Some Generic Selenium Questions

Question 1:
What is Selenium?
Answer:
Selenium is a browser automation tool which lets you automated operations like – type, click, and selection from a drop down of a web page.

Question 2:
How is Selenium different from commercial browser automation tools?
Answer:
Selenium is a library which is available in a gamut of languages i.e. java, C#, python, ruby, php etc while most commercial tools are limited in their capabilities of being able to use just one language. More over many of those tools have their own proprietary language which is of little use outside the domain of those tools. Most commercial tools focus on record and replay while Selenium emphasis on using Selenium IDE (Selenium record and replay) tool only to get acquainted with Selenium working and then move on to more mature Selenium libraries like Remote control (Selenium 1.0) and Web Driver (Selenium 2.0).
Though most commercial tools have built in capabilities of test reporting, error recovery mechanisms and Selenium does not provide any such features by default. But given the rich set of languages available with Selenium it very easy to emulate such features.

Question3:
What are the set of tools available with Selenium?
Answer:
Selenium has four set of tools – Selenium IDE, Selenium 1.0 (Selenium RC), Selenium 2.0 (WebDriver) and Selenium Grid. Selenium Core is another tool but since it is available as part of Selenium IDE as well as Selenium 1.0, it is not used in isolation.

Question 4:
Which Selenium Tool should I use?
Answer:
It entirely boils down to where you stand today in terms of using Selenium. If you are entirely new to Selenium then you should begin with Selenium IDE to learn Selenium location strategies and then move to Selenium 2 as it is the most stable Selenium library and future of Selenium. Use Selenium Grid when you want to distribute your test across multiple devices. If you are already using Selenium 1.0 than you should begin to migrate your test scripts to Selenium 2.0

Question 5:
What is Selenium IDE?
Answer:
Selenium IDE is a firefox plug-in which is (by and large) used to record and replay test is firefox browser. Selenium IDE can be used only with firefox browser.

Question 6:
Which language is used in Selenium IDE?
Answer:

Selenium IDE uses html sort of language called Selenese. Though other languages (java, c#, php etc) cannot be used with Selenium IDE, Selenium IDE lets you convert test in these languages so that they could be used with Selenium 1.0 or Selenium 2.0

Question 7:
What is Selenium 1.0?
Answer:
Selenium 1.0 or Selenium Remote Control (popularly known as Selenium RC) is library available in wide variety of languages. The primary reason of advent of Selenium RC was incapability of Selenium IDE to execute tests in browser other than Selenium IDE and the programmatical limitations of language Selenese used in Selenium IDE.

Question 8:
What is Selenium 2.0?
Answer:
Selenium 2.0 also known as WebDriver is the latest offering of Selenium. It provides
- better API than Selenium 1.0
- does not suffer from java script security restriction which Selenium 1.0 does
- supports more UI complicated UI operations like drag and drop

Question 9:
What are the element locators available with Selenium which could be used to locate elements on web page?
Answer:
There are mainly 4 locators used with Selenium –
- html id
- html name
- XPath locator and
- Css locators

Question 10:
What is Selenium Grid?
Answer:
Selenium grid lets you distribute your tests on multiple machines and all of them at the same time. Hence you can execute test on IE on Windows and Safari on Mac machine using the same test script (well, almost always). This greatly helps in reducing the time of test execution and provides quick feedback to stack holders.

Selenium IDE Questions

Question 11:
What are two modes of views in Selenium IDE?

Answer:
Selenium IDE can be opened either in side bar (View > Side bar > Selenium IDE) or as a pop up window (Tools > Selenium IDE). While using Selenium IDE in browser side bar it cannot record user operations in a pop up window opened by application.

Question 12:
Can I control the speed and pause test execution in Selenium IDE?
Answer:
Selenium IDE provides a slider with Slow and Fast pointers to control the speed of execution.

Question 13:
Where do I see the results of Test Execution in Selenium IDE?
Answer:
Result of test execution can be views in log window in Selenium IDE –

Question 14:
Where do I see the description of commands used in Selenium IDE?
Answer:
Commands of description can be seen in Reference section –

Question 15:
Can I build test suite using Selenium IDE?
Answer:
Yes, you can first record individual test cases and then group all of them in a test suite. Following this entire test suite could be executed instead of executing individual tests.

Question 16:
What verification points are available with Selenium?
Answer:
There are largely three type of verification points available with Selenium –
- Check for page title
- Check for certain text
- Check for certain element (text box, drop down, table etc)

Question 17:
I see two types of check with Selenium – verification and assertion, what's the difference between tow?
Answer:
- A verification check lets test execution continue even in the wake of failure with check, while assertion stops the test execution. Consider an example of checking text on page, you may like to use verification point and let test execution continue even if text is not present. But for a login page, you would like to add assertion for presence of text box login as it does not make sense continuing with test execution if login text box is not present.

Question 18:
I don't see check points added to my tests while using Selenium IDE☹, how do I get them added to my tests?
Answer:
You need to use context menu to add check points to your Selenium IDE tests –

Question 19:
How do I edit tests in Selenium IDE?
Answer:
There are two ways to edit tests in Selenium IDE; one is the table view while other looking into the source code of recorded commands –

Question 20:
What is syntax of command used in Selenium?
Answer:
There are three entities associated with a command –
- Name of Command

- Element Locator (also known as Target)
- Value (required when using echo, wait etc)

Question 21:
There are tons of Selenium Command, am I going to use all of them
Answer:
This entirely boils down to operations you are carrying out with Selenium. Though you would definitely be using following Selenium Commands more often –
- Open: opens a web page.
- click/clickAndWait: click on an element and waits for a new page to load.
- Select: Selects a value from a drop down value.
- verifyTitle/assertTitle: verifies/asserts page title.
- verify/assert ElementPresent: verifies/asserts presence of element, in the page.
- verify/assert TextPresent verifies/asserts expected text is somewhere on the page.

Question 22:
How do I use html id and name while using Selenium IDE
Answer:
Html id and name can be used as it is in selenium IDE. For example Google search box has name – "q" and id – "list-b" and they can be used as target in selenium IDE –

Question 23:
What is XPath? When would I have to use XPath in Selenium IDE?
Answer:

XPath is a way to navigate in xml document and this can be used to identify elements in a web page. You may have to use XPath when there is no name/id associated with element on page or only partial part of name/ide is constant.

Direct child is denoted with - /

Relative child is denoted with - //

Id, class, names can also be used with XPath –

- //input[@name='q']
- //input[@id='lst-ib']
- //input[@class=' lst']

If only part of id/name/class is constant than "contains" can be used as –

- //input[contains(@id,'lst-ib')]

Question 24:
What is CSS location strategy in Selenium?
Answer:

CSS location strategy can be used with Selenium to locate elements, it works using cascade style sheet location methods in which -

Direct child is denoted with – (a space)

Relative child is denoted with - >

Id, class, names can also be used with XPath –

- css=input[name='q']
- css=input[id='lst-ib'] or input#lst-ib
- css=input[class=' lst'] or input.lst

If only part of id/name/class is constant than "contains" can be used as –

- css=input[id*=' lst-ib ')]

Element location strategy using inner text

- css = a:contains('log out')

Question 25:
There is id, name, XPath, CSS locator, which one should I use?
Answer:
If there are constant name/id available than they should be used instead of XPath and CSS locators. If not then css locators should be given preference as their evaluation is faster than XPath in most modern browsers.

Question 26:
I want to generate random numbers, dates as my test data, how do I do this in Selenium IDE?
Answer:
This can be achieved by executing java script in Selenium. Java script can be executed using following syntax –

```
type
css=input#s
javascript{Math.random()}
```

And for date –

```
type
css=input#s
javascript{new Date()}
```

Question 27:
Can I store result of an evaluation and use it later in my test?
Answer:
You can use "store" command to achieve this. You can save result of an evaluation in a variable and use it later in your Selenium IDE script. For example we can store value from a text box as following, and later use it to type it in another text box –

```
storeText
css=input#s
var1
type
css=input#d
${var1}
```

Question 28:
I have stored result of an evaluation; can I print it in IDE to check its value?
Answer:
You can use echo command as following to check the stored value in Selenium IDE –

```
storeText
css=input#s
var1
echo
${var1}
```

Question 29:
Can I handle java script alert using Selenium?
Answer
You could use verify/assertAlert to check presence of alert on page. Since selenium cannot click on "Ok" button on js alert window, the alert itself does not appear on page when this check is carried out.

Question 30:
Selenium has recorded my test using XPath, how do I change them to css locator?
Answer
You can use drop down available next to Find in Selenium to change element locator used by Selenium –

Question 31:

I have written my own element locator, how do I test it?

Answer
You can use Find button of Selenium IDE to test your locator. Once you click on it, you would see element being highlighted on screen provided your element locator is right Else one error message would be displayed in log window.

Question 32:
I have written one js extension; can I plug it in Selenium and use it?
Answer
You could specify you js extension in "Options" window of Selenium IDE –

Question 33:
How do I convert my Selenium IDE tests from Selenese to another language?
Answer
You could use Format option of Selenium IDE to convert tests in another programming language –

Question 34:
I have converted my Selenium IDE tests to java but I am not able to execute them☹, execution options as well as Table tab of Selenium IDE is disabled☹☹

Answer
This is because Selenium IDE does not support execution of test in any other language than Selenese (language of Selenium IDE). You can convert Selenium IDE in a language of your choice and then use Selenium 1.0 to execute these tests in that language.

Question 35:
I want to use only Selenese as my test script language but still want to execute tests in other browsers, how do I do that?
Answer
You can execute you Selenese test in another browser by specifying the "-htmlSuite" followed by path of your Selenese suite while starting the Selenium Server. Selenium Server would be covered in details in question about Selenium RC.

Question 36:
I have added one command in middle of list of commands, how do I test only this new command?
Answer
You can double click on newly added command and Selenium IDE would execute only that command in browser.

Question 37:
Can I make Selenium IDE tests begin test execution from a certain command and not from the very first command?
Answer
You could set a command as "start" command from context menu. When a command is set as start command then a small green symbol appears before command. Same context menu can be used to toggle this optio

Question 38:
Are there other tools available outside Selenium IDE to help me tests my element locators
Answer
You could XPath checker - https://addons.mozilla.org/en-US/firefox/addon/xpath-checker/ to test you XPath locators and Firefinder (a firebug add on) to test you css locators – https://addons.mozilla.org/en-US/firefox/addon/firefinder-for-firebug/
Firefinder can also be used to test XPath locators.

Question 39:
What is upcoming advancement in Selenium IDE?
Answer
The latest advancement in Selenium IDE would be to have capabilities of converting Selenium IDE tests in Webdriver (Selenium 2.0) options. This would help generating quick and dirty tests for Selenium 2.0

Question 40:
How can I use looping option (flow control) is Selenium IDE
Answer
Selenese does not provide support for looping, but there is extension which could be used to achieve same. This extension can be download from here - http://51elliot.blogspot.com/2008/02/selenium-ide-goto.html
This extension can be added under "Selenium IDE Extension" section to use loop feature in Selenium IDE.

Question 41:
Can I use screen coordinate while using click command? I want to click at specific part of my element.
Answer
You would need to use clickAT command to achieve. clickAt command accepts element locator and x, y coordinates as arguments –
clickAt(locator, coordString)

Question 42:
How do I verify presence of drop down options using Selenium?
Answer

Use assertSelectOptions as following to check options in a drop down list –
 assertSelectOptions

Question 43:
Can I get data from a specific html table cell using Selenium IDE?
Answer
Use storeTable command to get data from a specific cell in an html table, following example store text from cell 0,4 from an html table –
 storeTable
 css=#tableId.0.4
 textFromCell

Question 44:
I want to make Selenium IDE record and display css locator followed by other locators, is it possible to give high priority to css locator in Selenium IDE?
Answer
You can change default behavior of Selenium IDE > element locator preference by crating js file with following–
```
LocatorBuilders.order = ['css:name', 'css:id', 'id', 'link', 'name',
'xpath:attributes'];
```

And add this file under "Selenium IDE Extension" under Selenium Options.

Question 45:
My application has dynamic alerts which don't always appear, how do I handle them?
Answer
If you want to simulate clicking "ok " on alert than use – chooseOkOnNextConfirmation and if you want to simulate clicking "cancel" on alert than use -
 http://draft.blogger.com/blogger.g?blogID=4608630209994205502chooseCancelOnNextConfir mation ( )

Question 46:
Can I right click on a locator?
Answer

You can use command - contextMenu (locator) to simulate right click on an element in web page.

Question 47:
How do I capture screen shot of page using Selenium IDE?
Answer

Use command – captureEntirePageScreenshot to take screen shot of page.

Question 48:

I want to pause my test execution after certain command.
Answer

Use pause command which takes time in milliseconds and would pause test execution for specified time – pause (waitTime)

Question 49:
I used open command to launch my page, but I encountered time out error☹
Answer

This happens because open commands waits for only 30 seconds for page to load. If you application takes more than 30 sec then you can use "http://draft.blogger.com/blogger.g?blogID=4608630209994205502setTimeout (timeout)" to make selenium IDE wait for specified time, before proceeding with test execution.
Question 50:
What's the difference between type and typeKeys commands?
Answer

type command simulates enter operations at one go while typeKeys simulates keystroke key by key.
typeKeys could be used when typing data in text box which bring options (like Google suggestion list) because such operation are not usually simulated using type command.

Selenium RC (Selenium 1.0) Questions

Question 51:
What is Selenium RC (also known as Selenium 1.0)?
Answer

Selenium RC is an offering from SeleniumHQ which tries to overcome following draw backs of Selenium IDE –
- Able to execute tests only with Firefox
- Not able to use full-fledged programming language and being limited to Selenese

Question 52:
What are the main components of Selenium RC?
Answer

Selenium RC has two primary components –
- Client libraries which let you writes tests in language of your preference i.e. java, C#, perl, php etc
- Selenium sever which acts as a proxy between browser and application under test (aut)

Question 53:
Why do I need Selenium Server?
Answer

Selenium uses java script to drives tests on a browser; Selenium injects its own js to the response which is returned from aut. But there is a java script security restriction (***same origin policy***) which lets you modify html of page using js only if js also originates from the same domain as html. This security restriction is of utmost important but spoils the working of Selenium. This is where Selenium server comes to play an important role.
Selenium server stands between aut and browser and injects selenium js to the response received from aut and then it is delivered to broswer. Hence browser believes that entire response was delivered from aut.

Question 54:
What is Selenium core? I have never used it!!!
Answer

Selenium core is the core js engine of Selenium which executes tests on browser, but because of same origin policy it needs to be deployed on app server itself, which is not always feasible. Hence Selenium core is not used in isolation. Selenium IDE as well as Selenium RC use Selenium core to drive tests while over coming same origin policy. In case of Selenium IDE tests are run in context of browser hence it is not hindered by same origin policy and with Selenium RC, Selenium Server over comes same origin policy.

Question 55:
Where is executable for Selenium RC, how do I install it?
Answer

Installation is a misnomer for Selenium RC. You don't install Selenium RC you only add client libraries to you project. For example in case of java you add client driver and Selenium server jars in Eclipse or IntelliJ which are java editors.

Question 56:
I have downloaded Selenium Server and Client libraries, how do I start Selenium Server?
Answer

To start Selenium Server, you need to navigate to installation directory of Selenium server and execute following command –
        Java -jar .jar
This will start Selenium server at port 4444 by default.
Notice that you need to have java version 1.5 or higher available on your system to be able to use Selenium server.

Question 57:
On my machine port 4444 is not free☹ . How do Use another port?

Answer

You can specify port while running the selenium server as –
        Java -jar .jar –port 5555

Question 58:
I am new to programming; can Selenium generate sample code from my Selenese scripts?
Answer

You can first record tests in Selenium IDE and then use format option to convert them in a language of your choice.

Question 59:
Can I start Selenium server from my program instead of command line
Answer

If you are using java then you can start Selenium server using SeleniumServer class. For this you need to instantiate SeleniumServer and then call start() method on it.

```
seleniumServer = new SeleniumServer();
seleniumServer.start();
```

Question 60:
And how do I start the browser?
Answer

While using java you need to create instance of DefaultSelenium class and pass it four parameters –

```
selenium = new DefaultSelenium(serverHost, serverPort, browser,
appURL);
            selenium.start();
```
Herein you need to pass,
      host where Selenium server is running,
      port of Selenium server,
      browser where tests are to be executed and
      application URL

Question 61:
I am not using java to program my tests, do I still have to install java on my system?

Answer

Yes, since Selenium server is written in java you need java to be installed on your system to be able to use it. Even though you might not be using java to program your tests.

Question 62:
What are the browsers offering from Selenium?
Answer

Following browsers could be used with Selenium –
```
   *firefox
   *firefoxproxy
   *pifirefox
   *chrome
   *iexploreproxy
   *iexplore
   *firefox3
   *safariproxy
   *googlechrome
   *konqueror
   *firefox2
   *safari
   *piiexplore
   *firefoxchrome
   *opera
   *iehta
   *custom
```
Here pi and proxy stand for proxy injection mode of Selenium server

Question 63:
During execution of tests I see two browser windows; one my test application another window shows commands being executed. Can I limit it to just one window?
Answer

Yes you can instruct Selenium Server to launch just one window. To do so you must specify – singleWindow while starting the Selenium server as –
        Java -jar .jar –singleWindow

Question 64:
My tests usually time out☹. Can I specify bigger time out?
Answer

This usually happens while using open method with Selenium. One way to overcome this is to use setTimeOut method in your test script another way is to specify time out while starting Selenium server as following –
        Java -jar .jar –timeout

Question 65:
My system is behind corporate network, how do I specify my corporate proxy?
Answer

You can specify your corporate proxy as following while starting Selenium Server –

java -jar .jar -Dhttp.proxyHost= -Dhttp.proxyPort= -Dhttp.proxyUser= -Dhttp.proxyPassword=

Question 66:
I am switching domains in my test scripts. I move from "yahoo.com" to "google.com" and my tests encounter permission denied error☹
Answer

Changing domains is also a security restriction from java script. To overcome this you should start you Selenium server in proxy injection mode as –
      java -jar .jar –proxyInjectionMode
Now Selenium server would act as proxy server for all the content going to test application

Question 67:
Can I execute my Selenese tests in another browser using Selenium server? I want to use only Selenese script my tests but still want to execute my test in non firefox browsers
Answer

Yes you can. To do so you need to specify following parameters while starting Selenium server –
      Browser
      Test domain
      Path to html suite (you Selenese tests) and
      Path to result
java -jar <>.jar -htmlSuite

Question 68:
Can I log more options during test execution?
Answer

If you want to log browser side option during test execution then you should start Selenium server as following –

      java -jar <>.jar –browserSideLog

Question 69:
Can I use Selenium RC on my UNIX/Mac also?
Answer
You can use Selenium RC on any system which is capable I running Java. Hence you can use it on RC and UNIX machines also

Question 70:

I want to test my scripts on new experimental browser, how do I do that?

Answer

You can specify *custom followed by path to browser to execute your tests on any browser while starting Selenium server

```
*custom
```

Question 71:

I executed my tests cases but where is the test report?

Answer

Selenium RC itself does not provide any mechanism for test reporting. Test reporting is driven from the framework you use for Selenium. For example with java client driver of Selenium –

- if you are using JUnit then you can use ant plug-in of JUnit to generate test report
- if you are using TestNG then TestNG generates reports for you

Same reporting option is available with PHP unit and other client libraries you are using.

Question 72:

How do I use recovery scenarios with Selenium? I used to use them with QTP.

Answer

Power of recovery scenarios lies with the programming language you use. If you are using java then you can use Exception handling to overcome same. For example if you are reading data from a file and file is not available then you should keep you code statements in "try catch" block so that test execution could continue even in the wake of errors. Such mechanism entirely boils down the errors you want to recover from, while being able to continue with test execution.

Question 73:

How do I iterate through options in my test script.

Answer

You can use loop features of the programming language, for example you can use "for" loop in java as following to type different test data in a text box –

```
// test data collection in an array
String[] testData = {"test1", "test2", "test3"};

// iterate through each test data
for (String s : testData) {
        selenium.type("elementLocator", testData);
}
```

Question 74:
Can I execute java script from my tests? I want to count number of images on my page.
Answer
You can use method getEval() to evaluate java script. For example if you want to count number of images then you can pass following dom statement to getEval() as following –

selenium.getEval("window.document.images.length;");
Or to get All anchor objects from a page
selenium.getEval("window.document.getElementsByTagName('a');");

Question 75:
Is there a way for me to know all available options when I start Selenium Server?
Answer
If you want to see all options available while starting Selenium server then you should use option "-h" while starting Selenium server -
Java –jar .jar –h
It would display you all the options which you can use while starting the Selenium server.

Question 76:
I have created my own firefox profile; can I execute my test scripts on it
Answer
You may like to create your own firefox profile because Selenium always created a clean firefox profile while executing the tests and none of your FF settings and plug-in are considered with this clean profile. If you want to execute tests on FF with your settings then you should create custom profile for FF.
To be able to execute tests on a custom firefox profile you should specify its path while starting Selenium server. For example if your new profile is stored at "awesome location" in your directory then you should start Selenium server as following –
```
Java –jar .jar -firefoxProfileTemplate   "awesome
location"
```

Question 77:
How do I capture server side log from Selenium server?
Answer
Start your Selenium Server as following –

java -jar .jar -log selenium.log

And Selenium would start logging server side info, i.e.

20:44:25 DEBUG [12] org.openqa.selenium.server.SeleniumDriverResourceHandler -
Browser 12345/:top frame1 posted START NEW

Question 78:
What are Heightened Privileges Browsers?
Answer
Firefox and IE have browser modes which are not restricted by java script's same origin policy. These browsers are known as browsers with elevated security privileges. In case of Firefox it is known as chrome (It's not the Google browser) and in case of IE it is known as iehta

Question 79:
My application has lots of pop up window, how do I work with them?
Answer
You need to know the Window ID of pop window to be able to work with them.
First you need to bring control on pop up window; execute selenium commands there, close the pop up window and then bring control back to main window. Consider following example where click on an image brings a pop up window –

```
        // click on image brings pop up window
    selenium.click("css=img");

// wait for pop up window identified using anchor target "ss"
        selenium.waitForPopUp("ss", getWaitPeriod());
        selenium.selectWindow("ss");

        // Some more operations on popup window
// Close the pop up window and Select the main application
window
        // Main window is selected by adding null as argument
        selenium.close();
        selenium.selectWindow("null");
        // continue with usual operation ☺
```


Question 80:
While trying to execute my tests with firefox I encountered following error – "Firefox Refused Shutdown While Preparing a Profile". How do I solve it?
Answer
This message simply means that Selenium is not able to launch FF browser as it is already running on your system. To overcome this you should close all running instances of FF browser. You should also check your system process if there is any hidden FF profile running which is not visible on screen. You should kill all FF processes and following this your tests should run smooth

Question 80:
While trying to execute my tests with firefox I encountered following error – "Firefox Refused Shutdown While Preparing a Profile". How do I solve it?

Answer

This message simply means that Selenium is not able to launch FF browser as it is already running on your system. To overcome this you should close all running instances of FF browser. You should also check your system process if there is any hidden FF profile running which is not visible on screen. You should kill all FF processes and following this your tests should run smooth

Question 81:

My application uses Ajax heavily how do I use Selenium RC to work with Ajax operations?

Answer

Ajax operations don't reload a page like normal form submission but they make http requests behind the scene. You cannot use waitForPageToLoad for such operations and instead should use conditional wait for change in state of application. This could as well mean waiting for presence of an element before continuing with test operations. Consider following example in which type operation triggers Ajax operation which is followed by conditional wait for presence of a text box –

```
// type operation brings element "q" on screen without loading
the page
selenium.type("elementLocator", "testData");

// conditional wait for element "q"
for (int second = 0;; second++) {
     if (second >= 60) fail("timeout");
try { if (selenium.isElementPresent("q")) break;
} catch (Exception e) {}
         Thread.sleep(1000);
}
```

Question 82:

How do I upload a file using Selenium? I need to upload a word file during test execution.

Answer

If you are using Firefox then you can use "type" command to type in a File Input box of upload file. But type operation does not work with IE and you would have to use "Robot" class in java to work make file upload work.

Question 83:

Why do I get "permission denied" error during execution of Selenium tests?

Answer

The primary reason of permission denied error is same origin policy restriction from java script. To overcome this error you can use browsers with elevated security privileges. In case of Firefox you should use *chrome and in case of IE you should use *iehta as browser for working with Selenium.

Question 84:
I am not able to use "style" attribute to locate element with IE browser☹
Answer
This is because IE expects attribute values to be in caps while other browsers expect it to be lower case letters. Hence

//tr[@style="background-color:yellow"] works with other browsers
//tr[@style="BACKGROUND-COLOUR:yellow"] works with IE

Question 85:
Are there any technical limitations while using Selenium RC?
Answer
Besides notorious "same origin policy" restriction from js, Selenium is also restricted from exercising anything which is outside browser. For example you cannot click on "Tools" option of your browser by just using Selenium.

Question 86:
But my tests need me to exercise objects outside browser, how do I achieve it?
Answer
You can use Robot class in java to achieve this, but it would be dirty solution even if you get through this.

Question 87:
Does Selenium have any offering for mobile browsers?
Answer
Selenium 2.0 (WebDriver) provides iPhone as well Android drivers which could be used to drive tests on mobile browsers

Question 88:
How does Selenium RC stand with other commercial tools?
Answer
The biggest advantage of Selenium RC is that it is absolutely free and has vast support of languages and browsers (almost always). Selenium lags when it comes to test reporting as Selenium does not have any in built reporting but this can be easily achieved using the programming language you decide to work on with Selenium. A bigger drawback is not being able to exercise objects which are outside browser window, for example clicking on folder on your desktop.

Question 89:
How does Selenium RC stand with other commercial tools?
Answer

The biggest advantage of Selenium RC is that it is absolutely free and has vast support of languages and browsers (almost always). Selenium lags when it comes to test reporting as Selenium does not have any in built reporting but this can be easily achieved

Question 90:
Can I just use Selenium RC to drive tests on two different browsers on one operating system without using Selenium Grid?
Answer
If you are using java client driver of Selenium then java testing framework TestNG lets you achieve this. You can set tests to be executed in parallel using "parallel=test" attribute and define two different tests, each using a different browser. Whole set up would look as (notice the highlighted sections for browser in test suite)–

```xml
xml version="1.0" encoding="utf-8"?>
DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name="Test Automation" verbose="10">

    <parameter name="serverHost" value="localhost" />
    <parameter name="appURL" value="http://tamil.yahoo.com"/>
    <parameter name="proxyInjection" value="false" />
    <parameter name="serverPort" value="4444"/>

    <test name="Web Test1">
        <parameter name="browser" value="*chrome" />
        <parameter name="m" value="1">parameter>
<classes><class name="com.core.tests.TestClass1">class>classes>
    test>

    <test name="Web Test2">
        <parameter name="browser" value="*iehta" />
        <parameter name="m" value="2">parameter>
<classes><class name="com.core.tests.TestClass1">class>classes>
    test>

suite>
```

Selenium Grid Questions

Question 91:
How do I cut down text execution time for my selenium tests? I want to execute my tests on a combination of different machines and browsers.
Answer

Selenium grid is your friend☺. Selenium grid lets you distribute tests across browsers and machines of your choice.

Question 92:
How does Selenium grid works?

Answer
Selenium grid uses combination of Selenium RC servers to execute tests in multiple browsers on different machine. Herein one Selenium RC server works as hub while other RC servers work as slaves, which could be controlled by hub. Whenever there is a request for a specific configuration for test execution then hub looks for a free RC slave server and if available then test execution begins on it. Once test execution is over then RC slave server would be available for next set of test execution.

Question 93:
Can you show me one diagram which describes functionality of Selenium grid?
Answer
In the following diagram Selenium hub is controlling three Selenium RC servers which are running for configurations –
- IE on Windows
- FF on Linux
- FF on windows

Question 94:
Which jar files are needed to works with Selenium GRID?
Answer
You need to download and add following jar files to your Selenium set up to be able to work with Selenium. These jar files are –
- selenium-grid-remote-control-standalone-.jar
- selenium-grid-hub-standalone-.jar
- selenium-grid-tools-standalone-.jar

Question 95:
How do I start Selenium Grid hub from my machine?
Answer
You should have "ant" set up on your system to be able to work with Grid. Once you have downloaded Selenium Grid, navigate to its distribution directory and execute following command -
        ant launch-hub

This would start grid hub on port 4444 locally. You can verify this by navigating to following URL -   http://localhost:4444/console

Question 96:
How do I start Selenium Grid Slave Server from my system?
Answer

Navigate to download directory of Selenium gird and execute following command –
        ant launch-remote-control
This would launch remote control server at port 555. At this point if you navigate
to http://localhost:4444/console then you would see this remote control listed under "Available
Remote Controls"

Question 97:
How do I start Selenium grid slave on a different port than 5555?
Answer
You can use option "-Dport" followed by port number to start grid slave on a specific port.
        ant -Dport=1111 launch-remote-control
        ant -Dport=2222 launch-remote-control

Question 98:
How do I start grid RC slaves on a different system than my local host so than hub could control
and contact a specific configuration?
Answer
You should specify following configuration while starting RC slave –
        ant -Dport= -Dhost= -DhubURL= launch-remote-control
Herein "hostname" is the host where RC slave is running and "hub url" is URL of machine
where grid hub is running.

Question 99:
How do I specify an environment while starting grid slave machine?
Answer
You could specify an environment using "-Denvironment" while starting a slave machine.
        ant -Denvironment="Safari on Mac" launch-remote-control
Herein Safari on Mac is the environment which would be used to recognize configuration of RC
slave.

Question 100:
How do I use machine specific configuration in my Selenium tests?
Answer
You could specify machine specific configuration while instantiating Selenium as –
Selenium = new DefaultSelenium("localhost", 4444, **'Safari on Mac'**, 'http://yahoo.com');
And then you use this selenium instance to carryout operation on your web application.

Question 101:
But how does my tests know that 'Safari on Mac' mean a safari browser? How does mapping
between names like 'Safari on Mac' and original browser options available in Selenium work?
Answer
Selenium grid uses a file called "grid_configuration.yml" which defines configurations of all
browsers. You would have to add this in your project. This file looks like –

Question 102:
How does Selenium grid hub keeps in touch with RC slave machine?

Answer
Selenium grid hub keeps polling all RC slaves at predefined time to make sure they are available for testing. If not then Selenium hub disconnect any unavailable RC slaves and makes it clear that any RC slave is not available for testing. The deciding parameter is called – "remoteControlPollingIntervalInSeconds" and is defined in "grid_configuration.yml" file.

Question 103:
My RC becomes unresponsive at times and my Selenium grid hub keeps waiting for RC slave to respond☹. How do I let hub know to give up on RC slave after a certain time?
Answer
You could state Selenium grid hub to wait for RC slave for a predefined time, and if RC slave does not responds with in this time then hub disconnects test execution on that slave. This parameter is called "sessionMaxIdleTimeInSeconds" and this parameter can be defined in "grid_configuration.yml" file.

Question 104:
What if my hub goes down while Selenium RC slaves are up and running?
Answer
There is one heart beat mechanism from RC slave to hub which is reciprocal to mechanism used by hub to slave machines. RC slaves use a parameter called "hubPollerIntervalInSeconds" to keep track of running grid hub. This parameter can be defined while starting the hub as –
ant -DhubPollerIntervalInSeconds= launch-hub
if hub does not respond within this time then RC slaves deregister themselves from hub.

Question 105:
Can Selenium grid be used for performance testing?
Answer
Selenium grid is for functional testing of application across different configuration. Performance testing is usually not carried out on actual devices but on a simulated http request/response mechanism. If you want to use Selenium Grid for performance testing then you would have to invest heavily on s/w and h/w infrastructure.

Question 106:
Are there additional logs available while working with Selenium grid?
Answer
You can find Selenium grid hub logs in "log/hub.log" and Remote Control logs in "log/rc-*.log" folder.

Question 107:
There are various options available while starting a Selenium server, how do I use them with Selenium grid?
Answer
You can use "`seleniumArgs`" Java property while launching the remote control and specify any of the option which you would with normal Selenium set up. For example you can run Selenium RC slave in single window mode using following command –
        ant -DseleniumArgs="-singleWindow -debug" launch-remote-control

Question 108:
I see duplicate entries in my hub console, same RC slave listed more than once :-O

Answer
This is because you are killing RC slave very ferociously. For example you if you just close the console without actually ending the process in more civil manner. To avoid this you should kill all RC slaves in more civil manner. If you encounter this error then you should restart Selenium hub and RC slaves would them register themselves to hub.

Question 109:
How do I specify my corporate proxy while starting Selenium grid hub or slave machines?
Answer
You could use setting for "http.proxyHost" abd "http.proxyPort" while starting hub and remote control machines –
ant -Dhttp.proxyHost= -Dhttp.proxyPort= launch-hub
ant -Dhttp.proxyHost= -Dhttp.proxyPort= launch-remote-control

Question 110:
How do I use Selenium Grid while using Java, .Net or Ruby
Answer
With java you can take advantage of parallel testing capabilities of TestNG to drive your Selenium grid tests
With .Net you can use "Gallio" to execute your tests in parallel
With Ruby you can use "DeepTest" to distribute your tests

Question 111:
How about the test report when I use Selenium grid for test execution?
Answer
This entirely boils down to framework you use to write your tests. For example in case of java, TestNG reports should suffice.

Web Driver (Selenium 2.0) Questions

Question 112:
What is Selenium 2.0? I have heard this buzz word many times.
Answer
Selenium 2.0 is consolidation of two web testing tools – Selenium RC and WebDriver, which claims to give best of both words – Selenium and WebDriver. Selenium 2.0 was officially released only of late.

Question 113:
Why are two tools being combined as Selenium 2.0, what's the gain?

Answer
Selenium 2.0 promises to give much cleaner API then Selenium RC and at the same time not being restricted by java script Security restriction like same origin policy, which have been haunting Selenium from long. Selenium 2.0 also does not warrant you to use Selenium Server.

Question 114:
So everyone is going to use Selenium 2.0?
Answer
Well no, for example if you are using Selenium Perl client driver than there is no similar offering from Selenium 2.0 and you would have to stick to Selenium 1.0 till there is similar library available for Selenium 2.0

Question 115:
So how do I specify my browser configurations with Selenium 2.0?
Answer
Selenium 2.0 offers following browser/mobile configuration –
- AndroidDriver,
- ChromeDriver,
- EventFiringWebDriver,
- FirefoxDriver,
- HtmlUnitDriver,
- InternetExplorerDriver,
- IPhoneDriver,
- IPhoneSimulatorDriver,
- RemoteWebDriver

And all of them have been implemented from interface WebDriver. To be able to use any of these drivers you need to instantiate their corresponding class.

Question 116:
How is Selenium 2.0 configuration different than Selenium 1.0?
Answer
In case of Selenium 1.0 you need Selenium jar file pertaining to one library for example in case of java you need java client driver and also Selenium server jar file. While with Selenium 2.0 you need language binding (i.e. java, C# etc) and Selenium server jar if you are using Remote Control or Remote WebDriver.

Question 117:
Can you show me one code example of setting Selenium 2.0?
Answer
Here is java example of initializing firefox driver and using Google Search engine –

```
protected WebDriver webDriver;

//@BeforeClass(alwaysRun=true)
public void startDriver(){
webDriver = new FirefoxDriver();
// Get Google search page and perform search on term "Test"
webDriver.get("http://www.google.com");
webDriver.findElement(By.name("q")).sendKeys("Test");
webDriver.findElement(By.name("btnG")).click();
```

Question 118:
Which web driver implementation is fastest?
Answer
HTMLUnitDriver. Simple reason is HTMLUnitDriver does not execute tests on browser but plain http request – response which is far quick than launching a browser and executing tests. But then you may like to execute tests on a real browser than something running behind the scenes

Question 119:
What all different element locators are available with Selenium 2.0?
Answer
Selenium 2.0 uses same set of locators which are used by Selenium 1.0 – id, name, css, XPath but how Selenium 2.0 accesses them is different. In case of Selenium 1.0 you don't have to specify a different method for each locator while in case of Selenium 2.0 there is a different method available to use a different element locator. Selenium 2.0 uses following method to access elements with id, name, css and XPath locator –

```
driver.findElement(By.id("HTMLid"));
driver.findElement(By.name("HTMLname"));
driver.findElement(By.cssSelector("cssLocator"));
driver.findElement(By.xpath("XPathLocator));
```

Question 120:
How do I submit a form using Selenium?
Answer
You can use "submit" method on element to submit form –

```
element.submit();
```

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

Question 121:
Can I simulate pressing key board keys using Selenium 2.0?
Answer
You can use "sendKeys" command to simulate key board keys as –

```
element.sendKeys(" and some", Keys.ARROW_UP);
```
You can also use "sendKeys" to type in text box as –

```
HTMLelement.sendKeys("testData");
```

Question 122:
How do I clear content of a text box in Selenium 2.0
Answer
You can use "clear" method on text box element to clear its content –

textBoxElement.clear();

Question 123:
How do I select a drop down value using Selenium2.0?
Answer
To select a drop down value, you first need to get the select element using one of element locator
and then you can select element using visible text –

```
Select selectElement
= new Select(driver.findElement(By.cssSelector("cssSelector")));
          selectElement.selectByVisibleText("India");
```

Question 124:
What are offering to deal with popup windows while using Selenium 2.0?
Answer
You can use "switchTo" window method to switch to a window using window name. There is
also one method "getWindowHandles" which could be used to find all Window handles and
subsequently bring control on desired window using window handle –

```
webDriver.switchTo().window("windowName");

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

Question 125:
How about handling frames using Selenium 2.0?
Answer
You can use "switchTo" frame method to bring control on an HTML frame –
      driver.switchTo().frame("frameName");
You can also use index number to specify a frame –
      driver.switchTo().frame("parentFrame.4.frameName");
This would bring control on frame named – "frameName" of the 4th sub frame names
"parentFrame"

Question 126:
Can I navigate back and forth in a browser in Selenium 2.0?
Answer
You can use Navigate interface to go back and forth in a page. Navigate method of WebDriver
interface returns instance of Navigation. Navigate interface has methods to move back, forward
as well as to refresh a page –
```
      driver.navigate().forward();
      driver.navigate().back();
      driver.navigate().refresh();
```

Question 127:
What is the order of fastest browser implementation for WebDriver?
Answer

HTMLUnitDriver is the fastest browser implementation as it does not involves interaction with a browser, This is followed by Firefox driver and then IE driver which is slower than FF driver and runs only on Windows.

Question 128:
Is it possible to use Selenium RC API with Selenium 2.0?
Answer
You can emulate Selenium 1.0 API with Selenium 2.0 but not all of Selenium 1.0 methods are supported. To achieve this you need to get Selenium instance from WebDriver and use Selenium methods. Method executions might also be slower while simulating Selenium 1.0 with in Selenium 2.0

Question 129:
Can you show me one example of using Selenium 1.0 in Selenium 2.0?
Answer
Code Sample:
```
    // Create web driver instance
    WebDriver driver = new FirefoxDriver();

    // App URL
    String appUrl = "http://www.google.com";

    // Get Selenium instance
Selenium selenium = new WebDriverBackedSelenium(driver, appUrl);

    // Tests using selenium
    selenium.open(appURL);
    selenium.type("name=q", "testData");
    selenium.click("name=btnG");

    // Get back the WebDriver instance
WebDriver driverInstance = ((WebDriverBackedSelenium)
selenium).getUnderlyingWebDriver();
```

Question 130:
I had support of lots of browsers while using Selenium 1.0 and it seems lacking with Selenium 2.0, for example how do I use < awesome> browser while using Selenium 2.0?
Answer
There is a class called Capabilities which lets you inject new Capabilities in WebDriver. This class can be used to set testing browser as Safari –
```
    //Instantiate Capabilities
    Capabilities capabilities = new DesiredCapabilities()
```

```
     //Set browser name
     capabilities.setBrowserName("this awesome browser");

     //Get your browser execution capabilities
CommandExecutor executor
= new SeleneseCommandExecutor("http:localhost:4444/", "http://ww
w.google.com/", capabilities);

     //Setup driver instance with desired Capabilities
WebDriver driver = new RemoteWebDriver(executor, capabilities);
```

Question 131:

Are there any limitations while injecting capabilities in WebDriver to perform tests on a browser which is not supported by WebDriver?

Answer

Major limitation of injecting Capabilities is that "fundElement" command may not work as expected. This is because WebDriver uses Selenium Core to make "Capability injection" work which is limited by java script security policies.

Question 132:

Can I change User-Agent while using FF browser? I want to execute my tests with a specific User-Agent setting.

Answer

You can create FF profile and add additional Preferences to it. Then this profile could be passed to Firefox driver while creating instance of Firefox –

```
     FirefoxProfile profile = new FirefoxProfile();
profile.addAdditionalPreference("general.useragent.override", "U
ser Agent String");
     WebDriver driver = new FirefoxDriver(profile);
```

Question 133:

Is there any difference in XPath implementation in different WebDriver implementations?

Answer

Since not all browsers (like IE) have support for native XPath, WebDriver provides its own implementation for XPath for such browsers. In case of HTMLUnitDriver and IEDriver, html tags and attributes names are considered lower cased while in case of FF driver they are considered case in-sensitive.

Question 134:

My application uses ajax highly and my tests are suffering from time outs while using Selenium 2.0☹.

Answer

You can state WebDriver to implicitly wait for presence of Element if they are not available instantly. By default this setting is set to 0. Once set, this value stays till the life span of

WebDriver object. Following example would wait for 60 seconds before throwing ElementNotFound exception –

```
      WebDriver driver = new FirefoxDriver();
driver.manage().timeouts().implicitlyWait(60, TimeUnit.SECONDS);
WebElement element = driver.findElement(By.id("elementID"));
```

Question 135:
What if I don't want to use implicit wait and want to wait only for presence of certain elements?
Answer
You can use explicit wait in this situation to wait for presence of certain element before continuing with test execution. You can use "WebDriverWait" and "ExpectedCondition" to achieve this –

```
      WebDriver driver = new FirefoxDriver();
WebElement myDynamicElement = (new WebDriverWait(driver,
60)).until(new ExpectedCondition<WebElement>(){

      @Override
      public WebElement apply(WebDriver d) {
            return d.findElement(By.id("myDynamicElement"));
      }});
```

This is going to wait up to 60 seconds before throwing ElementNotFound exception.

Question 136:
What is RemoteWebDriver? When would I have to use it?
Answer
RemoteWebDriver is needed when you want to use HTMLUnitDriver. Since HTMLUnitDriver runs in memory, you would not see a browser getting launched –

```
        // Create HTMLUnitDriver instance
        WebDriver driver = new HtmlUnitDriver();

        // Launch Yahoo.com
        driver.get("http://www.yahoo.com");
```

Question 137:
What all languages available to be used with WebDriver?
Answer
Java and C# are on the forefront of WebDriver languages. Support is also available for Python and Ruby. There is also one java script library available for Friefox.

Question 138:
How do I handle java script alert using WebDriver?
Answer
WebDriver would support handling js alerts using Alert interface.

```
        // Bring control on already opened alert
        Alert alert = driver.switchTo().alert();

        // Get the text of the alert or prompt
        alert.getText();

    // Click ok on alert
        alert.accept();
```

Question 139:
Could I safely execute multiple instances of WebDriver implementations?
Answer
As far as HTMLUnitDriver and FF drivers are concerned, each instance would be independent of other. In case of IE driver there could be only one instance of IE driver running on Windows. If you want to execute more than one instance of IE driver then you should consider using RemoteWebDriver and virtual machines.

Question 140:
Is it possible to interact with hidden elements using WebDriver?
Answer
Since WebDriver tries to exercise browser as closely as real users would, hence simple answer is No, But you can use java script execution capabilities to interact with hidden elements.

Question 141:
I have all my tests written in Selenium 1.0 (Selenium RC), why should I migrate to Selenium 2.0 (WebDriver)?

Answer
Because –
* WebDriver has more compact and object oriented API than Selenium 1.0
* WebDriver simulates user behaviour more closely than Selenium 1.0, for example if a text box is disabled WebDriver would not be able to type text in it while Selenium 1.0 would be
* WebDriver is supported by Browser vendor themselves i.e. FF, Opera, Chrome etc

Question 142:
My XPath and CSS locators don't always work with Selenium 2.0, but they used to with Selenium 1.0☹.
Answer
In case of XPath, it is because WebDriver uses native browser methods unless it is not available. And this cause complex XPath to be broken. In case of Selenium 1.0 css selectors are

implemented using Sizzle Library and not all the capabilities like "contains" are available to be used with Selenium 2.0

Question 143:
How do I execute Java Script in Selenium 2.0?
Answer
You need to use JavaScriptExecutor to execute java script in Selenium 2.0, For example if you want to find tag name of an element using Selenium 2.0 then you can execute java script as following –

```
WebElement element
= driver.findElement(By.id("elementLocator"));
String name = (String)
((JavascriptExecutor) driver).executeScript(
     "return arguments[0].tagName", element);
```

Question 144:
Why does not my java script execution return any value?
Answer
This might happen when you forget to add "return" keyword while executing java script. Notice the "return" keyword in following statement –

```
((JavascriptExecutor) driver).executeScript("return window.title
;");
```

Question 145:
Are there any limitations from operating systems while using WebDriver?
Answer
While HTMLUnitDriver, FF Driver and Chrome Driver could be used on all operating systems, IE Driver could be used only with Windows.

Question 146:
Give me architectural overview of WebDriver.
Answer
WebDriver tries to simulate real user interaction as much as possible. This is the reason why WebDriver does not have "fireEvent" method and "getText" returns the text as a real user would see it. WebDriver implementation for a browser is driven by the language which is best to driver it. In case of FF best fit languages are Javascript in an XPCOM component and in IE it is C++ using IE automation. Now the implementation which is available to user is a thin wrapper around the implementation and user need not know about implementation.

Question 147:
What is Remote WebDriver Server?
Answer

Remote WebDriver Server has two components – client and server. Client is WebDriver while Server is java servlet. Once you have downloaded selenium-server-standalone-.jar file you can start it from command line as –

```
java -jar selenium-server-standalone-<version-number>.jar
```

Question 148:
Is there a way to start Remote WebDriver Server from my code?
Answer
First add Remote WebDriver jar in your class path. You also need another server called "Jetty" to use it. You can start sever as following –

```
WebAppContext context = new WebAppContext();
context.setContextPath("");
context.setWar(new File("."));
server.addHandler(context);

context.addServlet(DriverServlet.class, "/wd/*");

SelectChannelConnector connector = new SelectChannelConnector();
connector.setPort(3001);
server.addConnector(connector);

server.start();
```

Question 149:
But what are the advantages of using Remote WebDriver over WebDriver?
Answer
You can use Remote WebDriver when –
• When you want to execute tests on a browser not available to you locally
• Introduction to extra latency to tests
But there is one disadvantage of using Remote WebDriver that you would need external servlet container.

Question 150:
Can you show me code example of using Remote WebDriver?
Answer

```
    // Any driver could be used for test
DesiredCapabilities capabilities = new DesiredCapabilities();

        // Enable javascript support
        capabilities.setJavascriptEnabled(true);

        // Get driver handle
        WebDriver driver = new RemoteWebDriver(capabilities);

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

Question 151:
What are the modes of Remote WebDriver
Answer
Remote WebDriver has two modes of operations –
Client Mode: This is where language bindings connect to remote instance. FF drive and
RemoteWebDriver clients work this way.
Server Mode: In this mode language bindings set up the server. ChromeDriver works this way.

Question 152:
What Design Patterns could be used while using Selenium 2.0?
Answer
These three Design Patterns are very popular while writing Selenium 2.0 tests –

1. Page Objects – which abstracts UI of web page
2. Domain Specific Language – which tries to write tests which could be understood by a normal
   user having no technical knowledge
3. Bot Style Tests – it follows "command-like" test scripting

Question 153:
So do I need to follow these Design patterns while writing my tests?
Answer
Not at all, these Design Patterns are considered best practices and you can write you tests
without following any of those Design Patterns, or you may follow a Design Pattern which suites
your needs most.

Question 154:
Is there a way to enable java script while using HTMLUnitDriver?
Answer
Use this –

```
HtmlUnitDriver driver = new HtmlUnitDriver();
      driver.setJavascriptEnabled(true);
```

or this –
```
      HtmlUnitDriver driver = new HtmlUnitDriver(true);
```


Question 155:
Is it possible to emulate a browser with HTMLUnitDriver?

Answer

You can emulate browser while using HTMLUnitDriver but it is not recommended as applications are coded irrespective of browser you use. You could emulate Firefox 3 browser with HTMLUnitDriver as –

```
HtmlUnitDriver driver
= new HtmlUnitDriver(BrowserVersion.FIREFOX_3);
```

Or you can inject desired capabilities while instantiating HTMLUnitDriver as –

```
HtmlUnitDriver driver = new HtmlUnitDriver(capabilities);
```

Question 156:

How do I use iPhone Driver?

Answer

You should start iPhone SDK and build iPhone driver. Down load iPhone development tools and provision profile. Now iPhone driver can connect through HTTP to the iphone simulator. You can also run simulator on another machine in your network and WebDriver could connect to it remotely.

Question 157:

Is it possible to convert Selenium IDE test to WebDriver test?

Answer

~~For now there is no formatter available to convert Selenium IDE tests to corresponding WebDriver tests, hence simple answer is No.~~ Yes WebDriver style of code can be generated from Selenium IDE

Question 158:

Can WebDriver handle UntrustedSSLCertificates?

Answer

This feature is currently supported in Firefox browser and is awaiting implementation in IE and Chrome drivers.

Question 159:

Can I carry out multiple operations at once while using WebDriver?

Answer

You can use Builder pattern to achieve this. For example if you want to move an element from one place to another you can use this –

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

```
Action dragAndDrop = builder.clickAndHold(element)
        .moveToElement(otherElement)
        .release(otherElement)
        .build();

dragAndDrop.perform();
```

Question 160:
How do I simulate keyboard keys using WebDriver?
Answer
There is a KeyBoard interface which has three methods to support keyboard interaction –

- sendKeys(CharSequence)- Sends character sequence

- pressKey(Keys keyToPress) - Sends a key press without releasing it.

- releaseKey(Keys keyToRelease) - Releases a modifier key

Question 161:
What about Mouse Interaction?

Answer
Mouse interface lets you carry out following operations –
- click(WebElement element) – Clicks an element

- doubleClick(WebElement element) - Double-clicks an element.

- void mouseDown(WebElement element) - Holds down the left mouse button on an element.

- mouseUp(WebElement element) - Releases the mouse button on an element.

- mouseMove(WebElement element) - Moves element form current location to another element.

- contextClick(WebElement element) - Performs a context-click (right click) on an element.

Question 162:
How does Android Webdriver works?
Answer
Android WebDriver uses Remote WebDriver. Client Side is test code and Server side is application installed on android emulator or actual device. Here client and server communicate using JSON wire protocol consisting of Rest requests.

Question 163:

What are the advantages of using Android WebDriver?

Answer

Android web driver runs on Android browser which is best real user interaction. It also uses native touch events to emulated user interaction.

But there are some drawbacks also like, it is slower than headless WebKit driver. XPath is not natively supported in Android web view.

Question 164:

Is there a built-in DSL (domain specific language) support available in WebDriver?

Answer

There is not, but you can easily build your own DSL, for example instead of using –

```
webDriver.findElement(By.name("q")).sendKeys("Test");
```

You can create a more composite method and use it –

```
public static void findElementAndType(WebDriver webDriver,
String elementLocator, String testData) {

webDriver.findElement(By.name(elementLocator)).sendKeys(testData
);
      }
```

And now you just need to call method findElementAndType to do type operation.

Question 165:

What is grid2?

Answer

Grid2 is Selenium grid for Selenium 1 as well as WebDriver, This allows to –

- Execute tests on parallel on different machines
- Managing multiple environments from one point

Question 166:

How do I start hub and slaves machines in grid 2?

Answer

Navigate to you selenium server standalone jar download and execute following command –

```
java -jar selenium-server-standalone-.jar -role hub
```

And you start Slave machine by executing following command –

```
Java -jar selenium-server-.jar –role webdriver  -hub
http://localhost:4444/grid/register -port 6666
```

Question 167:
And how do I run tests on grid?
Answer
You need to use the RemoteWebDriver and the DesiredCapabilities object to define browser, version and platform for testing. Create Targeted browser capabilities as –

```
DesiredCapabilities capability = DesiredCapabilities.firefox();
```

Now pass capabilities to Remote WebDriver object –

```
WebDriver driver = new RemoteWebDriver(new
URL("http://localhost:4444/wd/hub"), capability);
```

Following this, hub will take care of assigning tests to a slave machine

Question 168:
What parameters can be passed to grid2?
Answer
You can pass following parameters to grid 2 –

- -port 4444 (default 4444)

- -nodeTimeout (default 30) the timeout in seconds before the hub automatically releases a node that hasn't received any requests for more than the specified number of seconds.

- -maxConcurrent 5 (5 is default) The maximum number of browsers that can run in parallel on the node.

Selenium Tool Implementation Misc Questions

Question 169:
How do I implement data driven testing using Selenium?
Answer
Selenium, unlike others commercial tools does not have any direct support for data driven testing. Your programming language would help you achieving this. You can you jxl library in case of java to read and write data from excel file. You can also use Data Driven Capabilities of TestNG to do data driven testing.
Question 170:
What is equivalent to test step, test scenario and test suite in Selenium.
Answer

If you are using Java client driver of Selenium then TestNG test method could be considered equivalent to test step, a test tag could be considered as test scenario and a suite tag could be considered equivalent to a test suite.

Question 171:
How do I get attribute value of an element in Selenium?

Answer
You could use getAttribute method
With Selenium 1.0 –

```
String var = selenium.getAttribute("css=input[name='q']@maxlength");
System.out.println(var);
```

With Selenium 2.0 –

```
String var =
webDriver.findElement(By.cssSelector("input[name='q']")).getAttribute(
"maxlength")
System.out.println(var);
```

Question 172:
How do I do database testing using Selenium?
Answer
Selenium does not support database testing but your language binding does. For example while using java client driver you can use java data base connectivity (jdbc) to establish connection to data base, fetch/write data to data base and doing data comparison with front end.

Question 173:
I completed test execution and now I want to email test report.
Answer
If you are using "ant" build tool then you can use "mail" task to deliver your test results. Similar capabilities are available in Continuous Build Integration tools like – Hudson.

Question 174:
How do I make my tests more comprehensible?
Answer
Selenium tests which are written as –

```
selenium.click("addForm:_ID74:_ID75:0:_ID79:0:box");
```

Make it tough to understand the element which is being exercised upon.
Instead of hard coding element locator in tests you should externalize them. For example with java you can use properties file to contain element locators and then locator reference is given in test script. Following this approach previous test step would look as –

```
selenium.click(PolicyCheckbox);
```

And this is far more comprehensible.

Question 175:
Why should I use Page Object?
Answer
Page object is a design pattern which distinguishes the code carrying out operations on page and code which carries out tests (assertion/verification). While implementing page object you abstract functioning of a page or part of it in a dedicated "Classs" which is then used by test script to perform actions on page and reach a stage when actual test could be performed.

Advantage of using page object is the fact that if application lay out changes then you only need to modify the navigation part and test would function intact.