Manual Testing Interview Questions and Answers

I) Questions on SDLC Models:

Q) What is SDLC?
SDLC (Software Development Life Cycle) is a process or methodology used to develop software in a systematic approach.

Q) Explain Phases of SDLC?
SDLC has 6 phases.
a. Requirement gathering
b. Analysis and planning
c. Software Design
d. Coding
e. Testing
f. Release and Maintenance

a. Requirement gathering:
During this phase, Business requirements are gathered from customer and documented. The output of this phase is Business requirements specification (BRS) document.
b. Analysis and Planning:
During this phase, Business requirements are analyzed and System Requirements are documented. At the same time technology required to develop the system is also identified. Project plan is created to come up with project milestones and release date. The output of this phase is System Requirements Specification (SRS) document and Project Plan.
c. Software Design:
During this phase, based on SRS document, High Level Design (HLD) and Low Level Design (LLD) is done for the entire system and for individual components in the system. The output of this phase is Design documents.
d. Coding:
During this phase, based on the Design documents, programs are written in the selected programming language. The final output of this phase is code documents and executable software.
e. Testing:
During this phase, Test cases are created and executed by the testers to ensure that the system is developed as per the System requirements specifications(SRS). If there are any deviations, defects are reported to developers which are again fixed and given to testers. Testers perform Re and Regression testing and close the defects. The final output of this phase is Test Documents and Tested Software.

f. Release and Maintenance:
During this phase, tested software is released to production and the software is deployed in customer's environment by release team. Any defects/improvements identified thereafter by customer are addressed by maintenance team.

Q) What are different types of SDLC Models?
There are different types of SDLC Models.
a. Sequential Models
b. Iterative/Incremental Models
i. Prototyping model
ii. Spiral Model
iii. Agile Model

Q) What is Sequential SDLC Model?
SDLC Model in which the development activities are executed in a sequential manner. One phase starts only after the completion of its previous phase.

Q) What are different Sequential Models?
Below are different Sequential Models
i. Waterfall model
ii. V-Model

Q) What are the different phases in Waterfall Model?
Waterfall model has 6 phases similar to general SDLC.
a. Requirement gathering
b. Analysis and planning
c. Software Design
d. Coding
e. Testing
f. Release and Maintenance

Q) Who are the people (Roles) involved in different phases of Waterfall Model?
Roles involved in different phases depend on the organizational structure. In general, below are the people who get involved in different phases of SDLC.

i. Business analyst or Business Team in Requirements Gathering phase.
ii. Technical Manager or Senior Developers in Analysis phase, Project Manager in Planning phase.
iii. System Architect or Technical Lead in Design Phase.
iv. Development Leads and Developers in Coding phase.
v. Test Leads, Testers in Testing Phase.
vi. Project Manager, release team and Maintenance Team in Release and Maintenance phase.

In addition to above, Development Managers and QA Managers are also involved to manage the respective teams.

Q) What are the advantages and disadvantages of Waterfall model?
Advantages of Waterfall model:
————————————
1. The main strength of Waterfall model is its simplicity and it is very easy to use.
2. Output generated at the end of each phase is reviewed and approved and hence it has high visibility.
3. Minimal resources are required.
4. It is less expensive as minimal resources are required.
5. This model is preferred for smaller projects where requirements are well understood.
Disadvantages of Waterfall model:
————————————
1. It is difficult to implement any changes in the requirements.
2. No working software is produced until the last stage of the life cycle.
3. It is expensive to incorporate any changes once the project is in to testing phase.
4. Not suitable for complex projects, long term projects.
5. Not suitable when requirements are not clear.

Q) What is V-Model?
V-Model is also known as Verification and Validation Model. In V-Model, testing is done in parallel with corresponding development activity.

Q) What are the different levels of testing in V-Model?
i. Unit / Component Testing
ii. Integration Testing
iii. System Testing
iv. Acceptance Testing

Q) Who perform integration testing in V-Model?
In general, developers perform integration testing in V-model.

Q) What are the advantages and disadvantages of V-Model?
Advantages of V-model:
————————-
1. Quality of the software gets improved as the model as testing activities for the respective development activity.
2. Avoids the Defect multiplication.
3. Saves more time as testing activities happen well before coding.
4. Defects can be identified in the early stages of the life cycle.

Disadvantages of V-model:
————————-
1. Expensive as it involves more number of resources
2. Adopting any changes in requirements is not possible.

Q) What are the advantages of V-model over Waterfall Model?
Software quality gets improved in V-Model as it has testing activities in parallel for the respective development activity. Defects can be identified in early stages of life cycle and defects multiplication can be avoided.

Q) What is the importance of SDLC in Software Development?
Quality software can be delivered when SDLC is followed.

Q) When does Testing activity starts in Waterfall model?
Testing activity starts only after development phase is completed i.e, Testing is post development activity in Waterfall model.

Q) When does Testing activity starts in V-Model?
Testing activity starts during Requirements stage itself in V-model.

Q) What is Iterative/Incremental SDLC Model?
Process of developing software through repeated cycles (Iterative) and in smaller portions at a time (incremental) is called Iterative/Incremental SDLC.

Q) What are different Iterative SDLC Models?
i. Prototype Model
ii. Spiral model
iii. Agile Model

Q) What is Prototype Model?
In this model, based on the requirements, prototype (Non-functional system) is developed and delivered to customer for his evaluation. Based on customer feedback, prototype is refined and again given for customer evaluation. The process repeats until developers get clarification on the requirements. Once the prototype is finalized, the requirements are also frozen.

Q) What is Prototype?
Prototype is the sample implementation of the actual system. It can be also called as Non-functional system.

Q) What are the advantages and disadvantages of Prototype Model?
Advantages of Prototype model:
——————————
1. Clarity on requirements from the customer, success rate of the project is higher.
2. Customer can see the system and he provide the feedback immediately.
3. Errors can detected much earlier as customer is getting involved.

Disadvantages of Prototype model:
——————————
1. Customer may think that prototype is the real system.
2. Time taking model as the actual development does not start until the prototype is finalized.
3. Adopting changes or adding new requirements is difficult.
4. Prototype is of no use once the development of actual system is started.

Q) What is Agile Model?
Meaning of Agile is “Moving Quickly”. When applied to Software development, it means that delivering the software that meets customer requirements in shortest possible time.

Agile software life cycle is an iterative process where software is ready at each iteration but can always be improved in next iteration.

Q) What are the advantages and disadvantages of Agile Model?
Advantages of Agile Model:
——————————
1. Success rate of the project very high compared to any other models.
2. Can adopt changes in requirements at any point of time.
3. Working software is delivered frequently.
4. It emphasizes on responding to change rather than extensive planning and documentation.
5. It is recommended for Product Development.

Disadvantages of Agile Model:

——————————

1. Expensive Model as more number of resources are required.
2. Complex in Managing.
3. There is lack of emphasis on necessary designing and documentation.
4. The project can easily get taken off track if there is any communication gap.

Q) What are different Types of Agile Models?

i. Extreme Programming
ii. Scrum Model

Q) What is Extreme Programming Model?

Extreme Programming (XP) is a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. As a type of agile software development, it advocates frequent releases in short development cycles, which is intended to improve productivity and introduce checkpoints where new customer requirements can be adopted.

Q) What is Scrum Model?

Scrum is an iterative and incremental agile software development methodology for managing software development.
In this model, System is divided into small parts known as Sprints. The duration of each sprint can range from one week to three weeks. At the end of the sprint, team members and stake holders meet to assess the progress of the project and identify further plan of action. This assessment helps in taking stalk of the current state and rework the line of work and complete the project on time and not just speculate or predict the further outcome.

Q) What is Spiral Model?

The spiral model is a incremental model with more emphasis placed on risk analysis. This model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations called Spirals in this model.

Q) What are the advantages and disadvantages of Spiral Model?

Advantages of Spiral model:

——————————

1. High amount of risk analysis and hence avoidance of Risk is enhanced.
2. Good for large and mission-critical projects.
3. Strong approval and documentation control.
4. Additional Functionality can be added at a later date.
5. Software is produced early in the software life cycle.

Disadvantages of Spiral model:

——————————

1. Can be a costly model to use.
2. Risk analysis requires highly specific expertise.
3. Project's success is highly dependent on the risk analysis phase.
4. Doesn't work well for smaller projects.

Q) Why is Agile Model so popular than other SDLC models?

Agile Model is popular because of its flexibility in adopting changes in requirements and at the same time delivering software in shortest possible time.

Q) What are the advantages of Iterative Models over Sequential Models?

1. In iterative model we can only create a high-level design of the application before we actually begin to build the product and define the design solution for the entire product. Later on we can design and built a skeleton version of that, and then evolved the design based on what had been built.
2. In iterative model we are building and improving the product step by step. Hence we can track the defects at early stages and avoid the downward flow of the defects.
3. In iterative model we can get the reliable user feedback. When presenting sketches and blueprints of the product to users for their feedback, we are effectively asking them to imagine how the product will work.
4. In iterative model less time is spent on documenting and more time is given for designing.

Q) What is the difference between Sequential and Iterative Models?

1. In Sequential models, Software is developed at once and delivered where as in Iterative models, software is divided into increments and developed.
2. In Sequential models, working software is not produced until late in the life cycle, where as in Iterative models, software is developed and delivered early in increments.

II) Software Test Levels:

Q) What are different Test Levels?

There are 4 test levels
a. Unit/Component/Program/Module Testing
b. Integration testing
c. System Testing
d. Acceptance Testing

Q) Explain Unit Testing?

Unit testing is to verify the correctness of an individual component/module of the system/software. This type of testing is done during the development of a software.

Q) What are different names for Unit Testing?

Unit testing is also called Component Testing, Program Testing or Module Testing.

Q) What is integration Testing?

Individual modules/components of the software are integrated and tested as a group to ensure whether they are functioning as expected with their counterpart.

Q) What are different types of Integration Testing?

i. Big Bang approach
ii. Top-down approach
iii. Bottom-up approach
iv. Hybrid approach

Q) What is Big-Bang Integration Testing?

In Big Bang approach, all the developed modules are coupled together to form a complete system and it is tested as a whole. There is no need to simulate the components as everything is complete before integration testing starts.

Q) What is Top-down Integration Testing?

In Top-down Integration testing, highest level modules are integrated first and testing takes place from top to down. Any unfinished modules are substitute by stubs.

Q) What are the advantages and disadvantages of Top-Down Integration Testing?

Advantages of Top-down Integration testing:
————————————————-
1. It is easy to locate errors.
2. Possibility to obtain an early prototype.
3. It is easy to maintain the code.

Disadvantages of Top-down Integration testing:
————————————————
1. Stubs should be created for unfinished modules and might be more complicated than they first appear to be.
2. Lower level modules may not be tested as much as the upper level modules.

Q) What is Bottom-Up Integration Testing?

In Bottom-up approach, testing of low level components is done first and move towards the high level components. any unfinished components are substituted by Drivers.

Q) What are the advantages and disadvantages of Bottom-Up Integration Testing?

Advantages of Bottom-up Integration testing:

————————————————-

1. Bugs can be found easily.
2. It is easy to maintain the code there is a more clear structure of how to do things.

Disadvantages of Bottom-up Integration testing:

————————————————–

1. When releasing a prototype you can not see a working prototype until nearly all the program has been completed so that may take a long time .
2. Need to create driver modules.

Q) What is Hybrid approach of Integration Testing?

Hybrid approach is combination of Top-down and Bottom-up Integration approaches. Testing can start as and when modules become available.

Q) What is Driver in Integration Testing?

A driver is a software component that replaces a component that takes care of the control and/or the calling of a component or system.

Q) What is Stub in Integration Testing?

A Stub is a special-purpose implementation of a software development, used to develop or test a component that calls or is otherwise dependent on it.

Q) What type of integration testing is preferred?

That depends Project as well as Company.

Q) What is System Testing?

System testing is testing the system as a whole after integrating all the components. It is performed on the entire system to ensure that it is working as end user expected.

Q) What is Acceptance Testing?

Acceptance testing is the testing done in the real time environment by the intended audience or business representatives.

Q) What are different types of Acceptance Testing?

i. User Acceptance Testing
ii. Operational Maintenance testing
iii. Certified testing
iv. Contract acceptance testing
v. Alpha testing
vi. Beta testing

Q) What is User Acceptance Testing?

User acceptance testing focuses mainly on the functionality thereby validating the functionality of the system based on the business requirement specifications.

Q) What is Operational Acceptance Testing?

Operational acceptance testing validates whether the system meets the requirements for operation. System administrators will perform the operational acceptance test shortly before the system is released. The operational acceptance test may include testing of backup/restore, disaster recovery, maintenance tasks and periodic check of security vulnerabilities.

Q) What is Certification Testing?

Certified testing also known as Compliance acceptance testing is performed against the regulations which must be adhered to, such as governmental, legal or safety regulations.

Q) What is Alpha Testing?

A cross-section of potential users and members are invited to use the system. Developers observe the users and note problems

Q) What is Beta Testing?
Beta testing or field testing sends the system to a cross-section of users who install it and use it under real-world working conditions. The users send records of incidents with the system to the development organizations where the defects are fixed.

Q) Who does User Acceptance Testing?
User Acceptance Testing is performed by the users/customers, business representatives and application managers

Q) Who does Operational Acceptance Testing?
System administrators will perform the operational acceptance testing.

Q) Who does Certification Testing?
Certified Testers who are aware of the governmental, legal and safety regulations will perform Certified testing.

III) Software Test Types:

Q) What is verification?
Verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. It takes care of set of activities to address the question "Are we building the product right?"

Q) What is Validation?
Validation is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. It takes care of set of activities to address the question "Are we building the right product?"

Q) What is Quality Assurance?
Quaity Assurance aims to prevent defects with a focus on the process of producing the software. It is defect Prevention oriented.

Q) What is Quality Control?
Quality control aims to identify defects in the developed software. It is defect-detection and correction oriented.

Q) What is Testing?
It is a process of executing a program or application with the intent of finding defects and it helps to measure quality of software.

Q) What is Static Testing?
Static Testing is done during verification process. This testing includes reviewing of the documents (including source code) and static analysis. This is useful and cost effective way of testing. For example: reviewing, walk through, inspection, etc.

Q) What is Dynamic Testing?
In dynamic testing the software code is executed to demonstrate the result of running tests. It's done during validation process. For example: unit testing, integration testing, system testing, etc.

Q) What are different types of Testing?
i. Functional testing
ii. Non-functional Testing
iii. Structural testing
iv. Experience based testing

Q) What is Functional Testing?
Functional testing involves testing an applications's functionality and features based on analysis of the requirements specifications. It helps to verify what the system is supposed to do.

Q) What is Non-Functional Testing?
Non-functional testing involves testing the application's non-functional characteristics such as reliability, scalability etc. It helps to determine how well the system works.

Q) What is Sanity Testing?
Sanity testing is done to ensure that basic functionality is working fine and whether it is reasonable to proceed with further testing or not. It can also called as "Build Verification test"

Q) What is the difference between Sanity Testing and Smoke Testing?
According to the Industry standards, there is no difference between Smoke testing and Sanity testing.

Q) What is Re-Testing?
Re-Testing or Confirmation testing is the process of executing test cases that failed the last time when they were run, in order to verify the correctiveness of the fixes provided.

Q) What is Regression Testing?
Regression testing is done to ensure that the changes made in the software for fixing defects or to enhance the functionalities does not affect the exiting functionality.

Q) What is the difference between Re-testing and Regression Testing?
Re-Testing is done to verify defect fixes and Regression testing is done to check if the defect fixes have disturbed existing functionality that was working fine before making the changes.

Q) What is Performance Testing?
Testing performed on software in order to determine how a system performs in terms of responsiveness and stability under a particular workload.

Q) What is Load Testing?
Load testing is performed to understand the behavior of the system under a specific expected load. Objective is to identify the performance bottlenecks in the system.

Q) What is Stress Testing?
Stress testing is performed to evaluate a system beyond the limits of specified requirements or resources, to ensure that system does not break. Objective is to identify the breakpoint of the system.

Q) What is the difference between Performance Testing and Load Testing?
Load testing is sub-type of performance testing.

Q) What is Adhoc Testing?
Testing done without using any formal testing technique is called Ad-hoc testing.

Q) What is Exploratory Testing?
It is a formal testing process where there are no test cases are available to test the application. Functionality of the application is understood by exploring the application and understand the functionalities.

Q) What is Internationalization Testing?
Testing the application to know how it displays the messages or various languages and how it handles the inputs, strings and sorting items.

Q) What is Localization Testing?
Testing whether the software functions as expected in localized environment after translating the messages, documents and customizing the resources .

Q) What is Exhaustive Testing?
Testing which covers all combination of input values and preconditions for an element of the software under test is called Exhaustive testing.

Q) What is Usability Testing?
The testing that validates the ease of use, speed, and aesthetics of the software from user's point of view is called Usability testing.

Q) What is Compatibility Testing?
Testing whether software is compatible with other elements of a system with which it should operate, e.g browsers, operating systems etc.

Q) What is Reliability Testing?
Reliability testing is to check the ability of software to function under given environmental conditions for a particular amount of time.

Q) What is Volume Testing?
Volume testing refers to testing a software application with a certain amount of data. This amount can, in generic terms, be the database size or it could also be the size of an interface file that is the subject of volume testing.

Q) What is Spike Testing?
Spike testing is done by suddenly increasing the number of or load generated by, users by a very large amount and observing the behaviour of the system. The goal is to determine whether performance will suffer, the system will fail, or it will be able to handle dramatic changes in load.

Q) What is Endurance Testing?
Endurance testing is usually done to determine if the system can sustain the continuous expected load. That is, to ensure that the throughput and/or response times after some long period of sustained activity are as good or better than at the beginning of the test.

Q) What is Configuration Testing?
Tests are created to determine the effects of configuration changes to the system's components on the system's performance and behaviour. A common example would be experimenting with different methods of load-balancing.

Q) What is Security Testing?
Security testing is a process to determine that an information system protects data and maintains functionality as intended.

Q) What is Installation Testing?
Installation testing focuses on what users will need to do to install and set up the new software successfully. The testing process may involve full, partial or upgrades install/uninstall processes

Q) What is Risk based testing?
Risk based testing is prioritizing and executing the tests of features and functions based on the risk of their failure.

Q) What is Recovery Testing?
Recovery testing is the activity of testing how well an application is able to recover from crashes, hardware failures and other similar problems.

Q) What is Database Testing?
Database testing mainly deals with finding errors in the databases so as to eliminate them. This will improve the quality of the database.

Q) What is GUI Testing?
Graphical user interface testing is the process of testing a product's graphical user interface to ensure it meets its written specifications.

Q) What is Positive Testing?
Testing aimed at showing software works. Also known as "Test to pass".

Q) What is Negative Testing?
Testing aimed at showing software does not work. Also known as "Test to fail".

Q) What is Soak Testing?
Soak testing involves testing a system with a significant load extended over a significant period of time, to discover how the system behaves under sustained use.

IV) Software Test Design Techniques:

Q) What is Test Design technique?
Test design is a procedure to select few test cases out of many with the likelihood of identifying defects

Q) What are different types of Test Design Techniques?
Specification-based or Black-box Testing
Structure-based or White-box Testing
Experience based Testing

Q) What is Black Box Testing?
Black-box test design technique is a procedure to derive or select test cases based on an analysis of the specification, either functional or non-functional, of a component without reference to its internal structure.

Q) What are different types of Black box Testing Techniques?
i. Equivalent partitioning or Equivalent class
ii. Boundary Value analysis
iii. Decision Tables
iv. State Transition testing
v. Use cases testing

Q) What is White Box Testing?
White-box test design technique is a procedure to derive or select test cases based on an analysis of the internal structure of a component or system.

Q) What are different types of White box testing Techniques?
i. Statement Coverage
ii. Decision Coverage

Q) What is Equivalence Partitioning or Equivalent Class?
Equivalence partitioning involves identifying a small set of input values that produce many different output conditions as possible. This reduces number of permutations and combinations of input, output values used for testing thereby increasing the coverage and reducing the effort involved in testing.

Q) What is Boundary Value Analysis?
Boundary value analysis is based on testing the boundaries between partitions. Test Cases are designed based on boundary values.

Q) What is Decision Table testing?
Decision table testing is a test design technique in which test cases are designed to execute the combinations of inputs shown in a decision table.

Q) What is State Transition Testing?
Sate Transition Testing is a test design technique in which test cases are designed to execute valid and invalid state transitions.

Q) What is Use Case Testing?
Use case testing is a technique that helps us identify test cases that exercise the whole system on a transaction by transaction basis from start to finish.

Q) What is Statement coverage?
Statement coverage is the percentage of executable statements that have been exercised by a test suite.

Q) What is Decision Coverage?
Decision coverage is the percentage of decision outcomes that have been exercised by a test suite. 100% decision coverage implies 100% statement coverage.

Q) What is condition Coverage?
The percentage of condition outcomes that have been exercised by test suite. 100% code coverage requires each single condition in every decision statement to be tested as True or False.

Q) What is Multi Condition Coverage?
Design test cases for each combination of conditions is called multi condition testing.

Q) What is Experience-based testing or Informal Testing?
Experience-based testing is a procedure to select test cases based on the tester's experience, knowledge and intuition.

Q) What are different types of Informal Testing Techniques?
i. Error guessing
ii. Exploratory testing

Q) What is Error guessing?
Error guessing is a technique where the experience of the tester is used to anticipate what defects might be present in the component or system under test as a result of errors made, and to design tests specifically to expose them.

Q) What is Exploratory Testing?
Exploratory Testing is a test design technique where the tester actively controls the design of the tests as those tests are performed and uses information gained while testing to design new and better tests.

Q) When to use Decision table technique?
When different combinations of inputs result in different actions being taken, it would be more difficult to show using equivalence partitioning and boundary value analysis, which tend to be more focused on the user interface.

Q) What is Decision Table?
A table showing combinations of inputs and stimuli(causes) with their associated outputs and/or actions (effects), which can be used to design Test cases.

Q) What is State diagram?
A diagram that depicts the states that a component or system can assume and shows the events or circumstances that cause or result from a change from one state to another.
Software Test Process:

V) Software Test Process or STLC

Q) What is Software Test Process or STLC?
It is a systematic approach to test the Software. Main objective is to monitor and control testing activities.

Q) What are different phases of STLC?
There are four different phases of STLC.
i. Test Planning
ii. Test Design
iii. Test Execution
iv. Test Closure

Q) What is the difference between SDLC and STLC?
STLC is part of SDLC. SDLC involves the complete verification and validation of a software whereas STLC involves only Validation of the system.

Q) What is Test Planning?
The activity of establishing or updating a Test Plan.

Q) What is Test Plan?
A document describing the detailed approach to test the software and what the eventual work flow will be. It consists of features to be tested, features not to be tested, approach, entry criteria, exit criteria, suspension criteria, resume criteria, test environment, training needs, resources, roles and responsibilities, risks and contingency plan.

Q) What are the tasks involved in Test Planning?
1. Understand and analyze the requirements
2. Risk analysis
3. Test Strategy Implementation
4. Test Estimations
5. Team formation
6. Test Plan documentation
7. Configuration Management planning
8. Traceability Matrix
9. Define Test Environment set up

Q) Who prepares Test Plan?
Test Lead prepares Test Plan.

Q) What is the Input for creating Test Plan?
Mandatory inputs are Requirements documents, Project plan, Test Strategy. Optional inputs are Design documents, Process guideline documents, Corporate standard documents.

Q) What is Test Strategy?
Test Strategy is a organization level document that describes the testing approach during SDLC.

Q) What is Entry Criteria?
Criteria that describes when to Start testing.

Q) What is Exit Criteria?
Criteria that describes when to Stop testing.

Q) What is Suspension Criteria?
Criteria that describes when to stop testing temporarily.

Q) What is Resume Criteria?
Criteria that describes when to Resume or Re-start testing.

Q) What is the Output of Test Plan?
Test Plan document

Q) Who approves Test Plan?
Project Manager

Q) What are the tasks involved in Test Design phase?
i. Creating Test scenarios
ii. Test case documentation
iii. Test data collection

Q) What are the inputs to Test Design Phase?
Requirements Specification, Test Plan, Test Scenario Template, Test Case Template, Test Data Template

Q) What is the output of Test Design phase?
Test Cases documents, Test Data

Q) What is Traceability Matrix?
Document showing the relationship between Requirements and Test Cases.

Q) Who prepares and updates Traceability Matrix?
Test Lead or Team Lead creates Traceability Matrix and Testers update Traceability Matrix throughout the STLC.

Q) What are the tasks involved in Test Execution Phase?
1. Creating Test batches
2. Verifying Test Environment set up
3. Test Execution
4. Test reporting
5. Defect Reporting
6. Re & Regression testing

Q) What are the inputs for Test Execution Phase?
Requirements Specification, Test Plan, Test Case documents, Test Data, Defect Report, Test Report

Q) What is the output of Test Execution Phase?
Test Reports, Opened/Closed Defect Reports

Q) What are the tasks involved in Test Closure phase?
1. Evaluating Exit Criteria
2. Collecting all from Test activities
3. Sending Test Deliverables to the customer
4. Improvement suggestions for future projects

Q) What are the inputs for Test Closure phase?
Requirements, Test Plan, Test Reports, Open/Closed Defect Reports, Test Summary report Template

Q) What is the output for Test Closure phase?
Test Summary report, Test Deliverables (Test Plan, Test Scenarios, Test Cases, Test Data, Test Reports, Opened/Closed Defect Reports)

Q) What are Test Metrics?
Test Metrics is used in decision making and

Q) What are Test Deliverables?
Any test work product that must be delivered to someone other than the test work product's author.

Q) What are base lined documents?
Base lined documents are the documents that are approved and finalized documents.

Q) What is Defect report?
A document reporting any flaw in a component or system that can cause the system to fail to perform its required function.

Q) What is Test Summary report?
A document summarizing testing activities and results. It also contains an evaluation of the corresponding test items against exit criteria.