



CTFL

ISTQB Certified Tester - Foundation Level

[Processexam.com](https://www.processexam.com)

Exam Summary

Syllabus

Questions

Table of Contents

Key to success in CTFL Exam on ISTQB Certified Tester - Foundation Level 2

ISTQB CTFL Certification Details: 2

ISTQB CTFL Exam Syllabus:..... 3

CTFL Sample Questions: 6

Answers to CTFL Exam Questions:..... 8

Key to success in CTFL Exam on ISTQB Certified Tester - Foundation Level

To achieve the professional designation of ISTQB Certified Tester - Foundation Level from the ISTQB, candidates must clear the CTFL Exam with the minimum cut-off score. For those who wish to pass the ISTQB CTFL certification exam with good percentage, please take a look at the following reference document detailing what should be included in ISTQB Tester Foundation Exam preparation.

The ISTQB CTFL Exam Summary, Body of Knowledge (BOK), Sample Question Bank and Practice Exam provide the basis for the real ISTQB Certified Tester - Foundation Level exam. We have designed these resources to help you get ready to take ISTQB Certified Tester - Foundation Level (CTFL) exam. If you have made the decision to become a certified professional, we suggest you take authorized training and prepare with our online premium [ISTQB Tester Foundation Practice Exam](#) to achieve the best result.

ISTQB CTFL Certification Details:

Exam Name	ISTQB Certified Tester - Foundation Level
Exam Code	CTFL
Exam Fee	USD \$200
Exam Duration	60 Minutes
Number of Questions	40
Passing Score	65%
Format	Multiple Choice Questions
Books / Trainings	Trainings
Schedule Exam	Pearson VUE
Sample Questions	ISTQB CTFL Exam Sample Questions and Answers
Practice Exam	ISTQB Certified Tester - Foundation Level Practice Test

ISTQB CTFL Exam Syllabus:

Domain	Details
Fundamentals of Testing	
Why is Testing Necessary?	<ul style="list-style-type: none"> - Describe, with examples, the way in which a defect in software can cause harm to a person, to the environment or to a company - Distinguish between the root cause of a defect and its effects - Give reasons why testing is necessary by giving examples - Describe why testing is part of quality assurance and give examples of how testing contributes to higher quality - Explain and compare the terms error, defect, fault, failure and the corresponding terms mistake and bug, using examples
What is Testing?	<ul style="list-style-type: none"> - Recall the common objectives of testing - Provide examples for the objectives of testing in different phases of the software life cycle - Differentiate testing from debugging
Seven Testing Principles	<ul style="list-style-type: none"> - Explain the seven principles of testing
Fundamental Test Process	<ul style="list-style-type: none"> - Recall the five fundamental test activities and respective tasks from planning to closure
The Psychology of Testing	<ul style="list-style-type: none"> - Recall the psychological factors that influence the success of testing - Contrast the mindset of a tester and of a developer
Testing Throughout the Software Life Cycle	
Software Development Models	<ul style="list-style-type: none"> - Explain the relationship between development, test activities and work products in the development life cycle, by giving examples using project and product types - Recognize the fact that software development models must be adapted to the context of project and product characteristics - Recall characteristics of good testing that are applicable to any life cycle model
Test Levels	<ul style="list-style-type: none"> - Compare the different levels of testing: major objectives, typical objects of testing, typical targets of testing (e.g., functional or structural) and related work products, people who test, types of defects and failures to be identified
Test Types	<ul style="list-style-type: none"> - Compare four software test types (functional, non-functional, structural and change-related) by example - Recognize that functional and structural tests occur at any test level - Identify and describe non-functional test types based on non-functional requirements - Identify and describe test types based on the analysis of a software system's structure or architecture - Describe the purpose of confirmation testing and regression testing
Maintenance Testing	<ul style="list-style-type: none"> - Compare maintenance testing (testing an existing system) to testing a new application with respect to test types, triggers for testing and amount of testing - Recognize indicators for maintenance testing (modification,

	migration and retirement) - Describe the role of regression testing and impact analysis in maintenance
Static Techniques	
Static Techniques and the Test Process	- Recognize software work products that can be examined by the different static techniques - Describe the importance and value of considering static techniques for the assessment of software work products - Explain the difference between static and dynamic techniques, considering objectives, types of defects to be identified, and the role of these techniques within the software life cycle
Review Process	- Recall the activities, roles and responsibilities of a typical formal review - Explain the differences between different types of reviews: informal review, technical review, walkthrough and inspection - Explain the factors for successful performance of reviews
Static Analysis by Tools	- Recall typical defects and errors identified by static analysis and compare them to reviews and dynamic testing - Describe, using examples, the typical benefits of static analysis - List typical code and design defects that may be identified by static analysis tools
Test Design Techniques	
The Test Development Process	- Differentiate between a test design specification, test case specification and test procedure specification - Compare the terms test condition, test case and test procedure - Evaluate the quality of test cases in terms of clear traceability to the requirements and expected results - Translate test cases into a well-structured test procedure specification at a level of detail relevant to the knowledge of the testers
Categories of Test Design Techniques	- Recall reasons that both specification-based (black-box) and structure-based (white-box) test design techniques are useful and list the common techniques for each - Explain the characteristics, commonalities, and differences between specification-based testing, structure-based testing and experience-based testing
Specification-based or Black-box Techniques	- Write test cases from given software models using equivalence partitioning, boundary value analysis, decision tables and state transition diagrams/tables - Explain the main purpose of each of the four testing techniques, what level and type of testing could use the technique, and how coverage may be measured - Explain the concept of use case testing and its benefits
Structure-based or White-box Techniques	- Describe the concept and value of code coverage - Explain the concepts of statement and decision coverage, and give reasons why these concepts can also be used at test levels other than component testing (e.g., on business procedures at system level) - Write test cases from given control flows using statement and

	decision test design techniques - Assess statement and decision coverage for completeness with respect to defined exit criteria.
Experience-based Techniques	- Recall reasons for writing test cases based on intuition, experience and knowledge about common defects - Compare experience-based techniques with specification-based testing techniques
Choosing Test Techniques	- Classify test design techniques according to their fitness to a given context, for the test basis, respective models and software characteristics
Test Management	
Test Organization	- Recognize the importance of independent testing - Explain the benefits and drawbacks of independent testing within an organization - Recognize the different team members to be considered for the creation of a test team - Recall the tasks of typical test leader and tester
Test Planning and Estimation	- Recognize the different levels and objectives of test planning - Summarize the purpose and content of the test plan, test design specification and test procedure documents according to the 'Standard for Software Test Documentation' - Differentiate between conceptually different test approaches, such as analytical, model-based, methodical, process/standard compliant, dynamic/heuristic, consultative and regression-averse - Differentiate between the subject of test planning for a system and scheduling test execution - Write a test execution schedule for a given set of test cases, considering prioritization, and technical and logical dependencies - List test preparation and execution activities that should be considered during test planning - Recall typical factors that influence the effort related to testing - Differentiate between two conceptually different estimation approaches: the metrics-based approach and the expert-based approach - Recognize/justify adequate entry and exit criteria for specific test levels and groups of test cases
Test Progress Monitoring and Control	- Recall common metrics used for monitoring test preparation and execution - Explain and compare test metrics for test reporting and test control (e.g., defects found and fixed, and tests passed and failed) related to purpose and use - Summarize the purpose and content of the test summary report document according to the 'Standard for Software Test Documentation'
Configuration Management	- Summarize how configuration management supports testing

Risk and Testing	<ul style="list-style-type: none"> - Describe a risk as a possible problem that would threaten the achievement of one or more stakeholders' project objectives - Remember that the level of risk is determined by likelihood (of happening) and impact (harm resulting if it does happen) - Distinguish between the project and product risks - Recognize typical product and project risks - Describe, using examples, how risk analysis and risk management may be used for test planning
Incident Management	<ul style="list-style-type: none"> - Recognize the content of an incident report according to the 'Standard for Software Test Documentation' - Write an incident report covering the observation of a failure during testing
Tool Support for Testing	
Types of Test Tools	<ul style="list-style-type: none"> - Classify different types of test tools according to their purpose and to the activities of the fundamental test process and the software life cycle - Explain the term test tool and the purpose of tool support for testing
Effective Use of Tools: Potential Benefits and Risks	<ul style="list-style-type: none"> - Summarize the potential benefits and risks of test automation and tool support for testing - Remember special considerations for test execution tools, static analysis, and test management tools
Introducing a Tool into an Organization	<ul style="list-style-type: none"> - State the main principles of introducing a tool into an organization - State the goals of a proof-of-concept for tool evaluation and a piloting phase for tool implementation - Recognize that factors other than simply acquiring a tool are required for good tool support

CTFL Sample Questions:

01. Which of the following is a project risk?

- a) Skill and staff shortages
- b) Poor software characteristics (e.g. usability)
- c) Failure-prone software delivered
- d) Possible reliability defect (bug)

02. Which of these tasks would you expect to be performed during the Test Analysis and Design phase of the Fundamental Test Process?

- a) Defining test objectives
- b) Reviewing the test basis
- c) Creating test suites from test procedures
- d) Analyzing lessons learned for process improvement

03. As a test manager, you are asked for a test summary report. Concerning test activities, and according to the IEEE 829 Standard, what should be the MOST important information to include in your report?

- a) The number of test cases executed and their results.
- b) An overview of the major testing activities, events and the status with respect to meeting goals
- c) Overall evaluation of each development work item
- d) Training taken by members of the test team to support the test effort

04. Which of the following statements BEST describes non-functional testing?

- a) Non-functional testing is the process of testing an integrated system to verify that it meets specified requirements.
- b) Non-functional testing is the process of testing to determine system compliance with coding standards.
- c) Non-functional testing is testing without reference to the internal structure of a system.
- d) Non-functional testing is testing system characteristics, such as usability, reliability, or maintainability.

05. Which of the following can be categorized as a product risk?

- a) Low quality of requirements, design, code and tests.
- b) Political problems, and delays in especially complex areas in the product.
- c) Error-prone areas, potential harm to the user, poor product characteristics.
- d) Problems in defining the right requirements, potential failure areas in the software or system.

06. You are working on a project that has poor specifications and time pressure. Which of the following test techniques would be the most useful approach to use?

- a) Use Case Testing
- b) Statement Testing
- c) Exploratory Testing
- d) Decision Testing

07. Which of the following BEST describes how tasks are divided between the test manager and the tester?

- a) The test manager plans testing activities and chooses the standards to be followed, while the tester chooses the tools and controls to be used.
- b) The test manager plans, organizes, and controls the testing activities, while the tester specifies and executes tests.
- c) The test manager plans, monitors, and controls the testing activities, while the tester designs tests and decides about the approval of the test object
- d) The test manager plans and organizes the testing, and specifies the test cases, while the tester prioritizes and executes the tests.

08. Which of the following is a characteristic of good testing and applies to any software development life cycle model?

- a) Acceptance testing is always the final test level to be applied.
- b) All test levels are planned and completed for each developed feature.
- c) Testers are first involved when first piece of code can be executed.
- d) For every development activity there is a corresponding testing activity

09. Which of the following statements BEST describes one of the seven key principles of software testing?

- a) By using automated testing it is possible to test everything.
- b) With sufficient effort and tool support, exhaustive testing is feasible for all software.
- c) It is normally impossible to test all input/output combinations for a software system.
- d) The purpose of testing is to demonstrate the absence of defects.

10. Which of the following is NOT a goal of a pilot project for tool evaluation?

- a) To evaluate how the tool fits with existing processes and practices
- b) To determine use, management, storage, and maintenance of the tool and testware
- c) To assess whether the benefits will be achieved at reasonable cost
- d) To reduce the defect rate in the pilot project

Answers to CTFL Exam Questions:

Question: 01 Answer: a	Question: 02 Answer: b	Question: 03 Answer: b	Question: 04 Answer: d	Question: 05 Answer: c
Question: 06 Answer: c	Question: 07 Answer: b	Question: 08 Answer: d	Question: 09 Answer: c	Question: 10 Answer: d

Note: If you find any typo or data entry error in these sample questions, we request you to update us by commenting on this page or write an email on feedback@processexam.com