



H2K Infosys is **E-Verify** business based in Atlanta, Georgia – United States
Providing World Class Live Online & Onsite I.T. training on Real Time Projects
Job Placement Assistance | Tech Support

www.H2KINFOSYS.com

USA - +1-(770)-777-1269

Training@H2KInfosys.com / H2KInfosys@Gmail.com

www.h2kinfosys.com | training@h2kinfosys.com

US - (770) 777 1269

DISCLAIMER

H2K Infosys, LLC
Dream, Strive & Achieve Victory

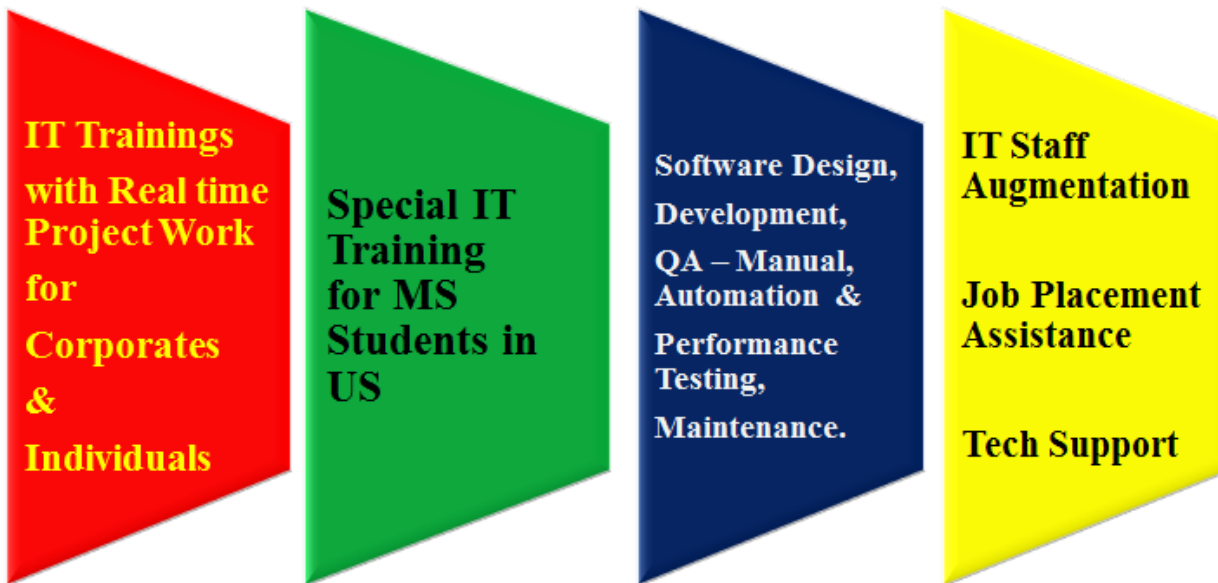


H2K Infosys, LLC (hereinafter "H2K") acknowledges the proprietary rights of the trademarks and product names of other companies mentioned in any of the training material including but not limited to the handouts, written material, videos, power point presentations, etc. All such training materials are provided to H2K students for learning purposes only. H2K students shall not use such materials for their private gain nor can they sell any such materials to a third party. Some of the examples provided in any such training materials may not be owned by H2K and as such H2K does not claim any proprietary rights for the same. H2K does not guarantee nor is it responsible for such products and projects. H2K acknowledges that any such information or product that has been lawfully received from any third party source is free from restriction and without any breach or violation of law whatsoever.

www.h2kinfosys.com | training@h2kinfosys.com

US - (770) 777 1269 | UK - (020) 3371 7615

H2K INFOSYS PROVIDES WORLD CLASS SERVICES IN



Q 1) 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 2) 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 Bar code 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 though the user can find numerous helping communities.
- User is expected to possess prior programming language knowledge.

Q 3) What are the different types of locators in Selenium?

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 4) 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 5) 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 6) 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 as Selenium 1

Q 7) 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

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, 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 8) 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 9) How to select value in a dropdown?

Value in the drop down 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 10) 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 11) How to handle frame in WebDriver?

An inline frame acronym as iframe is used to insert another document with in 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");
```

frame(Name of Frame)

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

Q 12) 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 13) 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 14) 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 15) How can I read test data from excels?

Test data can efficiently be read from excel using JXL or POI API

Q 16) 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 17) 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 18) Tell me some TestNG Annotations.

@Test, @Parameters, @Listeners, @BeforeSuite, @AfterSuite, @BeforeTest, @AfterTest,
@DataProvider, @BeforeGroups, @AfterGroups, @BeforeClass, @AfterClass,
@BeforeMethod, @AfterMethod, @Factory

Q 19) What are desired capabilities?

Desired Capabilities help to set properties for the Web Driver. A typical use case would be to set the path for the Firefox Driver if your local installation doesn't correspond to the default settings.

Q 20) Describe your framework.

Q 21) Which is the best way to locate an element?

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

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

Q 22) In what situation selenium finding element get fails?

- Element loading issue
- Dynamic id of web element

Q 23) What is the difference between "GET" and "NAVIGATE" to open a web page in selenium web driver?

Get method will get a page to load or get page source or get text that's all whereas navigate will guide through the history like refresh, back, forward. For example if we want to move forward and do some functionality and back to the home page this can be achieved through `navigate()` only. `driver.get` will wait till the whole page gets loaded and `driver.navigate` will just redirect to that page and will not wait

Q 24) How we can retrieve the dynamically changing Ids?

Q 25) What is the difference between driver.Close() and driver.Quit () method?

Close() - It is used to close the browser or page currently which is having the focus.

Quit() - It is used to shut down the web driver instance or destroy the web driver instance
(Close all the windows)

Q 26) What is the basic use of Firefox profiles and how can we use them using selenium?

A profile in Firefox is a collection of bookmarks, browser settings, extensions, passwords, and history; in short, all of your personal settings.

We use them to change user agent, changing default download directory, changing versions etc.

Q 27) How to handle Ajax popup window?

By using `getWindowHandles()` and `obj.switchTo.window(windowid)` we can handle popups using

`explicit wait` and `driver.switchTo.window("name")` commands for your requirements.

Q 28) How to change user agent in Firefox by seleniumweb driver.

`FirefoxProfile profile = new FirefoxProfile();`

`profile.setPreference("general.useragent.override", "some UA string");`

`Web Driver driver = new FirefoxDriver(profile);`

Q 29) What is the MOST challenging test problem in my career in Automation?

In my career

- Changing XPATHS' between testing server and production server-by keeping generic xpath
- Keep separate property files for production and UAT
- automating flash apps
- Mobile Automation

Q 30) What is the difference between @before method and @beforeclass.

In JUnit4 @Before is used to execute set of preconditions before executing a test. For example, if there is a need to open some application and create a user before executing a test, then this annotation can be used for that method. Method that is marked with @Before will be executed before executing every test in the class. If a JUnit test case class contains lot of tests which all together need a method which sets up a precondition and that needs to be executed before executing the Test Case class then we can utilise "@BeforeClass" annotation.

Q 31) How to mouse hover on an element?

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

Q 32) How to switch between the windows?

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

Q 33) What is actions class in web driver?

Actions class with web Driver help is Sliding element, Resizing an Element, Drag & Drop, hovering a mouse, especially in a case when dealing with mouse over menus.

Q 34) Difference between find element () and findelements ()?

findElement() :

Find the first element within the current page using the given "locating mechanism".
Returns a single WebElement.

findElements() :

Find all elements within the current page using the given "locating mechanism".
Returns List of Web Elements.

Q 35) Write down scenarios which we can't automate?

Barcode Reader, Captcha etc

Q 36) Differences between jxl and ApachePOI.

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

Q 37) Does Selenium support https protocols?

Yes

Q 38) What is the difference between Method Overloading and Method Overriding?

Q 39) What are access modifiers ?

www.H2KINFOSYS.com || Training@H2KInfosys.com || H2KInfosys@Gmail.com
USA: +1-770-777-1269

www.H2KINFOSYS.com || Training@H2KInfosys.com || H2KInfosys@Gmail.com
USA: +1-770-777-1269