

Software Testing Foundation: A Comprehensive Hands-On Introduction - 4 Days

Preparing for the ISTQB Certified Tester Exam

Course 316 Overview

You Will Learn How To

- Apply general software testing principles and fundamental test processes
- Implement test levels and types to various software development models
- Conduct static techniques using proper roles, responsibilities and tools
- Perform specification- and structure-based test design techniques
- Manage tests including planning, estimating, monitoring and controlling
- Prepare for the ISTQB Certified Tester Foundation Level Exam

Course Benefits

The proper testing of software can save an organization time, effort and money. In this course, software professionals and managers gain thorough knowledge of testing approaches that can be integrated into the software life cycle. Through hands-on exercises, you learn how to build testing methods into your work process to correctly design products that are functionable and maintainable.

Who Should Attend

Software testers, programmers, test leaders, quality specialists and those who would like to earn the ISTQB Certified Tester Foundation Level certification. A familiarity with software development concepts is assumed.

Hands-On Training

Hands-on exercises provide you with practical experience in software testing, including:

- Recognizing the value of a defined test process
- Deciphering when to apply static and dynamic techniques
- Recognizing equivalence partitions
- Performing boundary value analysis
- Conducting state transition testing
- Ensuring statement, decision and condition coverage
- Organizing test development processes
- Creating a test policy and writing a test plan

Software Testing Foundation: A Comprehensive Hands-On Introduction - 4 Days

Preparing for the ISTQB Certified Tester Exam

Course 316 Outline

Fundamentals of Software Testing

- Assessing the goals of testing
- Identifying causes of software defects
- Organizing testing processes
- Planning
- Controlling
- Analyzing
- Designing
- Implementing and executing
- Evaluating exit criteria and reporting

Ensuring Testing throughout the Software Life Cycle

Key objectives of testing

- Finding defects during the life cycle
- Implementing test levels
- Component
- Integration
- System
- Acceptance

Recognizing key concepts in maintenance testing

- Identifying reasons for maintenance testing
- Performing maintenance testing

Comparing the four test types

- Functional
- Nonfunctional
- Structural
- Retesting

Coping with the psychology of testing

- Contrasting the mind-set of developers and testers
- Deciphering levels of independence

Implementing Static Analysis Techniques

Determining when to apply each technique

- Defining roles and responsibilities
- Comparing formal and informal reviews
- Discussing the types of review
- Walkthrough
- Inspection
- Technical

Leveraging Test-Design Techniques

Differentiating various "specifications"

- Test design
- Test case
- Test procedure

Applying specification-based techniques

- Equivalence partitioning
- State transition
- Boundary value analysis
- Use case
- Decision table

Utilizing structure-based techniques

- Statement
- Decision
- Condition

Deploying experience-based knowledge

- Intuition
- Experience
- Knowledge

Test Management

The importance of a test policy

- Defining goals and objectives
- Assigning roles and responsibilities
- Independence
- Test leader
- Tester
- Standardizing test documentation
- Monitoring and controlling test progress

Structuring a test plan

- Writing a test plan
- Identifying objectives
- Assessing the entry and exit criteria
- Assigning appropriate resources
- Resolving defects

Interpreting a test summary report

- Evaluating summary report content
- Applying common metrics

Managing incidents

- Recording
- Analyzing
- Closing

Addressing project and product risks

- Contractual
- Organizational
- Technical
- Assess
- Determine
- Implement

Implementing Configuration Management (CM)

Defining the functions of CM

- Change control
- Version control
- Traceability
- Configuration identification and audits

Evaluating objectives of CM

- Ensuring proper version control
- Generating incident reports

Adopting Test Support Tools

Classifying different types of test tools

- Test management
- Static testing
- Test specification
- Executing and logging
- Performance and monitoring
- Other

Introducing a tool into an organization

- Recognizing potential benefits and risks
- Considering special circumstances