**Q: What is exploratory testing?**

A: It’s a type of software testing where the test cases are not scripted earlier but it’s an on the go process where tester thinks of something to test and notes it and then tests it simultaneously. It helps in discovering new problems and solve it.

**Q: What is traceability matrix?**

A: A traceability matrix is a worksheet that contains the requirements with its all possible test scenarios and cases showing whether it passed or failed. As it covers all possible cases, testing can be done thoroughly.

**Q: What is boundary value testing?**

A: It is a methodology of software testing which emphasis on creating test cases for values near valid range i.e. boundary of the range. It contains a value of boundary, a next value in valid boundary range and a value before boundary range. It is done for both starting and ending boundary. As most of the errors can occur around boundary it best to test there. So by testing only six values the results can be known. It is a part of black box testing.

**Q: What is Equivalence partitioning testing?**

A: It is a methodology of software testing in which a set of valid inputs are treated equal and any one value of input is tested from the group to verify the set of inputs along with one or two invalid values i.e. values outside the set. It is also a part of black box testing.

**Q: What is Integration testing?**

A: It is a type of testing where small modules are connected logically and the connection between these modules are tested. Its main aim is to check the data flow between the modules and also to check whether new problems arises or not when different modules are interconnected.

**Q: What determines the level of risk?**

A: The level of risk can be determined by variety of factors like likelihood of event occurring, the potential consequences when it occurs and the ability to manage the risk.

**Q: What is Alpha testing?**

A: It is a type of testing which is done at software development site. It can be done by developers or an independent testing team. It is performed in a virtual environment.

**Q: What is beta testing?**

A: Beta testing is done in real time environment by the customers at their own site using their own data. It is always open to market and public. It is a kind of black box testing and also acceptance testing. Beta testing can also be considered as ‘pre-release’ testing.

**Q: What is component testing?**

A: Component testing is a type of software testing where components are first tested individually before they are integrated with other components to check whether the components are functioning properly or not. It is also referred as module testing or unit testing.

**Q: What is functional system testing?**

A: It is a type of testing in which it is checked whether the features of the system is working properly as mentioned in the requirements. Each function is checked according to the requirement to assure output is consistent with the expectations.

**Q: What is Non-Functional testing?**

A: Non-functional testing is focused on the non-functional aspects of software application such as speed, stability, scalability etc. It is important to test these things before the launch of product because failing in any of these things can cause negative impact in market and can also cause revenue loss.

**Q: What is GUI testing?**

A: GUI testing is the kind of testing in which testing is done of visible things which can be seen on the display such as to check the alignment, fonts, components functioning properly, color, same page in different environment etc. So basically source code is not required in this type of testing.

**Q: What is Adhoc testing?**

A: It is a type of unstructured testing in which testing is done on any part of the application. It does not follow any documentation. It is normally done by a technique called Error Guessing. It is a technique in which the experienced tester randomly selects a part to test which in their expertise can cause problems. So this type of testing is mainly done by experienced testers.

**Q: What is load testing?**

A: It is a type of performance testing in which the system is tested by applying load to it. The goal is to check how system responds when a large amount of load is applied. It helps to improve performance of system. It is done before the deployment of the system.

**Q: What is stress testing?**

A: The main goal of stress testing is to break the system. In this testing a load beyond normal expected load is applied to the system to test it beyond its capacity. So a max point at which system can hold the load is known. Also how the system works at such loads can be evaluated and when in actual conditions if system breaks then they can be prepared for the next steps to stabilize the system.

**Q: What is white box testing and list the types of white box testing.**

A: White box testing is a testing related to the internal structure of the system. It focuses on how the system works. In this testing the tester needs to look inside the source code and determine which part of the code is misbehaving.

The types of white box testing are Unit testing, Testing for memory leaks, Penetration testing, Mutation testing etc.

**Q: What is black box testing? What are the different black box testing techniques?**

A: In this type of testing the internal source code is not required. The tester only focuses on input and the output of the system and not onto the internal system of how it is done. It only checks if required output is achieved when the inputs are given.

The different black box testing techniques are:

* Equivalence Partitioning
* Boundary value analysis
* Decision table
* State transition
* UML diagram

**Q: Mention what are the categories of defects?**

A: The categories of defect are:

* Data Quality/Database defect: Deals with improper handling of database
* Critical Functionality defect: The defect in functional part which can cause severe effect is called critical functionality defect
* Functionality defect: The defects that affects the functionality of the product are called functionality defect
* Security defect: Defects that occur during authentication or authorization are called security defects. These types of defects are given highest priority
* User Interface Defect: The defects which belongs to user interface are called user interface defects. Normally these defects are less severe.

**Q: Mention what bigbang testing is?**

A: In this type of testing all the components and modules are integrated and testing is done on whole integrated system. The main drawback in this testing is that when an error occurs then it is difficult to find which part of code is misbehaving.

**Q: What is the purpose of exit criteria?**

A: The purpose of exit criteria is to determine when to stop testing. It can be achieved when all the functions are tested or the time has ended for testing or all the defects have been fixed. So after such criteria the tester needs to stop testing.

**Q: When should “Regression Testing” be performed?**

A: Regression testing is performed when there is a modification in a program or the environment of system is changed. It is done to ensure that with the modification or change in environment new problems are not arising in the program.

**Q: What is 7 key principles? Explain in detail.**

A: 7 key principles are the general testing principles. These principles are:

1. 100% defect free is impossible: Testing shows that there are defects but it cannot prove that there are no defects. There may be some uncovered defects which can not be found. With more testing more defects can be found but 100% defect free is impossible.
2. Exhaustive testing is impossible: In some cases there may be lots of input combinations to test which could take a lot of time and resource. And if there is a time scale then all these combinations are impossible to test. So a tester needs to prioritise based on risk based approach. So exhaustive testing is impossible.
3. Early testing: Testing should be started as early as possible because if a defect is found early the it can be solved at early stage of the process which will help in development of program easily and it reduced time wastage if the same defect is found later.
4. Defect clustering: Defects are not evenly spread in the program. Normally a small segment could contain maximum number of defects. So it is said that the defects are detected in clusters.
5. Pesticide paradox: If the same tests are performed again and again then new defects cannot be found. So the test cases should be updated regularly to find new defects.
6. Testing is context dependent: Testing is done differently in different contexts. Like different websites are tested differently. So it is said that testing is context dependent.
7. Absence of errors fallacy: Even if a system is error free but it is not performing the functions which it was supposed to do then the system becomes unusable. System should perform according to needs and expectations. So in making a system it is equally important about the requirements as well as defect free system.

**Q: Difference between QA vs QC vs Tester.**

A: QA includes activities that ensure the implementation of processes, procedures and standards in context verification of intended software and its requirements. QC ensures the verification of developed software according to requirements. Tester ensures the identification of bugs, defects, errors in the software.

QA focuses on processes and procedures rather than actual testing. QC focuses on actual testing by executing the software to find bugs or errors through implementation of processes and procedures. Tester focuses on actual testing.

QA has process oriented activities while QC and Tester have product oriented activities.

QA is a subset of Software Test Life Cycle(STLC). QC is a subset of QA. Testing is a subset of QC.

**Q: Difference between smoke and sanity?**

A: Smoke testing is done during builds to check the functionalities of the program. So it is done before the whole program is built. If there is some defect then it is pointed out and send to developer to correct it.

Sanity testing is done after the build goes through some changes to determine there are no new defects in the program. If the build contains defects again then the build is rejected to save time and cost involved in more rigorous testing.

So basically smoke testing verifies critical functionalities of a software while Sanity testing verifies new functionality like bug fixes.

**Q: Difference between verification and validation.**

A: Verification is a development phase process whereas validation is done after coding process.

Verification process is focused on documentation whereas validation is focused on the software development process.

The objective of verification is to ensure that the product is built according to requirements and design specifications. The objective of validation is to ensure that the product meets the actual requirements and performs as desired.

The items that are evaluated in verification are plans, codes, test cases, Design. In validation the evaluation item is the actual product.

Verification activities involve reviews, walkthroughs and inspections. Validation activities involve testing.

**Q: Explain types of performance testing.**

A: There are mainly two types of performance testing:

1. Load testing
2. Stress testing

In load testing the system is put through a load which is expected amount of load during peak time and the performance of system is checked. It helps in knowing how system responds to the load and its stability and performance during the load.

In stress testing the system is put through a load which is more than expected amount of load. Its main aim is to break the system and check how system reacts when it breaks. So the max limit is known at which the system works properly. Also when system breaks how much times it takes to get back system is also measured. It helps the team to work efficiently when the system breaks in actual environment.

**Q: What is error, defect, bug and failure?**

A: A mistake in coding is called error. Error found by tester is called defect. Defect accepted by development team is called bug. If build does not meet the requirements then it is called failure.

**Q: Difference between priority and severity.**

A: Priority is an order in which the developer should resolve the defect while severity is the degree of impact defect has on the operation of the product.

Priority is categorized in three types: low, medium and high while severity is categorized into five types critical, major, moderate, minor and cosmetic.

Priority status is based on customer requirements whereas severity status is based on technical aspect of the product.

High priority status is resolved first whereas high severity status solution depends on priority of the defect.

**Q: What is bug life cycle?**

A: The time span between the first time bug is found and it is resolved successfully is called bug life cycle. When a bug is discovered it goes through several states and eventually reaches one of the terminal states where it becomes inactive and closed.

**Q: Explain the difference between functional testing and non functional testing.**

A: Functional testing is done to check the functional components of the system whereas non functional testing is done to check the performance of the product.

Functional testing checks whether the program is working as per requirements while non functional testing checks the speed, stability and scalability of the product.

Functional testing is done earlier than non functional testing.

Manual testing or automation testing both can be used for functional testing whereas automation testing is preferred for non functional testing.

Functional testing describes what the product does while non functional testing describes how good the product works.

Types of functional testing are smoke testing, sanity testing, unit testing, integration testing, black box testing, white box testing etc. Types of non functional testing are performance testing, volume testing, load testing, stress testing, security testing etc.

**Q: What is difference between STLC(Software Testing Life Cycle) and SDLC(Software Development Life Cycle)**

A: STLC is related to software testing while SDLC is related to software development.

STLC only focuses on testing while SDLC focuses on development and testing.

STLC requires less number of members while SDLC requires more members.

In STLC, testing team makes the plans and design while in SDLC development team makes plans and designs.

Goal of STLC is to complete successful testing of software while goal of SDLC is to complete successful development of software.

STLC helps in making software defect free while SDLC helps is making good quality software.

**Q: What is the difference between test scenarios, test cases and test scripts?**

A: Test scenarios are derived from SRS document. It focuses on all the functionalities of the software that needs to be tested. It tells you what things to be tested. It requires less time for execution.

Test cases are derived from test scenarios. If focuses on set of actions needed to test the functionality of the software. Test case helps in exhaustive testing of the software. It is focused on what to test and how to test. It requires more time and resources for the execution.

Test script is a small set of instructions or a short program to test the functionality of the product. It is an automatic approach of testing. It is beneficial for constant execution. It requires less time for testing scripts.

**Q: Explain what test plan is? What is the information that should be covered?**

A: A test plan is a detailed document that catalogs the test strategies, objectives, schedules, estimations, deadlines and resources required to complete the product. It is a kind of blueprint for running tests. The information covered in the test plan are scope, schedule, resource allocation, Environment, Tools, Defect management, Risk management and exit parameters.

**Q: What is priority?**

A: Priority means the level of importance assigned to the item. It shows how fast the problem should be solved. The different levels of priority are low, high and critical. Critical holds the highest priority level.

**Q: What is severity?**

A: Severity means how much a defect can cause the functionality of the product. There can be some defects which can cause severe effect in the functionality while there can be some defect which causes minimum effect in functionality of the product. So according to these the severity is categorized in different categories such as low, high, critical and blocker.

**Q: Bug categories are?**

A: Bug categories are: Security, Database, Functionality(Critical/General), UI.

**Q: Difference between priority and severity.**

A: Priority means the level of importance assigned to the item while Severity means how much a defect can cause the functionality of the product.

Priority decides which defect should be solved first while severity shows how much a defect can effect the functionality of the program.

Priority can be categorized in three categories: low, high and critical while severity in four categories: low, high, critical and blocker.

Priority is based on customer requirements while severity is based on technical aspect of the product.

**Q: What are the different methodologies of Agile development model?**

A: The different methodologies of Agile development model are scrum and kanban.

Scrum is an agile development model which concentrates particularly on how to manage tasks within a team based environment. It consists of three roles and their responsibilities Scrum master, product owner and scrum team. Scrum master is responsible for setting up the team, sprint meeting and removes obstacles to progress. Product owner creates product backlog, prioritizes the backlog and is responsible for the delivery of the functionality of each iteration. Scrum team manages its own work and organizes the work to complete the sprint or cycle.

Kanban is a very popular framework for development in agile software development methodology. It provides a transparent way of visualizing the tasks and work capacity of a team. It mainly uses physical and digital boards to allow the team members to visualize the current state of the project they are working on. The kanban board has columns and story cards. The columns are nothing but workflow states and cards are nothing but a demonstration of the actual task a team member is performing.

**Q: Explain the difference between authorization and authentication in web testing. What are the common problems faced in web testing?**

A: Authorization defines the access rights of the user while authentication verifies the identity of the user.

Authentication is done before authorization process.

In authentication users login details are required while in authorization users privilege levels are needed.

The authentication part can be changed by the user but authorization part cannot be changed by the user.

The common problems faced in web testing are: Integration, Interoperability, security, performance and usability.

Integration testing exposes problems when different builds are integrated.

Interoperability is also a common issue when end to end testing is done.

Security is one of the most important factor. It deals with unsecured communications, removing malicious files etc. Data integrity is dependent on security of the product.

Performance depends on the speed of the product. A good functional product with slow speed is of no importance so performance is one of the common problems.

There are so many different environment where a user can use the website so proper usability of the website on different platforms is essential. So usability is also a common problem testers face during web testing.