**Software Quality control interview questions**

**Questions in green color is for seniors**

1. Fundamental

**Ques.1. What is Software Testing?**   
Ans. [Software testing](https://artoftesting.com/manualTesting/what-is-software-testing.html) is the process of evaluating a system to check if it satisfies its business requirements.it is used for ensuring the quality of software to the stakeholders of the application.

**Ques.2. Why is testing required?**   
Ans. We need software testing for following reasons-

Testing provides an assurance to the stakeholders that product works as intended.

Avoidable defects leaked to the end user/customer without proper testing adds bad reputation to the development company.

Defects detected earlier phase of SDLC results into lesser cost and resource utilization of correction.

Saves development time by detecting issues in earlier phase of development.

Testing team adds another dimension to the software development by providing a different view point to the product development process.

**Ques.3. When should we stop testing?**   
Ans. Testing (both manual and automated) can be stopped when one or more of the following conditions are met-

After test case execution - Testing phase can be stopped when one complete cycle of test cases is executed after the last known bug fix with agreed upon value of pass-percentage.

Once the testing deadline is met - Testing can be stopped after deadlines get met with no high priority issues left in system.

Based on Mean Time Between failure (MTBF)- MTBF is the time interval between two inherent failures. Based on stakeholders decisions, if the MTBF is quite large one can stop the testing phase.

Based on code coverage value - Testing phase can be stopped when the automated code coverage reaches a specific threshold value with sufficient pass-percentage and no critical bug.

**Ques.4. State main difference Between Quality Assurance and Quality Control?**

|  |  |
| --- | --- |
| QA | QC |
| QA is more planned and systematic method of monitoring the quality of the process. This process is followed to produce quality software and application. | Quality Control is related to the quality of the product. QC team never find the defects and suggests improvements. |

**Ques.6. What is the difference between Verification and Validation?**   
Ans. Following are the major differences between verification and validation-

|  |  |
| --- | --- |
| **Verification** | **Validation** |
| Verification is the process of evaluating the artifacts as well as the process of software development in order to ensure that the product being developed will comply to the standards. | Validation is the process of validating that the developed software product conforms to the specified business requirements. |
| It is static process of analyzing the documents and not the actual end product. | It involves dynamic testing of software product by running it. |
| Verification is a process oriented approach. | Validation is a product oriented approach. |
| Answers the question - "Are we building the product right?" | Answers the question - "Are we building the right product?" |
| Errors found during verification require lesser cost/resources to get fixed as compared to be found during validation phase. | Errors found during validation require more cost/resources. Later the error is discovered higher is the cost to fix it. |

**Ques.7. What is SDLC?**   
Ans. [SDLC](https://artoftesting.com/manualTesting/software-development-life-cycle-sdlc.html) stands for Software Development Life Cycle. It refers to all the activities performed during software development - requirement gathering, requirement analysis, designing, coding or implementation, testing, deployment and maintenance.



**Ques.8. Explain STLC - Software Testing life cycle.**   
Ans. [Software testing life cycle](https://artoftesting.com/manualTesting/software-testing-life-cycle-stlc.html) refers to all the activities performed during testing of a software product. The phases include-

* Requirement analyses and validation - In this phase the requirements documents are analysed and validated and scope of testing is defined.
* Test planning - In this phase test plan strategy is defined, estimation of test effort is defined along with automation strategy and tool selection is done.
* Test Design and analysis - In this phase test cases are designed, test data is prepared and automation scripts are implemented.
* Test environment setup - A test environment closely simulating the real world environment is prepared.
* Test execution - The test cases are prepared, bugs are reported and retested once resolved.
* Test closure and reporting - A test closure report is prepared having the final test results summary, learning and test metrics.

**Ques.9. What are the different types of testing?**   
Testing can broadly be defined into two types-

* **Functional testing** - Functional testing involves validating the functional specifications of the system.
* **Non-Functional testing** - Nonfunctional testing is a type of testing that involves testing of non-functional requirements of the system such as performance, scalability, security, endurance, portability etc.

Going by the way the testing is done, it can be categorized as-

* [**Black box testing**](https://artoftesting.com/manualTesting/black-box-testing.html) - In black box testing, the tester need not have any knowledge of the internal architecture or implementation of the system. The tester interact with the system through the interface providing input and validating the received output.
* [**White box testing**](https://artoftesting.com/manualTesting/white-box-testing.html) - In white box testing, the tester analyses the internal architecture of the system as well as the quality of source code on different parameters like code optimization, code coverage, reusability etc.
* **Gray box testing** - In gray box testing, the tester has partial access to the internal architecture of the system e.g. the tester may have access to the design documents or database structure. This information helps tester to test the application better.

**Ques.10. What is a test plan?**   
Ans. A [test plan](https://artoftesting.com/manualTesting/test-plan-document-template.html) is a formal document describing the scope of testing, the approach to be used, resources required and time estimate of carrying out the testing process. It is derived from the requirement documents(Software Requirement Specifications).

**Ques.11. What is a test scenario?**   
Ans. A [test scenario](https://artoftesting.com/manualTesting/test-scenario.html) is derived from a use case. It is used for end end to end testing of a feature of an application. A single test scenario can cater multiple test cases. The scenario testing is particularly useful when there is time constraint while testing.

**Ques.12. What is a test case?**   
Ans. A [test case](https://artoftesting.com/manualTesting/test-case.html) is used to test the conformance of an application with its requirement specifications. It is a set of conditions with pre-requisites, input values and expected results in a documented form.

**Ques.13. What are some attributes of a test case?**   
Ans. A test case can have following attributes-

(Testcase Id.-Test Summary- Description - Prerequisite or pre-condition -Test Steps- Expected result - Actual result -Test Result - Date - Executed by)

**Ques.14. What is a test script?**   
Ans. A test script is an automated test case written in any programming or scripting language. These are basically a set of instructions to evaluate the functioning of an application.

**Ques.15. What is a bug?**   
Ans. A bug is a fault in a software product **detected at the time of testing**, causing it to function in an unanticipated manner.

**Ques.16. What is a defect?**   
Ans. A defect is non-conformance with the requirement of the product **detected in production** (after the product goes live).

**Ques.18. What are some defect reporting attributes?**   
Ans. Some of the attributes of a Defect report are-

(Defect Id - Defect Summary -Defect Description - Steps to reproduce - Expected Result - Actual Result- Defect Severity - Priority )

**Ques.19. What are some of the bug or defect management tools?**   
Ans. Some of the most widely used Defect Management tools are - Jira, Bugzilla, Redmine, Mantis, Quality Center etc.

**Ques.20. What is defect density?**   
Ans. Defect density is the measure of density of the defects in the system. It can be calculated by dividing number of defect identified by the total number of line of code(or methods or classes) in the application or program.

**Ques.21. What is defect priority?**   
Ans. A defect priority is the urgency of the fixing the defect. Normally the defect priority is set on a scale of P0 to P3 with P0 defect having the most urgency to fix.

**Ques.22. What is defect severity?**   
Ans. Defect severity is the severity of the defect impacting the functionality. Based on the organisation, we can have different levels of defect severity ranging from minor to critical or show stopper.

**Ques.23. Give an example of Low priority-Low severity, Low priority-High severity, High priority-Low severity, High priority-High severity defects.**   
Ans.

1. **Low priority-Low severity** - A spelling mistake in a page not frequently navigated by users.
2. **Low priority-High severity** - Application crashing in some very corner case.
3. **High priority-Low severity** - Slight change in logo color or spelling mistake in company name.
4. **High priority-High severity** - Issue with login functionality.

**Ques.24. What is a blocker?**   
Ans. A blocker is a bug of high priority and high severity. It prevents or blocks testing of some other major portion of the application as well.

**Ques.25. What is a critical bug?**   
Ans. A critical bug is a bug that impacts a major functionality of the application and the application cannot be delivered without fixing the bug. It is different from blocker bug as it doesn't affect or blocks the testing of other part of the application.

**Ques.26. Explain bug life cycle or the different states of a bug.**   
Ans. A bug goes through the following phases in software development-

(New - Assigned - Open - Rejected/Not a bug - Deferred - Fixed - Test - Reopened - Verified - Closed )

**Ques.27. What are the different test design techniques?**   
Ans. Test design techniques are different standards of test designing which allow systematic and widely accepted test cases. The different test design techniques can be categorized as static test design technique and dynamic test design technique.

1. **Static Test Design Techniques** - The test design techniques which involves testing without executing the code. The various static test design techniques can be further divided into two parts manual and using tool-

* Manual static design techniques-
* Walk through
* Informal reviews
* Technical reviews
* Audit
* Inspection
* Management review
* Static design techniques using tool-
* Static analysis of code - It includes analysis of the different paths and flows in the application and different states of the test data.
* Compliance to coding standard - This evaluates the compliance of the code with the different coding standards.
* Analysis of code metrics - The tool used for static analysis is required to evaluate the different metrics like lines of code, complexity, code coverage etc.

1. **Dynamic Test Design Techniques** - Dynamic test design techniques involves testing by running the system under test.

* **Specification based** - Specification based test design techniques are also referred to as blackbox testing. These involve testing based on the specification of the system under test without knowing its internal architecture.
* **Structure based** - Structure based test design techniques are also referred to as white box testing. In this techniques the knowledge of code or internal architecture of the system is required to carry out the testing.
* **Experienced based** - The experienced based techniques are completely based on the experience or intuition of the tester. Two most common forms of experienced based testing are - adhoc testing and exploratory testing.

**Ques.28. Explain the different types of specification-based test design technique?**   
Ans. Specification based test design techniques are also referred to as Blackbox testing. It involves testing based on the specification of the system under test without knowing its internal architecture. The different types of specification-based test design or black box testing techniques are-

* **Equivalence partitioning** - Grouping test data into logical groups or equivalence classes with the assumption that all the data items lying in the classes will have same effect on the application.
* **Boundary value analysis** - Testing using the boundary values of the equivalence classes taken as the test input.
* **Decision tables** - Testing using decision tables showing application's behavior based on different combination of input values.
* **State transition testing** - Testing based on state machine model.
* **Use case testing** - Testing carried out using use cases.

**Ques.29. Explain equivalence class partitioning.**   
Ans. Equivalence class partitioning is a specification based black box testing techniques. In [equivalence class partitioning](https://artoftesting.com/manualTesting/equivalence-class-partitioning.html), set of input data that defines different test conditions are partitioned into logically similar groups such that using even a single test data from the group for testing can be considered as similar to using all the other data in that group. E.g. for testing a Square program(program that prints the square of a number- the equivalence classes can be-   
Set of Negative numbers, whole numbers, decimal numbers, set of large numbers etc.

**Ques.30. What is boundary value analysis?**   
Ans. [Boundary value analysis](https://artoftesting.com/manualTesting/boundary-value-analysis.html) is a software testing technique for designing test cases wherein the boundary values of the classes of the equivalence class partitioning are taken as input to the test cases e.g. if the test data lies in the range of 0-100, the boundary value analysis will include test data - 0,1, 99, 100.

**Ques.31. What is decision table testing?**   
Ans. Decision table testing is a type of specification based test design technique or black box testing technique in which testing is carried out using decision tables showing application's behavior based on different combination of input values. Decision tables are particularly helpful in designing test cases for complex business scenarios involving verification of application with multiple combinations of input.

**Ques.32. What is a cause effect graph?**   
Ans. A cause effect graph testing is black box test design technique in which graphical representation of input i.e. cause and output i.e. effect is used for test designing. This technique uses different notations representing AND, OR, NOT etc relations between the input conditions leading to output.

**Ques.33. What is state transition testing?**   
Ans. State transition testing is a black box test design technique based on state machine model. State transition testing is based on the concept that a system can be defined as a collection of multiple states and the transition from one state to other happens because of some event.

**Ques.34. What is use case testing?**   
Ans. A use case testing is a black box testing approach in which testing is carried out using use cases. A use case scenario is seen as interaction between the application and actors(users). These use cases are used for depicting requirements and hence can also serve as basis for acceptance testing.

**Ques.35. What is structure based testing?**   
Ans. Structure based test design techniques are also referred to as white box testing. In this techniques the knowledge of code or internal architecture of the system is required to carry out the testing. The various kinds of testing structure based or white testing techniques are-

* Statement testing - Test scripts are designed to execute code statements and coverage is the measure of line of code or statements executed by test scripts.
* Decision testing/branch testing - Measure of the percentage of decision points(e.g. if-else conditions) executed out of the total decision points in the application.
* Condition testing- Testing the condition outcomes(TRUE or FALSE). So, getting 100% condition coverage required exercising each condition for both TRUE and FALSE results using test scripts(For n conditions we will have 2n test scripts).
* Multiple condition testing - Testing the different combinations of condition outcomes. Hence for 100% coverage we will have 2^n test scripts. This is very exhaustive and very difficult to achieve 100% coverage.
* Condition determination testing - It is an optimized way of multiple condition testing in which the combinations which doesn't affect the outcomes are discarded.
* Path testing - Testing the independent paths in the system(paths are executable statements from entry to exit points).

**Ques.36. What is Statement testing and statement coverage in white box testing?**   
Ans. Statement testing is a white box testing approach in which test scripts are designed to execute code statements.   
Statement coverage is the measure of the percentage of statements of code executed by the test scripts out of the total code statements in the application. The statement coverage is the least preferred metric for checking test coverage.

**Ques.37. What is decision testing or branch testing?**   
Ans. Decision testing or branch testing is a white box testing approach in which test coverage is measured by the percentage of decision points(e.g. if-else conditions) executed out of the total decision points in the application.

**Ques.38. What are the different levels of the testing?**   
Ans. Testing can be performed at different levels during the development process. Performing testing activities at multiple levels help in early identification of bugs. The different levels of testing are -

1. Unit Testing
2. Integration Testing
3. System Testing
4. Acceptance Testing

**Ques.39. What is unit testing?**   
Ans. [Unit testing](https://artoftesting.com/manualTesting/unit-testing.html) is the first level of testing and it involves testing of individual modules of the software. It is usually performed by developers.

**Ques.40. What is integration testing?**   
Ans. [Integration testing](https://artoftesting.com/manualTesting/integration-testing.html) is performed after unit testing. In integration testing, we test the group of related modules. It aims at finding interfacing issues between the modules.

**Ques.41. What are the different types of integration testing?**   
Ans. The different type of integration testing are-

1. Big bang Integration Testing - In big bang integration testing, testing starts only after all the modules are integrated.
2. Top-down Integration Testing - In top down integration, testing/integration starts from top modules to lower level modules.
3. Bottom-up Integration Testing - In bottom up integration, testing starts from lower level modules to higher level module up in the hierarchy.
4. Hybrid Integration Testing - Hybrid integration testing is the combination of both Top-down and bottom up integration testing. In this approach, the integration starts from middle layer and testing is carried out in both the direction

For details check [Integration testing](http://artoftesting.com/manualTesting/integration-testing.html).

**Ques.42. What is stub?**   
Ans. In case of top-down integration testing, many a times lower level modules are not developed while beginning testing/integration with top level modules. In those cases Stubs or dummy modules are used that simulate the working of modules by providing hard-coded or expected output based on the input values.

**Ques.43. What is driver?**   
Ans. In case of bottom-up integration testing, drivers are used to simulate the working of top level modules in order to test the related modules lower in the hierarchy.

**Ques.44. What is a test harness? Why do we need a test harness?**   
Ans. A test harness is a collection of test scripts and test data usually associated with unit and integration testing. It involves stubs and drivers that are required for testing software modules and integrated components.

**Ques.45. What is system testing?**   
Ans. [System testing](https://artoftesting.com/manualTesting/system-testing.html) is the level of testing where the complete software is tested as a whole. The conformance of the application with its business requirements is checked in system testing.

**Ques.46. What is acceptance testing?**   
Ans. [Acceptance testing](https://artoftesting.com/manualTesting/acceptance-testing.html) is a testing performed by the potential end user or customers to check if the software conforms to the business requirements and can be accepted for use.

**Ques.47. What is alpha testing?**   
Ans. [Alpha testing](https://artoftesting.com/manualTesting/alpha-testing.html) is a type of acceptance testing that is performed end users at the developer site.

**Ques.48. What is beta testing?**   
Ans. [Beta testing](https://artoftesting.com/beta-testing) is the testing done by end users at end user's site. It allows users to provide direct input about the software to the development company.

**Ques.49. What is adhoc testing?**   
Ans. Adhoc testing is an unstructured way of testing that is performed without any formal documentation or proper planning.

1. Mobile testing

**1) Explain what is the difference between Web testing and WAP testing?**

* **WAP Testing:** It is the testing the WAP (Wireless Application Protocol) used in network applications
* **Web Testing:** It is related mainly to the testing of web applications such as websites and portals

**2) List out some of the automated mobile testing tools?**

For mobile testing, two kinds of automation tools are available to test mobile Applications.

* Object based mobile testing tools: Jama solution, Ranorex,
* Image based mobile testing tools: RoutinBot, Egg Plant, Sikuli

**3) Explain what is the difference between simulator and emulator?**

* **Simulator:** It is an electronic network simulation equipment or a base station equipment for CDMA/CMA mobile phones. It helps in latching home networks without roaming services and can make Voice; Data calls, SMS,
* **Emulator:** It is a software to test mobile application without a live handset   
  **4) List out the types of mobile app testing?**

The types of mobile app testing includes

* [Usability Testing](https://www.guru99.com/usability-testing-tutorial.html)
* [Compatibility Testing](https://www.guru99.com/compatibility-testing.html)
* Interface testing
* Services testing
* Low-level resource testing
* [Performance Testing](https://www.guru99.com/performance-testing.html)
* Operational testing
* Installation tests
* [Security Testing](https://www.guru99.com/what-is-security-testing.html)

**5) Mention what is the Android testing strategy?**

The standard Android testing strategy must include the following test

* Unit Test
* Integration Test
* Operation Test
* System Test

**6) Explain Android testing framework?**

Android testing framework includes three segments

* **Application Package:** It is the target application that requires to be tested
* **Instrumentation TestRunner:** It is a[Test Case](https://www.guru99.com/test-case.html)runner that runs test cases on target application. It includes an **SDK tools** for building test and a **tool that provides APIs** for writing program that control an android device, for example, MonkeyRunner
* **Test Package:** It includes two classes, **Test case classes,** and **Mock objects**. Test case classes include test methods to perform on target application, while mock object includes mock data that will be used as sample input for test cases.

**7) List out the best practices for Android Testing?**

* Developer should prepare the test cases at the same time when they are writing the code
* Together with source code all test cases should be stored
* Use continuous integration and execute tests every time the code is changed
* Avoid using rooted devices and emulators

**8) Mention what are the common bugs found while mobile testing?**

* **Critical:** Your phone system crash when testing particular feature in your device
* **Block:** Unable to do anything though phone is on unless you reboot your device
* **Major:**Unable to perform a function of a particular feature
* **Minor:** Under minor bugs usually GUI bugs fall.

**9) Explain what is Robo-electric testing framework?**

Testing done on Android Testing Framework for emulators or device is difficult. Running and building test case sometimes takes lots of development effort. Robo-electric framework allows you to run Android test directly on JVM without the need of a device or an emulator.

**10) Explain how A/B testing is done for ios app?**

A/B testing for ios includes three steps

* **Configure a test:** It prepares two versions of your iOS app (A&B) and test metric
* **Test:** Tests two iOS versions above on devices simultaneously
* **Analyze:** It select and measure better version to release

**11) While performing end to end mobile testing what are the major criteria, you have to take in consideration?**

* Installation
* Application launching without having network
* Uninstallation of app
* Orientation of app if it supports
* Testing application performance on a different kind of devices and network scenarios
* Testing the application response how it is responding

**12) List out the features does monkey tool provides?**

Monkey tools provide features like

* Basic configuration options
* Operational constraints
* Event types and frequencies
* Debugging options

**13) Mention what should be the selecting criteria for Test Automation Tool for mobile Testing?**

For mobile testing, the test automation tool should have following criteria

* **Multi-platform support:** Ensure that the tool does support your current and future target platform
* **Script Usability:** Object-based tools provides a high degree of the script usability
* **Jailbreak Requirement:** If the tool uses rooted devices, it may not support latest OS version and may be incompatible with MDM policies
* **Source Code Changes:** Sharing source code may not be possible always
* **Lead time for New OS version:** How soon tool can support new iOS/android/other OS version

**14) When to choose automation testing and when manual testing?**

**Manual Testing**

* If the application has new functionality
* If the application requires testing once or twice

**Automate Testing**

* If the regression tests are repeated
* Testing app for complex scenarios

**15) List out the most common problem that tester faces while doing mobile testing in Cloud Computing?**

Challenges that tester faces while doing mobile testing are

* Subscription model
* High Costing
* Lock-in
* Internet connectivity issues
* Automation is image based and time-consuming
* Automation cannot be used outside the framework

**16) Explain what does mobile security testing includes?**

Mobile security testing includes

* Checks for multi-user support without interfering with the data between them
* Checks for access to files stored in the app by any unintended users
* Decryption or Encryption method used for sensitive data communication
* Detect sensitive areas in tested application so that they do not receive any malicious content

**17) List out mobile App testing?**

* Testing in all web browsers
* Browsers very significantly across the devices
* May support xHTML,HTML,WML,AJAX
* Difficulty in Benchmarking the performance due to highly fragmented Market
* Emulators do not capture all the attributes or characteristics of a device
* Implementation of the specification may not be consistent across vendors and devices
* In some situation, transcoder may not respect user experience factors

**18) Explain what is port testing?**

This testing is done to test the same functionality on different devices with different platforms. It is classified into two categories

* Device Testing
* Platform Testing

**19) List out some iPhone and iPad testing tools?**

* iPhone tester: Test your web interface in an i-phone sized frame
* Appium: It is a test automation tool used with native and hybrid ios application
* iPad Peek: Test your web application using an iPad interface
* Test Studio: It enables you to record, build and run automated tests for your iPad and iPhone applications.

1. Performance

**1)  What is performance testing?**

Performance testing is done for quality assurance. It involves testing Software application to make sure that the software is working well under their expected workload.

**2) Name different types of performance testing**

* Load testing
* Stress testing
* Endurance testing
* Spike testing
* Volume testing
* Scalability testing

**3) What are the common performance problem does user’s face?**

* Longer loading time
* Poor response time
* Poor Scalability
* Bottlenecking (coding errors or hardware issues)

**4) What are the common performance bottlenecks?**

Some common performance bottlenecks include

* CPU Utilization
* Memory Utilization
* Networking Utilization
* S limitation
* Disk Usage

**5) What are the Important performance testing tool?**

* HP Loader
* HTTP Load
* Proxy Sniffer
* Rational Performance Tester
* JMeter
* Borland Silk Performer

**6) Why does JMeter become a natural choice of the tester when it comes to performance testing?**

JMeter tool has benefits like

* It can be used for testing both static resources like [HTML](https://career.guru99.com/top-50-html-interview-questions/) and JavaScript, as well as dynamic resources like Servlets, Ajax, JSP, etc.
* JMeter has a tendency to determine the maximum number of concurrent users that your website can handle
* It provides a variety of graphical analyses of performance reports

**7) What all thing involves in Performance Testing Process?**

Performance Testing lifecycle includes the following steps/phases

* **Right testing environment:** Figure out the physical test environment before carrying performance testing, like hardware, software and network configuration
* **Identify the performance acceptance criteria:**It contains constraints and goals for throughput, response times and resource allocation
* **Plan and design Performance tests:** Define how usage is likely to vary among end users and find key scenarios to test for all possible use cases
* **Test environment configuration:**Before the execution, prepare the testing environment and arranges tools, other resources, etc.
* **Test design implementation:**According to your test design, create a performance test
* **Run the tests:**Execute and monitor the tests
* **Analyze, tune and retest:**Analyze, consolidate and share test results. After that, fine tune and test again to see if there is any enhancement in performance. Stop the test, if CPU is causing bottlenecks.

**8)  Name important parameters considered for performance testing?**

* Memory usage
* Processor usage
* Bandwidth
* Memory pages
* Network output queue length
* Response time
* CPU interruption per second
* Committed memory
* Thread counts
* Top waits, etc.

**9)  What are the important factors you must consider before selecting performance tools?**

* Customer preference tool
* Availability of license within customer machine
* Availability of test environment
* Additional protocol support
* License cost
* Efficiency of tool
* User options for Manual Testing
* Vendor support

**10) What is the difference between JMeter and SOAPUI?**

|  |  |
| --- | --- |
| **JMeter** | **SoapUI** |
| * It is used for load and performance testing HTTP, [JDBC](https://career.guru99.com/top-50-jdbc-interview-questions-and-answers/), JMS, Web Service(SOAP), etc. * It supports distributed load testing * ——– | * It is specific for web services and has a more user-friendly IDE * It does not support distributed load testing * For most IDE, it has plugin support |

**11) What is the difference between performance testing and Functional Testing?**

|  |  |
| --- | --- |
| **Functional Testing** | **Performance Testing** |
| * It helps to verify the accuracy of the software with definite inputs against expected output, and functional Testing is done. * This Testing can be done manually or automated * One user performs all the operations * Customer, Tester and Development involvement is required * Production sized test environment is not necessary, and H/W requirements are minimal | * To validate the behavior of the system at various load conditions performance testing is done. * It gives the best result if automated * Several users perform desired operations * Customer, Tester, Developer, DBA, and N/W management team * Requires close to a production test environment and several H/W facilities to populate the load |

**12) What are the benefits of LoadRunner on testing tools?**

Benefits of LoadRunner testing tools is

* Versatility
* Test Cases Results
* Easy Integrations
* Robust reports
* Enterprise Package

**13) What is Endurance Testing and Spike Testing?**

* **Endurance Testing:**It is one type of performance testing where the Testing is conducted to evaluate the behavior of the system when a significant workload is given continuously
* **Spike Testing:**It is also a type of performance testing that is performed to analyze the functioning of the system when the load is increased substantially.

**14)  What are the common mistakes done in Performance Testing?**

The common mistakes done in Performance Testing are

* Direct jump to multi-user tests
* Test results not validated
* Unknown workload details
* Too small run durations
* Lacking long duration sustainability test
* Confusion on a definition of concurrent users
* Data not populated sufficiently
* The significant difference between test and production environment
* Network bandwidth not simulated
* Underestimating software testing schedules
* Incorrect extrapolation of pilots
* Inappropriate base-lining of configurations

**15)  Explain the steps required in JMeter to create a performance test plan**

To create a performance test plan in JMeter

* Add thread group
* Add JMeter elements
* Add Graph result
* Run test & get the result

**16) How you can execute spike testing in JMeter?**

In JMeter, spike testing can be done by using Synchronizing Timer.  The threads are jammed by synchronizing the timer until a specific number of threads have been blocked and then release at once, creating a large instantaneous load.

**17) What is the throughput in Performance Testing?**

In performance testing, throughput is referred to the amount of data transported to the server in response to the client request at a given period of time. It is calculated in terms of requests per second, calls per day, reports per year, hits per second, etc. Performance of application depends on throughput value, higher the value of throughput -higher the performance of the application.

**18) What are the phases for automated performance testing?**

Phases for automated performance testing includes

* Design or Planning
* Build
* Execution
* Analyzing & Software Tuning

**19) What is the difference between the benchmark testing and baseline testing?**

* **Benchmark Testing:** It is the method of comparing the performance of your system performance tuning against an industry standard that is set by other organization
* **Baseline Testing:**It is the procedure of running a set of tests to capture performance information. When future change is made in the application, this information is used as a reference.

**20) What is concurrent user hits in load testing?**

In load testing, without any time difference when multiple users hit on the same event of an application under the load test is called a concurrent user hit.

1. API

**1) What are the tools used for API testing?**

The tools used for various API testing are

* SoapUI Pro
* PostMan
* Alertsite API monitoring

**2) What is API testing?**

API (Application Programming Interface) specifies how some software components should interact with other, in other words it’s a set of functions and procedures that allows the creation of applications which access the features or data of an application or [operating system](https://career.guru99.com/top-50-operating-system-interview-questions/).   Testing of these functions is known as API testing.

**3) What are the common tests performed on API’s?**

The common tests performed on API’s

* Verification of the API whether it is updating any [data structure](https://career.guru99.com/top-50-data-structure-interview-questions/)
* Verify if the API does not return anything
* Based on input conditions, returned values from the API’s are checked
* Verification of the API whether it triggers some other event or calls another API

**4) Mention the key difference between UI level testing and API testing?**

UI ( User Interface) refers to testing graphical interface such as how user interacts with the applications, testing application elements like fonts, images, layouts etc. UI testing basically focuses on look and feel of an application.While, API enables communication between two separate software systems. A software system implementing an API contains functions or sub-routines that can be executed by another software system

**5) Explain what is SOAP?**

SOAP-stands for Simple Object Access Control, and it is an [XML](https://career.guru99.com/xml-interview-questions/) based protocol for exchanging information between computers.

**6) Explain what is REST API?**

It is a set of functions to which the developers performs requests and receive responses. In REST API interaction is made via HTTP protocol

REST – stands for Representational State Transfer, it is quickly becoming defacto standard for API creation.

**7) Difference API and Unit Testing?**

|  |  |
| --- | --- |
| API testing | UNIT testing |
| * API is owned by QA team | * Unit testing is owned by development team |
| * API is mostly black box testing | * Unit testing is white box testing |
| * Full functionality of the system is considered in API testing as it will be used by the end-user (external developers who will use your API ) | * Unit testing is done to verify whether each unit in isolation performs as expected or not |
| * API test are often run after the build is ready and authors do not have access to the source code | * For each of their module the developers are expected to build unit tests for each of their code modules and have to ensure that each module pass unit test before the code is included in a build |

**8) How to test API’s ?**

To test the API’s you should follow the following steps

* Select the suite in which you want to add the API test case
* Choose test development mode
* Develop test cases for the desired API methods
* Configure application control parameters
* Configure test conditions
* Configure method validation
* Execute API test
* View test reports
* Filter API test cases
* Sequence API test cases

**9) Mention what the main areas to be taken in consideration while writing API document ?**

The key area to be considered when writing API documents are

* Source of the content
* Document plan or sketch
* Delivery layout
* Information required for each function in the document
* Automatic document creation programs

**10) In API document explain how to document each function ?What are the tools used for documentation?**

* **Description:** Small description about what a function does
* **Syntax:** Syntax about the parameter of the code, the sequence in which they occur, required and optional elements etc.
* **Parameters:** Functions parameters
* **Error Messages:** Syntax of error messages
* **Example Code:** Small snippet of code
* **Related Links:** Related functions

Popular tools used for API documentations are JavaDoc (for Java code ) Doxygen (for .Net code)

**11) Explain API framework?**

API framework is self-explanatory. Values for test run and for holding the configurable parts, config file is used.  Automated test cases must represent in “ parse-table” format within config file.  When testing API, it is not necessary to test each API so the config file have some section whose all API are activated for that specific run.

**12) How does the API Builder work?**

API Builder is a PLSQL program consists of four [SQL](https://www.guru99.com/sql-server-questions.html) files

* For setting API parameters and starting the process one file is responsible
* Two files are created for temporary tables and Master package to create the outputted code
* Fourth file creates “spooled” output of the code into a file called “output\_script\_.sql”

**13) Explain what is TestApi ?**

TestApi is a library of utility and test APIs that enables testers and developers to create testing tools and automated tests for .NET and Win32 application.  It provides a set of common test building blocks, types, data-structure and algorithms.

**14) What is Input injection and what are different ways of doing it ?**

Input Injection:  It is the act of simulating user input, in several ways you can simulate user input.

* Direct Method Invocation
* Invocation using an accessibility interface
* Simulation using low-level input
* Simulation using a device driver
* Simulation using a robot

**15) What are the main challenges of API testing?**

The main challenges in API testing is

* Parameter Selection
* Parameter Combination
* Call sequencing

**16) What is API testing with runscope ?**

Runscope is a web application that provides backend services and easy to use interface for testing APIs.

**17) Explain what are the principles of API test design?**

The principle for API test design are

* **Setup :** Create objects, start services, initialize data etc
* **Execution:** Steps to exercise API or scenario, also logging
* **Verification:** Oracles to evaluate execution outcome
* **Reporting:** Pass, failed or blocked
* **Clean up:** Pre-test state

**18) What are the types of Bugs will API testing finds?**

The types of Bugs, API will find

* Missing or duplicate functionality
* Fails to handle error conditions gracefully
* Stress
* Reliability
* Security
* Unused flags
* Not implemented errors
* Inconsistent error handling
* Performance
* Multi-threading issues
* Improper errors

**19) What are the tools used for API test automation?**

While testing Unit and API testing,  both target source code, if an API method is using code  based on .NET then the tool which is supporting should have .NET

Automation tools for API testing can be used are

* NUnit for .NET
* JUnit for Java
* HP UFT
* Soap UI

**20) Mention the steps for testing API ?**

API testing steps

* Select the test case that has to be fulfilled
* For API call develop a test case
* To meet the test case configure the API parameters
* Determine how will you validate a successful test
* Using programming language like PHP or .NET execute the API call
* Allow the API call to return the data to validate

**21) What are the common protocols that are testing in API tesing ?**

* HTTP
* JMS
* REST
* SOAP
* UDDI

1. Security

Q1. What is security testing?

**Answer:**

Security testing is a process where test cases are executed to reveal the defects in the security mechanism of the information systems. Tester plays an important role as attackers and playgrounds the system to find the defects related to security mechanisms. The goal of the security testing is to find the vulnerability in any application or system and protect their data from attackers.

Q2. What is SQL injection?

**Answer:**

SQL injection is a code injection technique which is used to attack data-driven systems in which malicious SQL statements are inserted into the entry field for execution. It is mostly known as an attack vector for websites but can be used to attack any type of sql database. SQL injection attacks allow attackers to spoof identity, tamper with existing data, cause repudiation issues such as voiding transactions or changing balances, allow the complete disclosure of all data on the system, destroy the data or make it otherwise unavailable, and become administrators of the database server.

Q3. What is vulnerability?

**Answer:**

Vulnerability is nothing but a weakness of any system. Using the vulnerability, bugs or attackers can easily attack the system. It can be avoided by performing security testing. If time to time, security testing is performed, then the chances of the vulnerability decreases. To protect the system from vulnerability or any attack, security testing is a must.

Q4. What is intrusion detection?

**Answer:**

Intrusion detection is a system which helps to determine possible attacks and deal with them. It collects information from various systems and sources. Then it analyzes this information and finds the possible ways of the attack on the system. It also checks the abnormal activities. Checks whether data of the system is altered or not.

Q5. What are the attributes of security testing?

**Answer:**

There are 7 attributes of security testing, namely authentication, authorization, integrity, non repudiation, confidentiality, availability, and resilience.

* **Authentication:** In authentication, the user’s identity are checked to provide access to the system.
* **Authorization:** In authorization, the authorities of the user are checked to access the resources.
* **Integrity:** Integrity ensures that data of the system are not altered.
* **Non repudiation:** Non repudiation is assurance that someone cannot deny the action they have done.
* **Confidentiality:** It ensures that information are kept private to authenticate users only.
* **Availability:** It ensures that system, application, and data are available for the user when they need them.
* **Resilience:** It is the ability of the entity to continuously deliver the intended outcome despite adverse cyber events.

1. Database testing

**1) What is Database Testing?**

Database Testing is also known as Backend Testing.   
Database[Testing](https://www.guru99.com/software-testing.html)is segmented into four different categories.

* Testing of Data Integrity
* Testing of Data Validity
* Data base related performance
* Testing of functions, procedure and triggers

**2) In database testing, what do we need to check normally?**

Normally, the things that we check in DB Testing are:

* Constraint Check
* Validation of a Field size
* Stored procedure
* Matching application field size to database
* Indexes for performance based issues

**3) Explain what is data driven test?**

In a data-table, to test the multi numbers of data, data-driven test is used. By using this it can easily replace the parameters at the same time from different locations.

**4) What are joins and mention different types of joins?**

Join is used to display two or more than two table and the types of joins are:

* Natural Join
* Inner Join
* Outer Join
* Cross Join

The outer join is divided again in two:

* Left outer join
* Right outer joinhrough HR Interviews

**5) What are indexes and mention different types of indexes?**

Indexes are database objects and they are created on columns. To fetch data quickly they are frequently accessed. Different types of indexes are:

* B-Tree index
* Bitmap index
* Clustered index
* Covering index
* Non-unique index
* Unique index

**6) While testing stored procedures what are the steps does a tester takes?**

The tester will check the standard format of the stored procedures and also it checks the fields are correct like updates, joins, indexes, deletions as mentioned in the stored procedure.

**7) How would you know for database testing, whether trigger is fired or not?**

On querying the common audit log you would know, whether, a trigger is fired or not. It is in audit log where you can see the triggers fired.

**8) In data base testing, what are the steps to test data loading?**

Following steps need to follow to test data loading

* Source data should be known
* Target data should be known
* Compatibility of source and target should be checked
* In[SQL](https://www.guru99.com/sql.html)Enterprise manager, run the DTS package after opening the corresponding DTS package
* You have to compare the columns of target and data source
* Number of rows of target and source should be checked
* After updating data in the source, check whether the changes appears in the target or not.
* Check NULLs and junk characters

**9) Without using Database Checkpoints, how you test a SQL Query in QTP?**

By writing scripting procedure in VBScript, we can connect to database and can test the queries and database.

**10) Explain how to use SQL queries in QTP ?**

In[QTP](https://www.guru99.com/quick-test-professional-qtp-tutorial.html)using output database check point and database check, you have to select the SQL manual queries option. After selecting the manual queries option, enter the “select” queries to fetch the data in the database and then compare the expected and actual.

**11) What is the way of writing testcases for database testing?**

Writing a testcases is like functional testing. First you have to know the functional requirement of the application. Then you have to decide the parameters for writing testcases like

* Objective: Write the objective that you would like to test
* Input method: Write the method of action or input you want to execute
* Expected: how it should appear in the database

**12) To manage and manipulate the test table what are the SQL statements that you have used in Database testing?**

The statements like SELECT, INSERT, UPDATE, DELETE are used to manipulate the table, while ALTER TABLE, CREATE TABLE and DELETE TABLE are used to manage table.

**13) How to test database procedures and triggers?**

To test database procedures and triggers, input and output parameters must be known. EXEC statement can be used to run the procedure and examine the behaviour of the tables.

* Open the database project in solution explorer
* Now in View menu, click the database schema
* Open the project folder from schema View menu
* Right click on the object that has to be tested, and then click on the dialog box that says Create Unit Tests
* After that create a new language test project
* Select either a) Insert the unit test or b) Create a new test and then click OK
* Project that has to be configured will be done by clicking on the Project Configuration dialog box.
* Once it configured click on OK

**14) How you can write testcases from requirements and do the requirements represents exact functionality of AUT (Application Under Test)?**

To write a testcases from requirements, you need to analyse the requirements thoroughly in terms of functionality. Thereafter you think about the appropriate testcases design techniques like Equivalence partitioning, Black box design, Cause effect graphing etc. for writing the testcases.Yes, the requirements represent exact functionality of AUT.

**15) What is DBMS?**

DBMS stand for Database management system, there are different types of DBMS

* Network Model
* Hierarchical Model
* Relational Model

**16) What is DML?**

DML stands for Data Manipulation Language, It is used to manage data with schema objects. It is a subset of SQL.

**17) What are DCL commands? What are the two types of commands used by DCL?**

DCL stands for Data Control Language, it is used to control data.

The two types of DCL Commands are:

Grant: By using this command user can access privilege to database

Revoke: By using this command user cannot access the database

**18) What is white box testing and black box testing?**

Black box testing means testing the software for the outputs on giving particular inputs. This testing is usually performed to see if the software meets the user’s requirements. There is no specific functional output expected for running this test.

The white box testing is done to check the accuracy of code and logic of the program. This testing is done by the programmer who knows the logical flow of the system.

**19) How does QTP evaluate test results?**

Once the testing is done, QTP will generate a report. This report will show the checkpoints, system message and error that were detected while testing. The test results window will show any mismatches encountered at the checkpoints.

**20) Explain the QTP testing process?**

* QTP testing process is based on following steps:
* Creating GUI (Graphical User Interface) Map files : Identifies the GUI object which has to be tested
* Creating test scripts: Test scripts are recorded
* Debug tests: Test should be debugged
* Run tests: Testcases should be run.
* View results: The results reflects the success or failure of the tests
* Report detects: If the test is failed, the reasons will be recorded in the report detect file

**21) What is load testing and give some examples of it?**

To measure the system response, load testing is done. If the load exceeds the users pattern it is known as stress testing. Examples of load testing are downloading the set of large files, executing multiple applications on a single computer, subjecting a server to large number of e-mails and allotting many tasks to a printer one after another.

**22) How to test database manually?**

Testing the database manually involves checking the data at the back end and to see whether the addition of data in front end is affecting the back end or not, and same for delete, update, insert etc.

**23) What RDBMS stands for and what are the important RDMBS that SQL use?**

RDBMS stands for Relational Database Management Systems that use SQL, and the important RDBMS that SQL uses are Sybase, Oracle, Access ,Ingres, Microsoft SQL server etc.

**24) What is performance testing and what are the bottlenecks of performance testing?**

Performance testing determines the speed of the computer system performance. It includes the quantitative tests like response time measurement. The problem in performance testing is that you always need a well-trained and experienced man power also the tools you use are expensive.

**25) What is DDL and what are their commands?**

To define database structure, a Developer uses DDL. DDL stands for Data Definition Language. The various DDL commands include Create, Truncate, Drop, Alter, Comment and Rename.

1. Automation

1. What is Automation Testing?

Automation testing is the process of testing a software or application using an automation testing tool to find the defects. In this process, executing the test scripts and generating the results are performed automatically by automation tools. It is required when we have a huge amount of [regression test cases](https://www.softwaretestingmaterial.com/regression-testing/). Some most popular tools to do automation testing are HP QTP/UFT, [Selenium WebDriver](https://www.softwaretestingmaterial.com/install-selenium-webdriver/), etc.ve Common Automation Problems Using Katalon

2. What are the benefits of Automation Testing?

This is one of the common interview questions in any Automation testing job.

1. Saves time and money. Automation testing is faster in execution.
2. Reusability of code. Create one time and execute multiple times with less or no maintenance.
3. Easy reporting. It generates automatic reports after test execution.
4. Easy for compatibility testing. It enables parallel execution in the combination of different OS and browser environments.
5. Low-cost maintenance. It is cheaper compared to manual testing in a long run.
6. Automated testing is more reliable.
7. Automated testing is more powerful and versatile. Automation tools allow us to integrate with [Cross Browser Testing](https://www.softwaretestingmaterial.com/run-selenium-tests-on-browserstack/) Tools, [Jenkins](https://www.softwaretestingmaterial.com/setup-integration-jenkins-ci-tools/), [Github](https://www.softwaretestingmaterial.com/selenium-continuous-integration/" \t "_blank), etc.,
8. It is mostly used for regression testing. Supports execution of repeated test cases.
9. Minimal manual intervention. Test scripts can be run unattended.
10. Maximum coverage. It helps to increase the test coverage.

3. What are the challenges and limitations of Selenium WebDriver?

As we all know Selenium WebDriver is a tool that automates the browser to mimic real user actions on the web. Selenium is a free open source testing tool. Some of the challenges with Selenium WebDriver are as follows

1. We cannot test the windows application
2. We cannot test mobile apps
3. Limited reporting
4. Handling dynamic Elements
5. Handling page load
6. Handling pop up windows
7. Handling captcha

4. What type of tests have you automated?

Our main focus is to automate test cases to do [Regression testing](https://www.softwaretestingmaterial.com/regression-testing/), [Smoke & Sanity testing](https://www.softwaretestingmaterial.com/smoke-testing-vs-sanity-testing/). Sometimes based on the project and the test time estimation, we do focus on End to End testing.

5. How many test cases you have automated per day?

It is one of the Selenium Tricky Interview Questions.

Actually, it depends on Test case scenario complexity and length. I did automate 2-5 test scenarios per day when the complexity is limited. Sometimes just 1 or fewer test scenarios in a day when the complexity is high.

6. What is a Framework?

A framework defines a set of rules or best practices that we can follow in a systematic way to achieve the desired results. There are different types of automation frameworks and the most common ones are:

* [Data-Driven Testing Framework](https://www.softwaretestingmaterial.com/data-driven-framework-selenium-webdriver/)
* Keyword Driven Testing Framework
* Hybrid Testing Framework
* Behavioural Driven Framework

7. What type of test cases to be automated?

Types of Test Cases To Automate are

* Data-driven test cases
* Test cases with higher complexity
* Test case with many database updates
* The test execution rate is high
* Smoke/Critical tests
* Tests with several combinations
* Graph test cases
* Higher manual execution time

8. What type of test cases not to be automated?

Types of Test Cases Not To Be Automated are

* Subjective Validation
* New Functionalities
* Strategic Development
* User Experience
* Complex Functionality
* Quality Control
* Low return on investment
* Installation and setup testing

9. What are the advantages of the Test Automation Framework?

1. Reusability of code.
2. Easy reporting.
3. Low-cost maintenance.
4. Maximum Coverage
5. Minimal manual intervention

10. Have you created any Framework?

**If you are a beginner:**You can say “No, I didn’t get a chance to create a framework from the scratch. I have used the framework which is already available. My contribution is mostly in creating test cases by using the existing framework.”

**If you are a beginner but have good knowledge of creating framework:** You can say “Yes, I have involved in developing framework along with other automation testers in my company.”

**If you are an experienced tester:**You can say “I have contributed to developing framework.” or You can say “Yes, I have created a framework from the scratch. There was no automation process in my previous company. I designed the framework from the scratch.”.

11. Why do you prefer Selenium Automation Tool?

I prefer Selenium Automation Tool because some of the benefits of Selenium to do automation testing are

* **Free and open source –** It is a free open source tool. There is no need to allot budget for this tool
* **Help –** Have large user base and helping communities.
* [**Cross-browser compatibility**](https://www.softwaretestingmaterial.com/what-is-cross-browser-testing/)**–** It works on almost all popular browsers such as Chrome, Firefox, Internet Explorer, and Safari.
* **Cross Platform compatibility –** It works on platforms such as Windows, Linux, Mac.
* **Multiple programming languages –** It supports programming languages such as [Java](https://www.softwaretestingmaterial.com/java-tutorial/), Phyton, Perl, Php, C#, Ruby, etc.,
* **Parallel Execution –** Selenium Grid supports parallel execution of Selenium Scripts.
* **Continuous Integration –** We can achieve nightly execution using Jenkins.

12. What is Selenium?

Selenium is an open source (free) automated testing suite to test web applications. It supports different platforms and browsers. It has gained a lot of popularity in terms of web-based automated testing and giving a great competition to the famous commercial tool HP QTP (Quick Test Professional) AKA HP UFT (Unified Functional Testing).

Selenium is a set of different software tools. Each tool has a different approach in supporting web based automation testing.

It has four components namely,

1. Selenium IDE (Selenium Integrated Development Environment)
2. Selenium RC (Selenium Remote Control)
3. Selenium WebDriver
4. Selenium Grid

13. What is Selenium IDE?

Selenium IDE (Integrated Development Environment) is a Firefox plugin. It is the simplest framework in the Selenium Suite. It allows us to record and playback the scripts. Even though we can create scripts using Selenium IDE, we need to use Selenium RC or Selenium WebDriver to write more advanced and robust test cases.

14. What is Selenese?

Selenese is the language that is used to write test scripts in Selenium IDE.

15. Which is the only browser that supports Selenium IDE to be used?

Firefox and Chrome. However, as Selenium IDE is community-powered, regular updates and compatibility with new browser versions cannot be ensured.

Back in 2017 when it no longer worked with Firefox’s latest version, users switched to [Katalon Recorder](https://www.softwaretestingmaterial.com/go/katalon-selenium-interview-questions/%22%20/t%20%22_blank" \t "_blank). It supports the same commands, extension scripts, data-driven testing, and advanced test reporting platform with TestOps.

16. What is Selenium RC?

Selenium RC AKA Selenium Remote control / Selenium 1. Selenium Remote Control was the main Selenium project for a long time before the WebDriver merge brought up Selenium 2. Selenium 1 is still actively supported (in maintenance mode). It relies on JavaScript for automation. It supports Java, Javascript, Ruby, PHP, Python, Perl, and C#. It supports almost every browser out there.

17. What is Selenium WebDriver?

Selenium WebDriver AKA Selenium 2 is a browser automation framework that accepts commands and sends them to a browser. It is implemented through a browser-specific driver. It controls the browser by directly communicating with it. Selenium WebDriver supports Java, C#, PHP, Python, Perl, Ruby.

18. What is the difference between Selenium 3 and Selenium 4?

We all know that Selenium 4 was released as a stable version on October 13, 2021. So here in this post, we have covered Selenium 4 Interview Questions & Answers. First Let’s see the difference between Selenium 3 and Selenium 4.

**Selenium 3 –** JSON wire protocol was used to communicate between the Selenium Webdriver APIs and the browser native APIs. All the requests and responses communicated across the protocol were encoded & decoded.

**Selenium 4 –** Follows the W3C standard protocol. Due to this request and the response communicated across the protocol doesn’t require the encoding and decoding API.

19. What is Selenium Grid?

Selenium Grid is a tool used together with Selenium RC to run tests on different machines against different browsers in parallel. That is, running multiple tests at the same time against different machines running different browsers and operating systems.

In simple words, it is used to distribute your test execution on multiple platforms and environments concurrently.

20. When do you use Selenium Grid?

Selenium Grid can be used to execute same or different test scripts on multiple platforms and browsers concurrently so as to achieve distributed test execution

21. What are the advantages of Selenium Grid?

It allows running test cases in parallel thereby saving test execution time.   
It allows multi-browser testing   
It allows us to execute test cases on multi-platform

22. What is a hub in Selenium Grid?

A hub is a server or a central point that controls the test executions on different machines.

23. What is a node in Selenium Grid?

Node is the machine which is attached to the hub. There can be multiple nodes in Selenium Grid.

24. What are the types of WebDriver APIs available in Selenium?

* Firefox Driver
* Gecko Driver
* InternetExplorer Driver
* Chrome Driver
* HTMLUnit Driver
* Opera Driver
* Safari Driver
* Android Driver
* iPhone Driver
* EventFiringWebDriver

25. Which WebDriver implementation claims to be the fastest?

The fastest implementation of WebDriver is the HTMLUnitDriver. It is because the HTMLUnitDriver does not execute tests in the browser. Starting a browser and running test cases took more time compared to running the scripts without a browser. HTMLUnitDriver took a simple HTTP request-response mechanism for test case execution.

26. What are the Programming Languages supported by Selenium WebDiver?

* [Java](https://www.softwaretestingmaterial.com/java-tutorial/)
* C#
* Python
* Ruby
* Perl
* PHP

27. Which language is not supported by selenium?

Selenium supports all major programming languages such as Java, C#, Perl, Python, Ruby, PHP, Scala and Groovy. As of today, others are not compatible.

28. What are the Operating Systems supported by Selenium WebDriver?

* Windows
* Linux
* Mac OS X
* iOS
* Android

29. What are the testing types that can be supported by selenium?

Testing types that can be supported by Selenium are as follows:

* Functional Testing
* Regression Testing
* Retesting
* Acceptance Testing
* End-to-End Testing
* Smoke Testing
* Sanity Testing
* Responsive Testing
* Cross Browser Testing
* UI Testing
* Integration Testing

30. What are the Open-source Frameworks supported by Selenium WebDriver?

* JUnit
* TestNG

31. What are the Locators available in Selenium?

In Selenium WebDriver, there are 8 different types of locators:

1. ID – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-id-locator/)
2. ClassName – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-class-name-locator/)
3. Name – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-name-locator/)
4. TagName – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-tag-name-locator/)
5. LinkText – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-link-text-and-partial-link-text-locator/)
6. PartialLinkText – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-link-text-and-partial-link-text-locator/)
7. XPath – [Practical example](https://www.softwaretestingmaterial.com/how-to-locate-element-by-xpath-locator/)
8. CSS Selector – [Practical example](https://www.softwaretestingmaterial.com/css-selector-selenium-webdriver-tutorial/)

32. What is an XPath?

XPath is used to locate the elements. Using XPath, we could navigate through elements and attributes in an XML document to locate web elements such as textbox, button, checkbox, Image etc., in a web page.

33. When you use these locators ID, Name, XPath, Or CSS Selector?

**ID** & **Name** locators will be used when there are unique identifiers & unique names available on the web page.   
**CSS Selector** can be used for performance and when ID & Name locators are not unique.   
**XPath** is used when there is no preferred locators.

34. What is the difference between “/” and “//”

**Single Slash “/” –**Single slash is used to create XPath with absolute path i.e. the XPath would be created to start selection from the document node/start node.

**Double Slash “//” –** Double slash is used to create XPath with relative path i.e. the XPath would be created to start selection from anywhere within the document.

35. What is the difference between Absolute Path and Relative Path?

Absolute XPath starts from the root node and ends with desired descendant element’s node. It starts with top HTML node and ends with input node. It starts with a single forward slash(/) as shown below.

Relative XPath starts from any node in between the HTML page to the current element’s node(last node of the element). It starts with a double forward slash(//) as shown below.

36. What should you do when even XPath functions can’t identify the web element?

In the early stages of software developement, developers change identifiers and elements quite often. During the execution, the web elements may change dynamically and we cannot identify the web elements. To overcome this we use XPath axes along with XPath functions.

37. What are XPath Axes?

XPath axes are used to search for the multiple nodes in the XML document from the context (current) node.

XPath axes are used to find dynamic elements that would otherwise be impossible using standard locators.

38. What is a Context Node?

The context node is the node the XPath processor is currently looking at.

39. What is the difference between Assert and Verify in Selenium?

**Assert:** In simple words, if the assert condition is true then the program control will execute the next test step but if the condition is false, the execution will stop and further test step will not be executed.

**Verify:** In simple words, there won’t be any halt in the test execution even though the verify condition is true or false.

40. What are Soft Assert and Hard Assert in Selenium?

Soft Assert: Soft Assert collects errors during @Test Soft Assert does not throw an exception when an assert fails and would continue with the next step after the assert statement.

Hard Assert: Hard Assert throws an AssertException immediately when an assert statement fails and test suite continues with next @Test

41. What is the difference between setSpeed () and sleep () methods?

Both sleep() and setSpeed() are used to delay the execution speed.

**setSpeed():** It set up speed that will apply a delay time before every Selenium operation.

**Example:** setSpeed(“5000”) – It waits for 5 seconds

**sleep():** It set up wait only for once when called in our Selenium script.

**Example:** sleep(5000) – It waits for 5 seconds

**Note:**setSpeed method is applicable to Selenium IDE and Selenium RC. We cannot use setSpeed in Selenium WebDriver.

42. What are the verification points available in Selenium?

In Selenium IDE, we use Selenese Verify and Assert Commands as Verification points   
In Selenium WebDriver, there is no built-in features for verification points. It totally depends on our coding style. some of the Verification points are

* To check for page title
* To check for certain text
* To check for certain element (text box, button, drop down, etc.)

43. What is the super interface of WebDriver?

SearchContext acts as the super interface of Web Driver.

44. What is WebElement selenium?

WebElement in Selenium represents an HTML element. It basically represents a DOM element in a HTML document.

45. What are the different exceptions you have faced in Selenium WebDriver?

Some of the exceptions I have faced in my current project are

1. ElementNotVisibleException
2. StaleElementReferenceException

**Element Not visible Exception:**

This exception will be thrown when you are trying to locate a particular element on webpage that is not currently visible eventhough it is present in the DOM. Also sometimes, if you are trying to locate an element with the xpath which associates with two or more element.

**Stale Element Reference Exception:**

A [stale element reference exception](https://www.softwaretestingmaterial.com/stale-element-reference-exception-selenium-webdriver/) is thrown in one of two cases, the first being more common than the second.

The two reasons for Stale element reference are

1. The element has been deleted entirely.
2. The element is no longer attached to the DOM.

We face this stale element reference exception when the element we are interacting is destroyed and then recreated again. When this happens the reference of the element in the DOM becomes stale. Hence we are not able to get the reference to the element.

Some other exceptions we usually face are as follows:

* WebDriverException
* IllegalStateException
* TimeoutException
* NoAlertPresentException
* NoSuchWindowException
* NoSuchElementException

46. What are the types of waits available in Selenium WebDriver?

In Selenium we could see three types of waits such as Implicit Waits, Explicit Waits and Fluent Waits.

* Implicit Waits – [Click to view detailed post](https://www.softwaretestingmaterial.com/implicit-waits-selenium-webdriver/)
* Explicit Waits – [Click to view detailed post](https://www.softwaretestingmaterial.com/webdriverwait-selenium-webdriver/)
* Fluent Waits – [Click to view detailed post](https://www.softwaretestingmaterial.com/selenium-fluentwait/)

47. What is Implicit Wait In Selenium WebDriver?

Implicit waits tell to the WebDriver to wait for a certain amount of time before it throws an exception. Once we set the time, WebDriver will wait for the element based on the time we set before it throws an exception. The default setting is 0 (zero). We need to set some wait time to make WebDriver to wait for the required time.

48. What is WebDriver Wait In Selenium WebDriver?

WebDriverWaitis applied on a certain element with defined expected condition and time. This wait is only applied to the specified element. This wait can also throw an exception when an element is not found.

49. What is Fluent Wait In Selenium WebDriver?

FluentWait can define the maximum amount of time to wait for a specific condition and frequency with which to check the condition before throwing an “ElementNotVisibleException” exception.

50. Is Selenium Server needed to run Selenium WebDriver Scripts?

When we are distributing our Selenium WebDriver scripts to execute using Selenium Grid, we need to use Selenium Server.

51. List some scenarios which we cannot automate using Selenium WebDriver?

1. Bitmap comparison is not possible using Selenium WebDriver   
2. Automating Captcha is not possible using Selenium WebDriver   
3. We can not read bar code using Selenium WebDriver

52. What is Object Repository in Selenium WebDriver?

Object Repository is used to store element locator values in a centralized location instead of hard coding them within the scripts. We do create a property file (.properties) to store all the element locators and these property files act as an object repository in Selenium WebDriver.

53. What is Page Object Model in Selenium?

[Page Object Model](https://www.softwaretestingmaterial.com/page-object-model/) is a Design Pattern which has become popular in Selenium Test Automation. It is widely used design pattern in Selenium for enhancing test maintenance and reducing code duplication. Page object model (POM) can be used in any [kind of framework](https://www.softwaretestingmaterial.com/types-test-automation-frameworks/) such as modular, [data-driven](https://www.softwaretestingmaterial.com/data-driven-framework-selenium-webdriver/), keyword driven, hybrid framework etc.  A page object is an object-oriented class that serves as an interface to a page of your Application Under Test(AUT). The tests then use the methods of this page object class whenever they need to interact with the User Interface (UI) of that page. The benefit is that if the UI changes for the page, the tests themselves don’t need to change, only the code within the page object needs to change. Subsequently, all changes to support that new UI is located in one place.

54. What is Page Factory?

We have seen that ‘Page Object Model’ is a way of representing an application in a test framework. For every ‘page’ in the application, we create a Page Object to reference the ‘page’ whereas a ‘Page Factory’ is one way of implementing the ‘Page Object Model’.

55. What is the difference between Page Object Model (POM) and Page Factory?

Page Object is a class that represents a web page and hold the functionality and members.   
Page Factory is a way to initialize the web elements you want to interact with within the page object when you create an instance of it.

56. What are the advantages of Page Object Model Framework?

**Code reusability** – We could achieve code reusability by writing the code once and use it in different tests.

**Code maintainability** – There is a clean separation between test code and page specific code such as locators and layout which becomes very easy to maintain code. Code changes only on Page Object Classes when a UI change occurs. It enhances test maintenance and reduces code duplication.

**Object Repository** – Each page will be defined as a java class. All the fields in the page will be defined in an interface as members. The class will then implement the interface.

**Readability** – Improves readability due to clean separation between test code and page specific code

57. How to Upload a file in Selenium WebDriver?

There are two cases which are majorly used to upload a file in Selenium WebDriver such as using SendKeys Method and using AutoIT Script.

58. How to Download a file in Selenium WebDriver?

By using AutoIT script, we could download a file in Selenium WebDriver.

59. How to connect a Database in selenium?

As we all know Selenium WebDriver is a tool to automate User Interface. We could only interact with Browser using Selenium WebDriver.

We use JDBC Driver to connect the Database in Selenium (While using Java Programming Language).

60. What is Continuous Integration?

Continuous Integration is abbreviated as CI. Continuous Integration is a development practice that aims to make sure the correctness of software. After each commit, a suite of tests run automatically and test the software to ensure whether the software is running without any breaks. If any test fails, we will get immediate feedback say “build is broken”.

In simple words, continuous integration is a process of verifying the correctness of a software.

Some of the continuous integration tools are Jenkins, TeamCity, Bamboo, Travis, Circle Ci, Bitbucket.

We can schedule the test suite execution using these CI Tools.

61. Name some CI tools available in the Market?

Some of the best continuous testing softwares to use in your project.

* Selenium
* Katalon Studio
* Appium
* Unified Functional Testing
* Travis CI
* Egg Plant
* Watir
* Tricentis Tosca
* Test Sigma
* IBM Rational Functional Tester
* Test Complete
* QuerySurge
* JMeter
* Jenkins
* Bamboo
* Docker
* PagerDuty
* JIRA
* GitHub