**1) What is Automation Testing?**

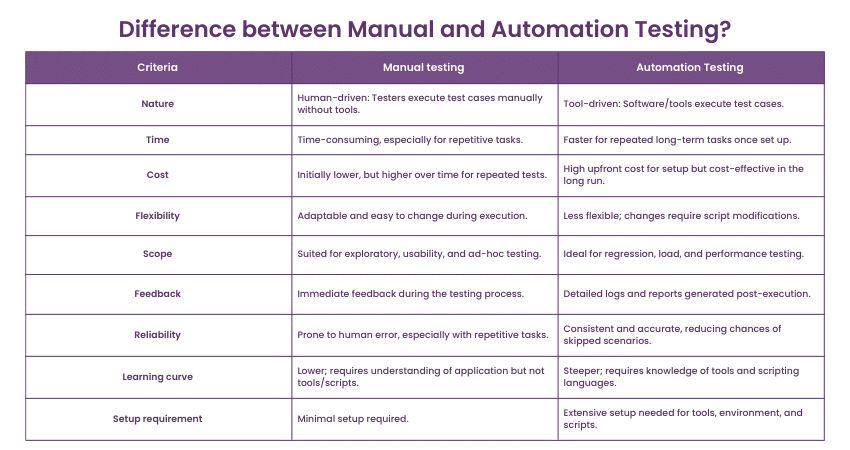
**Answer:**It involves using scripts and software tools to Test the software and then comparing the actual outcomes with expected results.

**2) Why is Automation Testing important?**

**Answer:** It increases the speed and accuracy of Testing, allows for frequent execution of Test cases, supports Regression Testing, and is suitable for load and Performance Testing.

**3) What is the difference between Manual and Automation Testing?**

**Answer:** Manual Testing involves human intervention in executing the Test cases and verifying the results, while Automation Testing uses scripts and tools to execute Test cases.



**4) Can you explain the Automation Testing lifecycle?**

**Answer:** The Automation Testing lifecycle involves phases like requirement analysis, Test planning, Test script development, Test execution, and result analysis.

**5) What are the key components of Automation Testing?**

**Answer:** The key components include the Automation tool, Test script, Test data, and the application under Test.

**6) Which Tests can be automated?**

**Answer:** Tests that are repetitive, time-consuming, and require multiple data sets are ideal  for Automation.

**7) Which Tests cannot be automated?**

**Answer:** Tests that are newly designed and not executed manually at least once and Tests that require constant human observation cannot be automated.

*Sharpen your Manual Testing skills with our*[***Manual Testing Training***](https://www.theknowledgeacademy.com/courses/software-testing-training/manual-testing-training/)*– Sign up today!*

**8) What are some popular tools used for Automation Testing?**

**Answer:** Tools like Selenium, TestComplete, Katalon Studio, and Appium are widely used in Automation Testing.

**9) What is Selenium?**

**Answer:** Selenium is a popular open-source tool used for automating web browsers.

**10) What are the advantages of Selenium?**

**Answer:** Selenium supports multiple programming languages, browsers, and platforms. It also allows parallel Test execution and integrates with tools like TestNG and JUnit for managing Test cases and generating reports.

**11) What is a Test script?**

**Answer:** A Test script is a set of instructions written in a programming language to run a Test on a software application.

**12) What is a Test suite?**

**Answer:**A Test suite is a collection of Test cases that are required to Test a software program to ensure it meets the specified requirements.

**13) What is regression Testing?**

**Answer:** Regression Testing is performed to ensure that a recent program or code change hasn't adversely affected existing features.

**14) What is the difference between Selenium and QTP?**

**Answer:** While Selenium is an open-source tool used for web application Testing, QTP (Quick Test Professional), now known as UFT (Unified Functional Testing), is a commercial tool used for desktop, web, and mobile application Testing.

**15) What is keyword-driven testing?**

**Answer:** Keyword-driven Testing is an Automation Testing framework which separates the Test creation process from the actual Test execution.

**16) What is Data Driven Testing?**

**Answer:** Data-driven Testing is a framework where Test input and output values are examined from data files (databases, excel files, CSV files, etc.) and are loaded into variables in captured or manually coded scripts.

**17) What is the difference between Keyword Driven and Data Driven Testing?**

**Answer:** While Keyword Driven Testing separates scripting from Test data, Data Driven Testing separates Test instructions from Test data.

**18) What is a hybrid Testing framework?**

**Answer:** It's a combination of more than one Testing framework methodologies, i.e., Data Driven, Keyword Driven, and Modular Driven frameworks.

**19) What is TestNG, and how is it better than JUnit?**

**Answer:** TestNG is a Testing framework inspired by JUnit. It is more powerful and easier to use than JUnit. It supports parallel execution of Test methods, flexible Test configuration, support for data-driven Testing, and a powerful execution model.

**20) What is JUnit?**

**Answer:**JUnit is a unit Testing structure for Java. It plays a crucial role in the development of Test-driven development frameworks.

**21) What are the various types of frameworks?**

Answer: The various types of frameworks are Data Driven Testing Framework, Keyword Driven Testing Framework, Hybrid Testing Framework, and Behaviour Driven Development Framework (BDD).

**22) What is the role of an Automation Tester?**

**Answer:** The role of an Automation Tester involves the creation, execution, and maintenance of automated scripts, as well as reporting the results of automated Tests.

**23) What is a Locator in Selenium?**

**Answer:**A Locator in Selenium is used to find and match the elements of a webpage that we want to interact with.

**24) Name the types of Locators in Selenium.**

**Answer:** The types of Locators in Selenium include the following : ID, Name, Class, Tag Name, Link Text, Partial Link Text, CSS Selector, and XPath.

**25) What is XPath?**

**Answer:**XPath is a language used for navigating through features and attributes in an XML document. It's used to find the location of any features on a webpage using HTML DOM structure.

**26) What is the Test Automation Pyramid?**

**Answer:**The Test Automation Pyramid is a method used for determining how to effectively apply different Types of Automation Testing  methods. It was first developed and released by Martin Fowler in the year 2012. The fundamental principle behind it is to have as many User Interface (UI) unit Tests as possible with fewer end-to-end Tests. This helps in prioritising fast and easy-to-run Unit Tests over the complex and expensive end-to-end Tests.

**27) What is browser automation?**

**Answer:**Browser Automation is the process of using Software tools to automate repetitive tasks in a web browser. This helps you carry out certain tasks with increased precision and accuracy, as well as control your web browser.

**28) What is a Test environment?**

**Answer:**A Test environment is nothing but a controlled and isolated setup used for Testing Software applications and codes. It is an extremely important component in the Software Testing process as it can simulate real-world scenarios and infrastructure. From local development environments to production-oriented setups, they vary greatly in terms of scope and complexity. Let’s explore the benefits of using a Test environment:

1) Ensure software quality

2) Minimise production defects

3) Provide a safe space for Testing

**What are some Automation Testing Interview Questions for experienced professionals?**

For our working professional readers, if you are looking for a change or aiming for a better company, you can take a glance at this set of Automation Testing Interview Questions. These Automation Testing Interview Questions for experienced professional focuses more on the depth of industrial technical knowledge. These questions have been provided with some expert answers for your ease. Let’s explore these questions with the proper answers:

**1) What is the difference between Selenium and QTP?**

**Answer:** Selenium is open source, supports multiple languages and browsers, and is suitable for web application Testing. QTP (now UFT) is a commercial tool, supports fewer languages, and is suitable for both web and desktop application Testing.

**2) How can you handle frames in Selenium WebDriver?**

**Answer:**Frames can be handled using the switchTo().frame() method in Selenium WebDriver.

**3) What is the role of TestNG in Automation Testing?**

**Answer:**TestNG provides better structure and flexibility to Tests, allows parallel Test execution, generates Test reports, and supports data-driven Testing.

**4) What is a Page Object Model?**

**Answer:**Page Object Model is a design pattern where each web page is represented as a class. This allows better maintainability and reusability of code.

**5) How would you automate a CAPTCHA?**

**Answer:**CAPTCHA is designed to prevent automation; hence, it should not be automated. The ideal approach is to bypass it during Test execution.

**6) How can you handle AJAX calls in Selenium?**

**Answer:** AJAX calls can be handled using WebDriverWait and ExpectedConditions classes in Selenium.

**7) What is XPath, and how do you use it?**

**Answer:** XPath is a language used to navigate through elements in an XML document. It is used in Selenium to locate elements in a webpage.

**8) How can you handle Windows-based pop-ups in Automation Testing?**

**Answer:**Windows-based pop-ups can be handled using tools like AutoIT and Robot class in Java.

**9) Explain the concept of Keyword Driven Framework.**

**Answer:** In Keyword Driven Framework, each function on a webpage is represented by a keyword. The Test data and keywords are stored separately and are combined during Test execution.

**10) What are the different types of waits in Selenium?**

**Answer:**The different types of waits in Selenium are Implicit Wait, Explicit Wait, and Fluent Wait.

**11) How can you handle alerts or pop-ups using Selenium WebDriver?**

**Answer:** Alerts or pop-ups can be handled using the switchTo()—alert () method in Selenium WebDriver.

**12) What is the difference between driver.close() and driver.quit() methods in Selenium?**

**Answer:**driver. close() closes the current browser window, while driver.quit() closes all the browser windows opened by the WebDriver.

**13) What is the significance of the Desired Capabilities class in Selenium WebDriver?**

**Answer:** Desired Capabilities are used to set the properties of the browser or platform used for Testing.

**14) How can you achieve Parallel Testing using Selenium WebDriver?**

**Answer:**Parallel Testing can be achieved using TestNG's parallel execution feature or Selenium Grid.

**15) Explain the Stale Element Reference Exception.**

**Answer:**This exception occurs when an element is interacted with after it has been deleted or modified.

**16) What is the Robot Framework?**

**Answer:**Robot Framework is an open-source Automation Testing framework that supports both keyword-driven and Data Driven Testing.

**17) How can you handle Dropdowns in Selenium?**

**Answer:** Dropdowns can be handled using the Select class in Selenium WebDriver.

**18) Explain the difference between Assert and Verify in Selenium.**

**Answer:**Assert stops the Test execution if the condition is false, while Verify continues the Test execution even if the condition is false.

**19) What is Cucumber, and how is it used in Automation Testing?**

**Answer:** Cucumber is a tool used for Behaviour-Driven Development. It allows writing Tests in a natural language that can be understood by non-technical stakeholders.

**20) How can you handle dynamic web elements?**

**Answer:** Dynamic web elements can be handled using dynamic XPath or CSS Selectors.

**21) What is the role of Jenkins in Automation Testing?**

**Answer:**Jenkins is a Continuous Integration tool used to develop and Test software projects continuously and to integrate changes to the project.

**22) What is the use of the @FindBy annotation in Selenium?**

**Answer:** The @FindBy annotation is used in Page Factory to locate elements on a webpage.

**23) How can you handle multiple windows in Selenium WebDriver?**

**Answer:** Multiple windows can be handled using the getWindowHandles() and switchTo().window() methods in Selenium WebDriver.

**24) What is the use of the JavaScriptExecutor interface in Selenium WebDriver?**

**Answer:** The JavaScriptExecutor interface is used to execute JavaScript code in Selenium WebDriver.

**25) How can you handle cookies in Selenium WebDriver?**

**Answer:**Cookies can be handled using the addCookie(), getCookieNamed(), getCookies(), and deleteCookie() methods in Selenium WebDriver.

**26) What is CAPTCHA?**

**Answer:**A completely Automated Public Turing Test to tell Computers and Humans Apart (CAPTCHA) is a security technique. It is primarily used for distinguishing human users and automated bots on websites and online platforms. CAPTCHAs can prevent automated bots from engaging in harmful activities like the following:

1) Spamming

2) Brute force attacks

3) Web scraping

4) Other malicious activities

**27) What is Protractor?**

**Answer:**Protractor is an open-source automated Testing tool used for Testing web applications. It is an end-to-end Testing framework primarily used for Testing Angular and AngularJS web applications. It provides the necessary tools and functionalities for Testing Angular applications and simplifies the process of writing and running Tests.

**28) What are the Test library APIs provided by the Robot Framework?**

**Answer:**The Robot Framework is an extremely popular open-source Test Automation framework. It provides various Test library APIs that help Testers and Developers create and execute automated Test cases. Let’s explore some of these API Test libraries below:

**1) Dynamic API:**In Dynamic APIs, the keyword names and their implementations are determined at runtime. So, their behaviour can change based on the specific conditions of Testing.

**2) API hybrid:**API hybrid is a combination of both Static and Dynamic APIs. It has the benefits of both fixed, predictable Testing and context-driven Testing.

**3) Static API:** Static API is a set of predefined methods and classes that map directly to specific keyword names. These methods typically take fixed arguments and have a one-to-one relationship with the keywords they implement.

**Scenario-based Automation Testing Interview Questions and Answers**

By now, you would have got a basic idea of some important technical questions and answers. It is now time for you to look at some Automation Testing Interview Questions and answers on real-life scenarios. These Automation Testing Interview Questions will help you answer practical questions, even if you haven’t had the opportunity for any practical experience. Let’s find out more:

**1) How would you automate login functionality for a website?**

**Answer:**The approach to automating login functionality for a website involves finding the elements of the login page using locators. A script is then written to input a username and password and click on the login button. Subsequently, the script verifies if the login was successful by checking if the user is redirected to the correct page or by looking for a specific element on the page.

**2) If a Test fails, what will be your next step?**

**Answer:**In the event of a Test failure, the appropriate course of action is to first analyse the failed Test case. This involves reviewing the logs, screenshots, or video of the Test execution if available. Then, the script is debugged to check if the failure is due to changes in the application or an issue in the script itself.

**3) If the application has minor changes, what would be your approach to modifying the Automation scripts?**

**Answer:**The approach to modifying the Automation scripts would depend on the changes in the application. If the changes are related to the UI, then the respective element locators in the scripts should be updated. If there are changes in the functionality,  the logic of the script should be modified accordingly.

**4) How would you handle dynamic elements on a webpage?**

**Answer:**Dynamic elements on a webpage can be handled by using dynamic XPath or CSS selectors that can identify elements based on their relationships with surrounding elements or attributes.

**5) How would you automate a Test scenario where you need to check if an email is sent after a user registration?**

**Answer:**This can be done with two simple steps. Firstly, the user registration process is automated. Then, an API is used to connect to the email server and verify if an email is received in the inbox.

**6) If there is a scenario that takes a long time to execute, would you prefer Manual Testng or Automation Testing? Why?**

**Answer:** In such scenarios, Automation Testing is preferred. Automation allows for the execution of the Test scenario multiple times without human intervention, ensuring accuracy and time efficiency.

**7) How would you automate a scenario where you need to validate the contents of a downloaded file after clicking a button on a webpage?**

**Answer:**In order to automate a scenario for validating the contents of a downloaded file after clicking a button, the first step involves automating the click event on the button. Subsequently, a programming language's file-handling capabilities are used to open and validate the contents of the downloaded file.

**8) How would you automate a scenario where you need to verify a specific colour, font, and position of an element on a webpage?**

**Answer:**This scenario can be automated by using the WebDriver’s JavaScript executor. JavaScript is executed code to get the colour, font, and position of an element, and it is then validated in the Test script.

**9) How would you handle pop-up windows or alert boxes in your Automation script?**

**Answer:**Pop-up windows or alert boxes can be handled using the Alert interface provided by Selenium WebDriver.  The interface offers methods like accept(), dismiss(), getText(), and sendKeys(), which can be used based on the requirement.

**10) How would you automate a scenario where you need to verify if a user is able to scroll down a webpage until the footer section is visible?**

**Answer:**In order to automate this scenario, the WebDriver's JavaScript executor is used to scroll down the page. The visibilityOf() method from the ExpectedConditions class is employed to check if the footer section is visible.

**11) You've been asked to automate a legacy application. What is your approach?**

**Answer:**Automating a legacy application begins with an assessment of existing manual Test cases. The Tests best suited for Automation are identified, and a suitable Automation tool is then selected. Then, the Automation framework is designed and developed to ensure maintainability and scalability for future needs.

**12) A script you wrote was working fine yesterday but is failing today. How do you troubleshoot?**

**Answer:**In the case of a script that was functioning correctly but is now failing, the first step is to check for changes in the application that might affect the script. The script is then debugged to address issues related to element locators or timing. Additionally, it is important to verify the correctness of the Test data and the environment.

**13) Your Automation scripts are running slowly. How can you improve the speed?**

**Answer:** In order to enhance the speed of Automation scripts, strategies such as reducing the use of implicit waits, utilising headless browsers for non-UI Tests, parallelising Tests, and optimising code for performance can be employed.

**14) You are automating a complex workflow with multiple steps. How do you ensure that the test is maintainable?**

**Answer:** Ensuring the maintainability of a complex workflow involves breaking down the workflow into smaller, reusable functions. The Page Object Model is used to separate the Test logic from the page elements.

**15) Your Automation script is failing due to a change in the application. How do you handle this?**

**Answer:**If a script fails due to an expected change in the application, the script is updated to accommodate the new behaviour. In the case of unexpected changes, the issue is communicated to the development team, and the script is updated once the issue is resolved.

**16) The Automation script needs to run on multiple browsers and platforms. How do you handle this?**

**Answer:**Automation scripts can be configured to run on multiple browsers and platforms using tools like Selenium Grid. This can also be done with cloud-based services such as Sauce Labs or BrowserStack. It is essential to ensure that the scripts are written in a way that makes them independent of the specific browser or platform.

**17) You are asked to automate a functionality that is not yet fully developed. How do you handle this?**

**Answer:**Automating a functionality that is not yet fully developed involves collaboration with the development team to understand the expected behaviour. The Automation script is designed based on this understanding, and stubs or mock objects may be used to simulate the undeveloped functionality.

**18) You are working with a team that follows the Agile methodology. How do you fit Automation into this process?**

**Answer:** In an Agile environment,  the focus of Automation is on Test cases that are part of the regression suite and Test cases related to the current sprint's user stories. These Test cases may be included in the Continuous Integration (CI)/Continuous Deployment (CD) pipeline for immediate feedback.

**19) How do you handle a situation where the Automation tool does not support a particular control or object in the application?**

**Answer:** If the Automation tool does not support a specific control or object in the application, the first step is to explore potential workarounds. If no viable workaround is available, the issue is communicated to the development and business teams, along with suggestions for possible alternatives.

**20) You are asked to estimate the time required for automating a Test case. How do you approach this?**

**Answer:**Estimating the time required for automating a Test case involves considering the complexity of the Test case, the maturity of the Automation framework, and familiarity with the application and Automation tool. The estimation also accounts for time allocated to debugging, maintenance, and reporting.

**Some essential tips for Automation Testing Interviews**

Along with all these interview questions, we have also prepared ten pointers that will act as essential tips before your Automation Testing Interviews. Let’s explore them below:

**1) Understand the basics:**Make sure you have a solid understanding of Automation Testing basics, including knowledge of different Testing types, tools, methodologies, and concepts like SDLC and STLC.

**2) Know your tools:**Be familiar with the tools that are commonly used in Automation Testing, such as Selenium, JMeter, TestComplete, or QTP. Understand their features and capabilities and how to apply them in different Testing scenarios.

**3) Learn about frameworks:**Understand the different types of Automation frameworks like Keyword-Driven, Data-Driven, Hybrid, or Page Object Model. Know their advantages, disadvantages, and where they are most effectively used.

**4) Brush up your coding skills:**While Automation Testing isn't all about coding, good programming skills can be a major advantage. Depending on the tools you work with, enhance your coding skills in languages like Java, Python, or JavaScript.

**5) Know how to design Test Cases:** Be ready to explain how you would design Test cases for Automation. This includes identifying which cases to automate, writing the scripts, and maintaining them.

**6) Understand CI/CD:**Knowledge of Continuous Integration/Continuous Delivery (CI/CD) pipelines and how Automation Testing fits into them is a huge plus. Familiarise yourself with tools like Jenkins or CircleCI.

**7) Stay updated:**The field of Automation Testing is constantly evolving. Stay updated with the latest tools and best practices in the industry. This will help you give latest and accurate answers for QA Automation Interview Questions.

**8) Work on real projects:**Hands-on experience is key. If you haven't worked on real projects, consider working on open-source projects or creating your own projects to demonstrate your skills.

**9) Prepare for behavioural questions:** Besides technical skills, interviewers will also assess your soft skills. Be prepared to discuss your experiences working in a team, handling tight deadlines, and solving challenging problems.

**10) Review and practice:**Finally, review common Automation Tester Interview Questions  and practice your responses. This will help you to be confident and articulate during your interview.