ONE PAGE SUMMARY

IT & Software › Software Testing › Testing Fundamentals

Introduction to Testing Fundamentals

Master the Basics of Software Testing Fundamentals!

DEFINITION

> Software testing fundamentals involve the processes and principles used to evaluate software quality. It ensures that software meets requirements and functions correctly before release.

KEY CONCEPTS

Testing Levels

Different stages of testing include unit, integration, system, and acceptance testing, each focusing on specific aspects of software.

Testing Types

Types of testing such as manual, automated, functional, and non-functional help identify various defects and ensure quality.

Test Cases

Test cases are specific conditions under which a tester checks whether a software application behaves as expected.

Defect Lifecycle

The defect lifecycle outlines the stages a defect goes through from identification to resolution and closure.

Test Planning

Test planning involves defining the scope, approach, resources, and schedule for testing activities.

EXAMPLES

→ Creating test cases for a login feature to check valid and invalid credentials.

→ Running automated tests to verify performance under high load conditions.

→ Conducting user acceptance testing with real users to gather feedback.

MEMORY TIPS

★ Remember 'T-L-T-D' for Testing Levels: Unit, Integration, System, Acceptance.

★ Use 'F-M-NF' to recall Testing Types: Functional, Manual, Non-Functional.

★ Think of 'D-C-R' for Defect Lifecycle: Detection, Correction, Resolution.

COMMON MISTAKES

✗ Neglecting to document test cases properly, leading to incomplete testing.

✗ Overlooking edge cases that may cause software failures.

✗ Failing to involve stakeholders in the testing process for feedback.

QUICK RECAP

Software testing fundamentals are essential for ensuring software quality and functionality. Key concepts include testing levels, types, and the defect lifecycle. Understanding these principles helps in creating effective test plans and cases.