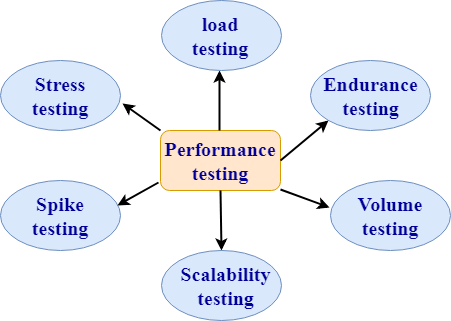
What are the types of performance testing?

**Performance testing**: Performance testing is a testing technique which determines the performance of the system such as speed, scalability, and stability under various load conditions. The product undergoes the performance testing before it gets live in the market.

**Types of software testing are:**



**1. Load testing:**

* Load testing is a testing technique in which system is tested with an increasing load until it reaches the threshold value.

#### **Note: An increasing load means the increasing the number of users.**

* The main purpose of load testing is to check the response time of the system with an increasing amount of load.
* Load testing is non-functional testing means that the only non-functional requirements are tested.
* Load testing is performed to make sure that the system can withstand a heavy load

**2. Stress testing:**

* Stress testing is a testing technique to check the system when hardware resources are not enough such as CPU, memory, disk space, etc.
* In case of stress testing, software is tested when the system is loaded with the number of processes and the hardware resources are less.
* The main purpose of stress testing is to check the failure of the system and to determine how to recover from this failure is known as recoverability.
* Stress testing is non-functional testing means that the only non-functional requirements are tested.

**3. Spike testing:**

* Spike testing is a subset of load testing. This type of testing checks the instability of the application when the load is varied.
* There are different cases to be considered during testing:
  + The first case is not to allow the number of users so that the system will not suffer heavy load.
  + The second case is to provide warnings to the extra joiners, and this would slow down the response time.

**4. Endurance testing:**

* Endurance testing is a subset of load testing. This type of testing checks the behavior of the system.
* Endurance testing is non-functional testing means that the only non-functional requirements are tested.
* Endurance testing is also known as Soak testing.
* Endurance testing checks the issues such as memory leak. A memory leak occurs when the program does not release its allocated memory after its use. Sometimes the application does not release its memory even after its use and this unusable memory cause memory leak. This causes an issue when the application runs for a long duration.
* Some of the main issues that are viewed during this testing are:
  + Memory leaks occurred due to an application.
  + Memory leaks occurred due to a database connection.
  + Memory leaks occurred due to a third party software.

**5. Volume testing:**

* Volume testing is a testing technique in which the system is tested when the volume of data is increased.
* Volume testing is also known as flood testing.
* Volume testing is non-functional testing means that the only non-functional requirements are tested.
* For example: If we want to apply the volume testing then we need to expand the database size, i.e., adding more data into the database table and then perform the test.

**6. Scalability testing**

* Scalability testing is a testing technique that ensures that the system works well in proportion to the growing demands of the end users.
* Following are the attributes checked during this testing:
  + Response time
  + Throughput
  + Number of users required for performance test
  + Threshold load
  + CPU usage
  + Memory usage
  + Network usage

### What is the difference between functional and non-functional testing?

|  |  |  |
| --- | --- | --- |
| **Basis of comparison** | **Functional testing** | **Non-functional testing** |
| Description | Functional testing is a testing technique which checks that function of the application works under the requirement specification. | Non-functional testing checks all the non-functional aspects such as performance, usability, reliability, etc. |
| Execution | Functional testing is implemented before non-functional testing. | Non-functional testing is performed after functional testing. |
| Focus area | It depends on the customer requirements. | It depends on the customer expectations. |
| Requirement | Functional requirements can be easily defined. | Non-functional requirements cannot be easily defined. |
| Manual testing | Functional testing can be performed by manual testing. | Non-functional testing cannot be performed by manual testing. |
| Testing types | Following are the types of functional testing:   * Unit testing * Acceptance testing * Integration testing * System testing | Following are the types of non-functional testing:   * Performance testing * Load testing * Stress testing * Volume testing * Security testing * Installation testing * Recovery testing |

### Smoke Testing | Software Testing

**Smoke Testing** is a software testing method that determines whether the employed build is stable or not. It acts as a confirmation of whether the quality assurance team can proceed with further testing. Smoke tests are a minimum set of tests run on each build.

In other words, we verify whether the important features are working and there are no showstoppers in the build that is under testing. It is a mini and quick regression test of major functionality. Smoke testing shows that the product is ready for testing. This helps in determining if the build is flawed so as to make any further testing a waste of time and resources.

### Sanity testing

**Sanity Testing** is a subset of regression testing. Sanity testing is performed to ensure that the code changes that are made are working as properly. Sanity testing is a stoppage to check whether testing for the build can proceed or not.

**Functionality of Sanity Testing:**   
The major functionality of sanity testing is to determine that the changes or the proposed functionality are working as expected. If the sanity test fails, software product is rejected by the testing team to save time and money. It is performed only after the software product has passed the smoke test and Quality Assurance team has accepted for further testing.

### Exploratory Testing

Exploratory Testing is a type of software testing where Test cases are not created in advance but testers check system on the fly. They may note down ideas about what to test before test execution.

Simultaneous test design and execution against an application is called exploratory testing. In this testing, the tester uses his domain knowledge and testing experience to predict where and under what conditions the system might behave unexpectedly.

### What is traceability matrix?

The relationship between test cases and requirements is shown with the help of a document. This document is known as a traceability matrix.

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

Black box testing is the software testing method which is used to test the software without knowing the internal structure of code or program. This testing is usually done to check the functionality of an application. The different black box testing techniques are

1. Equivalence Partitioning
2. Boundary value analysis
3. Cause-effect graphing

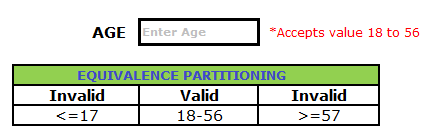
### Equivalence Partitioning

Equivalence Partitioning is also known as Equivalence Class Partitioning. In equivalence partitioning, inputs to the software or system are divided into groups that are expected to exhibit similar behavior, so they are likely to be proposed in the same way. Hence selecting one input from each group to design the test cases.

### Example on Equivalence Partitioning Test Case Design Technique:

**Example 1:**

Assume, we have to test a field which accepts Age 18 – 56

[](https://www.softwaretestingmaterial.com/wp-content/uploads/2016/03/Equivalence-Partitioning-1.png)

### Boundary Value Analysis

Boundary value analysis is a black-box testing technique. In this technique, we analyze the behavior of the application with test data residing at the boundary values of the equivalence classes. Software testing or rather exhaustive software testing is a very time and resource-intensive activity.

[Boundary Value Analysis](https://www.geeksforgeeks.org/boundary-value-analysis-triangle-problem/) is based on testing the boundary values of valid and invalid partitions.

**Valid Test cases:**Valid test cases for the above can be any value entered greater than 17 and less than 57.

* Enter the value- 18.
* Enter the value- 19.
* Enter the value- 37.
* Enter the value- 55.
* Enter the value- 56.

What is the difference between static and dynamic testing?

|  |  |
| --- | --- |
| **Static testing** | **Dynamic testing** |
| Static testing is a white box testing technique which is done at the initial stage of the software development lifecycle. | Dynamic testing is a testing process which is done at the later stage of the software development lifecycle. |
| Static testing is performed before the code deployment. | Dynamic testing is performed after the code deployment. |
| It is implemented at the verification stage. | It is implemented at the validation stage. |
| Execution of code is not done during this type of testing. | Execution of code is necessary for the dynamic testing. |
| In the case of static testing, the checklist is made for the testing process. | In the case of dynamic testing, test cases are executed. |

### What are verification and validation?

Verification is a process of evaluating software at the development phase. It helps you to decide whether the product of a given application satisfies the specified requirements. Validation is the process of evaluating software at the after the development process and to check whether it meets the customer requirements.

**What are the different test levels?**

There are four test levels

1. Unit/component/program/module testing
2. Integration testing
3. System testing
4. Acceptance testing

### What is a Test Case?

A **Test Case** is a set of actions executed to verify a particular feature or functionality of your software application. A Test Case contains test steps, test data, precondition, postcondition developed for specific test scenario to verify any requirement.

**Parameters of a Test Case:**

* **Module Name:**Subject or title that defines the functionality of the test.
* **Test Case Id:**A unique identifier assigned to every single condition in a test case.
* **Tester Name:**The name of the person who would be carrying out the test.
* **Test scenario: The test** scenario provides a brief description to the tester, as in providing a small overview to know about what needs to be performed and the small features, and components of the test.
* **Test Case Description:** The condition required to be checked for a given software. for eg. Check if only numbers validation is working or not for an age input box.
* **Test Steps:** Steps to be performed for the checking of the condition.
* **Prerequisite:**The conditions required to be fulfilled before the start of the test process.
* **Test Priority:**As the name suggests gives the priority to the test cases as in which had to be performed first, or are more important and which could be performed later.
* **Test Data:** The inputs to be taken while checking for the conditions.
* **Test Expected Result:** The output which should be expected at the end of the test.
* **Test parameters:**Parameters assigned to a particular test case.
* **Actual Result:** The output that is displayed at the end.
* **Environment Information:**The environment in which the test is being performed, such as operating system, security information, the software name, software version, etc.
* **Status:** The status of tests such as pass, fail, NA, etc.
* **Comments:** Remarks on the test regarding the test for the betterment of the software.

### Test Case vs Test Scenario

Below are some of the points of difference between a test case and a test scenario:

| S No. | **Test Case** | **Test Scenario** |
| --- | --- | --- |
| 1. | A test case is a defined format for software testing required to check if a particular application/software/module is working or not. Here we check for different conditions regarding the same. | Test Scenario provides a small description of what needs to be performed based on the use case. |

**What Test Plans consists of?**

Test design, scope, test strategies, approach are various details that Test plan document consists of.

1. Test case identifier
2. Scope
3. Features to be tested
4. Features not to be tested
5. Test strategy & Test approach
6. Test deliverables
7. Responsibilities
8. Staffing and training
9. Risk and Contingencies

### What is the difference between UAT and System testing?

System Testing: System testing is finding defects when the system undergoes testing as a whole; it is also known as end-to-end testing. In such type of testing, the application suffers from beginning till the end.

UAT: User Acceptance Testing (UAT) involves running a product through a series of specific tests which determines whether the product will meet the needs of its users.

### Mention the difference between Data Driven Testing and Retesting?

**Retesting:** It is a process of checking bugs that are actioned by the development team to verify that they are fixed.

**Data Driven Testing (DDT):**In data driven testing process, the application is tested with multiple test data. The application is tested with a different set of values.

**What are the valuable steps to resolve issues while testing?**

* Record: Log and handle any problems which have happened
* Report: Report the issues to higher level manager
* Control: Define the issue management process

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

Difference between test scenarios and test cases is that

**Test Scenarios:** A Test Scenario is any functionality that can be tested. It is also called Test Condition or Test Possibility.

**Test Cases:** It is a document that contains the steps that have to be executed; it has been planned earlier.

**Test Script:**It is written in a programming language and it’s a short program used to test part of the functionality of the software system. In other words a written set of steps that should be performed manually.

### What is Latent defect?

**Latent defect:**This defect is an existing defect in the system which does not cause any failure as the exact set of conditions has never been met

**What are the two parameters which can be useful to know the quality of test execution?**

To know the quality of test execution, we can use two parameters

* Defect reject ratio
* Defect leakage ratio

### Explain what Test Deliverables is?

Test Deliverables are a set of documents, tools and other components that have to be developed and maintained in support of testing.

There are different test deliverables at every phase of the software development lifecycle

* Before Testing
* During Testing
* After the Testing

**What is mutation testing?**

Mutation testing is a technique to identify if a set of test data or test case is useful by intentionally introducing various code changes (bugs) and retesting with original test data/ cases to determine if the bugs are detected.

**What all things you should consider before selecting automation tools for the AUT?**

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

**How will you conduct Risk Analysis?**

For the risk analysis following steps need to be implemented

1. Finding the score of the risk
2. Making a profile for the risk
3. Changing the risk properties
4. Deploy the resources of that test risk
5. Making a database of risk

**What are the categories of debugging?**

Categories for debugging

1. Brute force debugging
2. Backtracking
3. Cause elimination
4. Program Slicing
5. Fault tree analysis

**What is fault masking explain with example?**

When the presence of one defect hides the presence of another defect in the system, it is known as fault masking.

Example: If the “Negative Value” cause a firing of unhandled system exception, the developer will prevent the negative values input. This will resolve the issue and hide the defect of unhandled exception firing.

**Explain what Test Plan is? What is the information that should be covered in Test Plan?**

A test plan can be defined as a document describing the scope, approach, resources, and schedule of testing activities and a test plan should cover the following details.

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

**What is the common risk that leads to project failure?**

The common risk that leads to a project failure are

* Not having enough human resource
* Testing Environment may not be set up properly
* Limited Budget
* Time Limitations

**Explain what is testing type and what are the commonly used testing type?**

To get an expected test outcome, a standard procedure is followed which is referred to as Testing Type.

Commonly used testing types are

* Unit Testing: Test the smallest code of an application
* API Testing: Testing API created for the application
* Integration Testing: Individual software modules are combined and tested
* System Testing: Complete testing of the system
* Install/UnInstall Testing: Testing done from the point of client/customer view
* Agile Testing: Testing through Agile technique

**What does a typical test report contain? What are the benefits of test reports?**

A test report contains the following things:

* Project Information
* Test Objective
* Test Summary
* Defect

The benefits of test reports are:

* Current status of project and quality of product are informed
* If required, stakeholder and customer can take corrective action
* A final document helps to decide whether the product is ready for release

### In manual testing what are stubs and drivers?

Both stubs and drivers are part of incremental testing. In incremental testing, there are two approaches namely bottom-up and top-down approach. Drivers are used in bottom-up testing and stub is used for a top-down approach. In order to test the main module, the stub is used, which is a dummy code or program.

### What is the step you would follow once you find the defect?

Once a defect is found you would follow the step

a) Recreate the defect

b) Attach the screenshot

c) Log the defect

**Explain what N+1 testing is?**

The variation of regression testing is represented as N+1. In this technique, the testing is performed in multiple cycles in which errors found in test cycle ‘N’ are resolved and re-tested in test cycle N+1. The cycle is repeated unless there are no errors found.

**What is Fuzz testing and when it is used?**

Fuzz testing is used to detect security loopholes and coding errors in software. In this technique, random data is added to the system in an attempt to crash the system. If vulnerability persists, a tool called fuzz tester is used to determine potential causes. This technique is more useful for bigger projects but only detects a major fault.

### Mention what the difference between a “defect” and a “failure” in software testing is?

In simple terms when a defect reaches the end customer, it is called a failure while the defect is identified internally and resolved; then it is referred to as a defect.

### Explain which test cases are written first black boxes or white boxes?

Black box test cases are written first as to write black box test cases; it requires project plan and requirement document all these documents are easily available at the beginning of the project. While writing white box test cases requires more architectural understanding and is not available at the start of the project.

**Explain what the difference between latent and masked defects is?**

* **Latent defect:** A latent defect is an existing defect that has not caused a failure because the sets of conditions were never met
* **Masked defect:** It is an existing defect that has not caused a failure because another defect has prevented that part of the code from being executed

### Mention what bottom-up testing is?

Bottom-up testing is an approach to integration testing, where the lowest level components are tested first, then used to facilitate the testing of higher level components. The process is repeated until the component at the top of the hierarchy is tested.

**Mention what the basic components of defect report format are?**

The essential components of defect report format include

* Project Name
* Module Name
* Defect detected on
* Defect detected by
* Defect ID and Name
* Snapshot of the defect
* Priority and Severity status
* Defect resolved by
* Defect resolved on

**Explain in a testing project what testing activities would you automate?**

In testing project testing activities, you would automate are

* Tests that need to be run for every build of the application
* Tests that use multiple data for the same set of actions
* Identical tests that need to be executed using different browsers
* Mission critical pages
* A transaction with pages that do not change in a short time

### What is risk-based testing?

Risk-based[Testing](https://www.guru99.com/software-testing.html)is the term used for an approach to creating a[Test Strategy](https://www.guru99.com/how-to-create-test-strategy-document.html)that is based on prioritizing tests by risk. The basis of the approach is a detailed risk analysis and prioritizing of risks by risk level. Tests to address each risk are then specified, starting with the highest risk first.

How do we test a web application? What are the types of tests we perform on the web application?

To test any web application such as **Yahoo, Gmail**, and so on, we will perform the following testing:

* Functional testing
* Integration testing
* System testing
* Performance testing
* Compatibility testing ( test the application on the various operating systems, multiple browsers, and different version)
* Usability testing ( check whether it is user friendly)
* Ad-hoc testing
* Accessibility testing
* Smoke testing
* Regression testing
* Security testing
* Globalization testing ( only if it is developed in different languages)

For which and all types of testing do we write test cases?

We can write test cases for the following types of testing:

|  |  |
| --- | --- |
| **Different types of testing** | Test cases |
| **Smoke testing** | In this, we will write only standard features; thus, we can pull out some test cases that have all the necessary functions. Therefore, we do not have to write a test case for smoke testing. |
| **Functional/unit testing** | Yes, we write the test case for unit testing. |
| **Integration testing** | Yes, we write the test case for integration testing. |
| **System testing** | Yes, we write the test case for system testing. |
| **Acceptance testing** | Yes, but here the customer may write the test case. |
| **Compatibility testing** | In this, we don't have to write the test case because the same test cases as above are used for testing on different platforms. |
| **Adhoc testing** | We don't write the test case for the Adhoc testing because there are some random scenarios or the ideas, which we used at the time of Adhoc time. Though, if we identify the critical bug, then we convert that scenario into a test case. |
| **Performance testing** | We might not write the test cases because we will perform this testing with the help of performance tools. |
| **Usability testing** | In this, we use the regular checklist; therefore, we don't write the test case because here we are only testing the look and feel of the application. |
| **Accessibility testing** | In accessibility testing, we also use the checklist. |
| **Reliability testing** | Here, we don't write the manual test cases as we are using the automation tool to perform reliability testing. |
| **Regression testing** | Yes, we write the test cases for functional, integration, and system testing. |
| **Recovery testing** | Yes, we write the test cases for recovery testing, and also check how the product recovers from the crash. |
| **Security testing** | Yes, we write the test case for security testing. |
| **Globalization testing:** **Localization testing** **Internationalization testing** | Yes, we write the test case for L10N testing. Yes, we write the test case for I18N testing. |

How to test a pen?

We can perform both manual and automation testing. First, we will see how we perform manual testing:

|  |  |
| --- | --- |
| **Different types of testing** | **Scenario** |
| **Smoke testing** | Checks that basic functionality is written or not. |
| **Functional/unit testing** | Check that the Refill, pen body, pen cap, and pen size as per the requirement. |
| **Integration testing** | Combine pen and cap and integrate other different sizes and see whether they work fine. |
| **Compatibility testing** | Various surfaces, multiple environments, weather conditions, and keep it in oven and then write, keep it in the freezer and write, try and write on water. |
| **Adhoc testing** | Throw the pen down and start writing, keep it vertically up and write, write on the wall. |
| **Performance testing** | Test the writing speed of the pen. |
| **Usability testing** | Check whether the pen is user friendly or not, whether we can write it for more extended periods smoothly. |
| **Accessibility testing** | Handicapped people use them. |
| **Reliability testing** | Drop it down and write, and continuously write and see whether it leaks or not |
| **Recovery testing** | Throw it down and write. |
| **Globalization testing** **Localization testing** | Price should be standard, expiry date format. |
| **Internationalize testing** | Check whether the print on the pen is as per the country language. |

**What is Alpha testing?**

Pre-release testing by end user representatives at the developer’s site.

**What is beta testing?**

Testing performed by potential customers at their own locations.

### Alpha and beta testing:

Alpha testing is performed in the development environment before it is released to the customer. Input is taken from the alpha testers, and then the developer fixes the bug to improve the quality of a product. Unlike alpha testing, beta testing is performed in the customer environment. Customer performs the testing and provides the feedback, which is then implemented to improve the quality of a product.

### Mention what the difference between Pilot and Beta testing is?

The difference between a pilot and beta testing is that pilot testing is actually done using the product by the group of users before the final deployment, and in beta testing, we do not input real data, but it is installed at the end customer to validate if the product can be used in production.

What is QA and QC

## **What is Quality Assurance?**

Quality assurance is a proactive and preventative process that monitors and verifies that the methods used to manage and develop deliverables were followed and are operational. It detects weaknesses in the process and guarantees that the approaches, strategies, procedures, and methods designed for the projects are correctly implemented.

## **What is Quality Control?**

Quality control is a product-oriented software engineering process, used to guarantee that the approaches, techniques, methods, and processes designed in the project are implemented correctly.

## Defect Life Cycle Explained

1. Tester finds the defect
2. Status assigned to defect- New
3. A defect is forwarded to Project Manager for analyze
4. Project Manager decides whether a defect is valid
5. Here the defect is not valid- a status is given “Rejected.”
6. So, project manager assigns a status **rejected**. If the defect is not rejected then the next step is to check whether it is in scope. Suppose we have another function- email functionality for the same application, and you find a problem with that. But it is not a part of the current release when such defects are assigned as a **postponed or deferred**status.
7. Next, the manager verifies whether a similar defect was raised earlier. If yes defect is assigned a status **duplicate**.
8. If no the defect is assigned to the developer who starts fixing the code. During this stage, the defect is assigned a status **in- progress.**
9. Once the code is fixed. A defect is assigned a status **fixed**
10. Next, the tester will re-test the code. In case, the[Test Case](https://www.guru99.com/test-case.html)passes the defect is **closed.** If the test cases fail again, the defect is **re-opened** and assigned to the developer.
11. Consider a situation where during the 1st release of Flight Reservation a defect was found in Fax order that was fixed and assigned a status closed. During the second upgrade release the same defect again re-surfaced. In such cases, a closed defect will be **re-opened.**

### what is test bed in testing

* A Test Bed in Software Testing is a software development environment. It allows developers to test their modules without affecting the live production servers. Test bed is not only confined to developers but also used by testers. It is referred to as a test environment for rigorous and transparent testing of new technologies.

### Role of documentation

Software testing requires serialized and planned documentation. This is the reason most software developers concentrate on serious and regular documentation of software development from the beginning to the end. Appropriate and detailed documentation results inaccurate testing further making the processes of testing organized and easy. Eventually, the cost and time of the testing in usually minimised if appropriate documentation has been emphasized from the beginning.

In addition to this, proper documentation also helps the client to evaluate and review the software as well as its processes. The bottom line of the importance of documentation in software development is that documentation saves money, time and efforts of the organization. For giant software development companies like Adobe, Microsoft, TCS, oracle etc. it becomes easier for them to release their new product or software with the processes of documentation which eventually makes it easier for any user to understand the software. On the other hand, poor documentation in a project of software development seriously affects the standard of software as well as its applications.

### ****What is a Bug Report?****

In the course of the QA process, when a bug has been identified, it has to be documented and sent to developers to be fixed. Given that software is exceptionally complex, layered, and feature-heavy in the current digital environment, most QA pipelines generate multiple bugs.

**Elements of an Effective Bug Report**

When studying how to create a bug report, start with the question: What does a bug report need to tell the developer?

A bug report should be able to answer the following questions:

* What is the problem?
* How can the developer reproduce the problem (to see it for themselves)?
* Where in the software (which webpage or feature) has the problem appeared?
* What is the environment (browser, device, OS) in which the problem has occurred?

**How to write an Effective Bug Report**

An effective bug report should contain the following:

1. Title/Bug ID
2. Environment
3. Steps to reproduce a Bug
4. Expected Result
5. Actual Result
6. Visual Proof (screenshots, videos, text) of Bug
7. Severity/Priority

### what is configuration management

Configuration management is a process for maintaining computer systems, servers, and software in a desired, consistent state. It’s a way to make sure that a system performs as it’s expected to as changes are made over time.