1. **Que: What is Exploratory Testing?**

**Ans:** Exploratory Testing is a type of software testing in which the tester is free to select any possible methodology to test the software. It is an unscripted approach to software testing.

1. **Que: What is traceability matrix?**

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

1. **Que: What is Boundary value testing?**

**Ans:** Boundary Value Testing is one of the popular software testing mechanisms, where testing of data is done based on boundary values or between two opposite ends where the ends may be like from start to end, or lower to upper or from maximum to minimum.

1. **Que: What is Equivalence partitioning testing?**

**Ans:** a technique that divides the input domain of a system into partitions or classes that are expected to produce the same output or behaviour.

1. **Que: What is Integration testing?**

**Ans:** Integration testing is the testing performed to expose defect in the interfaces and in the interactions between integrated components or systems.

1. **Que: What determines the level of risk?**

**Ans:** Risk can be defined as the combination of the probability of an event occurring and the consequences if that event does occur. This gives us a simple formula to measure the level of risk in any situation.

1. **Que: What is Alpha testing?**

**Ans:** Alpha testing is the first end-to-end testing of a product to ensure it meets the business requirements and functions correctly.

1. **Que: what is beta testing?**

**Ans:** Beta testing is an opportunity for real users to use a product in a production environment to uncover any bugs or issues before a general release.

1. **Que: What is component testing?**

**Ans:** A minimal software item that can be tested in isolation.

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

**Ans:** A requirement that specifies a function that system or system component must perform

1. **Que: What is Non-Functional Testing?**

**Ans:** Non-functional testing is a type of software testing that verifies non-functional aspects of the product, such as performance, stability, and usability.

1. **Que: What is GUI Testing?**

**Ans:** GUI that means graphical user interface testing is the process of testing the system under test.

1. **Que: What is Adhoc testing?**

**Ans:** performing random testing without any plan is known as Adhoc testing

**Or**

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

1. **Que: What is load testing?**

**Ans:** load testing is a type of Performance Testing that determines the performance of a system, software product, or software application under real-life based load conditions.

1. **Que: What is stress Testing?**

**Ans:** stress testing is a system is stressed beyond its specifications to check how and when it fails. perform under the heavy load like putting large number beyond storage capacity, complex database queries, continuous input or system or database load.

1. **Que: What is white box testing and list the types of white box testing?**

**Ans:** white box testing based on an analysis of the internals structure of the component or system

**TYPES OF WHITE BOX TESTING:**

* Path Testing
* Loop Testing
* Conditional Testing
* Unit Testing
* Mutation Testing
* Integration Testing
* Penetration Testing
* Testing based on Memory Perspective.

1. **Que: What is black box testing?**

**Ans:** Black box testing, either functional or Non-functional, without reference to the internal structure of the component or system.

1. **Que: What are the different black box testing techniques?**

**Ans:** There are 4 specification-based or black-box

* Equivalence partitioning
* Boundary value analysis
* Decision tables
* State transition testing

1. **Que: Mention what are the categories of defects?**

**Ans:** there are 5 types of Defects.

* Data Quality / Database Defects
* Critical functionality Defects
* Functionality Defects
* Security Defects
* User Interface Defects

1. **Que: Mention what Bigbang testing is?**

**Ans:** Big Bang Integration Testing is an integration testing strategy wherein all units are linked at once, resulting in a complete system.

1. **Que: What is the purpose of exit criteria?**

**Ans:** Exit criteria is used to determine whether a given test activity has been completed or NOT.

1. **Que: When should "Regression Testing" be performed?**

**Ans:** after any feature (or application) enhancement, bug fix, or configuration changes.

1. **Que: What is 7 key principles?**

**Ans:**

* Testing shows the presence of defects.
* Exhaustive testing is not possible.
* Early testing.
* Defect clustering.
* Pesticide paradox.
* Testing is context-dependent.
* Absence of errors fallacy.

1. **Que: Explain in detail?**

#### Ans:

#### 1. Testing shows the presence of defects, not their absence

If your QA team reports zero defects after the testing cycle, it does not mean there are no bugs in the software. It means that there could be bugs, but your QA team did not find them. The reasons for not detecting any defects could be many, including the most common one - the test cases do not cover all scenarios. This principle helps in setting stakeholder expectations. Don't make claims that the software is [defect](https://tuskr.app/learn/defect)-free.

**2. Exhaustive testing is impossible**

Consider a simple real-world example in which you have a screen that takes two numbers as input and prints their sum. It would take infinite time to validate this screen for all possible numbers. If a simple screen like this is virtually impossible to test exhaustively, what about a full-fledged application? Trying to test exhaustively will burn time and money without affecting the overall quality. The correct way is to optimize the number of test cases using standard [black-box testing](https://tuskr.app/learn/black-box-testing) and [white-box testing](https://tuskr.app/learn/white-box-testing) strategies.

**3. Early testing saves time and money**

The proverb: a stitch in time saves nine, applies to most things in life, and software testing is no exception. As shown in a study conducted by IBM, what costs a dollar to fix in the design phase can cost fifteen in the testing phase and a whopping hundred if detected in a production system. Early testing also saves time. [Unit testing](https://tuskr.app/learn/unit-testing) and [integration testing](https://tuskr.app/learn/integration-testing) can quickly reveal design flaws, which can cause massive delays if detected later during [system testing](https://tuskr.app/learn/system-testing).

#### 4. Defects cluster together

This principle is an example of the 80:20 rule (also called the Pareto principle) – 80 percent of defects are due to 20 percent of code. While most believe this is some divine mandate, it is based on the observation that 80 percent of users use 20 percent of the software. It is this 20 percent of the software that will contribute most towards the defects.This principle can help your team focus on the right area - the most popular software part.

#### 5. Beware of the pesticide paradox

The use of a pesticide makes pests immune to its use. Similarly, subjecting a module to the same test cases can make the module immune to the test cases. This principle is compelling while testing complex algorithms.

For example, consider a resource scheduling software that schedules resources for tasks based on their work timings, time zones, and holidays. Testers wrote ten test cases related to scheduling, and after four rounds of testing, all the test cases passed. Does this mean the module is defect-free? Probably not, since it took four cycles to clear ten bugs.

**What's the solution?**

Rethink and write more and better test cases, especially when it is a crucial part of the application. Preferably, use a code-coverage tool to ensure that your test cases cover all code paths.

#### 6. Testing is context-dependent

There is no one-strategy-fits-all in software testing. Yes, testing a web application and ATM will be different but what this principle says is more profound.

For example, consider two companies making a ten-pack battery set. The first one is a premium one costing $50 while the other costs $5 - ten times cheaper. Do you think it makes sense for both companies to have the same QA processes? The answer is obvious - no. The first company is charging for quality and hence will have more robust standards and quality control. In your application, don't go for strategies without understanding your [cost of quality](https://tuskr.app/learn/cost-of-quality) - you don't want to spend more on quality than what you get in return. Put quality in the context of your product offering.

#### 7. Beware of the absence-of-errors fallacy

It is a common belief that a low defect rate implies the product is a success. This idea is the absence-of-errors delusion.Zero defects do not mean the software solves end-user problems successfully. Linux always had very few bugs, while Microsoft Windows was (is?) notorious for its bugs. However, most people used Microsoft Windows as their operating system because they found it easier to use and solved their problems better. Linux is becoming more and more mainstream today as it started focusing on end-user experience.

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

**Ans:**

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Quality assurance** | **Quality control** | **Testing** |
| 1 | Activity which ensures the implementation of processes, procedures and standards in context to verification of developed software and intended requirement | Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements. | Activities which ensure the identification of bugs/ error/defects in the software |
| 2 | Focuses on processes and procedures rather than conducting actual testing on the system | Focuses on actual testing by executing software with intend to identify bug/ defect through implementation of procedures and process. | Focuses on actual testing |
| 3 | Process oriented activities | Product oriented activities | Product oriented activities |
| 4 | Preventive activities | It is corrective process. | It is a preventive process. |
| 5 | It is a subset of Software Test life Cycle | QC can be considered as the subset of Quality Assurance. | Testing is the subset of Quality Control. |

1. **Que: Difference between Smoke and Sanity?**

**Ans:**

|  |  |
| --- | --- |
| **Smoke testing** | **Sanity testing** |
| Smoke testing is performed to ascertain that the critical functionalities of the program is working fine | Sanity testing is done to check the new functionality/bugs have been fixed |
| The | The objective of the testing is to verify the |
| This testing is performed by the developers or tester | Sanity testing is usually performed by testers |
| Smoke testing is usually documented or scripted is unscripted | Sanity testing is usually not documented and scripted |
| Smoke testing is a subset of regression testing | Sanity testing is a subset of acceptance testing |
| Smoke testing exercises the entire system from end to end | Sanity testing exercise only the particular component of the entire system |
| Smoke testing is like general health check up | Sanity testing is like specialized health |

1. **Que: Difference between verification and Validation.**

**Ans:**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Verification** | **Validation** |
| **Definition** | The process of evaluating work- product (not the actual final product) of a development phase to determine whether they meet the specified requirement for that phase. | The process of evaluating software during or at the development process to determine whether it satisfies specified business requirements. |
| **Objective** | to ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements | To ensure that the product actually meets the user’s needs, and that the specifications were, to demonstrated that the product fulfil its intended environment. |
| **Question** | Are we building the product, right? | Are we building the right product? |
| **Evaluation item** | Plans, requirement specs, design specs, code, test cases | The actual product/software |
| **Activities** | Reviews walk throughs inspections | Testing |

1. **Que: Explain types of Performance testing.**

**Ans:** There are 6 types of performance testing which is given below:

### **Load Testing**

[Load testing](http://searchsoftwarequality.techtarget.com/definition/load-testing) measures system performance as the workload increases. That workload could mean concurrent users or transactions. The system is monitored to measure response time and system staying power as workload increases. That workload falls within the parameters of normal working conditions.

### **Stress Testing**

Unlike load testing, [stress testing](http://searchsoftwarequality.techtarget.com/definition/stress-testing) also known as fatigue testing is meant to measure system performance outside of the parameters of normal working conditions. The software is given more users or transactions that can be handled. The goal of stress testing is to measure the software stability. At what point does software fail, and how does the software recover from failure?

### **Spike Testing**

Spike testing is a type of stress testing that evaluates software performance when workloads are substantially increased quickly and repeatedly. The workload is beyond normal expectations for short amounts of time.

### **Endurance Testing**

Endurance testing also known as soak testing is an evaluation of how software performs with a normal workload over an extended amount of time. The goal of endurance testing is to check for [system problems such as memory leaks](https://msdn.microsoft.com/en-us/library/ms859408.aspx). (A [memory leak](https://stackify.com/java-memory-leaks-solutions/) occurs when a system fails to release discarded memory. The memory leak can impair system performance or cause it to fail.)

### **Scalability Testing**

Scalability testing is used to determine if software is effectively handling increasing workloads. This can be determined by gradually adding to the user load or data volume while monitoring system performance. Also, the workload may stay at the same level while resources such as CPUs and memory are changed.

### **Volume Testing**

Volume testing determines how efficiently software performs with large projected amounts of data. It is also known as flood testing because the test floods the system with data

1. **Que: What is Error, Defect, Bug and failure?**

**Ans:**

* **Error:** We can say that a mistake made by a programmer during coding is called an error.
* **Defect:** Defects are defined as the deviation of the actual and expected result of system or software application.
* **Bug**: A software bug is an error, flaw or fault in the design, development, or operation of computer software that causes it to produce an incorrect or unexpected result, or to behave in unintended ways.
* **Failure:** When a defect reaches the end customer it is called a Failure.

1. **Que: Difference between Priority and Severity?**

**Ans:**

|  |  |  |
| --- | --- | --- |
| parameters | priority | Severity |
| Definition | Priority is the term that decided how fast we can fix a defect | Severity is a term that denote how severity defect can be affect the functionality of software |
| Parameters | Priority is basically a parameter that we decide the order in which we should fix the defect | Severity is the basically parameter that denote the total impact of a given defect on any software |
| Relation | Priorities relates to scheduling of defect to resolve them in a software | Severity relates to the standard of quality |
| Value | The value of priorities is subjective | The value of severity is objective |
| Change of value | The value of priorities changes from time to time | The value of severity changes continually from time to time |
| Who decided the defect | The product’s manager basically decided the defect’s priority level | The testing engineers basically decides a defect’s severity level |
| Types | There are three types of priority level: high, medium, low | There are five types of severities: cosmetic, minor, moderate, major and critical |

1. **Que: What is Bug Life Cycle?**

**Ans:** The developer first identifies the bug, then moves to the tester for testing, and the tester marks the stages based on the priority of the bug that needs to be fixed.

1. **Que: Explain the difference between Functional testing and Non-functional testing.**

**Ans:**

|  |  |
| --- | --- |
| **Functional testing** | **Non- functional testing** |
| Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements. | Non- functional testing checks the performance, reliability, scalability and other non-functional aspects of the software system |
| Functional testing is executed first | Non functional testing should be performed after functional testing |
| Manual testing or automation tools can be used for functional testing | Using tools will be effective for this testing |
| Business requirements are the inputs to functional testing | Performance parameters like speed, scalability are inputs to non-functional testing |
| Functional testing describe the what the product does | Non functional testing describe how good the product works |
| Easy to do manual testing | Tough to do manual testing |
| Types of functional testing are   * Unit testing * Smoke testing * Sanity testing * Integration testing * White box testing * User acceptance testing * Regression testing | Types of nonfunctional testing   * Performance testing * Load testing * Volume testing * Stress testing * Security testing * Installation testing * Penetrated testing * Compatibility testing * Migration testing |

1. **To create HLR & Testcase of (Instagram and Facebook) only first login page.**

**Ans:**

* Sheet 1 and 2 (Instagram HLR and Testcase)
* Sheet 3 and 4 (Facebook HLR and Testcase)

1. **Que: What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?**

**Ans:**

|  |  |  |
| --- | --- | --- |
| **SN** | **SDLC** | **STLC** |
| **1** | SDLC is mainly related to software development. | STLC is mainly related to software testing. |
| **2** | Besides development other phases like testing is also included. | It focuses only on testing the software. |
| **3** | SDLC involves total six phases or steps. | STLC involves only five phases or steps. |
| **4** | In SDLC, more number of members (developers) are required for the whole process. | In STLC, less number of members (testers) are needed. |
| **5** | in SDLC, development team makes the plans and designs based on the requirements. | In STLC, testing team (Test Lead or Test Architect) makes the plans and designs. |
| **6** | Goal of SDLC is to complete successful development of software. | Goal of STLC is to complete successful testing of software. |
| **7** | It helps in developing good quality software. | It helps in making the software defects free. |
| **8** | SDLC phases are completed before the STLC phases. | STLC phases are performed after SDLC phases. |
| **9** | Post deployment support, enhancement, and update are to be included if necessary. | Regression tests are run by QA team to check deployed maintenance code and maintains test cases and automated scripts. |
| **10** | Creation of reusable software systems is the end result of SDLC. | A tested software system is the end result of STLC. |

1. **Que: What is the difference between test scenarios, test cases, and test script?**

**Ans:**

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Test scenarios** | **Test cases** | **Test script** |
| **1** | The test scenarios are those derived from the **use case** and give **the one-line information about what to test**. | Test case is the step-by-step procedure to test any functionality of the software application/product | Test script is the instruction or a short programme to test any functionality of software application/product |
| **2** | Test scenario are one liner statements, but it is connected with several test cases | The case is a manual approach of a software testing | Test Script is an automatic approach of software testing. |
| **3** | writing the test scenario main objective is a cover end to end functionality of a software application | It is a set up that is used by the tester to test any specific function of the software product. | It is a program developed by the tester, intended to test any specific function of the software product. |
| **5** | These are **high-level actions.** | Test Cases are written manually. | Test Scripting is done by scripting format. |
| **6** | The test scenario is work on the basic to what to be tested | Test case is developed in form of templates. | Test script is developed in form of scripting. |
| **7** | The test scenario will help us in an agile way of testing throughout the functionality. | If the tester does not have a good understanding of how the program is used or about the recent risks to the program, then it will be difficult to use the test cases properly. | Active software projects frequently change. So testers have to make a continuous effort to update the scripts to match the changes of the new product. |
| **8** | Fewer resources are sufficient to write test scenarios as compared to the test cases. | Test Case is used in manual testing environment. | Test Script is used in automatic testing environment. |
| **9** | It can obtain from the **use cases**. | Test Cases are classified as delegated, positive, reusable, negative and UI test cases. | Test Script are characterized as manual test script and automation test scripts. |
| **10** | The test scenarios are a time saving process; that's why it is preferred by the new generation of software testing community. | Test Case comprises of many test qualities like conditions, test data, environment, test suite id, name and others. | In Test Script different commands can be used to develop scripts which is used as a part of performing repeatable and regression testing. |
| **11** | It takes **less time** as compared to test cases. | Requires more resources and time. | Requires less time for testing scripts. |

1. **Que: Explain what Test Plan is? What is the information that should be covered.**

**Ans:** Test planning in STLC is a phase in which a senior QA manager determines the test plan strategy along with efforts and cost estimates for the project.

* More over the resources test environment test limitations and the testing schedule are also determined.
* The test plan gets prepared and finalized in the same phase.

**Activities in requirement phase testing**

* Preparation of the test plan/strategy document for various types of testing
* Test tool selection
* Test effort estimation
* Resource planning and determining roles and responsibilities.
* Training requirement phase testing
* Test plan/strategy document
* Effort estimation documents

1. **Que: What is priority?**

**Ans:** One can define Priority as a parameter for deciding the order in which one can fix the defect.

1. **Que: Bug categories are….**

**Ans:**

* Functional Bugs.
* Logical Bugs.
* Workflow Bugs.
* Unit Level Bugs.
* System-Level Integration Bugs.
* Out of Bound Bugs.
* Security Bugs.

1. **Que: Advantage of Bugzilla?**

**Ans:**

* It improves the quality of the product.
* It enhances the communication between the developing team and the testing team.
* It has the capability to adapt to multiple situations.

1. **Que: What are the different Methodologies in Agile Development Model?**

**Ans:**

* [Kanban](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Kanban)
* [Scrum](https://u-next.com/blogs/product-management/types-of-agile-methodology/#scrum)
* [Extreme Programming (XP)](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Extreme-Programming)
* [Crystal](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Crystal)
* [Dynamic Systems Development Method (DSDM)](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Dynamic-Systems-Development-Method)
* [**Feature Driven Development (FDD)**](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Feature-Driven-Development)
* [**Lean Software Development**](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Lean-Software-Development)
* [**Scaled Agile Framework (SAFe)**](https://u-next.com/blogs/product-management/types-of-agile-methodology/#Scaled-Agile-Framework-(SAFe))

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

**Ans:**

|  |  |  |
| --- | --- | --- |
| **SN** | **Authorization** | **Authentication** |
| **1** | In the [authentication](https://www.geeksforgeeks.org/authentication-in-computer-network/) process, the identity of users is checked for providing the access to the system. | While in [authorization](https://www.geeksforgeeks.org/what-is-aaa-authentication-authorization-and-accounting/) process, a the person’s or user’s authorities are checked for accessing the resources. |
| **2** | In the authentication process, users or persons are verified. | While in this process, users or persons are validated. |
| **3** | It is done before the authorization process. | While this process is done after the authentication process |
| **4** | It needs usually the user’s login details. | While it needs the user’s privilege or security levels. |
| **5** | Authentication determines whether the person is user or not. | While it determines **What permission does the user have?** |
| **6** | Generally, transmit information through an ID Token. | Generally, transmit information through an Access Token. |
| **7** | The OpenID Connect (OIDC) protocol is an authentication protocol that is generally in charge of user authentication process. | The OAuth 2.0 protocol governs the overall system of user authorization process. |
| **8** | Popular Authentication Techniques-   * Password-Based Authentication * Password less Authentication * 2FA/MFA (Two-Factor Authentication / Multi-Factor Authentication) * [Single sign-on (SSO)](https://www.geeksforgeeks.org/introduction-of-single-sign-on-sso/) * Social authentication | Popular Authorization Techniques-   * Role-Based Access Controls (RBAC) * [JSON web token (JWT) Authorization](https://www.geeksforgeeks.org/json-web-token-jwt/) * SAML Authorization * OpenID Authorization * OAuth 2.0 Authorization |
| **9** | The authentication credentials can be changed in part as and when required by the user. | The authorization permissions cannot be changed by user as these are granted by the owner of the system and only, he/she has the access to change it. |
| **10** | The user authentication is visible at user end. | The user authorization is not visible at the user end. |
| **11** | The user authentication is identified with username, password, face recognition, retina scan, fingerprints, etc. | The user authorization is carried out through the access rights to resources by using roles that have been pre-defined. |
| **12** | **Example**: Employees in a company are required to authenticate through the network before accessing their company email. | **Example:** After an employee successfully authenticates, the system determines what information the employees are allowed to access. |

**Below are five web application testing challenges faced by web developers during the development process.**

* Integration. Integration testing exposes problems with interfaces among different program components before deployment. ...
* Interoperability. ...
* Security. ...
* Performance. ...
* Usability. ...
* Quality Testing, Exceptional Services.

1. **Que: To create HLR & Testcase of Web Based (WhatsApp web, Instagram)**

**Ans:**

* Sheet 5 and 6 (Whatsapp Web HLR and Testcase)
* Sheet 7 and 8 (Instagram Web HLR and Testcase)

1. **Que: To create HLR and Testcase on this Link.** [**https://artoftesting.com/**](https://artoftesting.com/)

**Ans:**

* Sheet no. 22 (HLR)
* Sheet no. 23 (Testcase)

1. **Que: Write a scenario of only WhatsApp chat messages**

**Ans:** Sheet no. 9

1. **Que: Write a Scenario of Pen.**

**Ans:** Sheet no. 10

1. **Que: Write a Scenario of Pen Stand.**

**Ans:** Sheet no. 11

1. **Que: Write a Scenario of Door.**

**Ans:** Sheet no. 12

1. **Que: Write a Scenario of ATM.**

**Ans:** Sheet no. 13

1. **Que: to used Usability Testing?**

**Ans:** Usability testing is a method of testing the functionality of a website, app, or other digital product by observing real users as they attempt to complete tasks on it.

1. **Que: What is the procedure for GUI Testing?**

**Ans:**

* Check Screen Validations.
* Verify All Navigations.
* Check usability Conditions.
* Verify Data Integrity.
* Verify the object states.
* Verify the date Field and Numeric Field Formats.

1. **Que: Write a scenario of Microwave Owen.**

**Ans:** sheet no. 14

1. **Que: Write a scenario of Coffee vending Machine.**

**Ans:** Sheet no. 15

1. **Que: Write a scenario of chair.**

**Ans:** Sheet no. 16

1. **Que: Online shopping to buy product (flipkart).**

**Ans:** Sheet no. 17

1. **Que: Write a Scenario of Wrist Watch.**

**Ans:** Sheet no. 18

1. **Que: Write a Scenario of Lift (Elevator).**

**Ans:** Sheet no. 19

1. **Que: Write a Scenario of WhatsApp Group (generate group).**

**Ans:** Sheet no. 20

1. **Que: Write a Scenario of Whatsapp payment.**

**Ans**: Sheet no. 21