1. What is Exploratory Testing?

- This may be the only type of technique used for low-risk systems, but this approach may be particularly useful under extreme time pressure – in fact this is one of the factors leading to exploratory testing.

2. 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.
- A software process should help you keeping the virtual table up-to-date.

3. What is Boundary value testing?

- Boundary value analysis is a methodology for designing test cases that concentrates software testing effort on cases near the limits of valid ranges Boundary value analysis is a method which refines equivalence partitioning.
- Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
- The trick is to concentrate software testing efforts at the extreme ends of the equivalence classes.

4. What is Equivalence partitioning testing?

- The techniques of equivalence partitioning and boundary value analysis are often applied to specific situations or inputs.
- However, if different combinations of inputs result in different actions being taken, this can be more difficult to show using equivalence partitioning and boundary value analysis, which tend to be more focused on the user interface.
- The other two specification-based software testing techniques, decision tables and state transition testing are more focused on business logic or business rules.
- A decision table is a good way to deal with combinations of things (e.g. inputs).

5. What is Integration testing?

- System Integration Testing is testing between the 'System' and 'Acceptance' phases.
- The System has already proven to be functionally correct, what remains to be tested is how the system reacts to other systems and/or organisations.

6. What determines the level of risk?

- A properly designed test that passes, reduces the overall level of Risk in a system.
- Risk – 'A factor that could result in future negative consequences; usually expressed as impact and likelihood'.
- When testing does find defects, the Quality of the software system increases when those defects are fixed.

7. What is Alpha testing?

- Alpha Testing is definitely performed and carried out at the developing organizations location with the involvement of developers.
- It comes under the category of both White Box Testing and Black Box Testing
- It is always performed within the organization. It is the form of Acceptance Testing.

8. What is beta testing?

- It is always performed by the customers at their own site.
- It is not performed by Independent Testing Team.
- Beta Testing is always open to the market and public.
- It is usually conducted for software product.
- It is performed in Real Time Environment.
- It is always performed outside the organization.

9. What is component testing?

- Component(Unit) – A minimal software item that can be tested in isolation. It means "A unit is the smallest testable part of software."
- Component Testing – The testing of individual software components.
- Unit Testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

10. What is functional system testing?

- Functional System Testing is a requirement that specifies a function that a system or system component must perform.
- A Requirement may exist as a text document and/or a model

11. What is 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

12. What is GUI Testing?

- Graphical User Interface (GUI) testing is the process of testing the system's GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.
- WHAT DO YOU CHECK IN GUI TESTING?
    - Check all the GUI elements for size, position, width, length and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
    - Check Font used in application is readable
    - Check the alignment of the text is proper
    - Check the Color of the font and warning messages is aesthetically pleasing
    - Check that the images have good clarity

13. What is Adhoc testing?

- Adhoc testing is an informal testing type with an aim to break the system.
- It does not follow any test design techniques to create test cases.
- In fact is does not create test cases altogether!
- This testing is primarily performed if the knowledge of testers in the system under test is very high.

14. What is load testing?

- Load Testing is a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system's response time degrades or fails.

- Load testing is a kind of performance testing which determines a system's performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.

15. What is stress Testing?

- System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.
- Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.

16. What is white box testing and list the types of white box testing?

- White Box Testing: Testing based on an analysis of the internal structure of the component or system.
- Structure-based testing technique is also known as 'whitebox' or 'glass-box' testing technique because here the testers require knowledge of how the software is implemented, how it works.

17. What is black box testing? What are the different black box testing techniques?

- Black-box testing: Testing, either functional or nonfunctional, without reference to the internal structure of the component or system.
- There are four specification-based or black-box technique:
    - Equivalence partitioning
    - Boundary value analysis
    - Decision tables
    - State transition testing
    - Use-case Testing
    - Other Black Box Testing

18. Mention what are the categories of defects?

- Defect is the variance from a desired product attribute (it can be a wrong, missing or extra data).
- It can be of two types –
- Defect from the product or a variance from customer/user expectations.
- It is a flaw in the software system and has no impact until it affects the user/customer and operational system.

19. Mention what bigbang testing is?

- In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole.
- Big Bang testing has the advantage that everything is finished before integration testing starts.

20. What is the purpose of exit criteria?

- Successful Testing of Integrated Application.
- Executed Test Cases are documented
- All High prioritized bugs fixed and closed

- Technical documents to be submitted followed by release Notes.

21. When should "Regression Testing" be performed?

- Regression Testing: Testing of a previously tested program following modification to ensure that defects have not been introduced or uncovered in unchanged areas of the software, as a result of the changes made. It is performed when the software or its environment is changed.

22. What is 7 key principles? Explain in detail?

- General Testing Principles
    1. Testing shows presence of Defects
    2. Exhaustive Testing is Impossible!
    3. Early Testing Defect Clustering
    4. The Pesticide Paradox
    5. Testing is Context Dependent
    6. Absence of Errors Fallacy

1. Testing shows presence of Defects

- Software testing reduces the prersence of defects.
- Software testing talks about the presence of defects and doesn't talk about the absence of
- Even multiple testing can never ensure that software is 100% Bug free

2. Exhaustive Testing is Impossible!

- Exhaustive testing is impossible mean the software can never test at every test cases.
- It can test only some test cases and assume that software is correct and it will produce the correct output in every cases

3. Early Testing

- The defects detected in early phases of SDLC will very less Expensive.
- For better performance of software, start software testing will start at initial phase.

4. Defect Clustering

- In a project a small number of the module can contain most of the defects
- Pareto principle to software testing state that 80% software defects comes from 20% of modules

5. The Pesticide Paradox

- Repeating the same test cases again and again will not find new bugs.
- It is Necessary to review the test cases and add or update test cases to find new bugs.

6. Testing is Context Dependent

- Testing approach depends on context of software developed.
- Different types of software need to perform different types of testing.
- The testing of the e-commerce site is different form the testing of the android application.

7. Absence of Errors Fallacy

- If a built software is 99% bug-free but it does not follow the user requirement then it is unusable.
- It is not only necessary that software is 99% bug-free but it also mandatory to fulfil all the customer requirements.

23. Difference between QA v/s QC v/s Tester

SN	Quality Assurance	Quality Control	Testing
1.	Activities which ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	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 a corrective process.	It is a preventive process.
5.	It is a subset of Software Test Life Cycle (STLC).	QC can be considered as the subset of Quality Assurance.	Testing is the subset of Quality Control.

24. Difference between Smoke and Sanity?

➢

SN	Smoke Testing	Sanity Testing
1.	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
2.	The objective of this testing is to verify "stability" of the system in order to with more rigorous testing	The objective of the testing is to verify the the "rationality" of the system in order proceed to proceed with more rigorous testing
3.	This testing is performed by the developers or testers	Sanity testing is usually performed by testers
4.	Smoke testing is usually documented or scripted	Sanity testing is usually not documented and is unscripted
5.	Smoke testing is a subset of Regression testing	Sanity testing is a subset of Acceptance testing
6.	Smoke testing is like General Health Check Up	Sanity Testing is like specialized health Check Up.
7.	Smoke testing exercises the entire system from end to end.	Sanity testing exercises only the particular component of the entire system

25. Difference between verification and Validation

➢

Criteria	Verification	Validation
Definition	The process of evaluating work-products (not the actual final product) of a development phase to determine whether they meet the specified requirements for that phase.	The process of evaluating software during or at the end of 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 correct in the first place. In other words, to demonstrate that the product fulfills its intended use when placed in its intended environment.
Question	Are we building the product right?	Are we building the right product?
Evaluation Items	Plans, Requirement Specs, Design Specs, Code, Test Cases	The actual product/software.
Activities	▪ Reviews ▪ Walkthroughs ▪ Inspections	▪ Testing

26. Explain types of Performance testing.

➢ Software performance testing is a means of quality assurance (QA). It involves testing software applications to ensure they will perform well under their expected workload.

➢ Types of Performance Testing
    - Load testing
    - Stress testing
    - Endurance testing
    - Spike testing
    - Volume testing
    - Scalability testing

▪ Load testing

✓ It checks the product's ability to perform under anticipated user loads. The Objective is to identify performance Congestion before the software product is launched in the market.

▪ Stress testing

✓ It involve testing a product under extreme workloads to see whether it handles high traffic or not. The Objective is to identify the breaking point of a software product.

27. What is Error, Defect, Bug and failure?

➢ A mistake in coding is called error.

➢ Error found by tester is called defect.

➢ Defect accepted by development team then it is called bug.

➢ Build does not meet the requirements then it is failure.

28. Explain the difference between Functional testing and Non-Functional testing

SN	Functional Testing	Non-Functional Testing
1.	Functional testing is performed using the functional specification provided by	Non-Functional testing checks the Performance, reliability, scalability and

	the client and verifies the system against the functional requirements.	other non-functional aspects of the software system.
2.	Functional testing is executed first	Non functional testing should be performed after functional testing
3.	Manual testing or automation tools can be used for functional testing	Using tools will be effective for this testing
4.	Business requirements are the inputs to functional testing	Performance parameters like speed , scalability are inputs to non-functional testing.
5.	Functional testing describes what the product does	Nonfunctional testing describes how good the product works
6.	Easy to do manual testing	Tough to do manual testing
7.	Types of Functional testing are ▪ Unit Testing ▪ Smoke Testing ▪ Sanity Testing ▪ Integration Testing ▪ White box testing ▪ Black Box testing ▪ User Acceptance testing ▪ Regression Testing	Types of Nonfunctional testing are ▪ Performance Testing ▪ Load Testing ▪ Volume Testing ▪ Stress Testing ▪ Security Testing ▪ Installation Testing ▪ Penetration Testing ▪ Compatibility Testing ▪ Migration Testing

29. To create HLR & TestCase of
    1. Instagram
    2. Facebook

- Instagram File
- Facebook.com

30. What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?

-

Parameter	SDLC	STLC
Origin	Development life cycle	Testing life cycle
Objective	The main project of SDLC life cycle is to complete successful development of the software including testing and other phases.	The only objective of the STLC Phase is testing
Requirement Gathering	In SDLC the business analyst gathers the requirements and create Development Plan	In STLC, the QA team analyze requirement documents like functional and non-functional documents and create System test Plan
High & Low Level Design	In SDLC, the development team create the high and low design plans	In STLC, the test analyst creates the integration test plan
Coding	The real code is developed and actual work takes place as per the design documents	The testing team prepares the test environment and executes them

Maintaince	SDLC Phase also includes post-deployment support and updates.	Testers, execute regression suits, usually automation script to check maintaince code deployed.

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

	Test scenario	Test Cases	Test Script
1.	The test scenario is just a document that is detailed and provides details about the assessment method, testing process, precondition, and anticipated output.	Test cases is a step by step procedure to test any functionality of the software application/prod uct.	Test script is set of instruction or a short program to test any functionality of software application/prod uct.
2.	The test scenarios are the ones based on the use situation and give one-line information one what to check.	Test cases is a manual approach of software testing.	Test script is an automatic approach of software testing.
3.	Test scenarios are one-liner statement, however, it is linked to a few test instances.	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.
4.	These are highlevel actions.	Point by point test case configuration encourages tester to test viably.	Automatic testing approach is beneficial for constant execution.
5.	Writing the test scenario's primary objective is an address end to get rid of functionality of a software program.	Test cases are written by manually.	Test scripting is done by scripting format.
6.	It will take less time as compared to test cases.	Test case is developed in form of templates.	Test script is developed in form of scripting.
7.	The test scenario will help us in a way that is nimble of through 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.	Test scenario are really easy to maintain due to their highlevel design.	Test case is used in manual testing environment.	Test script is used in automatic testing environment.
9.	The test scenarios tend to be work on the essential to "things to be tested".	Test cases are classified as delegated, positive, reusable, negative and UI test cases.	Test script are characterized as manual test script and automatic test scripts.

10. Explain what Test Plan is? What is the information that should be covered.

➢ 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.

- Moreover, the resources, test environment, test limitations and the testing schedule are also determined.
- The Test Plan gets prepared and finalized in the same phase.

11. What are the different Methodologies in Agile Development Model?

- Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes (small time frames) to deliver specific features for a release.
- The Agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle through a process of planning, executing, and evaluating.

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

-

Sr.No	Authentication	Authorization
1.	In the Authentication Process the identify of users are checked for providing the access to the system	In Authorization Process a the person or user authorities are checked for accessing the resources.
2.	Users or persons are verified.	Users or persons are validated
3.	It is done before the authorization process.	It is done after the authentication process.
4.	It need usually the user login details.	It need the user priviledge or security levels
5.	It determines whether the person is user or not.	It determines what permission does the user have?
6.	Generally, transmit information through an ID Token.	Generally, transmit information through an Access Token.

13. To create HLR & TestCase of WebBased (WhatsApp web , Instagram)
    1. WhatsApp Web : https://web.whatsapp.com/
    2. Instagram Web:- www.instagram.com/accounts/login/
    3. Art of Testing:- artoftesting.com

- Whatsapp Web:- Whatsapp File
- Instagram Web:- Instagram File
- Art of testing:- artoftesting File

14. Write a scenario of only Whatsapp chat messages
15. Write a Scenario of Pen
16. Write a Scenario of Pen Stand
17. Write a Scenario of Door
18. Write a Scenario of ATM
19. When to used Usablity Testing?
20. What is the procedure for GUI Testing?
21. Write a scenario of Microwave Owen
22. Write a scenario of Coffee vending Machine
23. Write a scenario of chair
24. To Create Scenario (Positive & Negative)
    1. Gmail Receiving Mail
    2. Online shopping to buy product (flipkart)

25. Write a Scenario of Wrist Watch
26. Write a Scenario of Lift(Elevator)
27. Write a Scenario of whatsapp Group (generate group)
28. Write a Scenario of Whatsapp payment