**Q 1. What is Selenium?**  
Selenium is a WebDriver automation tool that is used to test web-based applications via automating the test procedure. It supports various browsers, programming languages, and platforms.  
**Q 2. What is Automation Testing?**  
Automation Testing is a procedure for automating the manual steps or processes to test the application or software.  
It requires an additional testing tool to create test scripts that can be executed repeatedly.  
**Q 3. What are the advantages of automating the test procedure?**  
Advantages of automation testing are:

* Improvement in accuracy by eliminating human-generated errors
* Efficient and time-saving process
* Helps in testing a large application
* Tests can be repeatedly executed
* Allows another test to run simultaneously
* Provides test reports automatically

**Q 4. Name some automation tools which are used for application testing?**  
Automation tools used for application testing are

* Selenium
* TestingWhiz
* Katalon
* Tosca Testsuite
* TestComplete
* Ranorex

**Q 5. Why do we need software testing?**  
Once an application is developed, it is mandatory to check whether that application contains any errors or not.  
Software testing provides a tester to check for substantial errors or bugs in the application and resolves it.  
**Q 6. Explain different components of Selenium?**  
Selenium consists of four components:

* **Selenium Web Driver:**It is used for automation testing of web-based applications with the browser’s class method.
* **Selenium Integrated Development Environment (IDE):**It is a Firefox plugin used to record and playback test cases.
* **Selenium Remote Control (RC):**Selenium RC works on JavaScript to automate test procedures for web applications.
* **Selenium Grid:**It helps selenium to run different tests simultaneously.

**Q 7. What are the advantages of using Selenium?**  
Below are the advantages of using Selenium:

* It is an open-source automation tool that is available freely without any licensing cost.
* It supports multiple languages such as Java, Python, Ruby, etc.
* It also supports various browsers.
* It has a rich community that provides any type of information to any problem.
* It is user-friendly and even a beginner can easily understand and write the automated scripts for the test.

**Q 8. When will you choose to use Selenium Grid?**  
When there are multiple tests to be executed, we will use the selenium grid.  
It will enable us to run some test scripts on multiple platforms simultaneously and thus reducing the time consumption.  
**Q 9. Can you illustrate the drawbacks of Selenium?**  
Below are the drawbacks of Selenium:

* It cannot be used for desktop application testing.
* Also, it cannot perform tests on web services.
* To create vigorous scripts, knowledge of programming languages is required.
* External libraries are required to perform tasks in Selenium.

**Q 10. Name a few browsers that are supported by Selenium? Also, mention the name of the drivers.**  
Some common browsers supported by Selenium are:

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Browsers** | **Drivers** |
| **1.** | Google Chrome | Chrome Driver |
| **2.** | Mozilla Firefox | Firefox Driver |
| **3.** | Internet Explorer | Internet Explorer Driver |
| **4.** | Safari | SafariDriver |
| **5.** | HtmlUnit | HtmlUnitDriver |

**Q 11. Name the types of testing in Selenium?**  
Selenium supports the following types of testing that are:

* Regression Testing
* Functional Testing
* Load Testing

**Q 12. Name the different ways to find an element in a web page using Selenium?**  
Every object on a web page is referred to as an element in selenium. These can be found using different ways such as:

* ID
* CSS Selector
* Attribute
* Link text
* Xpath
* Name
* DOM
* ClassName
* Tag, etc.

**Q 13. Can you test APIs or web services using Selenium Web Driver and why?**  
No, we cannot test web services using Selenium. Web services are headless and it only uses browser’s class method to automate web applications. Thus it cannot automate web services.  
**Q 14. When will you choose to use Selenium IDE?**  
When there is repetition in the test procedure and we want to run the same sequence over and over again. Thus, Selenium IDE provides a record and playback feature that we can use to run tests repeatedly in the same sequence.  
**Q 15. What is X Path in Selenium?**  
X Path is a locator that uses an XML path to locate a web element in Selenium.  
It can also be used to locate HTML elements. X Path uses references from another element to find the specific element on a web page.  
**Q 16. Can you state the difference between Verify and Assert commands?**  
**Verify:**Verify command checks whether the given condition is true or not.  
Whatever the results, the program execution will not be halt. Even if the given condition is false, the program execution will not be stopped.  
**Assert:**Assert command also check whether the given condition is true or not.  
If the condition is true, the program will continue to execute to the next step. But, if the given condition is false, it will immediately halt the execution of the program.  
**Q 17. Can you state the difference between the use of a single slash (/) and a double slash (//) in X Path?**  
In X Path single slash is used to derive the absolute path from the root node whereas a double slash creates relative X Paths.  
**Q 18. What is the basic difference between absolute and relative X Path?**  
The basic difference between them is:  
**Absolute X Path:**It uses a complete path starting from the node of the root element to go to the desired element.  
**Relative X Path:**It only uses references from another element to go to the desired element.  
**Q 19.How will you launch the browser using Selenium Web Driver?**  
We will use the following syntax to launch the browser using Selenium Web Driver.

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Browser** | **Syntax** |
| **1.** | **Google Chrome** | WebDriver driver = **new**ChromeDirver(); |
| **2.** | **Mozilla Firefox** | WebDriver driver = **new**FirefoxDriver(); |
| **3.** | **Internet Explorer** | WebDriver driver = **new**InternetExplorerDriver(); |

**Q 20. Can you name the parameters which you have to pass in Selenium?**  
Yes, there are four parameters that need to be pass in Selenium. These are listed below:

* Port Number
* Host
* Browser
* URL

**Q 21. Have you ever automated test cases, If yes, how many per day?**  
Yes, I’ve automated test cases and on average I can automate 3-5 test cases each day. Although, these test cases can be sometimes complex and lengthy that can take a day to complete.  
**Q 22. Can we locate the elements by only using their text in XPath?**  
Yes, we can simply use the text() method to locate the element by using their text.

|  |
| --- |
| **Syntax:**xPathExpression = //\*[text()=’username’] |

**Q 23. Can you name the latest Selenium tool and its use?**  
WebDriver is the latest selenium tool used to automate web application testing and checks whether it is working as expected or not.  
**Q 24. How will you type in a textbox using Selenium and also give the syntax?**  
To enter the text in the textbox, we can use sendkeys(“Enter desired string”). The syntax is given below:

|  |
| --- |
| WebElement username = drv.findElement(By.id(“Email”)); // entering username username.sendKeys(“sth”); |

**Q 25. How will you find if an element is displayed on the screen?**  
Web  Driver lets the user check the visibility of the web elements including labels, radio buttons, checkboxes, drop boxes etc. with the following methods:

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Command** | **Syntax** |
| **1.** | **isEnabled()** | boolean searchIconEnabled = driver.findElement(By.id(“trial”)).isEnabled(); |
| **2.** | **isSelected()** | boolean buttonSelected = driver.findElement(By.id(“trial”)).isSelected(); |
| **3.** | **isDisplayed()** | boolean buttonPresence = driver.findElement(By.id(“trial”)).isDisplayed(); |

**Q 26. How can you launch different browsers in Selenium Web Driver?**  
We can easily use the given syntax to launch multiple browsers:

|  |
| --- |
| WebDriver driver = **new**FirefoxDriver(); |

**Q 27. Can you state the difference between driver.get(“URL”) and driver.navigate().to(“URL”) command? What is the use of them?**  
Both commands are similar and hence there is no difference between them.  
These commands are used to navigate to the URL which is declared in the command.  
**Q 28. How will you select the value in a drop-down?**  
We can easily select the value in a dropdown using the syntax given below:

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Command** | **Syntax** |
| **1.** | **selectByValue** | Select selectByValue = new Select(driver.findElement(By.id(“One”))); selectByValue.selectByValue(“greenvalue”) |
| **2.** | **selectByVisibleText** | Select selectByVisibleText = new Select (driver.findElement(By.id(“Two”))); selectByVisibleText.selectByVisibleText(“Lime”); |
| **3.** | **selectByIndex** | Select selectByIndex = new Select(driver.findElement(By.id(“Three”))); selectByIndex.selectByIndex(2); |

**Q 29. When will you use findElement() and findElements()?**  
**findElement():**This command is used to fetch the first element of the current webpage which is matching to the specified locator value in the syntax.

|  |
| --- |
| **Syntax:**WebElement element = driver.findElements(By.xpath(“//div[@id=’sample’]//ul//li”)); |

**findElements():**This command is used to fetch all the elements of the current webpage which is matching to the specified locator value in the syntax.

|  |
| --- |
| **Syntax:**List <WebElement> elementList = driver.findElements(By.xpath(“//div[@id=’sample’]//ul//li”)); |

**Q 30. Can you state the difference between driver.close() and driver.quit() command?**  
**driver.close():**This command is used to close the current web browser window opened by the user.  
**driver.quit():**This command is used to close all the web browser window opened by the user.

Both commands don’t need any parameters and do not return any value.  
**Q 31. Will Selenium be able to handle the Windows based pop up if it shows up?**  
No, Selenium is only meant for web-based applications thus it only supports web-based pop-ups.  
**Q 32. How will you assert the title on your web page?**  
With the use of given syntax, we can easily assert the title on a web page.

|  |
| --- |
| **Syntax:** assertTrue(“The title is incorrect.”,driver.getTitle().equals(“Title of the page”)); |

**Q 33. How will you clear the text which is written in a textbox?**  
We will use clear() command to clear the text which is written in a text box.

|  |
| --- |
| **Syntax:**driver.findElement(By.id(“elementLocator”)).clear(); |

**Q 34. Can you state the difference between regression and functional testing?**  
[Regression Testing](https://www.testbytes.net/regression-testing/)**:**Regression testing is a repeated test of an already tested program.  
Usually, it ensures to check the proper functioning of the application even if the minor modification is done that can create unexpected problems.  
**Functional Testing:**Functional testing usually ensures the functionality of the software program.  
The design and user experience remain untouched as the goal of functional testing is only to check the functionality of the program.  
**Q 35. What is the use of TestNG in Selenium?**  
[TestNG in Selenium](https://www.testbytes.net/blog/why-testng-awesome/) is used to cover a wide range of test categories such as functional, unit, end-to-end, etc.  
Using TestNG we can easily generate a proper report of test cases and can easily gather information on how many test cases were passed, failed, or skipped.  
**Q 36. Are you familiar with the term**[GeckoDriver in Selenium](https://www.testbytes.net/blog/selenium-with-gecko-driver/" \t "_blank)**? Why is it needed?**  
GeckoDriver is a web browser engine that forms a bridge between the Firefox browser and Selenium to interact with each other.  
GeckoDriver is needed in Selenium because, until Firefox version 47, Selenium uses the Firefox driver to interact with the browser. Now, Firefox has introduced a new version starting from version 48 that won’t allow any third party to directly interact with the browser.  
Thus, Selenium version 3 uses the driver to interact and run tests with the Firefox browser (version 48 onwards) which is GeckoDriver.  
**Q 37. Can we verify the image using Selenium?**  
No, we cannot verify the image in Selenium but we can easily verify whether the image is displayed or not using properties.  
**Q 38. Can you state the difference between ‘Type’ and ‘TypeAndWait’ command?**  
**Type:**When the user needs to enter the text into a text field, type command is used.  
**TypeAndWait:**This command is generally used to reload the web page as soon as the typing of the text is completed.  
**Q 39. What are the different types of mouse actions supported by Selenium?**  
Following are the various mouse actions that are supported by Selenium:

|  |  |
| --- | --- |
| **Sr.No** | **Syntax** |
| **1.** | click(WebElement element) |
| **2.** | contextClick(WebElement element) |
| **3.** | doubleClick(WebElement element) |
| **4.** | mouseUp(WebElement element) |
| **5.** | mouseDown(WebElement element) |
| **6.** | mouseMove(WebElement element) |
| **7.** | mouseMove(WebElement element, long xOffset, Long, yOffset |

**Q 40. Do you know who developed the Selenium and in which year?**  
It was [Jason Huggins](https://www.linkedin.com/in/jrhuggins/) who developed the Selenium in the year 2004.  
**Q 41. How will you take screenshots using Selenium?**  
We can easily take screenshot using TakeScreenShot function with getScreenshotAs() method. Example:

|  |
| --- |
| File scrFile = ((TakeScreenshot)driver).getScreenshotAs(screenshot.JPG); |

**Q 42. Can you go back and forth in the browser using Selenium?**  
Yes, we can easily use the following commands to move back and forth in the browser using Selenium:

* navigate().back()
* navigate().forward()

**Q 43. How would you delete the cookies using Selenium?**  
To delete the cookies, we will use the deletedAllCookies() command.

|  |
| --- |
| **Syntax:**driver.manage().deleteAllCookies(); |

**Q 44. Write the code to double-click an element?**

|  |
| --- |
| Actions action = new Actions(driver); WebElement element = driver.findElement(By.id(“elementId”)); action.doubleClick(element).perform(); |

**Q 45. Can you list all the navigation methods which are used in Selenium?**  
Yes, the following are the different navigation methods use in Selenium:

|  |  |
| --- | --- |
| **Sr. No.** | **Syntax** |
| **1.** | driver.navigate().to(String url); |
| **2.** | driver.navigate().refresh(); |
| **3.** | driver.navigate().forward(); |
| **4.** | driver.navigate().back(); |

**Q 46. How will you perform drag and drop in Selenium?**

|  |
| --- |
| Actions action = new Actions(WebDriver); action.dragAndDrop(sourceWE, destWE) |

**Q 47. Which command will you use to retrieve the color of an element in Selenium?**  
We will use the following command:

|  |
| --- |
| getCssValue(“Color”) getCssValue(“Background-color”) |

**Q 48. Which command will you use to copy the file from one location to another location?**  
We will use FileUtils.copyFile(srcLocation, destLocation) to copy the file from one location to another location.  
**Q 49. In what format the source view shows the script in Selenium IDE?**  
**The source** view shows the script in XML format.  
**Q 50. How will you verify whether the element is visible or not?**  
To verify whether the element is visible or not, we’ll use the following syntax:

|  |
| --- |
| WebElement e = driver.findElement(); boolean result = e.isDisplayed(); |

**Q 51. What are the exceptions available in the Selenium Web driver?**

Ans. Selenium web driver supports most of the exceptions available in another programming language. Some of these are:

* TimeoutException – it is triggered when an operation can not be performed within time.
* NoSuchElementException – it is triggered when an element with the mentioned properties is not found on the page.
* ElementNotVisibleException – It is triggered when the element is present in the page or DOM but its visible property is set to false.
* StaleElementException – it is triggered when the element is deleted from the DOM.

**Q 52. How do you navigate between different frames in Selenium?**  
Ans: switchTo() command can be used to switch between different frames in Selenium. It can be used to go to the default frame, parent frame and any specific frame based on the window ID or frame name as well.

**Q 53.How can you submit a form in Selenium?**  
Ans: There are 2 methods to submit a form using Selenium

* click() on the “Submit” button in the form
* submit() – submit method for the element itself.

Both work equally well.  
**Q54. What is the implicit and explicit wait?**  
Ans: Implicit wait directs the web driver to wait for a fixed time before throwing a “No Such Element Exception”. The default value is 0 and we need to set the wait time programmatically as below.  
*driver.manage().timeouts().implicitlyWait(TimeOut, TimeUnit.SECONDS);*  
Explicit wait, on the other hand, tells the web driver to wait for a certain condition before throwing the exception. These conditions can be element visible, element enabled, text present, title present etc.  
**Q55. What is a robot class in Selenium?**  
Ans: The robot class is used to perform actions using the keyboard or the mouse. It includes methods like KeyPress(), MouseMove(), MousePress() etc.  
**Q 56. When do you use robot class and methods?**  
Ans: Methods from the robot class can be used when the other functions do not work or there are issues with entering the value in the text box. Some examples:

* It can be used for swiping in the screens
* It can be used to type keys (shortcuts)

**Q 57. What are listeners in Selenium?**  
Ans: Listeners allow the customization of reports and log. They help analyze failures.  
There are 2 main listeners used with Selenium

* WebDriver Listener
* TestNG Listener

**Q 58. How can you get the size or set the size of a browser?**  
Ans: The maximize() method can be used to maximize the size of the browser.  
*driver.manage().window().maximize();*  
We can use getSize() and setSize methods to get and set the size of the browser.  
*System.out.println(driver.manage().window().getSize());*  
*Dimension d = new Dimension(420,600);*  
*driver.manage().window().setSize(d);*  
**Q 59. How can you upload a file in Selenium?**  
Ans: To upload the file, we first need to identify the web element and then we can just send the file path using the send\_keys method.  
*<input type=”file” name=”uploaded\_file” size=”50″ class=”pole\_plik”>*  
*element = driver.find\_element\_by\_id(”uploaded\_file”)*  
*element.send\_keys(“C:myfile.txt”)*  
**Q 60. Is it possible to skip a method or code block in TestNG? How?**  
Ans: Yes, it is possible to skip a method in TestNG. It can be done by setting the enabled parameter to false in the test annotation as below.  
*@Test(enabled = false)*  
**Q 61. What is a Group Test in TestNG?**  
Ans: Methods in TestNG can be categorized into groups. The speciality of these groups is that when a group is executed, all the methods in the group also get executed. A group can be executed by calling it in the @Test annotation.  
*@Test(groups={“xxx”})*  
**Q 62. How can you do database testing using Selenium?**  
Ans: Selenium does not support database testing, but it can still be used to do the testing partially with ODBC and JDBC connection strings.  
**Q 63. What is the latest version of Selenium?**  
Ans: The latest version is Selenium 4.0.0 Alpha 5, released in March of 2020.  
**Q 64. What is the “Same Origin Policy”?**  
Ans: The “Same Origin Policy” is like a security feature. It blocks scripts from other sites to access the contents of your site. According to this policy, the code loaded in a browser can only work within that domain and not outside.  
**Q 65. Name the JUnit annotations used with Selenium?**  
Ans: The JUnit annotations are:

* @Before: This method is executed before the main test. It is used to set the perquisites for the test and also for the initialization of the variables.
* @Test: Main test
* @After: This method is called after the completion of the main test. It is used to clear out the variables and kill the drivers. A clean up is what is done by the @After method.

**Q 66. What are the different types of locators available in Selenium?**  
Ans: The different type of locators are:

* id()
* name()
* tagName()
* className()
* linkText()
* partialLinkText()
* xpath()
* cssSelector()

**Q 67. What is a recovery scenario and how can you use it in Selenium?**  
Ans: A recovery scenario is used to recover from an error and continue with automation execution without the need for manual intervention.  
In Selenium, the recovery scenario used will be based on the programming language used. For Java, it can be done with the help of a simple “Try Catch Block”.  
**Q 68. What is a Selenese?**  
Ans: A Selenium set of commands used for running the tests is called Selenese.  
There are mainly 3 types of Selenese:

* Actions
* Assertions
* Accessors

**Q 69. How do you debug a test in Selenium?**  
Ans: You can debug a test by inserting a breakpoint. When you run the tests, the execution will pause at the breakpoint, from there you can run it one step at a time. You may also want to monitor the value of the variables at each step to debug and analyze the code.  
**Q 70. What is a regular expression and how is it used?**  
Ans: Regular expressions are like search strings which are used to search for a range of strings or patterns. In Selenium, this can be done with the help of the keyword “regexp” as a prefix with the search string.  
**Q 71. Can you work with multiple windows in Selenium? How?**  
Ans: Yes, we can work with multiple windows in Selenium. For switching between the windows we can use the selectWindow() method.  
**Q 72. When will use AutoIt with Selenium?**  
Ans: Selenium works well with web-based applications. It is not designed to handle windows pop-ups or non-HTML pop-ups. In cases where one needs to handle such pop-ups as part of the automation, we can make use of AutoIT tool.  
**Q 73. What function or method can be used to scroll up and down on the web page using Selenium?**  
Ans: scrollBy() and scrollIntoView() are the 2 methods that can be used to scroll through a long web page.  
*((JavascriptExecutor) driver).executeScript(“window.scrollBy(0,500)”);*  
*((JavascriptExecutor) driver).executeScript(“arguments[0].scrollIntoView();”, element);*  
**Q 74. How to press ALT/CTRL/SHIFT with other keys to perform special functions using Selenium?**  
Ans: In some cases, we may need to work with keyboard shortcuts using the ALT, CTRL, or SHIFT keys in combination with others. In such cases, we can keyUp() and keyDown() method. The keyDown() method assumes that the keys will remain pressed till the keyUp() method is called. This helps to automation a special combination key.  
*Actions builder = new Actions(driver);*  
*Action seriesOfActions = builder*  
*.moveToElement(txtUerName)*  
*.click()*  
*.keyDown(txtUserName, Keys.SHIFT)*  
*.sendKeys(txtUserName, “hello”)*  
*.keyUp(txtUserName, Keys.SHIFT)*  
*.doubleClick(txtUserName);*  
*.contextClick();*  
*.build();*  
*seriesOfActions.perform();*  
*}*  
**Q 75. What are the programming languages supported by Selenium?**  
Ans: Selenium supports:

* Java
* Python
* C-Sharp
* JavaScript
* Ruby
* PHP
* Perl

**Q 76. What is the main difference between xpath and css selectors?**  
Ans: With xpath you traverse in both directions, forward and backwards. But in the case of css you can traverse only in one direction, i.e. forward.  
**Q 77. Which are the open-source frameworks supported by Selenium?**  
Ans: Selenium can be easily integrated and widely used with the following frameworks:

* JUnit
* TestNG
* Maven
* FitNesse
* Xebium

**Q 78. Is it possible to perform a mouse hover using Selenium?**  
Ans: Actions class can be used to perform the hover over any object in the webpage. We can use the moveToElement() method for performing the hover.  
*actions.moveToElement(driver.findElement(By.id(“id of the object”))).perform();*  
**Q 79. What is POM (Page Object Model)?**  
Ans: Every webpage will have a page class that helps locate the page elements and perform the tasks on them. A POM is a design that makes use of this page class to build an object repository for these web elements. It helps to improve code reusability and readability.  
**Q 80. Can you use Selenium to automate captcha?**  
Ans: No, the idea of using the captcha is to prevent the use of the webpage using automation tools or bots. Also, Selenium is not very good when it comes to images.  
**Q 81. How can you handle authentication pop-up for Username and password in some web pages like SharePoint?**  
Ans. It can be done by first confirm the visibility of the authentication pop-up and then pass the authentication parameter using the alert class.  
*Alert alert = wait.until(ExpectedConditions.alertIsPresent());*  
*alert.authenticateUsing(new UserAndPassword(<username>, <password>));*  
**Q 82. What is the difference between typeKeys and type commands?**  
Ans: typeKeys is used to simulates a keystroke by collecting the key attributes using JavaScript. Type command on the other hand imitates the keypress just like a real user.  
**Q 83. What is the difference between setSpeed() and sleep() commands in Selenium?**  
Ans: Both the commands are used to alter the speed of execution of the automation script. In the case of sleep(), the execution is paused for a specific interval of time as passed in the method parameter. After this time elapses, the execution continues normally. In the case of setSpeed(), each line of code is delayed by a fixed time interval.  
**Q 84. What is a hub and node in Selenium?**  
Ans: A hub is a server or central point which can be used to run the automation scripts on different machines. The individual machines controlled by the hub are called nodes.  
**Q 85. Which WebDriver implementation is the fastest?**  
Ans. HTMLUnitDriver is the fastest WebDriver implementation because it does not execute the tests in the browser. It uses an HTML request and response mechanism for test case execution.  
**Q 86. What are the different commands that can be used to refresh the browser in Selenium?**  
Ans: There are multiple ways to refresh the browser, they are:

* *navigate().refresh()*
* *get(“url”) or driver.getCurrentUrl()*
* *navigate().to(“url”)*
* *sendKeys(Keys.F5)*

**Q 87. How can we handle hidden elements using WebDriver?**  
Ans: We can make use of the javaScript executor to work with the hidden objects as below:  
*(JavascriptExecutor(driver)).executeScript(“document.getElementsByClassName(ElementLocator).click();”);*  
**Q 88. How to find broken links on a web page?**  
Ans: In Selenium, we make use of request and response to find out the broken links on a web page. For that, we need to first collect all the links using the <a> tag. Once we collect these, we need to send an HTTP request to each and check the response received.  
Codes in the series of 400 and 500 indicate broken links.  
**Q 89. How can you handle the chrome browser notifications using Selenium?**  
Ans: To avoid the chrome browser notifications from popping up, we can programmatically disable them using Selenium.  
*ChromeOptions options = new ChromeOptions();*  
*options.addArguments(“disable-infobars”);*  
*WebDriver player = new ChromeDriver(options);*  
**Q 90.Name the listeners available in TestNG?**  
Ans: The different listeners available in TestNG are:

* IAnnotationTransformer
* IConfigurable
* IConfigurationListener
* IExecutionListener
* IHookable
* IInvokedMethodListener
* IInvokedMethodListener2
* IMethodInterceptor
* IReporter
* ISuiteListener
* ITestListener

**Q 91.How can you configure parallel execution using Selenium?**  
Ans: Parallel execution can be done in Selenium using the concept of hub and nodes. Hub controls the multiple nodes where the scripts are executed.  
**Q 92. How can you verify colour changes as verification points in Selenium?**  
Ans: For the color change type verifications, we can fetch the css color codes and verify.  
**Q 93. How can we move to the parent of an element using xpath?**  
Ans: “/..” is appended to the xpath of an expression to move to the parent element of the child element.  
**Q 94. How can we move to the nth-child position using xpaths?**  
Ans: There are 2 ways to navigate to the nth-child element.

* Using index within square brackets *div[2]*
* Using position(). *div[position()=2]*

**Q 95. How can you right-click an element in Selenium?**  
Ans: We can perform a right-click with the help of an action class. Here is the code.  
*Actions action = new Actions(driver);*  
*WebElement element=driver.findElement(By.id(“elementId”));*  
*action.contextClick(element).perform();*  
**Q 96. Name some TestNG annotations?**  
Ans: Some of the important TestNG annotations are:

* @Test – The test method
* @BeforeSuite – runs once before all the test in the test suites
* @AfterSuite – runs once after all the test in the test suites
* @BeforeClass – runs once before the first test method in the current class
* @AfterClass – runs once after all the test methods in the current class

**Q 97. What is the use of the testng.xml file?**  
Ans: The testing.xml is the file that is used for configuring the execution. This file enables the user to create test suites, test groups, indicate the tests to be run in parallel, details of the hubs and nodes etc. It is also used to pass the test parameters and trigger the test suite.  
**Q 98. What is the default priority of a test method in TestNG?**  
Ans: The default priority of a test method is 0. Now, if you have a test method with priority 1 and another test method with no priority mentioned, then the default value for the priority will be taken as 0 and it will be executed first.  
**Q 99. What is a properties file in Selenium?**  
Ans: It is a text file with key and value pairs. We can create a properties file with element name and element property. The benefit of using the property file is that if there is a change in the property (like a change in object name or text), we can simply make the change in the property file without having to touch the code.  
**Q 100. What are the DesiredCapabilities in Selenium WebDriver?**  
Ans: DesiredCapabilities file or simply the capability file contains key-value pairs and is used to set the properties for the WebDriver. This can be used to set the properties of the browser before launching like the version, cookie settings, size of the browser, and more.

Q 101. **What is headless browser testing?**

*Headless browser testing involves running automated tests without a visible browser window.*

Q 102**. What are the benefits of headless testing?**

* ***Resource efficiency:****No GUI rendering reduces resource consumption.*
* ***Cross-platform compatibility:****Tests run consistently across different platforms.*
* ***Faster execution:****Tests run faster without the overhead of a GUI.*
* ***Scalability:****Headless tests can be easily parallelized for faster execution.*

Q 10**3. What are the essential components for headless testing with Selenium in Java?**

* ***Selenium WebDriver JARs:****Provide libraries for interacting with web browsers.*
* ***ChromeDriver: A****WebDriver specifically designed for Chrome automation.*
* ***ChromeOptions:****Class for configuring ChromeDriver behavior.*

Q 10**4. How do I set up headless testing?**

* *Download Selenium WebDriver JARs and ChromeDriver for your platform.*
* *Set the system property “webdriver.chrome.driver” to point to the ChromeDriver location.*
* *Create a ChromeOptions instance and add the “headless” argument.*
* *Use the ChromeOptions instance to create a new ChromeDriver object.*

Q 10**5. How do I find and interact with elements in a headless test?**

* *Use the By class to locate elements by different locators like ID, name, CSS selector, etc.*
* *Use the WebDriver instance methods like findElement and click to interact with elements.*