1. **What is Error, Defect, Bug and Failure?**

A mistake in coding is called Error. Error is found by tester is called Defect. Defect accepted by development team then it is called Bug. Build does not meet the requirements then its failure.

1. **What is traceability matrix?**

To protect against changes you should be able to trace back from every system component to the original requirement that caused its presence.

1. **What is integration testing?**

Integration testing is expose defects in the interfaces and in the interaction between integrated components or systems.

1. **What is component testing?**

A minimal software item that can be tested in isolation.Its means a unit is the smallest testable part of software.

1. **What is functional system testing?**

A requirement that specifies a function that a system or system component must perform

1. **What is non functional testing?**

Testing the attributes of a component or system that do not relate to functionality. E.g. reliability, efficiency, usability, interoperability, maintainability and portability.

1. **Difference between QA v/s QC v/s Tester**

QA : QA is process oriented activities

Preventive activities.

QA is a subset od software test life cycle

QC : QC is product oriented activities

It is a corrective process.

QC can be considered as the subset of quality assurance

Tester: tester is a product oriented activities.

It is a preventive process.

Tester is the subset of the quality control

**8. What is Exploratory Testing?**

Exploratory testing is a concurrent process which test designing, execution and logging happen simultaneously

** What is Boundary value testing?**

Boundary testing is the process of testing between extreme ends or boundaries between partitions of the input values.

** What is Equivalence partitioning testing?**

**Equivalence Partitioning** or Equivalence Class Partitioning is type of black box testing technique which can be applied to all levels of [software testing](https://www.guru99.com/software-testing.html) like unit, integration, system, etc. In this technique, input data units are divided into equivalent partitions that can be used to derive test cases which reduces time required for testing because of small number of test cases.

** What determines the level of risk?**

Risk is a factor that could result in future negative consequence. The risk level can be **low, moderate or high**. Each enterprise risk has a risk level based on the impact and likelihood ranking of the risk. The risk level provides the basis for prioritization and action.

** What is Alpha testing?**

**Alpha Testing** is a type of software testing performed to identify bugs before releasing the product to real users or the public. Alpha Testing is one of the [u**ser acceptance tests**](https://www.geeksforgeeks.org/user-acceptance-testing-uat/)**.** This is referred to as[**Alpha testing**](NULL)only because it is done early on, near the end of the [development of the software](https://www.geeksforgeeks.org/what-is-software-development/).

** What is beta testing?**

Beta testing is defined as a phase in the software development lifecycle where a nearly completed product is released to a select group of external users for real-world testing. This stage follows alpha testing, where the product is tested internally by the developers and quality assurance team.

** What is GUI Testing?**

GUI testing, also known as User Interface Testing (UI testing), is the process of validating the visual elements of a software application. It focuses on verifying the visual components that users interact with, such as buttons, menus, icons, text boxes, lists, forms, images, and other interactive elements**.**

** What is Adhoc testing?**

Adhoc testing is a informal testing type with an aim to break the system

** What is load testing?**

Load testing is to test the system behavior under normal workload condition, and it is just testing simulating with the actual workload.

** What is stress Testing?**

Stress testing is to test the system behavior under extreme condition and is carried out till the system failure

** What is white box testing and list the types of white box testing?**

Whitebox testing is based on an analysis of the internal structure of the component or system

There are three types of technique:

1. Statement coverage
2. Decision coverage
3. Condition coverage

** What is black box testing? What are the different black box testing techniques?’**

Black box testing either functional or non- functional, without reference to the internal structure of the component or system

There are 4 type technique:

1. Equivalence partitioning
2. Boundary value analysis
3. decision table
4. state transition testing

** Mention what are the categories of defects?**

Data quality /database defect: deals with improper handling of data in the database.

Critical functionality defects: the occurrence of these bugs hampers the crucial functionality of the application

Functionality defects: these defects affect the functionality of the application

Security defects: application security defects generally involve improper handling of data sent from the user to the application.

User interface defects: as the name suggest, the bugs deal with problems related to UI are usually considered less severe.

** Mention what bigbang testing is?**

In big bang integration testing all components or module is integrated simultaneously, after which everything is tested as a whole

** What is the purpose of exit criteria?**

Software testing teams will use exit criteria to determine if a test plan or project can exit to the next stage or be considered complete. This isn't something that should be left up to the subjective and/or ad hoc decisions of a test admin or SQA engineer, as it can directly impact the success of the next stage or project as a whole.

Creating exit criteria helps:

* Align your teams on a common definition of test completion
* Ensure your product meets completion standards before entering the next stage, which avoids costly project delays
* Create clear parameters for test engineers to evaluate software

** When should "Regression Testing" be performed?**

[Regression testing should be performed in the following scenarios](https://www.bing.com/ck/a?!&&p=2f7c513cb6e1ea41JmltdHM9MTcyNDE5ODQwMCZpZ3VpZD0yMWFjMWVmYi02YmYyLTY0YWEtMzE1Ni0wYTUzNmEwMDY1YTEmaW5zaWQ9NTgyOQ&ptn=3&ver=2&hsh=3&fclid=21ac1efb-6bf2-64aa-3156-0a536a0065a1&psq=when+should+regression+testing+be+performed&u=a1aHR0cHM6Ly9ibG9nLmFpcmJyYWtlLmlvL2Jsb2cvc29mdHdhcmUtdGVzdGluZy9yZWdyZXNzaW9uLXRlc3Rpbmc&ntb=1)

* After any change is made to the code base.
* When a previously discovered issue has been marked as fixed and must be verified.
* When new functionality is added to the software application.
* When there is a requirement to change.
* When a defect is fixed.
* When a functional/performance issue is fixed.
* When there is a change in the environment

** What is 7 key principles? Explain in detail?**

1. Testing shows Presence of defect

This testing principle states that software testing concerns the presence of defects, not their absence. It involves identifying bugs in the software product and fixing them before deploying and releasing it to the market.

1. Exhausting testing is impossible

Before shedding light on the principle, let us understand [exhaustive testing](https://artoftesting.com/exhaustive-testing). It tests and verifies every functionality of the application using both valid and invalid inputs

1. Early testing

This testing principle says that it is beneficial to start testing as early as possible in the [software development life cycle](https://artoftesting.com/software-development-life-cycle-sdlc) (SDLC). In fact, validating requirements starts even before coding. This uncovers potential bugs and errors in the software design and prevents the cost required to fix them if they were found in the later stages of development.

1. Defect clustering

Defect clustering refers to an application’s modules or features containing more bugs, generally resulting in operational failure. It is very less likely that bugs are distributed evenly throughout the application. Hence, testing should concentrate on such modules and features

1. The pesticide paradox

When you spray pesticides on crops, insects get used to them and build immunity, making them ineffective to pesticides as time passes. When you repeat the same test cases repeatedly, they will stop finding new issues.

1. Testing is context dependent

Every software is developed with a different purpose. Hence, testing depends on the context of the software. As [different types of software](https://artoftesting.com/types-of-software) have distinct requirements, they require different types of testing. The same tests cannot detect all the major flaws in different types of applications, as users might have different ways of interacting with the product.

1. Absence of error fallacy

This principle states that it becomes unusable if your application is 99% bug-free but does not satisfy the end user’s requirements or business needs

** Difference between Smoke and Sanity?**

Smoke Testing is performed after software build to ascertain that the critical functionalities of the

program is working fine

➢ In Smoke Testing, the test cases chosen cover the most important functionality or component of the system.

3. Deference:

1. The objective of this testing is to verify "stability" of the system in order to with more rigorous testing

2. This testing is performed by the developers or testers

3. Smoke testing is usually documented or scripted

4. Smoke testing is a subset of Regression testing

5. Smoke testing exercises the entire system from end to end

6. Smoke testing is like General Health Check Up

Sanity Testing: After receiving a software build, with minor changes in code, or functionality, Sanity testing is performed to ascertain that the bugs have been fixed and no further issues are introduced due to these changes

➢ If sanity test fails, the build is rejected to save the time and costs involved in a more rigorous testing

1. The objective of the testing is to verify the "rationality" of the system in

order proceed to proceed with more rigorous testing

2. Sanity testing is usually performed by testers

3. Sanity testing is a subset of Acceptance testing

4. Sanity Testing is like specialized health Check Up

** Difference between verification and Validation**

**Verification:**

* 1. The process of evaluating work products of a development phase to determine whether they meet specified requirements for that phase
  2. Are we building the product right?
  3. Plans, requirement space, design specs, code, test cases
  4. Activities are reviews, walkthrough, inspection

**Validation:**

1. The process of evaluating software during or at the end of the development process to determine whether it satisfied business requirements
2. Are we building the right product?
3. The actual product/software evaluation items
4. Activities is testing

** Explain types of Performance testing.**

Types of Performance Testing :

1. Load testing

Load testing is a [type of testing](https://artoftesting.com/types-of-testing) which involves evaluating the performance of the system under the expected workload. A typical load test includes determining the response time, throughput, error rate, etc during the course of the load test.

2. Stress testing

Stress testing is a type of performance testing where we evaluate the application’s performance at a load much higher than the expected load. Another aspect of the stress testing is to determine the break-point of the application, the point at which the application fails to respond in the correct manner.

3. Endurance testing

Endurance testing is also known as ‘Soak Testing’. It is done to determine if the system can sustain the continuous expected load for a long duration. Issues like memory leakage are found with endurance testing.

4. Spike testing

In spike testing, we analyze the behavior of the system on suddenly increasing the number of users. It also involves checking if the application is able to recover after the sudden burst of users.

5.Volume testing

The volume testing is performed by feeding the application with a high volume of data. The application can be tested with a large amount of data inserted in the database or by providing a large file to the application for processing. Using volume testing, we can identify the bottleneck in the application with a high volume of data.

** Difference between Priority and Severity**

* **Severity** refers to the impact of a defect on the functionality of the software.
* **Priority** refers to the urgency of resolving the problem.
* Severity is an objective measure of a bug's impact.
* Priority reflects how much the software matters to the business or how likely the flaw is to occur.
* Managers or clients decide on the priority, while QA engineers determine the severity levels.

** What is Bug Life Cycle?**

- “A computer bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bugs arise from mistakes and errors, made by people, in either a program’s source code or its design. NEW, ASSIGNED, OPEN ,FIXED ,PENDING RETEST ,RETEST, VERIFIED, CLOSE

** Explain the difference between Functional testing and NonFunctional testing**

Functional Testing:

➢ Functional Testing: Testing based on an analysis of the specification of the functionality of a component or system.

➢ This testing mainly involves black box testing and it is not concerned about the source code of the application

➢ Functional testing is executed first

➢ Manual testing or automation tools can be used for functional testing

➢ Business requirements are the inputs to functional testing

➢ Functional testing describes what the product does

➢ Easy to do manual testing

➢ Types of Functional testing are

Unit Testing

Smoke Testing

White box testing

Black Box testing

Non-Functional Testing :

➢ Non-Functional Testing: Testing the attributes of a component or system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability

➢ Non functional testing should be performed after functional testing

➢ Using tools will be effective for this testing

➢ Nonfunctional testing describes how good the product works

➢ Tough to do manual testing

➢ Types of Nonfunctional testing are ∙

\* Performance Testing

\* Load Testing

\* Volume Testing

\* Stress Testing

**What isthe difference between the STLC (Software Testing Life Cycle) and SDLC(Software Development Life Cycle)?**

** What is the difference between test scenarios, test cases, and test script?**

** Explain what Test Plan is? What isthe information that should be covered.**

** What is priority?**

** What is severity?**

** Bug categories are…**

** Advantage of Bugzila .**

Explain the difference between Authorization and Authentication in Web testing.What are the common problems faced in Web testing?