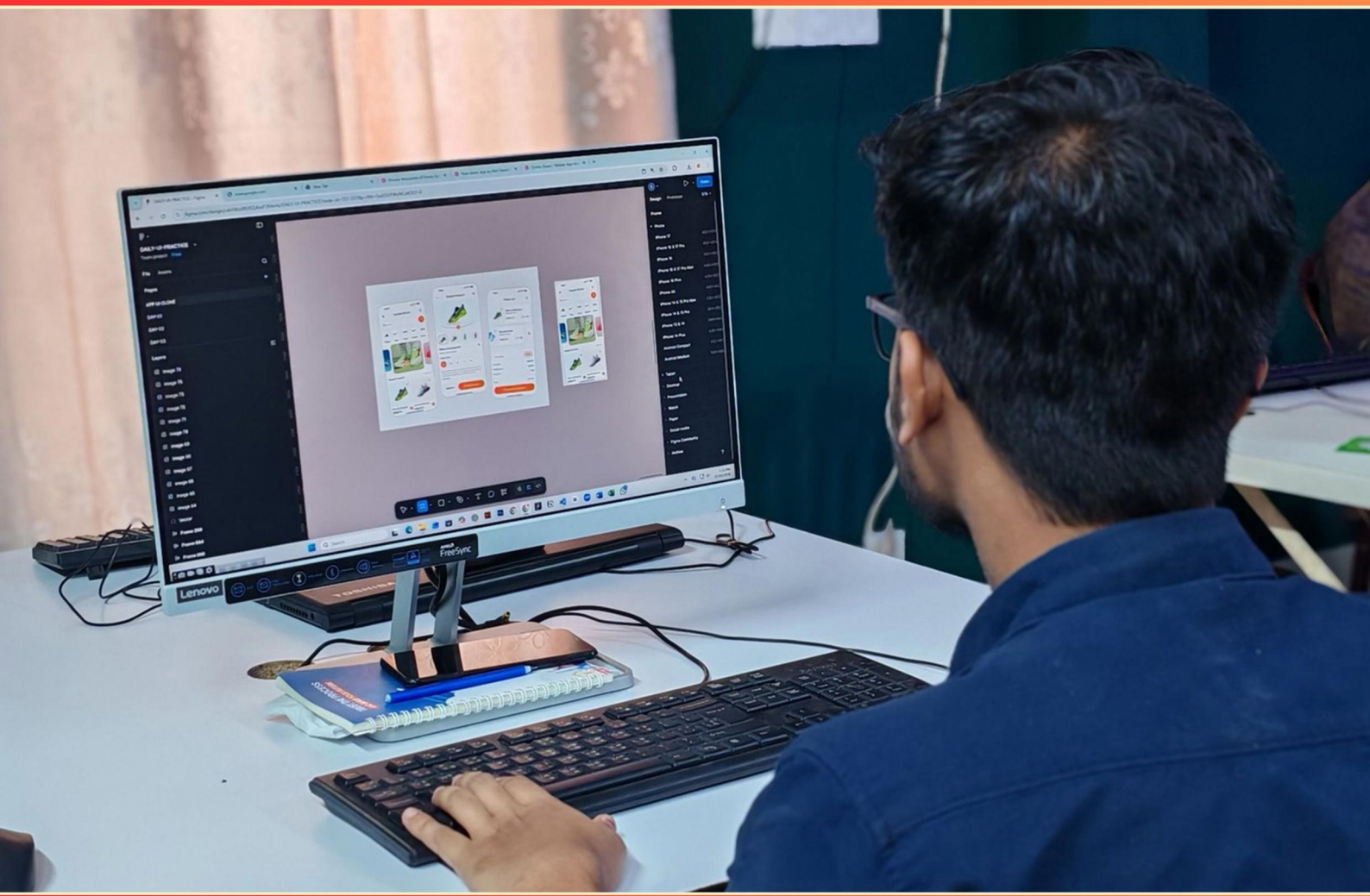
ASTQB FOUNDATION LEVEL PRACTICE EXAM (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://astqbfoundationlevel.examzify.com

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 8

Explanations 10

Next Steps 16

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.

SAMPLE

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!

SAMPLE

Questions

SAMPLE

1. What is the primary purpose of Test Design in the context of test specification tools?

A. To outline testing procedures

B. To prepare test data

C. To identify test cases

D. To measure test execution

2. What is a key factor contributing to a successful review?

A. Diverse technology tools

B. Predefined goals

C. Long duration of reviews

D. Frequent team changes

3. Which basic control flow construct is characterized by executing code repeatedly based on a specified condition?

A. Decision Point

B. Iteration

C. Sequence

D. Case Statement

4. What is the primary purpose of incident management?

A. To create user manuals for software

B. To recognize, investigate, and dispose of incidents

C. To design and develop new software products

D. To analyze market trends in software

5. How does the review process contribute to quality improvement in testing?

A. By eliminating the need for documentation

B. By focusing only on final product delivery

C. By enabling the identification of defects

D. By increasing software release frequency

SAMPLE

6. What triggers maintenance testing?

A. Only software updates

B. Modifications, migrations, and retirements

C. Only bug fixes

D. Software performance reviews

7. What does the test basis identify in the context of testing?

A. Test environments and budgets

B. Documents and artifacts for requirements

C. End-user functionalities and interfaces

D. Testing tools and resources needed

8. Which testing phase is performed to ensure components work together?

A. Integration testing

B. Unit testing

C. System testing

D. Acceptance testing

9. Which type of testing helps identify defects without executing the application?

A. Dynamic testing

B. Static testing

C. Performance testing

D. Usability testing

10. What type of flow does the system process flow diagram represent?

A. Low level execution paths

B. High level process flow

C. Flow of data only

D. Detailed instruction sequence

SAMPLE

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. What is the primary purpose of Test Design in the context of test specification tools?

A. To outline testing procedures

B. To prepare test data

C. To identify test cases

D. To measure test execution

The primary purpose of Test Design in the context of test specification tools is to identify test cases. This involves defining the conditions under which the tests will be conducted, specifying which functionalities or features will be tested, and developing specific scenarios or inputs that will be used during testing. Identifying test cases is a critical part of the testing process as it ensures that all aspects of the requirements are covered, aligning the tests with the desired outcomes. This phase helps testers to methodically break down the software features into specific testable components, thereby improving the effectiveness and efficiency of the testing process. While outlining testing procedures, preparing test data, and measuring test execution are all important components of the overall testing process, they do not capture the essence of the Test Design phase as precisely as identifying test cases does. The focus on test cases is what enables a structured approach to verifying that the software behaves as intended under various conditions.

2. What is a key factor contributing to a successful review?

A. Diverse technology tools

B. Predefined goals

C. Long duration of reviews

D. Frequent team changes

A key factor contributing to a successful review is having predefined goals. Setting clear, specific objectives for what the review aims to accomplish provides direction and focus. This allows team members to align their efforts and evaluations toward achieving those defined outcomes. When everyone understands the purpose of the review, it enhances collaboration, ensures that relevant aspects are adequately addressed, and ultimately leads to more effective and meaningful discussions. Predefined goals assist in measuring success, as the objectives can be assessed post-review to determine if they were met. This clarity also helps in prioritizing which areas to review thoroughly and which may require less attention, thereby streamlining the entire process. Overall, the presence of predefined goals is essential for guiding the team's efforts during a review and ensuring that it yields actionable insights.

SAMPLE

3. Which basic control flow construct is characterized by executing code repeatedly based on a specified condition?

A. Decision Point

B. Iteration

C. Sequence

D. Case Statement

The correct answer is iteration. In programming and software testing, iteration refers to the repetition of code execution as long as a specified condition remains true. This is fundamental in scenarios where the same set of instructions needs to be executed multiple times until a particular condition changes, such as processing items in a list or repeating a task until a user gives a specific input. Iteration allows for efficient and concise code, enabling developers to handle repetitive tasks without needing to manually write out similar blocks of code for each execution. Common structures that facilitate iteration include loops such as "for," "while," and "do-while." In contrast, a decision point refers to branching logic where the program chooses a path based on a condition, but it does not involve repeated execution of code. Sequence simply denotes the linear execution of statements one after another, and the case statement serves as a control structure for selecting one of many possible code paths but does not involve repetition based on conditions. Thus, iteration stands out as the construct specifically designed for executing a piece of code repeatedly contingent upon defined criteria.

4. What is the primary purpose of incident management?

A. To create user manuals for software

B. To recognize, investigate, and dispose of incidents

C. To design and develop new software products

D. To analyze market trends in software

The primary purpose of incident management is centered around recognizing, investigating, and resolving incidents that disrupt normal service operations. This process ensures that services are restored to operations as quickly as possible, minimizing impact on the business. It involves identifying incidents, classifying them according to their severity, and investigating their root causes to prevent recurrence. Effective incident management leads to improved service quality, enhances user experience, and promotes stability within IT operations. This capability is crucial for organizations that rely on software and IT services, as it helps maintain productivity and supports a continuous service improvement culture. The other choices do not align with the core objectives of incident management. Creating user manuals pertains to documentation and user support, rather than managing incidents. Designing and developing new software products focuses on the creation and development processes, while analyzing market trends is related to strategic business planning rather than operational incident response.

SAMPLE

5. How does the review process contribute to quality improvement in testing?

A. By eliminating the need for documentation

B. By focusing only on final product delivery

C. By enabling the identification of defects

D. By increasing software release frequency

The review process plays a crucial role in quality improvement in testing primarily by enabling the identification of defects. Through systematic reviews, such as code reviews, requirement reviews, and test case reviews, teams can detect issues at various stages of development before they escalate into more significant problems. This proactive approach helps ensure that any defects are addressed early, which not only enhances the overall quality of the product but also reduces potential costs and time associated with fixing those defects later in the development cycle. Identifying defects during reviews allows teams to gather feedback, improve processes, and refine testing strategies, ultimately leading to a more reliable software product. This reinforces the overall testing lifecycle by providing insights that can inform better practices and prevent similar issues from arising in the future. Thus, the review process is essential in fostering a culture of continuous improvement within software testing.

6. What triggers maintenance testing?

A. Only software updates

B. Modifications, migrations, and retirements

C. Only bug fixes

D. Software performance reviews

Maintenance testing is primarily triggered by various activities related to the ongoing management and evolution of software after its initial release. These activities include modifications to enhance functionality or address user needs, migrations that involve moving the software to a different environment or platform, and retirements which might require validation that existing functionalities are no longer affected during the decommission process. Each of these factors plays a significant role in maintaining the software's integrity, performance, and relevance to user requirements. For instance, when modifications are made, it's crucial to ensure that existing features are still functioning correctly and that new features integrate seamlessly. Similarly, during migrations, testing ensures that the software operates as expected in the new environment. Maintenance testing thus encompasses a broader range of activities beyond just bug fixes or software updates, ensuring that all aspects of the software continue to deliver reliable performance and fulfill user expectations during its lifecycle.

SAMPLE

7. What does the test basis identify in the context of testing?

A. Test environments and budgets

B. Documents and artifacts for requirements

C. End-user functionalities and interfaces

D. Testing tools and resources needed

The test basis is a crucial concept in software testing as it refers to the underlying documentation and artifacts that provide the foundation for the testing process. It encompasses all the information that can be used to derive test cases, ensuring that the testing effort aligns with the software requirements and specifications. This includes requirements documents, design documents, user stories, and any other relevant materials. By basing test cases on these documents, testers can verify that the software meets its intended functionality and that it behaves as expected according to the specified requirements. A solid test basis ensures that testing is thorough and systematically covers all necessary aspects of the software, which leads to improved quality and defect detection. In contrast, aspects like test environments and budgets, end-user functionalities, and testing tools and resources, while important, do not define the core of what the test basis is. The test basis specifically focuses on the documents and artifacts that outline what needs to be tested and why, making it fundamental to the creation and execution of effective testing strategies.

8. Which testing phase is performed to ensure components work together?

A. Integration testing

B. Unit testing

C. System testing

D. Acceptance testing

Integration testing is the phase of testing that is specifically designed to verify that individual components or modules of a system interact and function together correctly. This type of testing focuses on the interfaces and the interaction between integrated components, ensuring that they communicate as expected and that data is passed between them without issues. During integration testing, testers will typically examine how multiple units work together, looking for defects that arise only when components are combined. This is crucial because individual units may perform well in isolation during unit testing, but when brought together, their interactions can lead to unexpected behavior or failures. In contrast, unit testing focuses on testing individual components or functions in isolation, without considering their interaction with other components. System testing evaluates the complete and integrated software system to ensure it meets the specified requirements, while acceptance testing assesses the system's readiness for use from the user's perspective. Thus, integration testing is essential for confirming that all parts of the software work together harmoniously before moving on to broader testing phases.

SAMPLE

9. Which type of testing helps identify defects without executing the application?

A. Dynamic testing

B. Static testing

C. Performance testing

D. Usability testing

Static testing is a method of software testing that involves examining the code, documentation, and other project materials without actually executing the software application. This approach primarily focuses on detection of defects early in the development process, such as in the requirements or design phase, by reviewing files or executing tools that analyze source code. Static testing can include activities such as code reviews, static analysis, and inspections. Its key advantage is that it can identify issues like coding errors, deviations from standards, and design flaws before they turn into costly problems later in the development lifecycle. By identifying these defects without running the application, teams can address potential issues early, saving time and resources. In contrast, dynamic testing involves running the application and observing its behavior to find issues, while performance testing is concerned with how the application behaves under various load conditions. Usability testing focuses on how user-friendly and intuitive the application is. Thus, static testing stands out as the only method among the options that identifies defects without the need for execution.

10. What type of flow does the system process flow diagram represent?

A. Low level execution paths

B. High level process flow

C. Flow of data only

D. Detailed instruction sequence

A system process flow diagram typically illustrates the high-level overview of how processes in a system interact and flow. This type of diagram is beneficial for understanding the major steps involved in a process, how those steps connect, and the overall sequence in which operations occur. By representing these processes at a higher abstraction level, it allows stakeholders to grasp the big picture without getting bogged down in the minute details of each specific action. High-level process flows emphasize the main components and their interactions rather than focusing on low-level execution pathways, detailed sequences, or just the flow of data. This is important for stakeholders who need to understand system operations at a strategic level without delving into the complexities of execution paths or specific instructions within the system operations. Understanding this distinction is crucial when evaluating what information is presented within a flow diagram and how it relates to the broader context of system processes.

SAMPLE

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://astqbfoundationlevel.examzify.com

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

SAMPLE