Automation Test Engineer - Interview Preparation Questions

# I. Core Automation Concepts

* Introduce yourself.
* Explain your automation framework and its components.
* What is a Page Object Model (POM)? What are its advantages and disadvantages?
* What is the Singleton Design Pattern? How do you define a singleton class, and what are its uses?
* How do you prioritize test cases?
* What are the stages of automation testing?
* Is 100% automation possible? Which test cases should you automate, and which should you not?
* How do you calculate ROI for automation?
* How do you handle test data management in automation testing?
* Explain your roles and responsibilities in a project.
* Can you work as an individual contributor?
* How do you assign work when you have senior and junior team members?
* As a lead QA, what is your checklist for code review?
* Describe the different Agile ceremonies.
* Explain the points you consider while writing a good test case.
* Where do you write your test cases? (e.g., Excel, other tools)
* Describe any innovations you have introduced in your projects. How would you explore integrating Generative AI into your testing practices?
* How do you estimate the time required for automating a test case?
* How will you handle a scenario where a login button takes time to appear using Selenium best practices?
* Describe a challenge you faced during automation and how you resolved it.
* How do you prioritize your automation tasks when working on multiple modules?

# II. Selenium WebDriver

* What is Selenium? What are the differences between Selenium 3 and Selenium 4?
* What are the different types of locators in Selenium? When do you use XPath over CSS locators?
* Difference between findElement() and findElements()?
* How do you handle dropdowns in Selenium? How do you get a specific value from a dropdown and reuse it in verifications?
* How do you handle multiple windows in Selenium? Why is SET used with window handles?
* What is the difference between get() and navigate().to()?
* How do you handle alerts and pop-ups in Selenium?
* How do you handle frames and iframes? How do you switch between frames, and how do you detect frames? How do you switch to a parent frame?
* What is the difference between driver.quit() and driver.close()?
* What are the different types of waits in Selenium? Explain Implicit, Explicit, and Fluent Waits, and when do you use each? Provide a code snippet for Explicit Wait and Fluent Wait.
* How do you handle a StaleElementReferenceException?
* What is JavaScriptExecutor? How do you use it? How do you scroll to an element using Selenium? What other activities can you perform using JavaScriptExecutor?
* How do you take a screenshot in Selenium? How do you take a screenshot of a specific web element?
* How do you handle file uploads in Selenium?
* How do you validate broken links in Selenium?
* How do you capture network logs in Selenium?
* How do you handle browser notifications during test automation execution?
* How do you handle CAPTCHA in Selenium?
* How do you handle authentication pop-ups in Selenium?
* How do you handle dynamic elements on a webpage?
* How will you write xPath for disappearing elements in Selenium ?
* Print all values of Customer salary from a given graph using selenium?
* How to run Selenium scripts on an already opened browser?
* Provide pseudocode for automating the selection of all values from a dropdown on a webpage and verifying if they are sorted.
* Asked to share screen and write XPath for a given element.
* Use of the Actions class and Select class.
* What is Invalid Certificate Exception?

# III. Selenium Framework & Best Practices

* Explain your Selenium Framework in depth, including its architecture and components. Have you created a framework from scratch?
* What unit testing framework are you using? What is the difference between JUnit and TestNG?
* What is the Page Object Model (POM)? What is Page Factory? How is it different from POM? What are the advantages and disadvantages of POM?
* What is the difference between @FindBy and driver.findElement()?
* What are Selenium Grid and its advantages?
* What is the role of Desired Capabilities in Selenium?
* What are the different types of Assertions used in Selenium?
* What are the limitations of Selenium WebDriver?
* How do you integrate Selenium with TestNG or JUnit?
* What are TestNG Listeners, and how do you implement them? What is the role of listeners in TestNG?
* What are the best practices for designing a scalable Rest Assured framework?

# IV. TestNG

* What is TestNG? Describe the features and benefits of using TestNG.
* What is the execution format of tests in TestNG?
* What are some common TestNG annotations and their purposes?
* How do you generate and customize reports in TestNG?
* Can priority be negative for methods in TestNG? If yes, what is the execution flow as per priority?
* In TestNG, what is the order of execution for test methods annotated with priorities -2, -1, 0, 1, and 2? If no priority is explicitly set, what determines the order of execution (alphabetical or ASCII value)?
* What is the difference between dependsOnMethods and dependsOnGroups in TestNG?
* What are the different ways to exclude tests in TestNG?
* What does threadPoolSize mean in TestNG, and how does it work?
* Why do we need @BeforeSuite and @AfterSuite annotations?
* What is the purpose of Data Providers in TestNG? Provide a pseudocode example of their implementation.
* How do you rerun failed test cases in TestNG? Explain how to rerun failed test cases without utilizing the TestNg.failed.xml file. How can you run failed test cases using a TestNG Listener? Describe the implementation process.
* Suppose you have 200 test cases and 49 failed. How would you collect data from only failed test cases in TestNG? What is the best way?
* How do you do parallel testing using TestNG?

# V. Cucumber

* What is Cucumber, and how does it work?
* What is a Runner File in Cucumber? What's inside the Runner File in Cucumber?
* What are the options available in Cucumber options?
* What are tags in Cucumber?
* What is the purpose of the Background keyword in a Feature file?
* Differentiate between Scenario and Scenario Outline in Cucumber?
* Difference between Scenario vs Scenario Outline in Cucumber BDD.
* How can you run a scenario multiple times in Cucumber? How can you run certain activities for all scenarios multiple times?
* What is the use of a test runner in cucumber?

# VI. API Testing

* What is API Testing?
* What is the difference between API and unit testing?
* What is the difference between web services and APIs? What are REST, SOAP, and GraphQL in APIs?
* What do you test in standalone APIs? What do you test in 3rd-party integrated APIs?
* What are the components of an HTTP request? What is an HTTP response?
* How can we add validation points in Postman?
* When to use collections, environments, and global variables in Postman?
* How to execute a collection end-to-end in Postman?
* How to validate that an API response has the correct status code in Postman?
* What happens when an API response returns Form Data instead of JSON, and how do you validate it in Postman?
* How to set up Basic Auth in Postman? Where do you store environment credentials in Postman?
* How to save a demo response for an API request in Postman?
* How will you validate an API request if a VPN is required for it to work?
* How do you filter results in an API request using Postman?
* How to set custom headers in Postman?
* What do you understand by server-side validation?
* What is 3-tier architecture?
* What are the important validations you will put on an API?
* Some status code descriptions related to API testing.
* API status codes difference (400, 401, 201, 200, etc.).
* Difference between PUT vs PATCH request.
* What do you test in 3rd-party integrated APIs?

# VII. Rest Assured

* What is Rest Assured? Why is it used for API Testing? What are the key features of Rest Assured?
* How do you set the Base URI in Rest Assured?
* What is the difference between given(), when(), and then() in Rest Assured?
* How do you validate status codes and response bodies in Rest Assured?
* How do you perform GET, POST, PUT, and DELETE requests using Rest Assured?
* How can you send query parameters in a GET request with Rest Assured?
* How do you send JSON payload in POST requests with Rest Assured?
* How can you upload files using Rest Assured?
* What is the difference between pathParam() and queryParam() in Rest Assured?
* How do you handle Basic Authentication in Rest Assured?
* How do you implement OAuth 2.0 authorization using Rest Assured? What is the purpose of the auth() method in Rest Assured? How can you pass Bearer tokens in API requests using Rest Assured? What is the content type in OAuth 2.0?
* How do you validate JSON response body fields in Rest Assured?
* How can you assert response time using Rest Assured? What is the use of the assertThat() method in Rest Assured?
* How do you validate response headers in Rest Assured?
* How can you convert a Java object to JSON (Serialization) in Rest Assured?
* How do you deserialize a JSON response into a Java object?
* How do you perform data-driven testing using Rest Assured?
* How can you implement logging in Rest Assured for better debugging?
* How do you handle dynamic response values in Rest Assured?
* How can you test pagination logic in API testing with Rest Assured?
* How to chain multiple API requests in Rest Assured to create end-to-end test flows?
* How do you handle timeouts in Rest Assured? What strategies can you use to handle flaky API tests in Rest Assured? How do you debug Rest Assured test failures effectively?
* How can you integrate Rest Assured with TestNG or JUnit?

# VIII. Java

* Is the value of the String object "Akash" mutable? If not, explain why. What other Java classes allow modification of string values? Why is String immutable in Java?
* Can a Java class implement multiple interfaces? Explain.
* Write a Java program to find the second-highest number in an integer array.
* What are constructors? Describe different types of constructors in Java.
* What are the key differences between the StringBuffer and StringBuilder classes in Java? What is the difference between stringbuilder and stringbuffer? Difference between StringBuffer and StringBuilder.
* Compare and contrast ArrayList and LinkedList. In what scenario would you choose ArrayList over LinkedList for faster data retrieval? What is the difference between array list and Linked list? Difference between Array and ArrayList.
* What are the distinctions between HashMap and Hashtable in Java? Questions from HashMap, HashSet. Difference between Hashtable and HashSet.
* Explain the difference between operator == and equals() method in Java.
* Explain the exception hierarchy in Java.
* Write a program to create a custom exception in Java.
* What is an interface and an abstract class in Java? Abstract class vs Interface.
* Why is an abstract class allowed to have a constructor in Java?
* Difference between final, finally, and finalize in Java.
* Explain polymorphism in Java. What is compile time and run time polymorphism? How have you implemented them?
* Write a program to override a function in Java.
* Do you know about Java 8? What are Java Streams and Lambda Expressions? What is a functional interface?
* What are Java Streams and Lambda Expressions?
* What is a functional interface?
* Can you create an object of an interface? Can you call a method from an interface that you have not implemented in any class?
* Can you overload and override static methods?
* Difference between private, public, and protected access modifiers. default vs public access modifier.
* Explain the collections framework hierarchy in Java. What is Collection in Java? Difference between Collection and Collections.
* What is method hiding in Java?
* Write Java code to reverse a string and perform array manipulation. Reverse string using Java. Reverse a string without using inbuilt methods.
* Write Java code to print all the permutations of a given string?
* Write Java code to print all the array elements that appear at least 2 times.
* Write a program to validate usernames (alphanumeric, no spaces/special characters, no duplicates).
* Write a program to count character occurrences in a string. Calculate the number of occurrences of a character in a string using Java
* Find the highest number in an array without using inbuilt methods. Max/min in array
* Check prime numbers using Java. Prime check
* Swap numbers (no temp)
* Vowel/consonant count
* Char frequency
* Fibonacci & factorial
* Digit sum
* Remove duplicates
* Reverse words in string
* Even/odd in array
* String length without .length()
* String Integer conversion
* Even/odd index values
* Reverse array
* Sorted array check
* Case conversion

# IX. Git

* What are the different stages in committing the code to GitHub?
* Is it possible to revert changes in a remote repository? If yes, how?
* When do you commit your code? After committing, how do you validate that everyone has the updated code?
* How to merge stashed changes in a local repository?
* Why do we need a .gitignore file? How do you add files to it?
* List the Git commands you commonly use in your projects. Basic GitHub commands (git add, git commit, git diff). Git command: fetch, pull, clone use.
* How do you handle merge conflicts in Git? How do you resolve conflicts in Git? How do you handle merge conflicts without manually inspecting the files?
* Difference between git rebase and git merge. When to use git rebase?

# X. Jenkins

* How do you integrate Jenkins with Selenium?
* Explain Jenkins pipelines for continuous integration.
* Describe your process for executing automation scripts using Jenkins. How do you configure Jenkins for nightly builds?
* Do you know how to create a parameterized pipeline job in Jenkins?
* What is the typical number of test cases included in your daily regression suite execution?

# XI. Operating Systems

* Some questions on Linux commands?

# XII. SQL

* Write an SQL query using JOIN (two tables with columns given).
* Write an SQL query to retrieve the student\_id of students who are not enrolled in any courses, based on the provided table schemas.
* Write code for database connection?

# XIII. Application/Testing Types

* What is the difference between Smoke and Sanity Testing?
* Difference between Functional vs Non-Functional Testing.
* What is STLC in software testing?
* Difference between Alpha and Beta Testing.
* What do you test in standalone APIs?
* How do you perform cross browser testing?

# XIV. Error/Defect Handling

* Explain the exception hierarchy.
* How have you handled null pointer exceptions?
* What exceptions have you faced while working with Selenium? (Give 5 examples).
* How do you handle errors and exceptions in automation testing?
* What is the Bug Life Cycle?
* Important things to consider when logging a bug. Fields required to log a bug/defect.
* What is a test plan, and what does it include?
* Briefly explain the phases of SDLC.
* Difference between priority and severity.
* What is the strategy to start failure analysis from Day 1 itself in the current release?
* How do you handle flaky API tests?

# XV. Concurrency/Performance

* What is multithreading?

# XVII. Behavioral Questions

* Have you mentored juniors or collaborated with developers to debug automation failures?
* What will you do if your automation script passes, but a UI bug is reported by manual testers?