Practical Questions for selenium

1. How would you automate the process of logging into a website using Selenium WebDriver in Python?
2. Write code to search for a product on an e-commerce website using Selenium WebDriver in Python.
3. How do you verify that a certain element is present on a webpage using Selenium WebDriver in Python?
4. Write a script to navigate through different pages of a website and capture the URLs using Selenium WebDriver in Python.
5. How do you handle dynamic elements that have changing IDs or classes using Selenium WebDriver in Python?
6. Write a script to select multiple checkboxes on a webpage using Selenium WebDriver in Python.
7. How do you handle dropdown menus with a large number of options using Selenium WebDriver in Python?
8. Write code to capture all the links present on a webpage using Selenium WebDriver in Python.
9. How do you handle browser cookies using Selenium WebDriver in Python?
10. Write a script to upload a file to a website using Selenium WebDriver in Python.
11. How would you verify that a specific text is displayed on a webpage using Selenium WebDriver in Python?
12. Write code to scroll down to the bottom of a webpage using Selenium WebDriver in Python.
13. How do you handle pop-up windows that appear during website navigation using Selenium WebDriver in Python?
14. Write a script to capture a screenshot of a webpage using Selenium WebDriver in Python.
15. How would you handle alerts that appear after certain actions on a webpage using Selenium WebDriver in Python?
16. Write code to perform mouse hover actions on an element using Selenium WebDriver in Python.
17. How do you handle iframes embedded within a webpage using Selenium WebDriver in Python?
18. Write a script to capture the text from all the elements of a specific class on a webpage using Selenium WebDriver in Python.
19. How would you simulate keyboard actions such as pressing Enter or Tab using Selenium WebDriver in Python?
20. Write code to switch between different windows or tabs opened by the browser using Selenium WebDriver in Python.
21. How do you handle delays or timeouts while waiting for elements to load using Selenium WebDriver in Python?
22. Write a script to extract data from a table on a webpage using Selenium WebDriver in Python.
23. How would you handle the scenario of testing a website across multiple browsers using Selenium WebDriver in Python?
24. Write code to verify the presence of a specific attribute in an element using Selenium WebDriver in Python.
25. How do you handle the scenario of testing a website on different devices/resolutions using Selenium WebDriver in Python?
26. Write a script to perform drag and drop operations on a webpage using Selenium WebDriver in Python.
27. How would you handle the scenario of testing a website in different languages using Selenium WebDriver in Python?
28. Write code to capture the console logs from a webpage using Selenium WebDriver in Python.
29. How do you handle SSL certificate errors that may occur while accessing a webpage using Selenium WebDriver in Python?
30. Write a script to capture all the images present on a webpage using Selenium WebDriver in Python.
31. How would you handle the scenario of testing a website with different user roles using Selenium WebDriver in Python?
32. Write code to simulate browser navigation actions such as back, forward, and refresh using Selenium WebDriver in Python.
33. How do you handle the scenario of testing a website with different data sets using Selenium WebDriver in Python?
34. Write a script to verify the visibility of an element on a webpage using Selenium WebDriver in Python.
35. How would you handle the scenario of testing a website with different network conditions using Selenium WebDriver in Python?
36. Write code to capture network traffic generated by a webpage using Selenium WebDriver in Python.
37. How do you handle the scenario of testing a website with different time zones using Selenium WebDriver in Python?
38. Write a script to verify the responsiveness of a webpage by resizing the browser window using Selenium WebDriver in Python.
39. How would you handle the scenario of testing a website with different browser configurations using Selenium WebDriver in Python?
40. Write code to perform assertions on the attributes of an element using Selenium WebDriver in Python.
41. How do you handle the scenario of testing a website with different operating systems using Selenium WebDriver in Python?
42. Write a script to interact with JavaScript alerts, prompts, and confirmations using Selenium WebDriver in Python.
43. How would you handle the scenario of testing a website with different screen resolutions using Selenium WebDriver in Python?
44. Write code to simulate user interactions such as scrolling, zooming, and mouse movements using Selenium WebDriver in Python.
45. How do you handle the scenario of testing a website with different browser versions using Selenium WebDriver in Python?
46. Write a script to verify the correctness of form submissions on a webpage using Selenium WebDriver in Python.
47. How would you handle the scenario of testing a website with different accessibility requirements using Selenium WebDriver in Python?
48. Write code to handle dynamic elements that load asynchronously on a webpage using Selenium WebDriver in Python.
49. How do you handle the scenario of testing a website with different security configurations using Selenium WebDriver in Python?
50. Write a script to automate the entire checkout process on an e-commerce website using Selenium WebDriver in Python.

**Interview Questions**

1. What is Selenium WebDriver and why is it used in Python?
2. How do you install Selenium WebDriver for Python?
3. Explain the concept of browser automation using Selenium in Python.
4. How do you launch a browser using Selenium WebDriver in Python?
5. What are the different types of locators available in Selenium WebDriver for Python?
6. How do you find an element by ID using Selenium in Python?
7. Explain how to locate elements by class name in Selenium WebDriver with Python.
8. What is the purpose of XPath in Selenium WebDriver for Python, and how do you use it?
9. How do you find elements by tag name using Selenium WebDriver in Python?
10. Explain the use of CSS selectors for locating elements in Selenium WebDriver with Python.
11. How do you interact with web elements using Selenium WebDriver in Python?
12. What is the purpose of WebDriverWait in Selenium WebDriver for Python, and how do you use it?
13. How do you handle browser navigation (back, forward, refresh) using Selenium in Python?
14. Explain how to perform keyboard actions such as typing and pressing keys using Selenium WebDriver in Python.
15. How do you handle mouse actions like clicking, double-clicking, and right-clicking using Selenium in Python?
16. Explain how to handle alerts and pop-ups using Selenium WebDriver in Python.
17. How do you handle frames and iframes using Selenium WebDriver in Python?
18. What is the Page Object Model (POM), and how do you implement it in Selenium WebDriver with Python?
19. How do you capture screenshots of web pages using Selenium WebDriver in Python?
20. What is headless browsing, and how do you perform it using Selenium WebDriver in Python?
21. How do you handle dropdowns and select elements in Selenium WebDriver with Python?
22. Explain the concept of implicit and explicit waits in Selenium WebDriver for Python.
23. How do you perform drag and drop operations using Selenium WebDriver in Python?
24. What are the advantages of using Pytest with Selenium WebDriver?
25. How do you handle file uploads using Selenium WebDriver in Python?
26. Explain how to handle cookies using Selenium WebDriver in Python.
27. How do you switch between windows and tabs using Selenium WebDriver in Python?
28. What are desired capabilities in Selenium WebDriver for Python, and how do you use them?
29. How do you handle SSL certificate errors using Selenium WebDriver in Python?
30. What is the purpose of browser profiles in Selenium WebDriver, and how do you use them in Python?
31. How do you handle dynamic elements and AJAX calls using Selenium WebDriver in Python?
32. Explain how to perform scrolling and scrolling to specific elements using Selenium WebDriver in Python.
33. What is the purpose of the ActionChains class in Selenium WebDriver for Python?
34. How do you perform parallel testing using Selenium WebDriver in Python?
35. How do you handle browser notifications using Selenium WebDriver in Python?
36. Explain how to capture network traffic and analyze it using Selenium WebDriver in Python.
37. How do you handle window size and position using Selenium WebDriver in Python?
38. What are the limitations of Selenium WebDriver in Python?
39. How do you handle browser zoom levels using Selenium WebDriver in Python?
40. What is GeckoDriver, and when is it used with Selenium WebDriver in Python?
41. How do you handle authentication pop-ups using Selenium WebDriver in Python?
42. What is the difference between driver.get() and driver.navigate().to() methods in Selenium WebDriver for Python?
43. How do you handle synchronization issues and waits in Selenium WebDriver with Python?
44. Explain how to perform cross-browser testing using Selenium WebDriver in Python.
45. How do you handle dynamic dropdowns and auto-suggestions using Selenium WebDriver in Python?
46. What are the advantages of using the Python programming language with Selenium WebDriver?
47. How do you capture console logs and error messages using Selenium WebDriver in Python?
48. How do you simulate user interactions such as scrolling, zooming, and mouse movements using Selenium WebDriver in Python?
49. Explain how to execute JavaScript code using Selenium WebDriver in Python.
50. How do you integrate Selenium WebDriver tests with continuous integration tools like Jenkins in Python?