Q:1:- Testing planning and documentation:
- Learning how to create test plans,test cases,and test scripts?

Ans:- Test cases serve the purpose of confirming that an application or software system functions as intended. Test case authors develop these cases to enable testers to assess the correct operation of features within the app or system. It is crucial to conduct thorough testing of applications in order to observe their behaviour under various input scenarios. Having a comprehensive understanding of software functions and the testing process facilitates the creation of test cases that effectively identify defects. You can refer to the provided guide for detailed insights on crafting and formatting different types of test cases.

What is a QA Test Case?

Test cases provide guidelines for testers to verify the correct functioning of programs. They outline the expected behaviour of the software under normal, abnormal, or error conditions. The process of writing test cases involves translating user requirements into a series of test conditions and descriptions that assess the system's performance. A test suite in an automated test script can encompass multiple test cases combined together.

Test Case vs Test Scenario
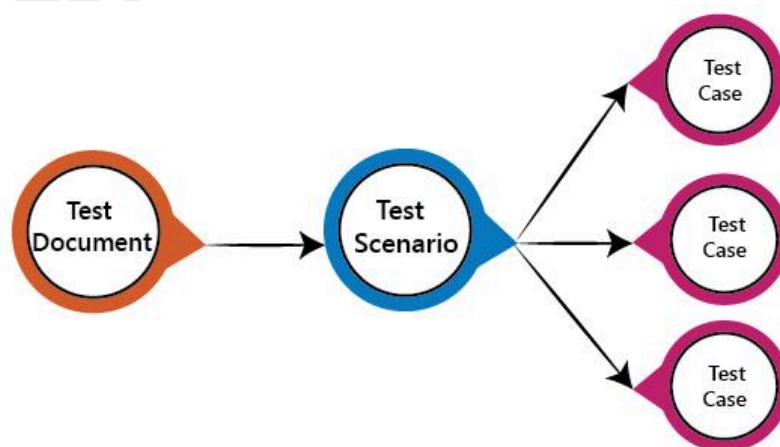
When formulating a test case, it is customary to consider every aspect of the expected behaviour. For instance, when developing a login system, a

test case might entail verifying that an error message is displayed when an incorrect email address is entered. Various tests can be designed to cover different scenarios, such as:

1. Omitting the email address altogether

2. Appending a space at the end of the email address

3. Using all uppercase letters for the email address

4. Capitalising only the first letter of the email address

Test scenarios, also referred to as test conditions or possibilities, encompass the typical tasks users may intend to accomplish using the software. A single test scenario may require multiple test cases to encompass the various potential outcomes. In the case of our login system, some scenarios could include "Successful login" or "Failed login due to incorrect email address entry."

Types of Test Cases
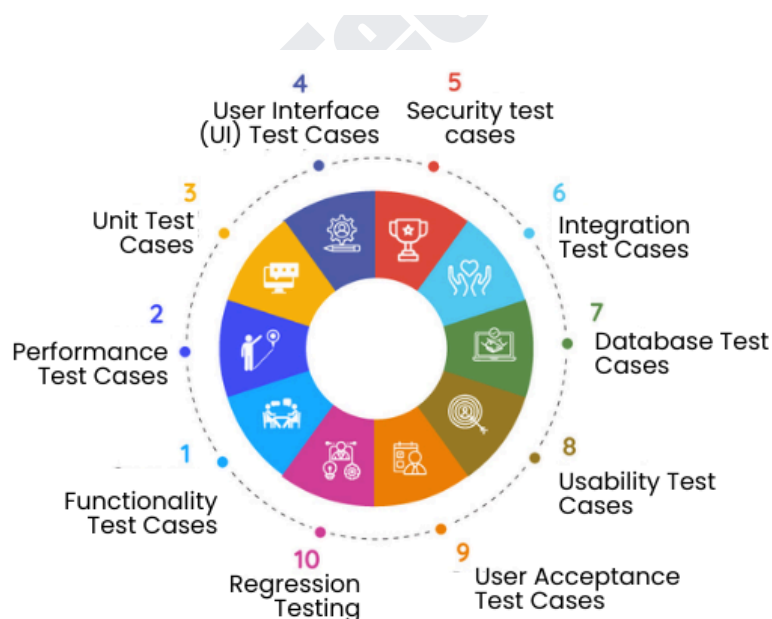
Test cases can be classified based on their purpose in the testing process, which is valuable for quality assurance professionals in focusing their efforts and selecting the appropriate test format.

1. Functionality Test Cases

These are fundamental and straightforward test cases that ensure the proper functioning of each feature in the system.

2. Performance Test Cases

These tests verify that the system operates at an acceptable speed and meets requirements related to scalability and stability.

3. Unit Test Cases

Typically written by software developers, unit tests evaluate individual units, such as modules, procedures, or functions, to ensure they perform as expected.

4. User interface (UI) Test Cases

UI test cases assess the correct functioning, appearance, and usability of each UI element, recognizing that the user interface is an integral part of the overall system.

5. Security Test Cases

These test cases validate that a product or system operates reliably under various conditions, including attempts by malicious users to gain unauthorised access or compromise the system. They prioritise the security, privacy, and confidentiality of data.

6. Integration Test Cases

Integration test cases confirm that the application components integrate seamlessly and function as intended when combined, forming a cohesive product.

7. Database Test Cases

These test cases ensure that the database meets both functional and non-functional requirements, verifying that the database management system (DBMS) adequately supports business needs.

8. Usability Test Cases

Usability test cases evaluate whether users can successfully interact with the application, assessing ease of use, intuitive navigation, and absence of confusion or difficulty in common procedures and functions.

9. User Acceptance Test Cases

User acceptance test cases determine whether an application meets the business requirements and receives user approval before its release into the live environment.

10.   Regression Testing

Regression test cases verify that modifications made during development do not cause any existing functionality to cease functioning properly. These tests are performed after implementing changes to ensure that all pre-existing or legacy features continue to operate as intended.

Who writes Test Cases?

The task of writing test cases is generally assigned to individuals within the software development, testing, or quality assurance teams. It is generally recommended to have someone who did not participate in writing the code to be responsible for writing the test cases, as they can offer a fresh perspective. The approach to QA testing may differ depending on the software development framework employed, such as Agile or Scrum.

How to write Test Cases: A step-by-step guide

Test cases serve as the guidelines for testers to follow, making it crucial for them to be clear, comprehensive, and accurate. Whether you are creating

new test cases or assessing existing ones, the following steps can be
helpful:
> Determine the specific scope you wish to address based on the test
  scenario.
> Ensure that the test case is easily understandable and executable by
  testers.
> Familiarise yourself with and apply relevant test designs that align
  with the testing requirements.
> Assign a unique test case ID to each case for proper identification.
> Use a requirements traceability matrix during testing to maintain
  visibility and track progress.
> Provide a clear and concise description within each test case.
> Include appropriate preconditions and postconditions to set the
  necessary context for the test.
> Clearly specify the expected result or outcome of each test case.
> Employ appropriate testing techniques suitable for the specific
  context.
> Seek a peer review of your test plan before proceeding to ensure its
 accuracy and effectiveness.

Best Practices for well-written Test Cases
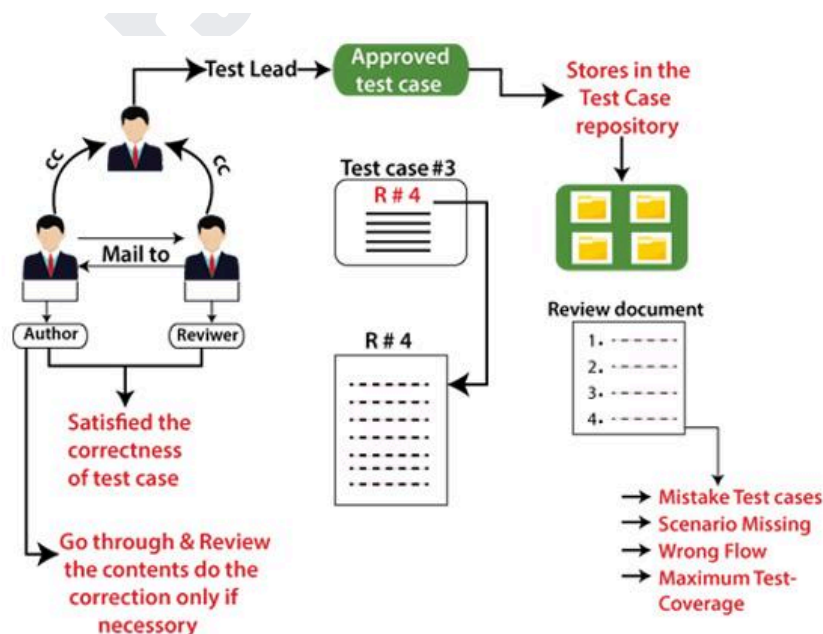
A test case template offers a versatile yet fundamental framework that can be tailored as per requirements. It can also function as a checklist to ensure all crucial elements are incorporated. Testers often utilise spreadsheets where each test is represented in a row and the test elements are organised in columns. Here are several elements that can be included in your test case template:

1. Test Case ID
2. Test Case Description
3. Pre-Conditions
4. Test Steps
5. Test Data
6. Expected Result
7. Post Conditions
8. Actual Result
9. Status

Benefits of Writing High-Quality Test Cases
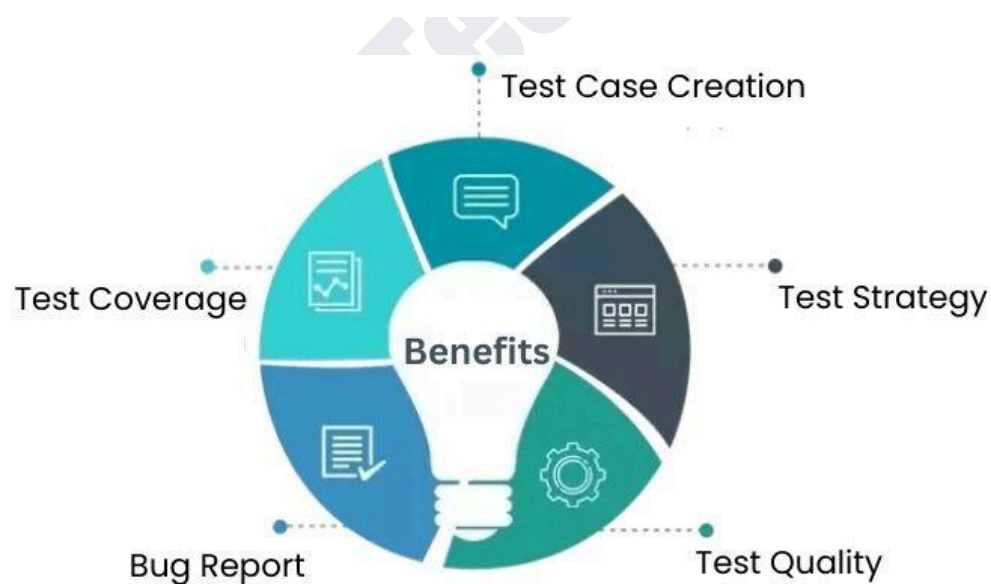
The process of writing test cases enables a thorough examination of all aspects of the software, facilitating the identification of any gaps that may exist during its development. There are several advantages associated with creating formal test cases:

➢ Documentation ensures comprehensive test coverage, allowing you to verify that all areas of the software have been adequately tested.

➢ By having well-defined test cases, you can minimise software maintenance, bug fixes, and future support costs.

➢ Test cases can be reused in both current and future projects, saving time and effort in test case creation.

➢ The quality of the software and the user experience can be enhanced through the implementation of effective test cases.

➢ A higher quality product results in greater customer satisfaction, ultimately leading to increased profits.