1. How to configure TestNG in eclipse IDE?

* Three ways we can to install TestNG in eclipse
* We can to install testNG in eclipse directly from the eclipse market place
* It is the most easier and quicker way to install any third party plugin in eclipse including the TestNG plugin, In the input field write TestNG and press enter to search, once search completed then install button will appear and press it to continue installing the TestNG for eclipse
* We can to installing testNG plug-in in eclipse using the Install New Software
* In this case, from the main menu bar, select the help menu and click on the Install New Software, then will appear link and fill the link and click the add button, once again will appear small popup dialog for name of a repository and fill by TestNG, after pressing the ok button, eclipse will query the above link and list the available version of the TestNG plugin, now tick on the TestNG with desired version and the plugin manager will check for the required component and dependencies
* Also we can to Install testNG in eclipse via offline jar files
* Download the latest version of TestNG jar file, open eclipse -> right click on the project and go to property -> build path -> configure build path and add the testNG jar file in the libraries using add external jar button

1. How to configure TestNG in Intelejia?

* Since TestNG is supported natively from intellij-7 onwards, there is no need for additional configuration for intelliJ users.
* We know that TestNG has various IDE plugins, in that case of IntelliJ the first step is to open IntelliJ and go
* to setting>plugins to check if the TestNG plugin is available, a list of available plugins will appear with TestNG-J
* If not available, click ‘Install JetBrains plugin’ and then install TestNG-j plugin
* Once you configure TestNG then you have to configure the options for the set to tests that will be running
* Package : specify a package to run, all the tests in this package and bellow will be included
* Group: specify a group to run
* Suite: specify an external testng.xml file to run test
* Class: run all tests in a single class
* Method: run a single test method
* Once you have the configuration, you can run it

1. What is TestNG ?

* TestNG is a test automation framework for Java, inspired by JUnit and NUnit and developed to overcome their limitations. Its purpose is to cover all categories of test automation – unit testing, functional test, end-to-end, integration testing
* Eliminating most of the limitations of the older framework like JUnit
* Test automation framework gives the developer the ability to write more flexibility and powerful tests
* Some new functionalities that make it more powerful and easier to use, such as annotations and support for data driven testing

1. What are the advantages of TestNG?

* The main advantage of using TestNG framework in selenium is its ease of running multiple tests from multiple classes using just one configuration (we can also have many configurations, which depends upon how we design our test). Testng.xml is an XML file that describes the runtime definition of a test suite
* Also Using Testng.xml file we can run test available in one or more class file and make them as a single group
* TestNG allows us to create parallel tests
* TestNG has a wider rangs of setup annotations, including @Before/AfterSuite, @Before/AfterTest and @Before/AfterGroup
* A list of attributes that we can pass to annotations, we used priority attribute, that tells us what priority has the test method for execution
* One important attributes that can be passed are dataProvider (used for parameterized)
* Asserts, that helps us to verify the conditions of the test and decide whether it failed or passed
* Generates logs
* It is gives the ability to produce HTML Reports of execution

1. What are the benefits of using annotations?

* Annotations are those things in TestNG which guides it for what to do next or which method should be executed next
* We can pass additional parameters to annotations
* Annotations are strongly typed, so the compiler will flag any mistakes right away
* Test classes no longer need to extend anything

1. Difference between TestNG and JUnit?

* We can perform groups test using TestNG
* JUnit cannot perform groups test
* TestNG allows us to create parallel tests
* JUnit is not possible to parallel tests
* No need to extend any class in TestNG
* JUnit need to extend other class
* TestNG allows us to parameterized objects
* JUnit not allow
* TestNG allow us to used dependency
* JUnit not supported dependency

1. Can we run test without using testng.xml file?

* Yes, we can run, but once your test become larger and requires configurations of multiple test at one place, the best option is testng.xml

1. What is the testng.xml file hierarchy?

* The root tag of this file is <suite>
* <suite> tag can contain one or more <test> tags
* <test> tag can contain one or more <classes>tags
* <classes> tag can contain one or more <method> tags

1. TestNG Annotations?

* @Test - that test method is being defined
* @BeforeTest - method to be executed before executing set of tests
* @AfterTest – method to be executed after executing set of tests
* @BeforeMethod – method to be executed before every test method
* @AfterMethod – method to be executed after every test method
* @BeforeClass – method to be executed before executing all test methods in a test class
* @AfterClass – method to be executed after executing all test methods in a test class
* @BeforeSuite – method to be executed before all tests in the suite
* @AfterSuite – method to be executed after all tests in the suite
* @BeforeGroups – method to be executed before a test belonging to specific group
* @AfterGroups – method to be executed after a test belonging to specific group
* @DataProvider – data generation of test
* @parameters – pass parameters to tests
* @Factory
* @Listeners

1. How to install maven in eclipse?

* We can to install Maven in eclipse directly from the eclipse market place
* It is the most easier and quicker way to install any third party plugin in eclipse including the Maven plugin, In the input field write Maven and press enter to search, once search completed then install button will appear with m2e-connector for maven-dependency-plugin and press it to continue installing the Maven for eclipse
* We can do installing Maven plug-in in eclipse using the Install New Software
* In this case, from the main menu bar, select the help menu and click on the Install New Software, then will appear name and link and fill the link and click the add button, once again will appear small popup dialog for name of a repository and fill by Maven, after pressing the ok button, eclipse will query the above link and list the available version of the Maven plugin, now tick on the Maven and the plugin manager will check for the required component and dependencies

1. How to create a new maven project in eclipse?

* First of all, open eclipse IDE and go for creating new project from New -> Other, it will open new project creation window
* Expand maven folder in new project creation wizard window and select Maven Project and then click on Next button, it will take you to project name and location selection screen
* On project name and location selection screen, you can choose work space location for maven project by clicking on browse button, if wants to use default work space location then select that check box and click on Next button, It will take you to archetype selection screen
* On archetype selection screen, Select Maven-artchetype-quickstart option and click Next Button
* On next screen, Enter Group ID, Artifact Id, Package, It will create new maven project in eclipse with default AppTest.java class
* Once maven project is created then you can see there is pom.xml file under your project
* Now you need to add selenium webdriver and TestNG dependencies with latest version in pom.xml file

1. What is the groupId and artifactId in maven?

* Group Id : Generally groupId refers to domain id, For best practices company name is used as groupId, it identifies the project uniquely
* ArtifactId: It is basically the name of the jar without version
* Version: this tag is used to create a version of the project

1. What is maven?

* Maven is build management tool
* Maven allows the developer to automate the process of the creation of the initial folder structure, performing the compilation and testing and the packaging and deployment of the final product
* Maven is essentially a project management and compression tool and such provides a way to help managing
* Builds
* Documentation
* Reporting
* Dependencies
* Release
* Distribution

1. Why Maven is used?

* Maven simplifies and standardizes the project build process
* It handles compilation, distribution, documentation, team collaboration and other tasks seamlessly
* Maven increases reusability and takes care of most of build related tasks
* Maven helps us to building reports, creating jar and use of repositories to manage jar files

1. Explain what is maven repository?

* A maven repository is location where all the project jars, library jars, plugins or any other particular project related artifacts are stored and can be easily used by maven

1. What is maven local repository?

* This is the place where maven stores all the project jars files or libraries or dependencies, by default the folder name is .m2

1. What is maven central repository?

* Maven central repository is the default location ‘mavenrepository.com’ for maven to download all the project dependency libraries
* For any library required in the project

1. What is dependency keyword?

* Dependencies are the libraries, which are required by the project, for example log4j jars, apache poi jars, selenium jars etc.

1. What are surefire plugin?

* The surefire plugin is used during the test phase of the build lifecycle to execute the unit tests of an application
* It generates in 2 different file formats like plain text file, xml files and html files as well
* Also used in TestNG and JUnit for reporting

1. Why maven plugins are used?

* Maven plugins are used to
* Create a jar file
* create war file
* compile code files
* unit testing of code
* documenting projects
* reporting

1. Explain maven POM file?

* POM is project object model XML file that contains information about the project and configuration details used by maven to build the project
* It contains default values for most project
* Some of the configuration that can be specified in the POM are the project dependencies, the plugins or goals that can be executed, the build profile

1. What are difference between src/main/java and scr/test/java?

* Src/main/java is the root directory for source code
* Src/test/java is the root directory for source code,
* Maven knows that the tests will be found in the src/test directory and source code will be found in the src/main directory

1. Resource directory and target directory in maven?

* Resource directory contains all the resources needed by the project
* Target directory is created by maven, it contains all the compiled classes, JAR files etc

1. Briefly explain maven build life cycle?

* Clean: deletes all artifacts and targets which are created already
* Compile: used to compile the source code of the project
* Test: test the compile code and these tests do not require to be packages or deployed
* Package: package is used to convert your project into a jar file or war etc
* Install : install the package into local repository for use of other project

1. Maven dependency details?

* Selenium-java :
* To get all java bindings for all browsers we need to use selenium-java dependency
* TestNG :
* TestNG is a test automation framework for Java, inspired by JUnit and NUnit and developed to overcome their limitations. Its purpose is to cover all categories of test automation – unit testing, functional test, end-to-end, integration testing
* Test automation framework gives the developer the ability to write more flexibility and powerful tests
* Some new functionalities that make it more powerful and easier to use, such as annotations and support for data driven testing
* Log4j :
* Log4j is fast, reliable logging framework and can be easily integrated with the code, it is possible to enable logging at run time
* Logging is essential part of programming, it helps developer to track code workflow and fix bugs efficiently, if we get any issue in the code, we check logs for that workflow for essentially
* POI :
* Sometimes we need to read data from Microsoft Excel Files or we need to generate reports in Excel format
* Also in data driven framework all of our test data is generated from some external files like Excel, CVS, XML or some database table
* To read or write an excel, apache provides a very famous library POI, this library is capable enough to read and write both XLS AND XLSX file format of Excel
* Apache also provides java interfaces and classes in POI for reading XLS and XLSX file
* XLS Classes: HSSFWorkbook, HSSFSheet, HSSFRow, HSSFCell
* XLSX Classes: XSSFWorkbook, XSSFSheet, XSSFRow, XSSFCell
* Workbook: is a interface where XSSFWorkbook and HSSFWorkbook classes is implement
* XSSFWorkbook: is a class representation of XLSX file
* HSSFWorkbook: is a class representation of XLS file
* Sheet: is a interface where XSSFSheet and HSSFSheet classes in implement
* XSSFSheet: is a class representing a sheet in an XLSX file
* HSSFSheet: is a class representing a sheet in an XLS file
* Row: is a interface where XSSFROW and HSSFRow classes is implemet
* XSSFROW: is a class representing a row in the sheet of XLSX file
* HSSFROW: is a class representing a row in the sheet of XLS file
* Cell: is a interface where XSSFCell and HSSFCell classes is implement
* XSSFCell: is a class representing a cell in the row of XLSX file
* HSSFCell: is a class representing a cell in the row of XLS file
* POI-OOXML :

1. Mention the difference between apache ant and maven?

* Ant is a toolbox
* Maven is a framework
* Ant does not have formal conventions like project directory structure
* Maven has conventions
* Ant does not have lifecycle
* Maven has a life cycle
* Ant script are not reusable
* Maven plugins are reusable

1. Where do you find the class files when you compile a maven project?

* You will find the class files ${basedir}target/classes.

1. Explain what would the jar goal do?

* Jar will not recompile sources, it will imply just create a JAR from the target/classes directory considering that everything else has been done

1. For POM what are the minimum required elements?

* The minimum required elements for POM are project root, modelVersion, groupID, artifactID, and version

1. How to run a single test in maven?

* We can run a single test class repeatedly, to run this through maven, we have to setup the test property to a specific test case, we can do that from console using the bellow command
* Mvn-Dtest=TestCircle test
* The value for test parameter is the name of test class

1. How to run multiple tests in maven?

* We can run set of methods in a single test class repeatedly, to run this through maven, we have to setup the test property to a specific test case, we can do that from console using the bellow command
* Mvn-Dtest=TestCircle #test\* test
* The value for test parameter is the name of test class

1. What is the benefit of testNG test suite in selenium?

* A test suite is a collection of test cases plan or design to test a behavior or set of behaviors of software program
* In testNG we cannot define a suite in testNG source code but it is represented by one XML file, as suite is the feature of execution, it also allows flexible configuration of the tests to be run
* A suite can contain one or more tests and is defined by <suite> tag

1. What is the testng.xml file?

* XML file where we will configure our test to run, set test dependency, include or exclude any test, method, class or package and set priority
* It is handle multiple test classes

1. What is the benefit of build plugins in maven?

* Build plugin will be executed during the build and then , they should be configure in the < build /> element
* There are many plugins that do something with your build, for example the maven-compiler-plugin which allows you to set java version for your project
* I would say build plugin goal is a various action defined by a plugin

1. What are selenium Web-Driver / Selenium 2?

* Selenium web-driver software testing tool is well designed object oriented API is developed to automate web and mobile application testing process
* We can automate our web applications software testing process using selenium web-driver
* Some limitations of selenium RC has been overcome in selenium web-driver software testing tool
* Selenium web-driver supported most of the popular browser like Firefox, IE, Chrome, Opera driver etc

1. Tell me web-driver supported mobile testing drivers?

* Selenium supported mobile app testing drivers are
* Android Driver
* Opera Mobile Driver
* I-phone Driver

1. What are the benefits of test automation?

* Save time and money:
* Automation test scripts will run itself so required very less human efforts compared to manual testing
* Also we can test application in different browsers using same test scripts so it will decrease rework
* Improve testing accuracy:
* Sometimes human can make mistake but machine cannot
* Once you record script in correct way, tool will run it every time as expected so it will improve your testing accuracy
* Re-usability of test cases:
* You need to prepare automation test cases only one time, then you can use same test case for all upcoming version release of software application

1. Why selenium as automation tool?

* It is open source testing tool
* Able to operate it on almost all OS
* Selenium support multiple languages to create test scripts so you can choose your preferred language
* Selenium support multiple browsers
* Selenium allow us to integrate other open source unit testing frameworks like TestNG, JUnit, NUnit
* Able to create frameworks as per your requirement like keyword driven, data driven and hybrid etc
* We can integrate selenium with maven, maven is helpful to compile source code, set dependencies so that it will download required files automatically
* Also we can integrate Jenkins with selenium for continuous integration
* We can use selenium to test web application in mobile too

1. Dose selenium web-driver support record and playback facility?

* No, web-driver do not have any record and playback facility, but you can record your test in one of the selenium version called selenium IDE and then you can export your recorded tests in web-driver compatible format as per your preferred language

1. Which open source framework is supported in web-driver with Java?

* JUnit
* TestNG

1. Can you tell me the syntax to open/launch Firefox browser in web-driver software testing tool?

* WebDriver driver = new FirefoxDriver();

1. How many types of waits available in selenium web-driver?

* There are two types of waits available in selenium web-driver
* Implicit wait
* Explicit wait

1. What is implicit wait in selenium web-driver?

* Sometimes, elements are taking time to be appear on web application page, using implicit wait in web-driver software testing test case, we can poll the DOM for certain amount of time when some element or elements are not available immediately on webpage
* Driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

1. What is explicit wait in selenium web-driver?

* Using explicit wait we can define to wait for a certain condition to occur before proceeding further test code execution
* WebDriverWait wait = new WebDriverWait(driver, 20);
* Wait.until(ExpectedConditions.elementToBeClickable(By.xpath(“//input[@id=’abcd’]”)))

1. I want to pause my test execution for fix 10 seconds at specific point. How can I do it?

* We can use java.lang.Thread.sleep method to pause the software test execution for specific time, we can do that using following syntax
* Thread.sleep(10000);

1. Do you need selenium server to run your tests in selenium Web-Driver?

* It depends, if you are using only selenium web-driver API to run your tests and you are using all your tests on same machine then you do not need selenium server because in this case, web-driver can directly interact with browser using browsers native support
* When you are using remote or virtual to run web-driver tests for software web application and that machine have specific browser version that is not your current machine
* When you are using selenium-grid to distribute your web-drivers test execution on different remote or virtual machines

1. Can we automate desktop software application testing using selenium web-driver?

* No, this is the biggest disadvantage of selenium Web-driver API
* We can automate only web and mobile software applications testing using selenium Web-Driver

1. What is the difference between driver.get() and driver.navigate() methods?

* Driver.get() method is generally used for Open URL of web application
* It will wait till the whole page gets loaded
* Driver.navigate() method is generally used for navigate to URL web application, navigate back, navigate forward, refresh the page
* It will just navigate to the page but wait not wait till the whole page gets loaded

1. Selenium Grid?

* Grid allows us to execute your tests in parallel by distributing them on different machines having different browser/OS combinations

1. What is AJAX?

* Asynchronous JavaScript and XML is full form of AJAX which is used for creating dynamic web pages very fast for software web applications
* Using AJAX we can update page behind the scene by exchanging small amount of data with server Asynchronously, that means, using AJAX we can update page data without reloading page

1. How to handle Ajax in selenium Web-Driver?

* To handle Ajax call in selenium Web-Driver software automation tests, we need to use web-driver Fluent-Wait method which can wait for specific amount of time with specific condition

1. What kind of software testing is possible using selenium Web-Driver?

* Generally we are using selenium Web-Driver for functional and regression testing of web application

1. Why xPath locator is much more popular than all other locator types in web-driver?

* It is very easy to learn and understand for any new user
* There are many functions to build xPath in different ways like contains, starts-with etc
* Presently many tools and add-ons are available to find xPath of any element

1. Can we perform drag and drop in selenium? Tell me syntax

* Yes,
* New Action(driver).dragAndDrop(X, Y).build().perform();

1. Do you have faced any technical challenges with selenium Web-Driver software test automation?

* Yes, I have faced technical challenges during selenium web-driver test cases development
* Sometimes not always, some element like text box, buttons etc, are taking more time then given implicit wait to appear on page of web application or to get enable on page, in such situation if I have used only implicit wait then my test case can run fine on first run but it may fail to find element on second run, to handle this kind of situation we can use explicit wait
* Also handling dynamic changing ID to locate element is tricky. If elements ID is changing every time when you reload the software web application page, in that case you have to use Xpath to like [start-with@id, ‘post-body’ or contains[@id, ‘post-body’] in xPath
* If you have to execute your test case in multiple browsers then one test case can run successfully in Firefox but same test case may fail in IE due to the timing related issues, we can resolve this issue by Increasing implicit wait time when you run your test in IE browser
* Sometime handling JQuery elements like moving slider, date picker, drag and drop etc is tricky, you should have knowledge of web-driver advanced user interactions API to perform all these actions
* Also working with multiple windows, frames, extracting data from web table, extraxting data from dynamic table are tricky
* There is not any direct command to upload or download files from web page using selenium web-driver, for downloading and uploading files using selenium web-driver, you need to create and set Firefox browser profile with web-driver test case

1. Can you tell me syntax to close current web-driver and to close all opened web-driver instance?

* Yes, to close current web-driver instance, we can use close() method like driver.close();
* If there are opened multiple web-driver instance and wants to close all of them then we can use web-drivers wait() method, like driver.quit();

1. Is it possible to execute javaScript directly during rest automation? If yes then tell me how to generate alert by executing javaScript in web-driver sript?

* Yes, we can execute javaScript during web-driver test automation,
* To generate you can write
* JavaScriptExecutor javascript = (JavaScriptExecutor) driver;

Javascript.executeScript(“alert(‘JavaScript Executed. ’);”);

1. Give me syntax to read JavaScript alert message string, clicking on button and clicking on cancel button?

* We can read alert message string as bellow
* String alrtmsg = driver.switchTo().alert().getText();
* We can click on ok button using bellow syntax
* Driver.switchTo().alert().accept();
* We can click on cancel button using bellow syntax
* Driver.switchTo().alert().dismiss();

1. Tell me a scenario which we cannot automate in selenium web-driver?

* Bitmap comparison is not possible using selenium web-driver software testing tool
* Automating captcha is not possible
* We cannot read bar code using selenium web-driver testing tool

1. What is JUnit?

* JUnit as unit testing framework to write repeatable tests for java programming language, it is very simple and open source framework and we can use it in selenium web-driver test script creation to manage them in well manner

1. What is the usage of testng.xml file?

* In selenium web-driver software testing tool, we are using testng.xml file to configure our whole test suite in single file. Few of the tasks which we can specify is testng.xml file like
* We can define software testing test suite using set of test cases to run them from single place
* Can include or exclude test methods from web applications test execution
* Can specify a group to include or exclude
* Can pass parameter to use in test case of web application
* Can specify group dependencies
* Can configure parallel test execution for web application
* Can define listeners

1. How to pass parameter in testng.xml file to use it in test case?

* We can define parameter in testng.xml using one of the syntax
* <parameter name=”param” value=”valueOfparam”>
* Here, name attribute defines parameter name and value defines value of that parameter, then we can use that parameter in selenium web-driver automation test case using bellow syntax
* @Parameters({"param”})

1. I have a test with two @Test methods. I want to exclude one @Test method from execution, can I do it?

* Yes, you need to specify @Test method exclusion in testng.xml file
* <!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >
* <suite name="Test Exclusion Suite">
* <test name="Exclusion Test" >
* <classes>
* <class name="Your Test Class Name">
* <methods>
* <**exclude name="Your Test Method Name To Exclude"**/>
* </methods>
* </class>
* </classes>
* </test>
* </suite>
* You need to provide @Test method name in exclude tag to exclude it from execution

1. How to set priority of @Test method? What is its usage?

* In your web application test case, you can set priority for your TestNG @Test annotated using @Test(priority=0) and @Test(priority=1)
* Using priority, we can control @Test method execution manner as per our requirement, that means @Test method with priority = 0 will be executed 1st and @Test method with priority = 1 will be executed 2nd and so on.

1. Tell me any 5 assertions of TestNG which we can use in selenium web-drier software testing tool?

* There are many different assertions available in testNG but generally some assertion are commonly use like
* **Assert.assertEquals(actual, expected):** This assertion is useful to compare expected and actual values in selenium web-driver, if both values match then its fine and will continue execution, but if fails then immediately it will mark that specific test method as fail and exit from that test method
* **Assert.assertNotEquals(actual, expected, message):** In selenium web-driver is asserNotEquals assertion, its function is opposite to assertEquals assertion, means if both sides values will not match then this assertion will pass else it will fail, here we can write our own message for failure condition
* **Assert.assertTrue(condition):** Is generally used for Boolean condition true, it will pass if condition return true, if it will return false then it will fail and skip test execution from that specific method
* **Assert.assertFalse(cpndition):** It will check Boolean value returned by condition and will pass if returned value is false, if returned value is pass then this assertion will fail and skip execution from current test method
* **Assert.asertNull(object):** It will check and verify that object is null, it will pass if object found null, if object found not null then it will return error message
* **Assert.asertNotNull(object):** assertNotNull assertion is designed to check and verify that values returned by object is not null, means it will be pass if returned value is not null, else it will fail

1. Can you tell me usage of testNG soft assertion in selenium web-driver testing tool?

* Using testNg soft assertion, we can continue our test execution even if assertion fails, that means on failure of soft assertion, remaining part of @Test method will be executed and assertion failure will be reported at the end of @Test method

1. What is the difference between find-Element and find-Elements?

* Find-Element is useful to locate and return single element from page of web application
* Find-Elements is useful to locate and return multiple elements from web page

1. Can we capture screenshot in selenium web-driver testing tool? How?

* Yes, we can use selenium web-driver TakeScreenShot method to capture scrrenshot, java file class will be used to store screenshot in your system local drive,

1. Selenium web-driver has any built in method using which we can read from excel file?

* No, selenium web-driver testing tool do not have any built in functionality using which we can read data from excel file

1. Do you know any external API name using which we can read data from excel file?

* We can use one more powerful API known as Apache POI API to read and write data in excel

1. Tell me any 5 web-driver common exceptions which you faced during software test case execution?

* Time-Out-Exception – This exception will be thrown when command execution does not complete in given time
* No-Such-Element-Exception – web-driver testing tool will throw this exception when element could not be found on page of software web application
* No-Alert-Present-Exception - This exception will be generated when web-driver ties to switch to alert popup but there is not any alert present on page
* Element-Not-Select-Table-Exception – it will be thrown when web-driver is trying to select unelectable element
* Element-Not-Visible-Exception – thrown when web-driver is not able to interact with element which is available in DOM but it is hidden

1. Tell me different ways to verify element present or not on page?

* We can verify 2 simple ways
* Using size() method
* Boolean elePresent = driver.findElements( By.id("ID of element") ).size() != 0;
* If above syntax return false means element is not present on page and true means element is present on page
* Using isEmpty() method
* Boolean elePresent = driver.findElements(By.id("ID of element")).isEmpty();
* If above returns true means element is not present on page and false means element is present on page

1. What is difference between get-Attribute () and get-Text ()?

* Get-Attribute () method is useful to read software web app elements attribute value like id, name, type etc
* Get-Text () method is useful to read text from element or alert

1. I have total 200 test cases, I want to execute only 20 test cases out of them, can I do it in selenium web-driver, how?

* Yes, if you are using TestNG with selenium web-driver software testing tool then you can do is using grouping approach, create separate group for those 20 test cases and configure testng.xml file according to run only those 20 test cases
* Also if you are using data driven framework then you can configure it in excel file

1. Can you tell me three different ways to refresh page, do not use , refresh() method

* We can refresh browser in many different ways to like
* driver.get(driver.getCurrentUrl());
* driver.navigate().to(driver.getCurrentUrl());
* driver.findElement(By.xpath("//h1[@class='title']")).sendKeys(Keys.F5)

1. I wants to pass parameter in software test case through testng.xml file, how can I do it?

* You can use <parameter> node under <test> node in testng.xml file with parameter name and value, then you can use @parameters annotation with parameter name in you test case of software web application

1. I want to set size and position of my browser window; do you know how to do it in selenium web-driver?

* We can use set-Size function to set size of window and set-Position function to set position of browser window

1. Is there any way to get size and position of browser window in selenium web-driver?

* Yes, we can do it using web-driver function get-Size and get-Position

1. I want to scroll my software web-application page by 300 pixels, tell me how can I do it?

* We can use javascript executor with window.scrollBy(x,y) to scroll page in x and y directions

1. How to select dropdown menu in selenium?

* First of all we have to created object for select class
* Once object created then we can pass element using method also we can pass element according to index, value and visible text

1. How to verify common text in selenium?

* First of all we could use Array-List to store Expected value
* Then we have to initialized list web-element interface which is represents an HTML element and using list scanned the whole DOM and stored empty text
* Then we can use iterate through the DOM and only store verified text into empty text
* After that we could use again Array-List to stored Actual value
* Finally we can use assertion to compare actual value and expected value

1. How to get total links in selenium?

* First of all we have to initialized List Web-Element, we know that Web-Element represents an HTML element, List will be scanned the whole DOM and store empty text into a variable
* Then we can use assertion size () method for counting links

1. How to select child window in selenium and get back to parent window after performance?

* First we need to get parents window URL and perform the click operation that opens new window
* In order to get unique ids of all the browsers we use getWindowHandles() method which return set<String>, so that we can stored all the unique ids in a set,
* Then we can use iterator() method that returns an each element in the collection
* For the first time when next() method is used, we will get the first browser unique id of the browser for the first time
* Then switch to new window we can use switchTo().window(nameOfTheChildWindow)
* Then if we want to perform any operation in child window, we could that
* Also if we want to come back to parents window we have to used close() method and again we need to use SwitchTo.window(nameOfTheParents) windows

1. How to verify Collapsed and Expanded Object in Selenium?

* For Collapsed or expanded web element in selenium provided us one method is called – getCssValue() which can used to read the value of any web elements attribute style in selenium web-driver in java,
* Using that method we can verify collapsed element and also we can verify expanded element like what is the result before and after expanded
* Then we can use assertion to verify the test is failed or passed

1. How to mouse over In Selenium?

* First of all we have to initialized Web-element reference that will represent all the HTML element
* Then in selenium provide us one class is called actions that class contains the actions and the action classes that are executed when executing these events like mouse over or click or move
* So action class has a method is called moveToElement() that will helps the mouse to the middle of the element

1. How to count total number of I-Frame in a web-page?

* There are two ways to find total number of iFrames in a web page
* First we can do that by executing JavaScriptExecutor class and second is by finding number of web elements with a tag name of iFrame

1. How to handle iFrame/iFrames with selenium web-Driver?

* Basically, we can switch over the elements in iFrames using 3 ways
* By Index
* By Name or Id
* By Web Element
* Index is one of the attributes for the iFrame through which we can switch to it, index of the iFrame starts with 0, suppose if there are 100 frames in page, we can switch to the frame by using index
* Driver.switchTo().frame(0)
* Driver.switchTo().frame(1)
* It is impossible to click iFrame directly through XPath since it is an iFrame, first we have to switch to the frame and then we can click using XPath
* We initialize the Firefox driver
* Navigate to the site which consist the frame
* Maximize the window
* We need to find out the id of the iframe by inspecting through firebug
* Then switch to the iframe through ID
* Then we can click the element using web-driver command

1. How to clone or get repository in github?

* First things we have to go in github and click on the repository, suppose I want to clone a project name like PageObjectModel for that we need to get the URL, when we will click on the repository then we can see on the right corner clone or download and we do click on that then copy to clipboard
* Once done we go back to any ID like eclipse then click on right mouse and we can see there is an option name as import, click on the import and we would see search input will be appeared and search by github then click on project from git then click Next also we would see some will be appeared as clone URI click on that and click on Next button then we would see by default URL click on next and it will be downloaded code from the code, once done sometimes we would see is not an empty directory that mean that directory already exist so for that we need to delete that repository and click on next again and finished, so that way we would clone any repository

1. How to automate iFrame in selenium?

* We initiate the driver it could be Firefox or Chrome
* Navigate to the desired site which consist the iFrame
* Maximized the window
* Then we need to find out the ID or NAME or sometimes don’t have name so in that case we need find out the iFrame number of the iFrame in inspecting through Firebug
* Then switch to the iFrame through ID
* Then we need to find out the xpath of the element to be clicked

1. How to drag and drop in selenium?

* We can perform easily using action class likes drag and drop, mouse over, click and hold so on
* Webdriver will provided method called dragAndDrop(source, destination) which is a method of action class

1. How to handle alert box in selenium webdriver?

* In selenium webdriver provided alert interface who has some of methods that will help us to handle alert box, like to click on cancel button of the alert, to click on the Ok button of the alert, to capture the alert message

1. How to handle dynamic table in selenium webdriver?

* First get the entire table and store into a variable of type web element
* Get all the rows with tag name tr and store all the elements in a list of web element, now all the elements with tag tr are stored in rows list
* Then loop through each row and get the list of elements with tag will give you list of column for the row
* Using iterate get all the details of each cell

1. How does help selenium javascriptexecutor for us?

* JavaScriptExecutor is an interface which provides mechanism to execute Javascript through selenium driver
* We know that selenium has some limitation like if we want to type in selenium without using sendKeys() method, how to click in selenium if button or radio button is disable, how to uncheck checkbox in selenium if checkbox is disable and ect
* To handle those problem Inside this interface we have some predefined method called executeScript so whaterver we will pass as a string it will be executed by JavaScriptExecutor

1. How to filter in selenium?
2. How to verify disable button in selenium?

|  |  |
| --- | --- |
|  |  |