Prepare for your software testing interview from our comprehensive list of over 100 interview questions. These interview questions are designed for both freshers and experienced. We will start with fairly simple questions and move to the more advanced level as the post progresses.

Ques.1. What is Software Testing?
Ans. Software testing is the process of evaluating a system to check if it satisfies its business requirements. It measures the overall quality of the system in terms of attributes like correctness, completeness, usability, performance etc. Basically, 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-

1. Testing provides an assurance to the stakeholders that product works as intended.
2. Avoidable defects leaked to the end user/customer without proper testing adds bad reputation to the development company.
3. Defects detected earlier phase of SDLC results into lesser cost and resource utilisation of correction.
4. Saves development time by detecting issues in earlier phase of development.
5. 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 can be stopped when one or more of the following conditions are met-

1. 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.
2. Once the testing deadline is met - Testing can be stoppped after deadlines get met with no high priority issues left in system.
3. 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.
4. 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. What is Quality Assurance?
Ans. Quality assurance is a process driven approach which checks if the process of developing the product is correct and conforming to all the standards. It is considered as a preventive measure as it identifies the weakness in the process to build a software. It involves activites like document review, test cases review, walkthroughs, inspection etc.

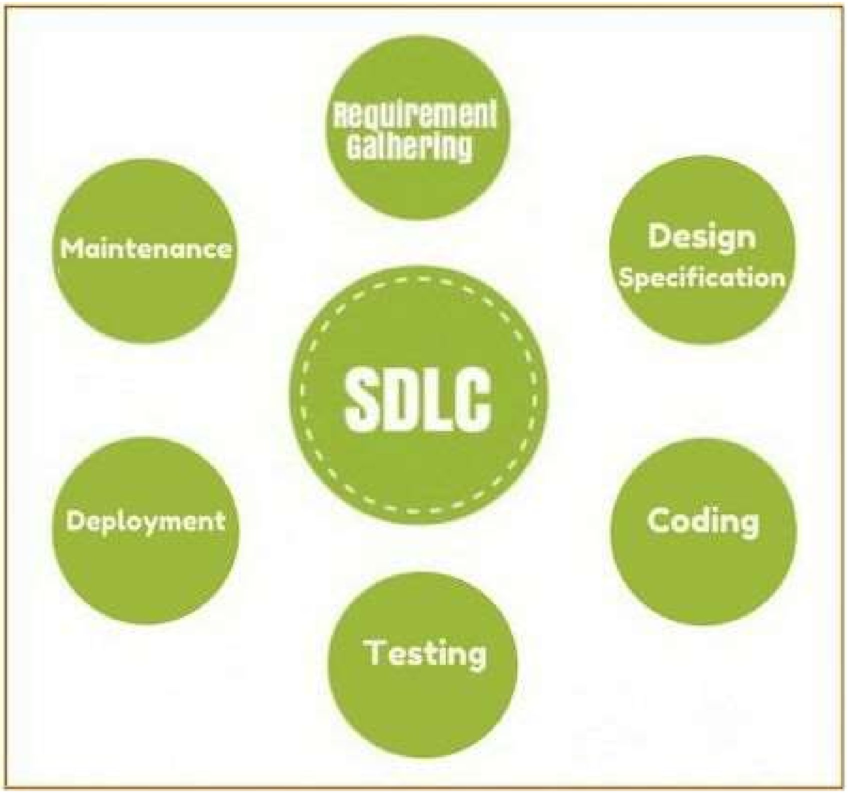
Ques.5. What is Quality Control?
Ans. Quality control is product driven approach which checks that the developed product conforms to all the specified requirements. It is considered as a corrective measure as it tests the built product to find the defects. It involves different types of testing like functional testing, performance testing, usability testing etc.

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

#	Verification	Validation
1.	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.
2.	It is static process of analysing the documents and not the actual end product.	It involves dynamic testing of software product by running it.
3.	Verification is a process oriented approach.	Validation is a product oriented approach.
4.	Answers the question - "Are we building the product right?"	Answers the question - "Are we building the right product?"

5.	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. Software Development Life Cycle refers to all the activities that are performed during software development, including - requirement analysis, designing, implementation, testing, deployment and maintenance phases.



Ques.8. Explain STLC - Software Testing life cycle.
Software testing life cycle 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, learnings 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** - Non functional testing includes testing the non-functional requirements of the system like performance, security, scalability, portability, endurance etc.

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

- **Black box testing** - 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** - 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, code 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 bed?
Ans. A test bed is a test environment used for testing an application. A test bed configuration can consist of the hardware and software requirement of the application under test including - operating system, hardware configurations, software configurations, tomcat, database etc.

Ques.11. What is a test plan?
Ans. A test plan 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.12. What is a test scenario?
Ans. A test scenario 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.13. What is a test case?
Ans. A test case 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.14. What are some attributes of a test case?
Ans. A test case can have following attributes-

1. TestCaseId - A unique identifier of the test case.
2. Test Summary - Oneliner summary of the test case.
3. Description - Detailed description of the test case.
4. Prerequisite or pre-condition - A set of prerequisites that must be followed before executing the test steps.
5. Test Steps - Detailed steps for performing the test case.
6. Expected result - The expected result in order to pass the test.
7. Actual result - The actual result after executing the test steps.
8. Test Result - Pass/Fail status of the test execution.
9. Automation Status - Identifier of automation - whether the application is automated or not.
10. Date - The test execution date.
11. Executed by - Name of the person executing the test case.

Ques.15. What is a test script?
Ans. A test script is an automated test case written in any programming or scripting langauge. These are basically a set of instructions to evaluate the functioning of an application.

Ques.16. 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.17. 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 resport are-

- DefectId - A unique identifier of the defect.
- Defect Summary - A one line summary of the defect, more like a defect title.
- Defect Description - A detailed description of the defect.
- Steps to reproduce - The steps to reproduce the defect.
- Expected Result - The expected behavior from which the application is deviating because of the defect.
- Actual Result- The current erroneous state of the application w.r.t. the defect.
- Defect Severity - Based on the criticality of the defect, this field can be set to minor, medium, major or show stopper.
- Priority - Based on the urgency of the defect, this field can be set on a scale of P0 to P3.

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 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 lifecycle or the different states of a bug.
Ans. A bug goes through the following phases in software development-

- New - A bug or defect when detected is in New state
- Assigned - The newly detected bug when assigned to the corresponding developer is in Assigned state
- Open - When the developer works on the bug, the bug lies in Open state
- Rejected/Not a bug - A bug lies in rejected state in case the developer feels the bug is not genuine
- Deferred - A deferred bug is one, fix of which is deferred for some time(for the next releases) based on urgency and criticality of the bug
- Fixed - When a bug is resolved by the developer it is marked as fixed

- Test - When fixed the bug is assigned to the tester and during this time the bug is marked as in Test
- Reopened - If the tester is not satisfied with issue resolution the bug is moved to Reopened state
- Verified - After the Test phase if the tester feels bug is resolved, it is marked as verified
- Closed - After the bug is verified, it is moved to Closed status.

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.
2. 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 intution 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 assumpation that any 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 behaviour based on different combination of input values.
- Cause-effect graph - Testing using graphical representation of input i.e. cause and output i.e. effect is used for test designing.
- 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, 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 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 behaviour 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 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 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 heirarchy.
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.

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 hardcoded 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 heirarchy.

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 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 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 is a type of acceptance testing that is performed end users at the developer site.

Ques.48. What is beta testing?
Ans. 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.

Ques.50. What is monkey testing?
Ans. Monkey testing is a type of testing that is performed randomly without any predefined test cases or test inputs.

Ques.51. How is monkey testing different from adhoc testing?
Ans. In case of adhoc testing although there are no predefined or

documented test cases still testers have the understanding of the application. While in case of monkey testing testers doesn't have any understanding of the application.

Ques.52. What is exploratory testing?
Ans. Exploratory testing is a type of testing in which new test cases are added and updated while exploring the system or executing test cases. Unlike scripted testing, test design and execution goes parallely in exploratory testing.

Ques.53. What is performance testing?
Ans. Performance testing is a type of non-functional testing in which the performance of the system is evaluated under expected or higher load. The various performance parameters evaluated during performance testing are - response time, reliability, resource usage, scalabilty etc.

Ques.54. What is load testing?
Ans. Load testing is a type of performance testing which aims at finding application's performance under expected workload. During load testing we evaluate the response time, throughput, error rate etc parameters of the application.

Ques.55. What is stress testing?
Ans. Stress testing is a type of performance testing in which application's behaviour is monitored under higher workload then expected. Stress testing is done to find memory leaks and robustness of the application.

Ques.56. What is volume testing?
Ans. Volume testing is a type of performance testing in which the performance of application is evaluated with large amount of data. It checks the scalability of the application and helps in identification of bottleneck with high volume of data.

Ques.57. What is endurance testing or Soak testing?
Ans. Endurance testing is a type of performance testing which aims at finding issues like memory leaks when an application is subjected to load test for a long period of time.

Ques.58. What is spike testing?
Ans. Spike testing is a type of performance testing in which the application's performance is measured while suddenly increasing the number of active users during the load test.

Ques.59. What is usability testing?
Ans. Usability testing is the type of testing that aims at determining the ease of using the application. It aims at uncovering the usability defects in the application.

Ques.60. What is Accessibility testing?
Ans. Accessibility is the type of testing which aims at determining the ease of use or operation of the application specifically for people with disabilities.

Ques.61. What is compatibility testing?
Ans. Testing software to see how compatible the software is with a particular environment - operating system, platform or hardware.

Ques.62. What is configuration testing?
Ans. Configuration testing is the type of testing used to evaluate the configurational requirements of the software along with effect of changing the required configuration.

Ques.63. What is localisation testing?
Ans. Localisation testing is a type of testing in which we evaluate the application's customization(localized version of application) in a particular culture, locale or country.

Ques.64. What is globalisation testing?
Ans. Globalisation testing is a type of testing in which application is evaluated for its functioning across the world in different cultures, languages, locale and countries.

Ques.65. What is negative testing?
Ans. Negative testing is a type of testing in which the application's robustness(graceful exiting or error reporting) is evaluated when provided with invalid input or test data.

Ques.66. What is security testing?
Ans. Security testing is a type of testing which aims at evaluating the integrity, authentication, authorization, availabilty, confidentiality and non-repudation of the application under test.

Ques.67. What is penetration testing?
Ans. Penetration testing or pen testing is a type of security testing in which application is evaluated(safely exploited) for different kinds of vulnerabilities that any hacker could expolit.

Ques.68. What is robustness testing?
Ans. Robustness testing is a type of testing that is performed to find the robustness of the application i.e. the ability of the system to behave gracefully in case of erroneous test steps and test input.

Ques.69. What is A/B testing?
A/B testing is a type of testing in which the two variants of the software product are exposed to the end users and on analysing the user behaviour on each variant, the better variant is chosen and used thereafter.

Ques.70. What is concurrency testing?
Ans. Concurrency testing is a multi-user testing in which an application is evaluated by analyzing application's behaviour with concurent users acccessing the same functioanity.

Ques.71. What is all pair testing?
Ans. All pair testing is a type of testing in which the application is tested with all possible combination of the values of input parameters.

Ques.72. What is failover testing?
Ans. Failover testing is a type of testing that is used to verify application's ability to allocate more resources(more servers) in case of failure and transfering of the processing part to back-up system.

Ques.73. What is fuzz testing?
Ans. Fuzz testing is a type of testing in which large amount of random data is provided as input to the application in order to find security loopholes and other issues in the application.

Ques.74. What is UI testing?
Ans. UI or user interface testing is a type of testing that aims at finding Graphical User Interface defects in the application and checks that the GUI conforms to the specifications.

Ques.75. What is risk analysis?
Ans. Risk analysis is the analysis of the risk identified and assigning an appropriate risk level to it based on its impact over the application.

Ques.76. What is the difference between regression and retesting?
Ans. Regression testing involves testing the application to verify that a new code change doesn't affect the other parts of the application. Whereas, in retesting, we verify if the fixed issue is resolved or not.

Ques.77. What is the difference between blackbox and whitebox testing?
Ans. Blackbox testing is a type of testing in which internal architecture of the code is not required for testing. It is usaually applicable for system and acceptance testing.
Whereas whitebox testing requires internal design and implementation knowledege of the application being tested. It is usually applicable for Unit and Integration testing.

Ques.78. What is the difference between smoke and sanity testing?
Ans. The difference between smoke and sanity testing is-

- Smoke testing is a type of testing in which the all major functionalities of the application are tested before carrying out exhaustive testing. Whereas, sanity testing is subset of regression testing which is carried out when there is some minor fix in application in a new build.
- In smoke testing, shallow-wide testing is carried out while in sanity narrow-deep testing (for a particular functionality) is done.

- The smoke tests are usually documented or are automated. Whereas, the sanity tests are generally not documented or unscripted.

Ques.79. What is code coverage?
Ans. Code coverage is the measure of the amount of code covered by the test scripts. It gives the idea of the part of the application covered by the test suite.

Ques.80. What is cyclomatic complexity?
Ans. Cyclomatic complexity is the measure of the number of independent paths in an application or program. This metric provides an indication of the amount of effort required to test complete functionality. It can be defined by the expression -
L – N + 2P, where:
L is the number of edges in the graph
N is the number of node
P is the number of disconnected parts

Ques.81. What is dynamic testing?
Ans. Testing performed by executing or running the application under test either manually or using automation.

Ques.82. What is an exit criteria?
Ans. An exit criteria is a formal set of conditions that specify the agreed upon features or state of application in order to mark the completion of the process or product.

Ques.83. What is requirement traceability matrix(RTM)?
Ans. In software testing, a traceability matrix is a table that relates the high level requirements with either detailed requirements, test plans or test cases. RTM helps in ensuring 100% test coverage.

Ques.84. What is pilot testing?
Ans. Pilot testing is a testing carried out as a trial by limited number of users to evaluate the system and provide their feedback before the complete deployment is carried out.

Ques.85. What is backend testing?
Ans. Backend testing is a type of testing that invloves testing the backend of the system which comprises of testing the databases and the APIs in the application.

Ques.86. What are some advantages of automation testing?
Ans. Some advantages of automation testing are-

1. Test execution using automation is fast and saves considerable amount of time.
2. Carefully written test scripts remove the chance of human error during testing.
3. Tests execution can be scheduled for nightly run using CI tools like Jenkins which can also be configured to provide daily test results to relevant stakeholders.
4. Automation testing is very less resource intensive. Once the tests are automated, test execution requires almost no time of QAs. Saving Qa bandwidth for other explratory tasks.

Ques.87. What are some disadvantages of automation testing?
Ans. Some advantages of automation testing are-

1. It requries skilled automation testing experts to write test scritps.
2. Additional effort to write scripts is required upfront.
3. Automation scripts are limited to verification of the tests that are coded. These tests may miss some error that is very glaring and easily identifiable to human(manual QA).
4. Even with some minor change in application, script updation and maintenance is required.

Ques.88. What is mutation testing?
Ans. Mutation testing is a type of white box testing in which the source code of the application is mutated to cause some defect in its working. After that the test scripts are executed to check for their correctness by verifying the failures caused the mutant code.

Ques.89. Write test cases for Pen.
Ans. Test cases of Pen

Ques.90. Write test cases for ATM Machine.
Ans. Test cases of ATM Machine

Ques.91. Write test cases for Login.
Ans. Test cases of Login Page

Ques.92. Write test cases for Lift.
Ans. Test cases of Lift

Ques.93. Write test cases for Lift.
Ans. Test cases of E-commerce application

> ## For other test scenarios check our Test Scenarios Examples section.

Ques.94. What should be the psychology testing?

The two main stakeholders in software development life cycle - Testers and Developers have different mindsets while approaching an application. Testers tend to have a more stringent approach of examining the software. Most of the time they are looking to "break the application". Whereas, developers have the mindset to "make the application work". ISTQB has defined certain psychological factors that influence the success of testing-

- Independence - Testers should enjoy a certain degree of independence while testing the application rather than following a straight path. This can be achieved by different apporaches like off-shoring the QA process, getting test strategies and other test plans by someone not following any sort of bias.
- Testing is often looked upon as destructive activity as it aims at finding flaws in system. But QA personnel should present testing as required and integral part, presenting it as constructive activity in overall sofware development lifecycle by mitigating the risks at early stages.
- More often than not, tester and developers are at the opposite end of spectrum. Testers need to have good interpersonal skills to

communicate their findings without induging in any sort of tussle with the developers.

- All the communication and discussions should be focussed on facts and figures(risk analysis, priority setting etc). Emphasizing that the collaborative work of developers and testers will lead to better software.
- Empathy is one characterstic that definitely helps in testing. Testers empathizing with developers and other project stakeholders will lead to better relation between the two. Also, empathizing with end users will lead to a software with better usability of the product.
- Reference- ISTQB Foundation Level Syllabus - The Certified Tester Foundation Level in Software Testing

Ques.95. What is the difference between testing and debugging?
Ans. Testing is the primarily performed by testing team in order to find the defects in the system. Whereas, debugging is an activity performed by development team. In debugging the cause of defect is located and fixed. Thus removing the defect and preventing any future occurrence of the defect as well.
Other difference between the two is - testing can be done without any internal knowledge of software architecture. Whereas debugging requires knowledge of the software architecture and coding.

Ques.96. Explain Agile methodology?
Ans. Agile methodology of software development is based on interative and increamental approach. In this model, the application is broken down into smaller build on which different cross functional team work together providing rapid delivery along with adapting to changing needs at the same time.

Ques.97. What is scrum?
Ans. A scrum is a process for implementing Agile methodology. In scrum, time is divided into sprints and on completion of sprints, a deliverable is shipped.

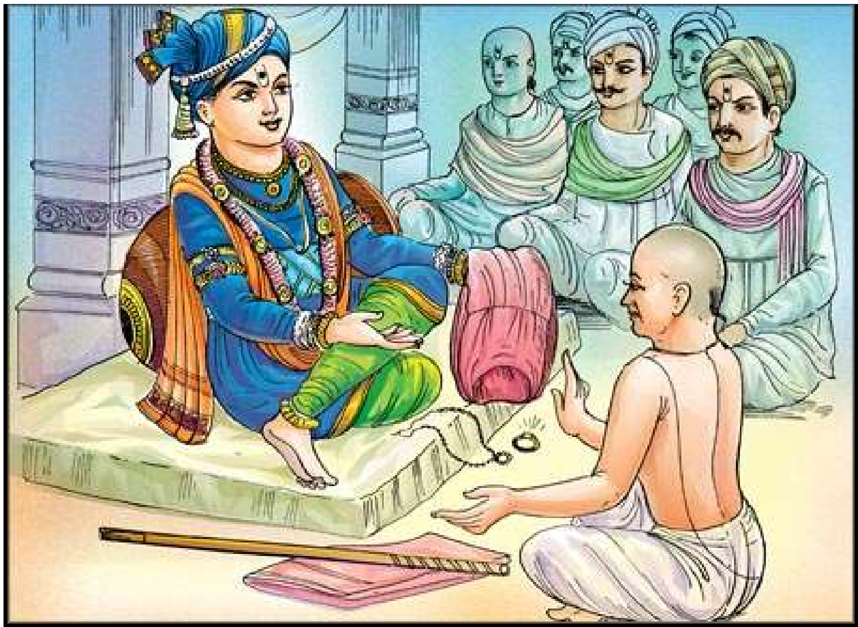
Ques.98. What are the different roles in scrum?
Ans. The different roles in scrum are -

1. Product Owner - The product owner owns the whole development of the product, assign tasks to the team and act as an interface between the scrum team(development team) and the stakeholders.

2. Scrum Master - The scrum master monitors that scrum rules get followed in the team and conducts scrum meeting.
3. Scrum Team - A scrum team participate in the scrum meetings and perform the tasks assigned.

Ques.99. What is a scrum meeting?
Ans. A scrum meeting is daily meeting in scrum process. This meeting is conducted by scrum master and update of previous day's work along with next day's task and context is defined in this meeting.

Ques.100. Explain TDD (Test Driven Development).
Ans. Test Driven Development is a software development methodology in which the development of the software is driven by test cases created for the functionality to be implemented. In TDD, first the test cases are created and then code to pass the tests is written. Later the code is refactored as per the standards.



Sundarji Suthar was the administrator for the King of Bhuj. Once, he came to Gondal with the wedding procession of the king's son. Learning that Shriji Maharaj was nearby in Bandhiya, he went for darshan. Maharaj asked, "Who has come?" Sundarji replied, "Your servant." Maharaj said, "What are the qualities of a servant?" Sundarji replied, "To do as told." Maharaj said, "Renounce these clothes and ornaments and become a sadhu." Sundarji Suthar became a sadhu! Then Maharaj told him to walk to Kashi. Maharaj asked Muktanand Swami, "I've shaved the head of the King's administrator. I have done a great thing today, haven't I?" Muktanand Swami replied, "What is great in that? We only had one devotee in Kutch. We would stay at his house. Now, we have no place!"

"Really?" replied Maharaj, "In that case, call him back." When he arrived, he bowed at Maharaj's feet. "Who has come?" inquired Maharaj. "Your servant," replied Sundarji. Maharaj said, "If you are a servant then put on the clothes and jewels you had before." Sundarji Suthar obeyed.

However, he was proud that he had followed Maharaj's commands. So, he told Maharaj, "It is okay that you have done this to me, but do not test others in this way. They will not be able to pass this kind of difficult test.”

Later, to dissolve his pride Shriji Maharaj wrote a letter to 18 well-to-do devotees commanding them to become sadhus and immediately go to Bhuj. All 18 left their homes, became sadhus and went to Bhuj. On seeing them Sundarji’s pride dissolved.

INSPIRATION: Even if we do something good in Satsang we should not harbour ego for doing it.

Sundarji Suthar was the administrator for the King of Bhuj. Once, he came to Gondal with the wedding procession of the king’s son. Learning that Shriji Maharaj was nearby in Bandhiya, he went for darshan. Maharaj asked, “Who has come?” Sundarji replied, “Your servant.” Maharaj said, “What are the qualities of a servant?” Sundarji replied, “To do as told.” Maharaj said, “Renounce these clothes and ornaments and become a sadhu.”

Sundarji Suthar became a sadhu! Then Maharaj told him to walk to Kashi. Maharaj asked Muktanand Swami, “I’ve shaved the head of the King’s administrator. I have done a great thing today, haven’t I?”

Muktanand Swami replied, “What is great in that? We only had one devotee in Kutch. We would stay at his house. Now, we have no place!”

“Really?” replied Maharaj, “In that case, call him back.” When he arrived, he bowed at Maharaj’s feet. “Who has come?” inquired Maharaj. “Your servant," replied Sundarji. Maharaj said, "If you are a servant then put on the clothes and jewels you had before." Sundarji Suthar obeyed.

However, he was proud that he had followed Maharaj's commands. So, he told Maharaj, "It is okay that you have done this to me, but do not test others in this way. They will not be able to pass this kind of difficult test."

Later, to dissolve his pride Shriji Maharaj wrote a letter to 18 well-to-do devotees commanding them to become sadhus and immediately go to Bhuj. All 18 left their homes, became sadhus and went to Bhuj. On seeing them Sundarji's pride dissolved.

INSPIRATION: Even if we do something good in Satsang we should not harbour ego for doing it