**1. Which Methodology do you follow in your Test Case?**

**A**. Phase Containment is consolidating QA into each of the stages of SDLC. It brings about Defect Prevention. If QA group performs Design Review, Requirements Review, and Code Review, errors would be not many when a real application is tested. That implies we have averted numerous errors by performing reviews at each phase of SDLC.

**2. What is Black box testing? What are the distinctive discovery Testing Strategies?**

**A**. Black box testing is the product testing technique which is applied to test the product without distinguishing the internal structure of program or code. This testing is typically done to validate the functionality of an application.

The various black box testing systems are:

* Boundary value analysis
* Equivalence Partitioning
* Cause-effect graphing

**3. What are Quality Control and Quality Assurance?**

[Quality Assurance](https://www.testbytes.net/blog/quality-assurance-process-methodology/) includes in process-oriented practices. It guarantees the counteractive action of bugs in the process utilized to make software application. So, the bugs don’t emerge when the software application is being produced.

[Quality Control](https://www.softwaretestinghelp.com/quality-assurance-vs-quality-control/) includes product-oriented practices. It administers the code or program to distinguish the bugs in the software application.

**4. How well do you deal with Ambiguity?**

**A.** Test cases are not generally straightforward, and QA Engineers should follow up on their own judgment. They must feel great with ambiguity.

**5. How well do you Function with others?**

**A**. Several QA teams have colleagues who are from all edges of the world. QA engineers must be content with speaking with individuals at all levels and all backgrounds.

**6. How might we perform Spike testing in JMeter?**

**A**. [JMeter](https://www.testbytes.net/blog/how-to-use-jmeter-for-load-testing/) accompanies a synchronizing timer which can deal with the demands of numerous threads. It can get the expected number of strings and discharge them immediately to cause a spike.

**7. What is the chief contrast among st Reactive and Preventative strategies of Testing?**

**A**. Reactive tests are planned after the product has been delivered and preventative tests are designed ahead.

**8. What are the significant steps to fix issues while Testing?**

**Record** – Log and handle any issues which have happened.

**Report** – Report the issues to high ranked manager.

**Control** – Define the issue management process.

**9. What is the distinction between Test scripts, Test scenarios, and Test cases?**

**A**. **Test Scripts** – They are composed in a programming language and it’s a short program used to test some portion of the functionality of the product framework. Simply speaking a written set of steps that ought to be performed manually.

**Test Scenarios** – [Test scenario](https://www.testbytes.net/blog/test-scenario/) is set up before the genuine testing begins, it incorporates plans for the testing software, environmental condition, number of colleagues, creating test cases, making test plans and every one of the characteristics that are to be tested for the software.

[**Test cases**](https://www.testbytes.net/blog/test-cases-for-mobile-applications/) – It is a record that contains the steps that must be performed, it has been planned before.

**10. What are the classes of Defects?**

**A**. Missing – There was a prerequisite given by the client and it was not done. This is a variance from the, a sign that particular was not performed, or a necessity of the client was not recorded appropriately.

Wrong – The prerequisites have been performed wrongly. This defect is different from the given.

Extra – A prerequisite included in the software that was not given by the end user. This is dependably a variance from the determination yet might be a trait wanted by the user of the software. But it is viewed as a defect due to its variance from the current prerequisite.

**11. List out the Test Deliverables.**

**A**.

* Test Strategy
* Test Plan
* Test Cases
* Test Scripts
* Test Scenarios
* Test Data
* Effort Estimation Report
* Requirement Traceability Matrix (RTM)
* Test Execution Report
* Defect Report/Bug Report
* Test summary report
* Test incident report
* Test closure report
* Test status report
* Weekly status report to Project manager and the client
* Graphs and Metrics
* Release Note
* User guide
* Configuration guide

**12. What is Rapid Application Development?**

**A**. Rapid Application Development (RAD) is formally a parallel development of functions and consequent integration. Functions are produced in parallel as though they were smaller than usual activities, the developments are time-boxed presented and after that congregated to a working prototype.

This can rapidly give the client an image to view and utilize and to give input with respect to the delivery and their prerequisites. Quick change and improvement of the software are probably applying this strategy.

In any case, the software particular should be created for the software sooner or later and the undertaking should be put under more formal controls before going into generation.

**13. What’s the difference between Acceptance testing and System testing?**

**A**. Acceptance testing checks the framework against the Prerequisites. It’s like System testing in which the entire framework is checked yet the vital contrast is the transition in focus.

System testing tests that the system that was indicated has been conveyed. Acceptance testing checks that the system will convey what was demanded. The client ought to dependably do Acceptance testing and not the engineer.

The client recognizes what is needed from the framework to accomplish value in the business and is the main individual equipped to make that determination.

This testing is more about guaranteeing that the product is produced as described by the client. It’sas getting a green light from the client that the product meets requirements and is prepared to be utilized.

**14. Clarify Random/Monkey Testing.**

**A**. A strategy of software testing where the system is checked by randomly ingesting inputs. This test does not follow any pre-defined set of principles. It’s done to test the application behavior.

**15. How to manage the low-frequency issues during the Testing?**

**A**. By low frequency, I trust you mean the issues that can’t be replicated reliably after some time. In case an issue isn’t coming up each time we repeat a similar series of steps, we do some drilling around to check whether we can discover any indication to the appearance of the bug (failure or logs messages) and when nothing else works, we report everything the same. As testers, we can’t leave anything without creating a report.

**16. As a component of which test process do you decide the exit criteria?**

**A**. The exit criteria are resolved on the bases of ‘Test Planning’.

**17. What is Mutation testing?**

**A**. This type of testing is used to detect if a set of the test case or test data is useful by intentionally presenting different code changes (bugs) and retesting with real test cases/data to decide whether the bugs are discovered.

**18. While controlling your venture what things you need to acknowledge?**

**A**. The things that must be taken in contemplations are:

* Is your venture on the plan?
* Are you working towards a similar career objective?
* Have you got enough resources?
* Are you overspending?
* Are there any warning indications of approaching issues?
* Is there any pressure from the administration to finish the undertaking quickly?

**19. What is Unit Testing?**

**A**. It is done to examine whether the individual module or unit of the source code is working legitimately. It is performed by the software developers in developer’s environment. Unit Testing is additionally known as Component Testing or Module Testing.

**20. What is Alpha Testing and Beta Testing?**

**A**. Alpha testing is finished by the in-house testers and engineers who built up the product. In some cases, alpha testing is completed by the customer or outsourcing group with the presence of testers or developers.

Beta testing is performed by a set number of end clients before delivery. Normally, it is done in the customer place.

**21. What is Risk Based Testing?**

**A**. To recognize the functionalities or modules which are in all probability to cause failures and to test them.

**22.What is Application Programming Interface (API)?**

**A**. A formalized assortment of programming calls and routines that can be mentioned by an application program to access to network systems or supporting system.

Also See Manual Testing Interview Questions and Answers by SendTods

**23. What is Agile Testing?**

**A**. Agile testing is a type of software testing practice that takes after the standards of an agile software development. It is an iterative program development procedure where necessities continue changing according to the client needs. Testing is done in parallel with the advancement of an iterative model. Test group gets frequent code changes from the dev group for testing a system.

**24. What are the various Methodologies in Agile Development Model?**

**A**. There are at present seven various agile systems that I know about:

* Scrum
* Extreme Programming (XP)
* Feature-Driven Development
* Lean Software Development
* Agile Unified Process
* Dynamic Systems Development Model (DSDM)
* Crystal

**25. Why we utilize the Decision Tables?**

**A**. The methods of boundary value analysis and equivalence partitioning are regularly connected to conditions or sources of info.

But if diverse combinations of input end in various steps are made, this can be harder to indicate utilizing equivalence partitioning and limit value examination, which tend to be more centered on the UI.

The other two specification-based methods, state transition testing, and decision tables are more centered around business rules or business logic. A decision table is a decent method to manage combinations of things.

Certain times this technique is additionally known as a cause-effect table. The reason is there’s a related logic graphing procedure known ascause-effect graphing which was at times used to help procure the decision table.

**26. What is Load Testing?**

**A**. It is to check that the application/system can deal with the required number of transactions and to confirm the system/application behavior under peak load and normal conditions both.

**27. What are the advantages of Test Reports?**

**A**. The advantages of test reports are:

* Current status of undertaking and nature of software are notified.
* In case required, partner and the client can make a corrective move.
* The last report supports to determine whether the software is ready for launch.

**28. In incremental testing what are Stubs and Drivers?**

The two stubs and drivers are a component of incremental testing. In incremental testing, there are two methodologies to be specifically bottom-up and top-down methods.

Drivers are utilized as a part of bottom-up testing and stub is utilized for the top-down method. With a specific goal to test the principle module, a stub is utilized, which is a dummy program or code.

**29. What is Soak Testing?**

**A**. Running an application at high load for a continued timeframe to distinguish the execution issues is termed [Soak Testing](https://www.testbytes.net/blog/soak-testing/).

**30. What is Performance Testing?**

**A**. This kind of testing decides or approves the speed, versatility, or durability qualities of the application or system under test.

Execution is concerned about accomplishing response times, resource usage levels, and throughput that meet the execution targets for the product or project.

**31. What are the standards of Software Testing?**

**A**.

* Testing indicates the presence of flaws
* Thorough testing is not possible
* Early testing
* Defect clustering
* Testing is setting depending
* Pesticide Paradox
* The absence of error false notion

**32. What is a Bug?**

**A**. In the event that software testers discover any contradiction in the application or system in testing stage then they describe it as a bug.

**33. What is DFD (Data Flow Diagram)?**

**A**. As a flow of data through a data system is graphically depicted then it is called Data Flow Diagram. It is likewise utilized for the representation of data handling.

**34. What is Fuzz testing and when will you use it?**

**A**. Fuzz testing is used to recognize security loopholes and coding mistakes in the product. In this method, random data is added to the application trying to crash the system.

In the event that vulnerability continues, a device called fuzz tester is utilized to decide possible causes. This strategy is more helpful for greater projects yet just recognizes a major mistake.

**36. Describe the PDCA cycle.**

**A**. Software testing plays an essential role in the software development method. In a typical software development, there are four vital steps in PDCA: Plan, Do, Check, Act, cycle. The four stages work as

Plan: Describe the objective and the arrangement for fulfilling that objective.

Do: Perform as per those plan methods which is arranged in the planning stage.

Check: Check to ensure that everything is working as per the plan and gets the normal outcomes.

Act: Act as indicated by that issue.

**37. What ought to be done after a bug is found?**

**A**. Configuration administration covers the procedures used to control, facilitate, and track: code, prerequisites, documentation, issues, change demands, designs, compilers/tools/libraries/patches, alterations made to them, and who rolls out the improvements.

**38. What is SQA testing? Tell us steps of SQA testing.**

**A**. Test Plan has these principal contents:

* Schedule
* Resource requirement
* Due date
* Financial plan
* Feature to be examined with reason
* Feature not to be examined with reason
* Tools to be utilized
* Scope of testing
* Test technique

**39. Tell the contrast between a “defect” and a “failure” in software testing?**

**A**. In basic terms when a defect approaches the end client it is known as a failure while the defect is recognized inside and settled then it is called as a defect.

**40. Q**. **Specify what is the reason behind doing end-to-end testing?**

**A**. End-to-end testing is performed after the functional testing. The reason behind performing end-to-end testing is:

* Testing application in genuine environment situation
* Testing of association between database and application
* To approve the product necessities and integration with outside interfaces

**41. What is Bug Leakage?**

**A**. A bug which is really missed by the testing group while testing and the build was discharged to the Production. In case that now that bug was found by the end user then we refer to it as Bug Leakage.

**42. What is the part of mediator in the review procedure?**

**A**. The mediator or review leader drives the review procedure. The moderator decides, in co-activity with the creator, the sort of review, approach and the creation of the review team.

He or she plays out the entry check and the follow-up on the modified work, keeping in mind the end goal to control the nature of the input and output of the review procedure.

The mediator likewise plans the meeting, scatters reports before the meeting, paces the meeting, mentor’s other colleagues, prompts conceivable discussions and stores the information that is gathered.

**43. When should tester perform the Regression Testing?**

**A**. After the product has changed or when an environment has changed Regression testing ought to be performed.

**44. What is HotFix?**

**A**. A bug which needs to be managed as a high priority bug and fix it instantly.

**45. What is the reason for a test completion criterion?**

**A**. The reason for test completion criterion is to decide when to quit testing.

**46. What are the advantages of Independent Testing?**

**A**. Independent testers are unprejudiced and recognize diverse flaws in the meantime.

**47. Why are dynamic testing and static testing depicted as complementary?**

**A**. Since they share the purpose of recognizing defects yet vary in the kinds of defect they spot.

**48. What is Error seeding?**

**A**. Error seeding is a procedure of intendedly including known errors in software to recognize the rate of error location. It helps during evaluating of the tester aptitudes of discovering bugs and furthermore to know the capacity of the application i.e. how well the application is working when it has errors.

**49. What is equivalence class partition?**

**A**. In equivalence class partition, inputs to the system or product are separated into groups that are required to display like behavior, so they are probably going to be proposed similarly. Thus, choosing one input from each group for designing the test case.

**50. What is positive and negative testing?**

**A**. A positive testing is a point at which you put in a valid info and anticipate that some activity will be finished as per the specification. While a negative test is a point at which you put in an invalid information and gets errors.

**51. Could inspections or reviews be viewed as part of the testing?**

**A**. Yes, indeed, as both assists identify errors and enhance quality.

**52. What amount of testing is sufficient?**

**A**. The appropriate response relies upon the contract, risk for your industry, and special needs.

**53. Explain Exploratory Testing?**

Exploratory testing is based on less planning and more test execution. It refers to performing requirement analysis, test cases designing, test cases execution and understanding the test results simultaneously.

**54. Explain verification in software testing?**

Verification ensures that the product is being built according to the software requirements, and there are no deviations from it. The verification process involves Inspections, Reviews, and Walk-through.

**55. Explain validation in software testing?**

Validation ensures that the software product that is built is as per the client’s requirements or not.

**56. What is the basic difference between verification and validation in software testing?**

Verification detects defects in requirement specifications & validation detects the errors in the implemented Software application.

**57. Explain Static Testing?**

Static Testing is to review the documents to detect the errors in the initial phases of SDLC.

**58. Explain Dynamic Testing?**

Dynamic testing is to run the AUT to validate the output from the expected outcome.

**59. What are the other names for white box testing?**

It is also known as Glass Box, Clear Box, and Structural Testing.

**60. What is white box testing?**

White box testing refers to the testing of the internal code structure.  white-box testing is done at the unit level and is based on the internal perspective of the code and requires programming skills to design test cases. It tests if the code is written properly and inspects every line of the code for being bug-free.

**61. Can you explain about test strategy?**

Test strategy is a document that includes test design and defines how testing is to be carried out.

**62. Name the components of the test strategy document?**

The various components of the test strategy document are

* Test plan id
* Features to be tested
* Test techniques
* Testing tasks
* fail criteria/ Features pass
* Test deliverables
* Test schedule
* Test resources and their Responsibilities
* Testing environment

**63. Explain the Test plan?**

A test plan defines the goals, resources, and procedures for testing a software product. It includes a thorough understanding of the testing workflow.

**64. How will you create a test plan?**

Follow the following steps to write a test plan;

* Analyze the product
* Design the Test Strategy
* Clearly defining the Test Objectives
* Develop the Test Criteria
* Define how the test environment would be
* Schedule and Estimation
* Determine the exact Test Deliverables

**65. Explain the term testbed?**

The testbed is an amalgamation of hardware and software components that are required to conduct software testing.

It consists of Operating system, specific hardware, and software, the product under test, network configuration, other system software, and application software.

**66. Explain the term test environment?**

An environment where testers execute test cases is test environment. Software and hardware environment together form a test environment and serves as a podium to conduct testing.

**67. Explain the term test Data?**

Test data are the inputs given to AUT by the testers to execute test cases and validate the results.

**68. Explain test closure?**

Test closure document is the final document prepared before testers mark an end to testing. It consists of details like several test cases, test cases executed, defects found, defects fixed, defects not fixed, defects rejected, etc.

**69. Explain test coverage?**

It the measurement of quantity of testing done by a set of test cases.

**70. Explain the term integration testing?**

Integration testing is a type of testing where smaller units of code are integrated and tested in a group.

**71. What is System Testing?**

Known as the end to end testing, system testing validates a completely integrated application to if the system is in accordance with the requirements or not.

**72. What is Big Bang testing?**

Big bang testing is based on integrating all the modules together and then testing them all at once.

**73. What are different strategies of integration testing?**

Integration testing can be carried out using two approaches:

* Big Bang Approach
* Incremental Approach
* Top-Down Approach
* Bottom-Up Approach
* Sandwich Approach

**74. What are the disadvantages of Big Bang Integration?**

* The disadvantages of big bang integration are:
* Time-consuming
* Tracing the cause of failures is difficult.
* Enhanced chanced of critical failures.
* In case of a bug, finding the root cause of it is difficult.

**75. What is functional testing?**

Functional testing is a type of black-box testing, which validates that every function of the software as per the requirements.

**76. What are the different types of functional testing?**

Different types of functional testing are:

* Smoke testing
* Sanity testing
* Regression testing
* Usability testing

**77. Tell the steps to be performed for functional testing?**

* Steps to perform functional testing are:
* Identify the required functionalities
* Creation of test data
* Determine the expected output
* Execute the test case
* Compare actual and expected outputs
* Validate the software works as per required functionalities

**78. What is non-functional testing?**

It tests the non-functional requirements of the system like the way it operates.

**79. What are the different types of non-functional testing?**

* Various types of non-functional testing are
* Performance Testing
* Load Testing
* Failover Testing
* Security Testing
* Compatibility Testing
* Usability Testing
* Stress Testing
* Maintainability Testing
* Scalability Testing
* Volume Testing
* Security Testing
* Disaster Recovery Testing
* Compliance Testing
* Portability Testing
* Efficiency Testing
* Reliability Testing
* Baseline Testing
* Endurance Testing
* Documentation Testing
* Recovery Testing
* Internationalization Testing
* Localization Testing

**80. What are the different types of testing?**

* Different types of testing are:
* Functional
* Smoke testing
* Sanity testing
* Regression testing
* Usability testing
* Non – Functional
* Performance Testing
* Load Testing
* Failover Testing
* Security Testing
* Compatibility Testing
* Usability Testing
* Stress Testing
* Maintainability Testing
* Scalability Testing
* Maintenance
* Regression
* Maintenance

**81. Tell the different test levels?**

* The four different test levels are:
* Unit/component/program/module testing
* Integration testing
* System testing
* Acceptance testing

**82. Name the various black box testing techniques?**

* The different black box testing techniques are:
* Equivalence Partitioning
* Boundary value analysis
* Cause-effect graphing

**83. What does a test plan consist of?**

* A test plan consists of
* Test case identifier
* Scope
* Features to be tested
* Features not to be tested
* Test strategy & Test approach
* Test deliverables
* Responsibilities
* Staffing and training
* Risk and Contingencies

**84. What is retesting?**

Retesting is a process of testing the code again to check if the defects detected in the previous round of testing are fixed correctly.

**85. What is Data-Driven Testing?**

Data-Driven Testing is testing the application with multiple test data. The complete testing is data-driven and the main aim of such type of testing is to verify output for multiple test data.

**86. How can you resolve issues during software testing?**

We can resolve issues by:

* Record: the defects are logged and recorded
* Report: The defects are reported to higher management and the development team
* Control: issue management process is defined.

**87. What is a test scenario?**

The test scenario is any functionality of the code that is to be tested.

**88. Define test cases?**

A test case describes the execution settings, testing method, input data, expected results to attain a testing objective.

**89. Define test scripts?**

Test scripts are small programs that are run manually to test a specific functionality of the code.

**90. Define Latent defect?**

Latent defects are the defects that have been present in the code for a long time but have never been detected before in the previous releases.

**91. Name two parameters that define the quality of testing?**

You can judge the quality of testing by:

* Defect reject ratio
* Defect leakage ratio

**92. What are the test deliverables?**

Test deliverables are the artifacts that are delivered to the stakeholders after every round of testing.

* Test strategy
* Test plan
* Test scenario
* Test cases
* Test data
* RTM
* Test metrics
* Test incident report
* Test status report
* Test summary report
* Release note
* Test closure report

**93. What is mutation testing?**

It is a technique to identify if test data/test case is useful by knowingly introducing a bug in the code and retesting the bug induced code with the original test data, a test case to check if it detects the bug.

**94. What things will you ruminate before choosing automation tools for the AUT?**

* Technical Feasibility
* Complexity level
* Application stability
* Test data
* Application size
* Re-usability of automated scripts
* Execution across environment

**95. How to perform Risk Analysis?**

For the risk analysis following steps need to be implemented

* Find risk score
* Create a risk profile
* Change the properties of the risk
* Install test risk resources
* Create a risk database

**96. Can you tell the debugging categories?**

The debugging categories are:

* Brute force debugging
* Backtracking
* Cause elimination
* Program Slicing
* Fault tree analysis

**97. What information should a test plan include?**

* Test Strategy
* Test Objective
* Exit/Suspension Criteria
* Resource Planning
* Test Deliverables

**98. State few risks that can result in project failure?**

* Insufficient resource for the project
* Improper testing environment
* Budget constraints
* Strict timelines

**99. How to do project estimation?**

* Break the project into small tasks
* Assign them to different team members
* Estimate efforts for each task
* Confirm the approximation

**100. What are the commonly used testing types?**

* Unit Testing
* API Testing
* Integration Testing
* System Testing
* Install/Uninstall Testing
* Agile Testing

**101. What are the things to consider while monitoring your project?**

* Schedule of your project is going fine?
* Budget is under the limit?
* Are resources enough?
* Are there any warning signs reflecting any future problem
* Are there any requirement changes?

**102. Common mistakes that lead to issues?**

* Improper resource allocation
* Poor Scheduling
* Underestimating
* Ignoring the small problems
* Not following the process

**103. What does a test report consist of?**

* Project Information
* Test Objective
* Test Summary
* Defect

**104. What are the benefits of the test report?**

* Gives information on the on-going status of the project
* Tells about the quality of product
* Stakeholder can take corrective measures whenever required
* The final report tells if the product is ready for the release.

**105. What are the best practices of software quality assurance?**

* Continuous Improvement
* Documentation
* Tool Usage
* Metrics
* Responsibility by team members
* Experienced SQA auditors

**106. What steps you should follow after you detect any defect?**

* Recreate the defect
* Attach the screenshot
* Log the defect

**107. When to Prepare Requirement Traceability Matrix?**

RTM is prepared before test case designing.

**108. Who prepares the test plan?**

Test lead or the test manager prepares the test plan.

**109. Give a few examples of the test environment?**

Some of the examples of the test environment:

* Application Type: Web Application
* OS: Windows
* Web Server: IIS
* Web Page Design: Dot Net
* Client-Side Validation: JavaScript
* Server Side Scripting: ASP Dot Net
* Database: MS SQL Server
* Browser: IE/FireFox/Chrome

**110. As a tester what would you do, if you find any defect?**

I would

* Reconstruct the bugs/defect
* Take and affix a screenshot
* Record the bugs/defect

**111. What is Fuzz Testing?**

Fuzz testing is carried out by feeding random inputs to an application under test, to validate if the AUT returns exceptions like failing built-in code assertions, crashes,  or potential memory leaks.

**112. Name different test coverage techniques?**

* Various test coverage techniques are:
* Statement coverage
* Decision coverage
* Path coverage

**113. Name the important modules of defect report format?**

The various components are

* Module Name
* Project Name
* The date defect was detected on
* Who detected the defect
* Defect ID
* Defect Name
* Screenshot of the defect
* Priority status
* Severity status
* Who resolved the defect
* When was defect resolved

**114. What does COTS stand for?**

COTS stands for Commercial Off The Shelf.

**115. Why do we use automation testing?**

* Eases the execution of test cases that re repeated
* Helps in the testing of the large test matrix
* Supports parallel execution of test cases
* Is beneficial for long run test cases as they can be left unattended without requiring any human intervention
* Enhances the accuracy of test cases
* Saves time and money.

**116. Can you tell us the advantages of selenium over other testing tools?**

* Supports various languages including Java, Python, C#, PHP, Ruby, Perl &.Net
* Supports different OS including Windows, Mac or Linux
* Supports multiple browsers including Mozilla Firefox, Internet Explorer, Google Chrome, Safari or Opera
* Easy integration with TestNG & JUnit, Maven, Jenkins & Docker
* It is an open-source

**117. Name the components of Selenium?**

* Selenium Integrated Development Environment (IDE)
* Selenium Remote Control (RC)
* Selenium WebDriver
* Selenium Grid

**118. Name the various locators in Selenium?**

* ID
* ClassName
* Name
* TagName
* link text
* PartialLinkText
* Xpath
* CSS Selector
* DOM

**119. Name various exceptions in Selenium web driver?**

* TimeoutException
* NoSuchElementException
* ElementNotVisibleException
* StaleElementException

**120. How will you launch the browser using WebDriver?**

We can use the following syntax:

**WebDriver driver = new FirefoxDriver();**

**WebDriver driver = new ChromeDriver();**

**WebDriver driver = new InternetExplorerDriver();**

**121. What is Regression Testing?**

Regression testing is done to validate that the code changes have not introduced any bugs in the existing code.

**122. How does automation testing help in agile methodology?**

It helps in maximum test coverage in minimal time.

**123. Which all test cases can be automated?**

* Smoke test cases
* Regression test cases
* Complex calculation test cases
* Data-driven test cases
* Non-functional test cases

**124. What is a workbench concept?**

It is a way to document how a specific task has to be performed.

**125. What are the tasks in workbench?**

* Input
* Execute
* Check
* Production output
* Rework

**126. What strategies you can follow to roll out the product to end-users?**

The strategies we can follow are:

* Pilot
* Gradual Implementation
* Phased Implementation
* Parallel Implementation

**127. What does PDCA in software testing stand for?**

PDCA in software testing stands for Plan, Do, Check, Act

**128. Which all types of testing teams can conduct software testing?**

* Isolated test team
* Outsource – we can hire external testing resources and do testing for our project.
* Inside test team
* Developers as testers
* QA/QC team

**129. What are the shortcomings of monkey testing?**

* They are not realistic
* Many of the tests are redundant and unrealistic
* You will spend more time analysing results
* You cannot recreate the test if you do not record what data was used for testing

**130. What does DFD stand for?**

DFD stands for Data Flow Diagram. IN DFD the flow of data is represented graphically.

**131. What does LCSAJ stand for?**

LCSAJ stands for ‘linear code sequence and jump.’

**132. Why is test independence done?**

Test independence eludes author preference in describing operative tests.

**133. When do you define the exit criteria?**

Exit criteria are described based on “Test Planning”.

**134. What is alpha testing?**

Alpha testing is pre-release testing by the end-users at the developer’s end.

**135. What is beta testing?**

Beta testing is testing done by a selected group of end-users at their location. It is done before the software is out in the market.

**136. What are the responsibilities of test management?**

The responsibilities of Test management are:

* To enhance software quality
* Creating and up keeping the product metrics
* Ensures that there is less design or coding faults

**137. The output of the requirement analysis used as an input for which phase?**

User Acceptance Test Cases

**138. What are the benefits of Independent testing?**

Independent testers are neutral and detect dissimilar bugs at the same time.

**139. In reactive testing approach, when does test design work begin?**

The test design work in reactive testing approach begins after the software is produced.

**140. Name different Agile testing methodologies?**

The different Agile methodologies are:

* Extreme Programming (XP)
* Lean Software Development
* Unified Process
* Scrum
* Feature-Driven Development
* AgileCrystal
* Dynamic Systems Development Model (DSDM)

**141. Which process involves the estimation of the testability of the requirements and system?**

‘Test Analysis’ and ‘Design’

**142. Name a few dynamic testing techniques?**

Equivalence Partitioning, Use Case Testing, Exploratory Testing, and Decision Testing are dynamic testing techniques

**143. Name a few static testing techniques?**

Data Flow Analysis and Inspections are static testing techniques.

**144. Which testing technique detects threats such as virus from malicious outsiders?**

Security Testing detects threats such as virus from malicious outsiders

**145. When to implement configuration management procedures?**

During test planning.

**146. What is an equivalence partition?**

Equivalence partition is dividing input and output values in a range such that only one value becomes a test case.

**147. What is load testing?**

Load testing tests the system performance under real-life load conditions.

**148. What is stress testing?**

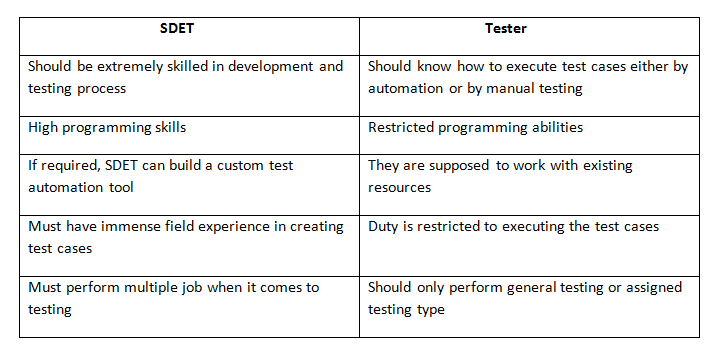
Stress testing tests system performance under unfavourable conditions/loads.

**149. When can you take a conclusion to end software testing?**

Some of the basic criteria to stop testing are:

* Deadlines are finished
* Test budget is over
* Bug rate is less than a certain level
* Passed Test cases percentage is above a pre-decided level
* Testing period for Alpha or beta testing is over
* Code Coverage, requirements are met to a required level.

**150. What’s the difference between a SDET and Tester?**



**151. What’s interrupt testing?**

Interruptions happen on our phone frequently. These sudden interruptions can have adverse effect on regular functioning of any app that’s being used.

Interruption testing will reveal how the app behaves to interruption and will it get back to the stopping point without any error?