



InterviewBit

Selenium Interview Questions for 5years Experience



To view the live version of the
page, [click here.](#)

© Copyright by Interviewbit

Contents

Selenium Interview Questions for Experienced (5+ years)

1. We know that we can maximize or minimize the browser window. However, we need to test the responsiveness of the UAT in different browser screen sizes. Can this be done in Selenium?
2. How to send ALT/SHIFT/CONTROL keys at once in Selenium WebDriver?
3. How can you upload a file using Selenium WebDriver?
4. What technique should you use throughout the script if there is neither a frame id nor a frame name?
5. You have written an automation code with both implicit and explicit wait commands. However, the elementNotVisible exception is still thrown. How do you fix this issue, without extending the wait timings in implicit or explicit waits?
6. How can you use regular expressions in Selenium?
7. You need to retrieve the message in an alert box, without using the getText() method. How will you handle this situation?
8. Consider a website that requires a username, password and a captcha for the user to log in. Your colleague suggests that this test execution can be redirected to the manual testing team? Do you agree with them? If yes, why?
9. Can you explain how you can handle colors in a web driver?
10. You are given a set of automated scripts that are highly confidential and must be run on the client's remote server only. You are asked to modify the scripts based on privacy needs, what will you do?
11. A fresher in your team encounters a 'StaleElement' exception while executing an automated test suite. How will you help them solve this issue?
12. How is network data capturing done in Selenium?
13. You have been asked to perform automation testing on a website that loads at a slow pace. You are able to execute the scenario manually without any impediments other than the speed issue. But while automation, NoSuchElementFound exception is thrown.

Selenium Interview Questions for Experienced (5+ years)

(.....Continued)

21. Your team needs to check if a warning message is displayed when the password entered by the user is moderately secure.
22. Which type command should you use if there is a reload event once your typing is completed?
23. The Selenium script runs in Chrome but not in IE. What can be done?
24. How do you write an Xpath locator to identify paragraph elements that are the immediate child of a div element, or the descendent of a div element?
25. How do you take a screenshot with Selenium Webdriver?

Let's get Started

If you have 5+ experience in the field of software testing and if automation testing is your niche, you are bound to know the in-depth concepts of the popular web automation testing tool [Selenium](#).

Are you an automation test engineer now wanting to switch to a different organization? Well, you have stumbled upon the right article then! You may have mastered Selenium, but are you prepared enough to attend a tricky and challenging interview? Since testing is a practical approach which requires quick thinking and a sharp eye, combined with great troubleshooting skills, you would need to come up with solutions that work the best for the UAT given to you.

Being an experienced professional, you would be a great guide to your juniors, once you master the art of providing a curated solution to their questions. For an experienced professional, questions in an interview are based on practical knowledge rather than theoretical concepts.

In this article, you will get a sneak peek at the popular Selenium interview questions for 5 years of experience and more with answers to grab your interviewer's attention for all the right reasons! This article will cover questions based on Selenium's advanced concepts like complex locating strategies, capabilities, javascript executors, different wait commands, and assert commands in Selenium. We have also included Selenium coding interview questions and scenario-based interview questions.

So, what are you waiting for? Start preparing for that job switch that you've dreamt of and ace the interview!

Selenium Interview Questions for Experienced (5+ years)

1. We know that we can maximize or minimize the browser window. However, we need to test the responsiveness of the UAT in different browser screen sizes. Can this be done in Selenium?

Yes, this can be done in Selenium. We can resize the browser window to a particular dimension using the `setSize()` method. We can customize the screen size by bypassing the desired width and height of the screen size in the parameters.

```
int width = 600;
int height = 400;
Dimension dimension = new Dimension(width, height);
driver.manage().window().setSize(dimension);
window.resizeTo() method and capability commands can also be used to achieve the same r
((JavascriptExecutor)driver).executeScript("window.resizeTo(600,400);");
```

2. How to send ALT/SHIFT/CONTROL keys at once in Selenium WebDriver?

Usually, we hold down these keys and click other buttons to achieve any unique functionality. To hold on to these keys, we need to define two more methods apart from `keys.ALT` or `keys.SHIFT` or `keys.CONTROL` functions. To include more than one key to be pressed or released, we can pass ALT, SHIFT, and CONTROL as parameters together in the modifier key argument.

Here is the syntax for the same:

```
Actions a = new Actions(driver)
a.keyDown(Keys.ALT).keyDown(Keys.SHIFT) .keyDown(Keys.CONTROL).sendKeys('test') .build()
```

3. How can you upload a file using Selenium WebDriver?

We can upload a file using the **element.sendKeys** (filepath) command. But before doing so, we need to use the HTML tag 'input' with the attribute type as a file. This helps Selenium identify the web element first and upload the file.

4. What technique should you use throughout the script if there is neither a frame id nor a frame name?

In this case, we can use the frame index. Let us consider a webpage that has 4 frames, where none of them have a frame id or name. Upon inspection of the frames, we can find the zero-based index of each of the frames, regardless of the presence of a name or id. The first frame would be at index '0', the second frame at index '1', and so on. Once the frame has been selected, all the subsequent calls on the selected web driver interface will be made to that frame.

```
driver.switchTo().frame(2);
```

Here, 2 is the index of the third frame.

5. You have written an automation code with both implicit and explicit wait commands. However, the `elementNotVisible` exception is still thrown. How do you fix this issue, without extending the wait timings in implicit or explicit waits?

In this case, we can use a command called fluent wait, which is an extension of the explicit wait command. With fluent wait, we can also set the frequency with which the condition is checked before throwing the `elementNotVisible` exception.

```
Wait wait = new FluentWait<WebDriver>(driver)
    .withTimeout(45, TimeUnit.SECONDS)
    .pollingevery(5, TimeUnit.SECONDS)
    .ignoring(NoSuchElementException.class);
```

While a timeout value is a time taken to wait for a condition, polling frequency is the frequency to check the success or failure of the condition.

6. How can you use regular expressions in Selenium?

Regexp is a keyword that can be prefixed to a text, to treat it as a regular expression in Selenium.

7. You need to retrieve the message in an alert box, without using the `getText()` method. How will you handle this situation?

`storeAlert()` command can be used to retrieve the message from the alert pop-up. The value is stored in a variable that can be returned using a `System.out.println` statement. You can also use `assertEquals()` to verify this message.

8. Consider a website that requires a username, password and a captcha for the user to log in. Your colleague suggests that this test execution can be redirected to the manual testing team? Do you agree with them? If yes, why?

Yes, I agree with my colleague's statement. Since the concept of a captcha is to ensure that no bots or automated scripts have access to sensitive information, Selenium does not support captcha automation. This is why, redirecting the test execution to the manual testing team is a feasible solution, as suggested by my colleague.

9. Can you explain how you can handle colors in a web driver?

We can use the command `getCSSValue(arg0)` to fetch the colours by sending the colour as the argument.

10. You are given a set of automated scripts that are highly confidential and must be run on the client's remote server only. You are asked to modify the scripts based on privacy needs, what will you do?

I understand that this scenario requires the tests to be run on a separate machine. Instead of using the WebDriver library and commands, I will use the RemoteWebDriver command to execute these tests. To run a remote WebDriver client, we need to configure it. This can be done by pointing the URL to the address of the remote server to execute the tests. Capabilities can be added to configure the tests further. Here is a sample code:

```
System.setProperty("Service_name", "selenium_java_client");
System.setProperty("Service_host", "localhost");
System.setProperty("Service_port", "14258");
ImmutableCapabilities = new ImmutableCapabilities("browserName", "chrome");
WebDriver driver=new RemoteWebDriver(new URL("http://sample.com"), capabilities);
```

11. A fresher in your team encounters a 'StaleElement' exception while executing an automated test suite. How will you help them solve this issue?

There are two causes for this exception. Element in the web page that was a part of the current instance has been refreshed or navigated to another web page. This causes the element to be destroyed or reconstructed. The second reason is that the element is no longer attached to the DOM. These are the breakthroughs that I will provide my team members with to solve this issue:

- You can try refreshing the webpage using the navigate command.

```
driver.navigate().refresh()
driver.findElement (By.xpath("sample")).click();
```

- You can use a try-catch block within a for loop if the element is not attached to the DOM

```
for(int i=0;i<=2;i++)
{ try { driver.findElement(By.xpath("sample")).click();
    break; }
  Catch (Exception e){
    Sysout(e.getMessage());
  }
}
```

- If you are sure that the element will be available, use the ExpectedConditions.refreshed method to avoid the StaleElementReference exception. This method updates the element by redrawing it and providing access to the referenced element.


```
wait.until(ExpectedConditions.refreshed(ExpectedConditions.stalenessOf("sample")));
```

12. How is network data capturing done in Selenium?

Network data capturing is done using an API in the default Selenium class. The response includes headers, status codes, timings, and AJAX requests. An instance, `Selenium.start("captureNetworkTraffic=true")` should be started before launching the application. After reaching a point where network traffic is to be captured, you can fire the method:

```
selenium.captureNetworkTraffic("xml");
```

You can also pass plain or JSON as a parameter to this method.

13. You have been asked to perform automation testing on a website that loads at a slow pace. You are able to execute the scenario manually without any impediments other than the speed issue. But while automation, `NoSuchElementException` is thrown.

Your test lead suggests making a change in the navigation commands. How will you handle this situation?

From my observation of the test scripts, I find that the `.navigateTo()` method has been used, which is not advisable in this case. `Driver.get` commands are perfect for a website that loads slowly, as it waits for the page to get loaded fully and then proceeds to the next steps. However, while using `driver.navigateTo()` method, the driver does not wait for the page to be loaded completely, and many components are loaded yet, which throws the `NoSuchElement` exception.

14. Which command takes you forward by one page in the browser history?

The **navigate().forward()** command takes you forward by one page (previous page) on the browser's history. This method enables the browser to click on the forward button in the existing browser window.

15. How will you enable a disabled textbox?

We can enable a disabled HTML element, be it a button, input field, text area, etc, by changing the attributes using the `.removeAttribute("disabled")` method. Here is a code to enable a text box using the Javascript executor.

```
String enable ="doc.getElementsByName("name").removeAttribute('disabled');"
javascript.executeScript(enable);
```

16. There are failures in your suite execution, where only a few test cases have passed. How will you optimize your upcoming test runs in Selenium?

In this case, re-running only the failed test cases would be an optimal solution. With the TestNG class, there are two methods to re-run only the failed test cases.

- In the test output folder, the testng-failed.xml suite file is generated. We can use this to run only the previously failed test cases.
- We can implement the TestNG IRetryAnalyzer interface. This can be done by creating a class (say RetryAnalyzer) and implementing IRetryAnalyzer. Next, create a class 'RetryListener' by implementing an 'IAnnotationTransformer' interface, where the transform method is called for every test during the test run. Later, include the mentioned listeners in the testng.xml file.

17. How can you debug the tests in the Selenium IDE?

Primarily, we need to insert the breakpoints where we need to debug and execute the test step by step. Next, run the test case, and we can find that the execution stops at the point where the breakpoint is kept. Click on the run button to continue executing the commands sequentially.

18. How to verify the specific position of a web element?

The commands “verifyElementPositionLeft”, “verifyElementPositionRight”, “verifyElementPositionTop”, “verifyElementPositionBottom” etc can be used to identify the position of the element with respect to the web page. The commands use pixel comparison to verify the positions.

19. How is desired capability useful in terms of Selenium?

A series of key-value pairs can be used to store information like the name and version of the browser, paths of browser drivers in the system, etc, to determine the browser's behaviour at run-time. This feature is called the desired capability. It is highly useful during cross-browser testing or testing across the different OS.

20. John is not sure if the warning text label is displayed on the web page of a blog. How will you help him using Selenium?

We can use the method `isDisplayed()` to confirm the presence of the element on the screen.

```
Boolean buttonPresence = driver.findElement(by.id("sample")).isDisplayed();
```

21. Your team needs to check if a warning message is displayed when the password entered by the user is moderately secure.

There are different datasets, where some passwords entered by the user are secure, so the warning message does not appear on the page. This should not halt the execution of the next steps in the test script. Can we use Assert commands to test this scenario?

No, we cannot use assert commands as it halts the further steps of the execution, in case of an assertion failure. Instead, we can use the verify command in this case. Verify commands, check the conditions if they are true or false, and also all the test phases get executed.

22. Which type command should you use if there is a reload event once your typing is completed?

The **typeAndWait** command can be used when there is a reload event in the software, post a typing action.

23. The Selenium script runs in Chrome but not in IE. What can be done?

The following steps can be done to fix this issue:

- Update the Selenium IE Driver
- Verify that the IE driver and working environment are compatible.
- Configure the IE driver with the setProperty method and import dependencies.
- Set the same value for the 'Enable Protected Mode' option for all zones from the Security tab
- Turn off the internet security settings in IE when running the script
- Use CSS Selectors to minimize exceptions
- Set a registry entry
- Avoid declaring the driver instance as static for running scripts on browsers parallelly.
- Use the latest Selenium jars
- Enable Javascript on the IE browser.
- Try using JavaScriptExecutor instead of a native click when clicking elements.

24. How do you write an Xpath locator to identify paragraph elements that are the immediate child of a div element, or the descendent of a div element?

An immediate child in XPath is indicated using "/". To find paragraph elements that are descendants to any div element. We can use "//" in the XPath.

25. How do you take a screenshot with Selenium Webdriver?

We can use the TakeScreenshot function to capture the screenshot and **getScreenshotAs()** command to save the screenshot.

Conclusion

In this article, we observe that many Selenium-based interview questions for 5+ years of experienced people are related to troubleshooting, using the right wait commands, exploring different options to verify a test case etc, generating reports etc. the Selenium scenario-based interview questions are mostly asked to test your prowess with the tool all these years. Dive into your practical knowledge of Selenium and practice various tricky scenarios to grab that job offer!

Useful Resources

- [Top Selenium Frameworks](#)
- [Selenium WebDriver Interview Questions](#)
- [Robot Framework Interview Questions](#)
- [RPA Interview Questions](#)
- [Automation Testing Interview Questions](#)
- [Complete Technical Interview Guide](#)

Links to More Interview Questions

[C Interview Questions](#)

[Php Interview Questions](#)

[C Sharp Interview Questions](#)

[Web Api Interview Questions](#)

[Hibernate Interview Questions](#)

[Node Js Interview Questions](#)

[Cpp Interview Questions](#)

[Oops Interview Questions](#)

[Devops Interview Questions](#)

[Machine Learning Interview Questions](#)

[Docker Interview Questions](#)

[Mysql Interview Questions](#)

[Css Interview Questions](#)

[Laravel Interview Questions](#)

[Asp Net Interview Questions](#)

[Django Interview Questions](#)

[Dot Net Interview Questions](#)

[Kubernetes Interview Questions](#)

[Operating System Interview Questions](#)

[React Native Interview Questions](#)

[Aws Interview Questions](#)

[Git Interview Questions](#)

[Java 8 Interview Questions](#)

[Mongodb Interview Questions](#)

[Dbms Interview Questions](#)

[Spring Boot Interview Questions](#)

[Power Bi Interview Questions](#)

[Pl Sql Interview Questions](#)

[Tableau Interview Questions](#)

[Linux Interview Questions](#)

[Ansible Interview Questions](#)

[Java Interview Questions](#)

[Jenkins Interview Questions](#)