercial tool
- Selenium is used specially for testing web based applications while QTP can be used for testing client server application also
- Selenium supports Firefox, IE, Opera, Safari on operating systems like Windows, Mac, Linux etc. however QTP is limited to Internet Explorer on Windows.
- Selenium supports many programming languages like Ruby, Perl, Python whereas QTP supports only VB script

Q #63) What are the four parameter you have to pass in Selenium?

Four parameters that you have to pass in Selenium are

- Host
- Port Number
- Browser
- URL

Q #64) What is the difference between setSpeed() and sleep() methods?

Both will delay the speed of execution.

Thread.sleep () : It will stop the current (java) thread for the specified period of time. Its done only once

- It takes a single argument in integer format

Ex: thread.sleep(2000)- It will wait for 2 seconds

- It waits only once at the command given at sleep

SetSpeed () : For specific amount of time it will stop the execution for every selenium command.

- It takes a single argument in integer format

Ex: selenium.setSpeed("2000")- It will wait for 2 seconds

- Runs each command after setSpeed delay by the number of milliseconds mentioned in set Speed

This command is useful for demonstration purpose or if you are using a slow web application

Q #65) What is same origin policy? How you can avoid same origin policy?

The "**Same Origin Policy**" is introduced for security reason, and it ensures that content of your site will never be accessible by a script from another site. As per the policy, any code loaded within the browser can only operate within that website's domain.

To avoid "Same Origin Policy" proxy injection method is used, in proxy injection mode the Selenium Server acts as a client configured **HTTP proxy** , which sits between the browser and application under test and then masks the AUT under a fictional URL

Q #67) How you can use "submit" a form using Selenium ?

You can use "submit" method on element to submit form-

www.resolve6training.ca

element.submit () ;

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

Q #68) What are the features of TestNG and list some of the functionality in TestNG which makes it more effective?

TestNG is a testing framework based on JUnit and NUnit to simplify a broad range of testing needs, from Unit Testing to Integration Testing. And the functionality which makes it efficient testing framework are

- Support for annotations
- Support for data-driven testing
- Flexible test configuration
- Ability to re-execute failed test cases

Q #69) Mention what is the difference between Implicit wait and Explicit wait?

Implicit Wait: Sets a timeout for all successive Web Element searches. For the specified amount of time it will try looking for element again and again before throwing a NoSuchElementException. It waits for elements to show up.

Explicit Wait : It is a one-timer, used for a particular search.

Q #70) Which attribute you should consider throughout the script in frame for “if no frame Id as well as no frame name”?

You can use.....driver.findElements(By.xpath("//iframe"))....

This will return list of frames.

You will need to switch to each and every frame and search for locator which we want.

Then break the loop

Q #71) Explain what is the difference between find elements () and find element () ?

find element ():

It finds the first element within the current page using the given “locating mechanism”. It returns a single WebElement

findElements () : Using the given “locating mechanism” find all the elements within the current page. It returns a list of web elements.

Q #72) Explain what are the JUnits annotation linked with Selenium?

The JUnits annotation linked with Selenium are

- @Before public void method() – It will perform the method () before each test, this method can prepare the test
- @Test public void method() – Annotations @Test identifies that this method is a test method environment
- @After public void method()- To execute a method before this annotation is used, test method must start with test@Before

Q #73) Explain what is Datadriven framework and Keyword driven?

Datadriven framework: In this framework, the test data is separated and kept outside the Test Scripts, while Test Case logic resides in Test Scripts. Test data is read from the external files (Excel Files) and are loaded into the variables inside the Test Script. Variables are used for both for input values and for verification values.

Keyworddriven framework: The keyword driven frameworks requires the development of data tables and keywords, independent of the test automation. In a keyword driven test, the functionality of the application under test is documented in a table as well as step by step instructions for each test.

Q #74) Explain how you can login into any site if it's showing any authentication popup for password and username?

Pass the username and password with url

- Syntax-`http://username:password@url`
- ex- `http://creyate:tom@www.gmail.com`

Q #75) Explain how to assert text of webpage using selenium 2.0 ?

```
WebElement el = driver.findElement(By.id("ElementID"))
```

```
//get test from element and stored in text variable
```

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

```
//assert text from expected
```

```
Assert.assertEquals("Element Text", text);
```

Q #76) Explain what is the difference between Borland Silk and Selenium?

Silk Test Tool	Selenium Test Tool
<ul style="list-style-type: none">• Borland Silk test is not a free testing tool	<ul style="list-style-type: none">• Selenium is completely free test automation tool
<ul style="list-style-type: none">• Silk test supports only Internet Explorer and Firefox	<ul style="list-style-type: none">• Selenium supports many browsers like Internet Explorer, Firefox, Safari, Opera and so on
<ul style="list-style-type: none">• Silk test uses test scripting language	<ul style="list-style-type: none">• Selenium suite has the flexibility to use many languages like Java, Ruby, Perl and so on
<ul style="list-style-type: none">• Silk test can be used for client server applications	<ul style="list-style-type: none">• Selenium can be used for only web application

Q #77) What is Object Repository ?

An object repository is an essential entity in any UI automations which allows a tester to store all object that will be used in the scripts in one or more centralized locations rather than scattered all over the test scripts.

Q #80) While injecting capabilities in webdriver to perform tests on a browser which is not supported by a webdriver what is the limitation that one can come across?

Major limitation of injecting capabilities is that "findElement" command may not work as expected.

Q #81) Explain how you can find broken images in a page using Selenium Web driver ?

To find the broken images in a page using Selenium web driver is

- Get XPath and get all the links in the page using tag name
- In the page click on each and every link
- Look for 404/500 in the target page title

Q #82) Explain how you can handle colors in web driver?

To handle colors in web driver you can use

Use `getCssValue(arg0)` function to get the colors by sending 'color' string as an argument

Q #83) Using web driver how you can store a value which is text box?

You can use following command to store a value which is text box using web driver

```
driver.findElement(By.id("your Textbox")).sendKeys("your keyword");
```

Q #84) Explain how you can switch between frames?

To switch between frames webdrivers [`driver.switchTo().frame()`] method takes one of the three possible arguments

- A number: It selects the number by its (zero-based) index
- A number or ID: Select a frame by its name or ID
- Previously found WebElement: Using its previously located WebElement select a frame

Q #85) Mention 5 different exceptions you had in Selenium web driver?

The 5 different exceptions you had in Selenium web drivers are

- `WebDriverException`
- `NoAlertPresentException`
- `NoSuchWindowException`
- `NoSuchElementException`
- `TimeoutException`

Q #86) Explain using Webdriver how you can perform double click ?

You can perform double click by using

- **Syntax- `Actions act = new Actions (driver);`**
- **`act.doubleClick(webelement);`**

Q #87) How will you use Selenium to upload a file ?

You can use "type" command to type in a file input box of upload file. Then, you have to use "Robot" class in JAVA to make file upload work.

Q #88) Which web driver implementation is fastest?

HTMLUnit Driver implementation is fastest, HTMLUnitDriver does not execute tests on browser but plain http request, which is far quick than launching a browser and executing tests

Q #89) Explain how you can handle frames using Selenium 2.0 ?

To bring control on HTML frame you can use "SwitchTo" frame method-

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

To specify a frame you can use index number

```
driver.switchTo().frame("parentFrame.4.frameName");
```

This would bring control on frame named- "frameName" of the 4th sub frame names "parentFrame"

Q #90) What is the difference between getWindowHandles() and getWindowHandle() ?

getWindowHandles(): It is used to get the address of all the open browser and its return type is Set<String>

getWindowHandle(): It is used to get the address of the current browser where the control is and return type is string

Q #91) Explain how you can switch back from a frame?

To switch back from a frame use method defaultContent()

Syntax-driver.switchTo().defaultContent();

Q #92) List out different types of locators?

Different types of locators are

- By.id()
- By.name()
- By.tagName()
- By.className()
- By.linkText()
- By.partialLinkText()
- By.xpath
- By.cssSelector()

Q #93) What is the command that is used in order to display the values of a variable into the output console or log?

- In order to display a constant string, command can be used is echo <constant string>
- If order to display the value of a variable you can use command like echo \${variable name}>>

Above is using PHP. If you are using Java, replace echo with System.out.println

Q #94) Explain how you can use recovery scenario with Selenium?

Recovery scenarios depends upon the programming language you use. If you are using Java then you can use exception handling to overcome same. By using "Try Catch Block" within your Selenium WebDriver Java tests

Q #95) Explain how to iterate through options in test script?

To iterate through options in test script you can loop features of the programming language, for example to type different test data in a text box you can use “for” loop in Java

```
// test data collection in an array
```

```
String[ ] testData = { “test1” , “test2” , “test3” } ;
```

```
// iterate through each test data
```

```
For (string s: test data) { selenium.type ( “elementLocator”, testData) ; }
```

Q #96) How can you prepare customized html report using TestNG in hybrid framework ?

There are three ways

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

Q #97) From your test script how you can create html test report?

To create html test report there are three ways

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

Q #98) Explain how you can insert a break point in Selenium IDE ?

In Selenium IDE to insert a break point

- Select “Toggle break point” by right click on the command in Selenium IDE
- Press “B” on the keyboard and select the command in Selenium IDE
- Multiple break points can be set in Selenium IDE

Q #99) Explain in Selenium IDE how can you debug the tests?

- Insert a break point from the location from where you want to execute test step by step
- Run the test case
- At the given break point execution will be paused
- To continue with the next statement click on the blue button
- Click on the “Run” button to continue executing all the commands at a time

Q #100) What is Selenese and what are the types of Selenese ?

Selenese is a selenium set of command which are used for running the test

There are three types of Selenese

- Actions: It is used for performing the operations and interactions with the target elements
- Assertions: It is used as a check points
- Accessors: It is used for storing the values in a variable

Q #101) Explain what are the limitations of Selenium IDE?

The limitations of Selenium IDE

- Exceptional handling is not present
- Selenium IDE uses only HTML languages
- External databases reading is not possible with IDE
- Reading from the external files like .txt, .xls is not possible
- Conditional or branching statements execution like if,else, select statements is not possible

Q #102) What are the two modes of views in Selenium IDE ?

Either Selenium IDE can be opened as a pop up window or in side bar

Q #103) In selenium IDE what are the element locators that can be used to locate elements on web page?

In selenium there are mainly 4 locators that are used

- X-path locators
- Css locators
- Html id
- Html name

Q #104) In Selenium IDE how you can generate random numbers and dates for test data ?

In Selenium IDE you can generate random numbers by using Java Script

type

css=input#s

javascript{Math.random()}

And for

type

css=input#s

javascript{new Date()}

www.resolve6training.ca

Q #105) How you can convert any Selenium IDE tests from Selenese to another language?

You can use the format option of Selenium IDE to convert tests into another programming language

Q #106) Using Selenium IDE is it possible to get data from a particular html table cell ?

You can use the “storeTable” command

Example store text from cell 0,2 from an html table

storeTable

Css=#table 0.2

textFromCell

Q #107) Explain what can cause a Selenium IDE test to fail?

- When a locator has changed and Selenium IDE cannot locate the element
- When element Selenium IDE waiting to access did not appear on the web page and the operation timed out
- When element Selenium IDE was trying to access was not created

Q #108) Explain how you can debug the tests in Selenium IDE ?

- Insert a break point from the location where you want to execute step by step
- Run the test case
- At the given break point execution will be paused
- To continues with the next step click on the Blue button
- To run commands at a time click on run button

Q #109) From Selenium IDE how you can execute a single line?

From Selenium IDE single line command can be executed in two ways

- Select “Execute this command” by right clicking on the command in Selenium IDE
- Press “X” key on the keyboard after selecting the command in Selenium IDE

Q #110) In which format does source view shows your script in Selenium IDE ?

In Selenium IDE source view shows your script in XML format

Q #111) Explain how you can insert a start point in Selenium IDE?

In two ways selenium IDE can be set

- Press “S” key on the keyboard and select the command in Selenium IDE

- In Selenium IDE right click on the command and then select "Set / Clear Start Point"

Q #112) What if you have written your own element locator and how would you test it?

To test the locator one can use "Find Button" of Selenium IDE, as you click on it, you would see on screen an element being highlighted provided your element locator is right or else an error message will be displayed

Q #113) What is regular expressions? How you can use regular expressions in Selenium ?

A regular expression is a special text string used for describing a search pattern. In Selenium IDE regular expression can be used with the keyword- **regexp**: as a prefix to the value and patterns needs to be included for the expected values.

Q #114) What are core extension ?

If you want to "extend" the default functionality provided by Selenium Function Library , you can create a Core Extension. They are also called "User Extension". You can even download ready-made Core Extension created by other Selenium enthusiasts.

Q #115) How will you handle working with multiple windows in Selenium ?

We can use the command **selectWindow** to switch between windows. This command uses the title of Windows to identify which window to switch to.

Q #116) How will you verify the specific position of an web element

You can use `verifyElementPositionLeft` & `verifyElementPositionTop`. It does a pixel comparison of the position of the element from the Left and Top of page respectively

Q #117) How can you retrieve the message in an alert box ?

You can use the `storeAlert` command which will fetch the message of the alert pop up and store it in a variable.