

ISTQB Certified Tester Foundation Level (CTFL) PRO Practice Test (Sample)

Study Guide



Everything you need from our exam experts!

Copyright © 2026 by Examzify - A Kaluba Technologies Inc. product.

ALL RIGHTS RESERVED.

No part of this book may be reproduced or transferred in any form or by any means, graphic, electronic, or mechanical, including photocopying, recording, web distribution, taping, or by any information storage retrieval system, without the written permission of the author.

Notice: Examzify makes every reasonable effort to obtain accurate, complete, and timely information about this product from reliable sources.

SAMPLE

Table of Contents

Copyright	1
Table of Contents	2
Introduction	3
How to Use This Guide	4
Questions	5
Answers	9
Explanations	11
Next Steps	17

SAMPLE

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 - 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

Questions

SAMPLE

- 1. Which of the following could be a disadvantage of independent testing?**
 - A. Developer and independent testing will overlap and waste resources.**
 - B. Communication is limited between independent testers and developers.**
 - C. Independent testers are too slow and delay the project schedule.**
 - D. Developers can lose a sense of responsibility for quality.**
- 2. How could engaging with the user community benefit testing phases?**
 - A. Provides additional funding for extra testing resources.**
 - B. Ensures user needs and expectations are understood and met.**
 - C. Reduces the overall number of test cases.**
 - D. Shortens the timeframes of test iterations.**
- 3. What enhances the likelihood of delivering requested functionality on time?**
 - A. Improving communication with the development team.**
 - B. Updating the project schedule with precise deadlines.**
 - C. Regularly reviewing the requirements and adjusting priorities.**
 - D. Having a dedicated testing team for each increment.**
- 4. What is a critical aspect of managing a testing team in a software development company?**
 - A. Ensuring the team works in isolation without developer intervention**
 - B. Promoting good communication and team dynamics**
 - C. Delegating all tasks to junior team members**
 - D. Focusing solely on meeting deadlines**

5. What factors should be considered to evaluate if sufficient testing has been completed?

- A. i and ii and iv and vi**
- B. i and ii and iii and vi**
- C. ii and iii and iv and v**
- D. i and ii and v and vi**

6. Which of the following is NOT a valid reason for using entry and exit criteria in a test strategy?

- A. The expectation is that development testing is not adequate.**
- B. Exit criteria are used to decide on when to stop testing.**
- C. Using entry and exit criteria will prevent poorly tested software from progressing.**
- D. Entry and exit criteria are a principal way for getting adequate resources.**

7. What purpose does a technical review serve?

- A. To check compliance with requirements**
- B. To ensure the team meets deadlines**
- C. To assess the performance of team members**
- D. To identify discrepancies in the functional design**

8. Which are the TWO most appropriate examples of entry criteria documented in the system integration test plan?

- A. The percentage of decision coverage achieved during unit testing.**
- B. The availability of the latest version of the capture-replay tool.**
- C. The sign-off of a performance test software release note.**
- D. The percentage of acceptance test procedures scheduled for execution.**

9. What is the Risk Priority Number for a specific risk item if it calculates to 63?

- A. 16**
- B. 2**
- C. 1**
- D. 63**

10. Which is a recommended approach when dealing with high-risk items during testing?

- A. Exploratory testing at all levels**
- B. Poor documentation practices**
- C. Structural testing techniques**
- D. Unit testing as the only method**

SAMPLE

Answers

SAMPLE

1. D
2. B
3. C
4. B
5. A
6. D
7. A
8. A
9. D
10. C

SAMPLE

Explanations

SAMPLE

1. Which of the following could be a disadvantage of independent testing?

- A. Developer and independent testing will overlap and waste resources.**
- B. Communication is limited between independent testers and developers.**
- C. Independent testers are too slow and delay the project schedule.**

D. Developers can lose a sense of responsibility for quality.

Independent testers play a crucial role in the quality assurance process by providing an unbiased perspective on the quality of the product. However, one possible disadvantage of independent testing is that developers may start to rely too heavily on the independent testing team, leading to a diminished sense of personal responsibility for the overall quality of their work. When developers know that an independent team will test their code, they might feel less compelled to ensure that their contributions meet quality standards before delivery. This dependency could result in developers feeling less accountable for producing high-quality code, which can ultimately compromise the overall quality of the product. In a well-integrated testing process, it is essential for developers to maintain ownership of the quality of their work. While independent testing is valuable for objectivity, a balance must be struck to ensure that developers remain engaged and responsible for quality outcomes, rather than deferring all responsibility to external testers.

2. How could engaging with the user community benefit testing phases?

- A. Provides additional funding for extra testing resources.**
- B. Ensures user needs and expectations are understood and met.**
- C. Reduces the overall number of test cases.**
- D. Shortens the timeframes of test iterations.**

Engaging with the user community during the testing phases is incredibly valuable, primarily because it helps ensure that user needs and expectations are understood and met. When testers and developers seek feedback from actual users, they gain insights into how the product will be used in real-world scenarios. This direct interaction allows teams to identify critical features and functionalities that might not be captured in the original requirements, as well as to uncover usability issues that could affect user satisfaction. By involving users early in the testing process, teams can adjust their testing strategy to focus on the aspects that matter most to the end user. This can lead to more relevant test cases that align with user expectations and ultimately enhance the product's quality. Furthermore, when the user community feels valued and their input is considered, it fosters better relationships and can lead to increased user satisfaction and loyalty. In summary, engaging with users not only clarifies their needs but also helps optimize the testing process, ultimately resulting in a more successful product launch.

3. What enhances the likelihood of delivering requested functionality on time?

- A. Improving communication with the development team.
- B. Updating the project schedule with precise deadlines.
- C. Regularly reviewing the requirements and adjusting priorities.**
- D. Having a dedicated testing team for each increment.

The correct choice relates to the importance of regularly reviewing the requirements and adjusting priorities in a project. This practice enhances the likelihood of delivering requested functionality on time because it ensures that the team is aligned with the most current project needs and stakeholder expectations. By continuously reviewing requirements, the team can identify which features are most critical and should be prioritized for development. This approach allows for flexibility in responding to changes in business needs or user feedback, which is often necessary in dynamic project environments. It also helps to manage the scope of the project effectively, avoiding last-minute changes that could lead to delays. This practice contrasts with other options that, while they may contribute positively to project management, do not directly address the fundamental issue of aligning work with current priorities. For instance, improving communication with the development team can foster better collaboration and understanding, but without proper prioritization of requirements, it may not necessarily lead to on-time delivery. Updating the project schedule with precise deadlines is important for tracking progress but doesn't inherently account for changes in project scope or priorities. Similarly, having a dedicated testing team can enhance quality assurance processes, but it doesn't specifically ensure timely delivery unless it is coupled with appropriate prioritization and requirement management.

4. What is a critical aspect of managing a testing team in a software development company?

- A. Ensuring the team works in isolation without developer intervention
- B. Promoting good communication and team dynamics**
- C. Delegating all tasks to junior team members
- D. Focusing solely on meeting deadlines

Promoting good communication and team dynamics is a critical aspect of managing a testing team in a software development company because effective communication fosters collaboration, transparency, and shared understanding among team members. In a testing environment, where feedback about software quality is vital, maintaining open lines of communication ensures that testers can share insights, discuss challenges, and align their efforts with the developers and other stakeholders. Good team dynamics also enhance morale and productivity, as team members feel valued and engaged in the process. When testers collaborate well, they can identify issues more efficiently, share knowledge about testing strategies and techniques, and support each other in achieving common goals. Therefore, the emphasis on nurturing communication and team dynamics contributes directly to the success of the software testing process and helps cultivate a culture of quality within the organization.

5. What factors should be considered to evaluate if sufficient testing has been completed?

- A. i and ii and iv and vi**
- B. i and ii and iii and vi**
- C. ii and iii and iv and v**
- D. i and ii and v and vi**

To determine whether sufficient testing has been completed, it is important to consider several key factors, typically including the defined requirements, the level of risk, the test coverage achieved, and stakeholder satisfaction. These aspects play an essential role in evaluating the comprehensiveness and effectiveness of the testing process. When looking at option A, it captures a comprehensive assessment based on the listed criteria. This choice implies that relevant factors such as requirements fulfillment (i), risk assessment (ii), test coverage (iv), and stakeholder feedback (vi) are being evaluated. Each of these factors contributes significantly to understanding whether the testing efforts align with the initial goals set out in the project. - Requirements completion ensures that all anticipated functionality and features have been tested. - Risk assessment involves measuring known risks and ensuring that testing has sufficiently addressed these areas. - Test coverage ensures that the tests conducted sufficiently cover the intended use cases and potential failure modes. - Stakeholder feedback verifies that the testing outcomes align with the expectations and needs of the project's stakeholders. By assessing these factors together, a more accurate picture of the overall testing effectiveness and sufficiency can be derived. This comprehensive approach helps ensure that the product not only meets its requirements but is also robust against potential issues, ultimately leading to better quality software

6. Which of the following is NOT a valid reason for using entry and exit criteria in a test strategy?

- A. The expectation is that development testing is not adequate.**
- B. Exit criteria are used to decide on when to stop testing.**
- C. Using entry and exit criteria will prevent poorly tested software from progressing.**
- D. Entry and exit criteria are a principal way for getting adequate resources.**

The choice stating that entry and exit criteria are a principal way for getting adequate resources is not a valid reason for using these criteria in a test strategy, because entry and exit criteria primarily serve to establish conditions that must be met before testing begins and conditions that determine when testing can be concluded. Entry criteria focus on the prerequisites needed to start a test phase or activity, ensuring that the testing environment is ready, and that necessary documents and requirements are in place. This might relate to the adequacy of the software to be tested, verification of requirements, and availability of test data. Exit criteria, on the other hand, define the conditions that must be satisfied to conclude the testing phase. They help the team decide when sufficient testing has been carried out, ensuring quality standards are met and determining whether the software can be considered for release. While having adequate resources is crucial for testing, entry and exit criteria do not directly address resource allocation or acquisition. Instead, they are focused on quality assurance and readiness for testing activities, which informs decisions surrounding testing depth and completion rather than resourcing.

7. What purpose does a technical review serve?

- A. To check compliance with requirements**
- B. To ensure the team meets deadlines**
- C. To assess the performance of team members**
- D. To identify discrepancies in the functional design**

The purpose of a technical review is primarily to check compliance with requirements. This process involves evaluating the technical artifacts, such as design documents, code, or specifications, to ensure that they align correctly with the established requirements and standards set for the project. By conducting a technical review, the team can identify potential issues early in the development lifecycle, ensuring that the product being developed meets the intended functionality and quality criteria. This proactive approach helps reduce the risk of defects and miscommunications later in the project, ultimately leading to a more successful outcome. While the other options may represent important aspects of project management and team dynamics, they do not specifically align with the core objective of a technical review, which is focused on verifying that technical work adheres to the defined requirements and quality standards.

8. Which are the TWO most appropriate examples of entry criteria documented in the system integration test plan?

- A. The percentage of decision coverage achieved during unit testing.**
- B. The availability of the latest version of the capture-replay tool.**
- C. The sign-off of a performance test software release note.**
- D. The percentage of acceptance test procedures scheduled for execution.**

The identification of entry criteria is vital in ensuring that the system is ready for the next phase of testing. In the context of system integration testing, entry criteria define the minimum requirements or conditions that must be met before testing can begin. One of the appropriate examples of entry criteria is the percentage of decision coverage achieved during unit testing. Achieving a specific level of decision coverage is crucial because it indicates that the unit tests have adequately exercised the various paths in the code. This ensures that the system components are functioning correctly before they are integrated with other parts of the system. This focus on coverage metrics helps mitigate the risk of issues arising from components that have not been sufficiently tested at the unit level, yielding a more stable environment for integration testing. The other options, while related to testing activities, do not directly address the prerequisites needed before beginning system integration testing. The availability of tools, sign-offs for software releases, and scheduled acceptance test procedures may contribute to the overall testing process but are not fundamental conditions that need to be satisfied before starting the integration tests.

9. What is the Risk Priority Number for a specific risk item if it calculates to 63?

- A. 16
- B. 2
- C. 1
- D. 63**

The Risk Priority Number (RPN) is a numerical rating calculated to prioritize risks based on their potential impact and the likelihood of occurrence. In this context, if a specific risk item has an RPN that calculates to 63, it means that, according to the defined methodology for assessing risk, this figure accurately represents the priority of the risk. The RPN is typically determined using a formula that considers various factors, such as severity, occurrence, and detection, often leading to a specific numerical result. Therefore, if the calculated RPN is 63, this number indicates the risk's importance relative to others, making it essential for the risk management process. Providing a straightforward answer, the figure of 63 directly corresponds to the RPN for that risk item, confirming that it is indeed the correct and relevant value to indicate its priority.

10. Which is a recommended approach when dealing with high-risk items during testing?

- A. Exploratory testing at all levels**
- B. Poor documentation practices**
- C. Structural testing techniques**
- D. Unit testing as the only method**

The recommended approach for dealing with high-risk items during testing is to use structural testing techniques. Structural testing, often referred to as white-box testing, focuses on the internal structure of the application. By utilizing this method, testers can examine the code and ensure that high-risk areas are thoroughly tested. This is essential because high-risk items may involve critical functionalities or complex algorithms that require detailed analysis to uncover potential flaws or vulnerabilities. Structural testing techniques allow testers to create test cases that specifically target the paths and conditions within the code that are most susceptible to errors, thus maximizing the probability of discovering defects in high-risk items. By ensuring that these critical areas are tested thoroughly, teams can mitigate risks effectively and enhance product quality. In contrast, other options do not align as effectively with a structured approach to testing high-risk items. For instance, exploratory testing, while valuable, lacks the rigorous and systematic emphasis that structural testing provides. Poor documentation practices can lead to misunderstandings and errors, making it more challenging to manage high-risk scenarios. Finally, relying solely on unit testing without considering other levels of testing can leave critical integration and system risks unaddressed. Therefore, structural testing techniques stand out as the most effective strategy for dealing with high-risk items during testing.

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

<https://istqbctflpro.examzify.com>

We wish you the very best on your exam journey. You've got this!

SAMPLE