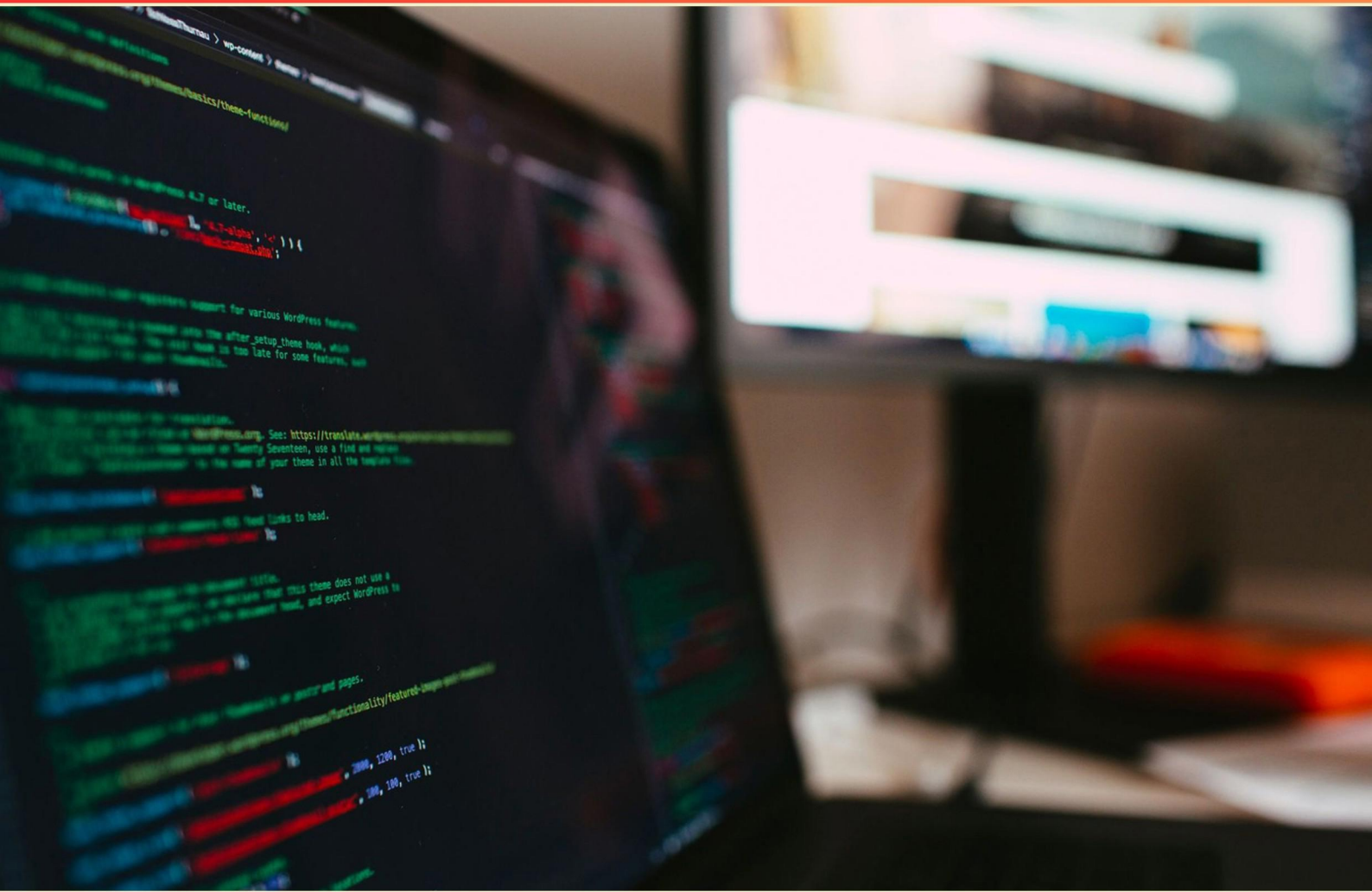


PIMA JTED SOFTWARE AND APP DESIGN PRACTICE TEST (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

**VISIT US ONLINE FOR
THE FULL VERSION STUDY GUIDE**

<https://pimajtedsoftwareappdesign.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.

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. What is a procedure in programming?**
 - A. A set of conditions for a program
 - B. A block of code or instructions that can run independently
 - C. A type of software documentation
 - D. A user interface element
- 2. What does an object represent in object-oriented programming?**
 - A. A class definition
 - B. An instance of a class
 - C. A set of functions
 - D. A variable type
- 3. What is the primary function of a browser?**
 - A. To edit documents
 - B. To connect devices in a local network
 - C. To view HTML pages/websites
 - D. To manage operating system files
- 4. What is a fundamental concept that aids in the organization and management of related data and functions in programming?**
 - A. Class
 - B. Encapsulation
 - C. Object
 - D. Structure
- 5. Which process relates to the automatic exchange of information between computing devices?**
 - A. Networking
 - B. Computing
 - C. Interfacing
 - D. Interconnectivity

- 6. What is the main advantage of using XML over HTML?**
- A. Faster page loading
 - B. Greater flexibility in data representation
 - C. More advanced styling options
 - D. Easier error checking
- 7. What does computer literacy encompass?**
- A. Understanding basic math
 - B. Knowledge of how to write code only
 - C. In-depth knowledge of how computers operate and solve problems
 - D. Expertise in operating systems
- 8. Which methodology involves a collaborative iterative software development process including sprints?**
- A. Agile methodology
 - B. Algorithm
 - C. API
 - D. Application
- 9. Which model of software development was developed by Winston W Royce?**
- A. Agile
 - B. Spiral
 - C. Waterfall
 - D. Incremental
- 10. What type of program is designed to detect and remove malware from a computer?**
- A. Anti-virus
 - B. Adware
 - C. Algorithm
 - D. Array

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. What is a procedure in programming?

- A. A set of conditions for a program
- B. A block of code or instructions that can run independently**
- C. A type of software documentation
- D. A user interface element

A procedure in programming is defined as a block of code or instructions that can run independently. This structure allows programmers to encapsulate a specific task or functionality within a defined set of commands, making it reusable throughout a program. Procedures, also known as functions or methods, can be called multiple times from different parts of the program, which helps in organizing and managing code more efficiently. By modularizing code into procedures, developers improve readability, reduce redundancy, and facilitate easier debugging and testing. In contrast, the other options do not accurately describe a procedure. The first option refers to conditions rather than executable code. The third option relates to software documentation, which is entirely different from the execution of code. Lastly, the fourth option describes a user interface element, which is also not related to the functionality or execution of a block of code in programming. Thus, the selection of a block of code or instructions that can run independently aligns with the fundamental concept of what a procedure is in programming.

2. What does an object represent in object-oriented programming?

- A. A class definition
- B. An instance of a class**
- C. A set of functions
- D. A variable type

In object-oriented programming (OOP), an object fundamentally represents an instance of a class. A class serves as a blueprint or template that defines properties (attributes) and behaviors (methods) which the objects created from it will possess. When a class is instantiated, the resulting object embodies the specific state defined by the class and can utilize the methods that the class includes. For example, consider a class called "Dog". This class may have attributes such as breed, size, and age, along with methods like bark() or fetch(). When you create an instance of the "Dog" class, such as "myDog" with the breed set to "Labrador", you are creating an object that represents that specific dog with its defined attributes and behaviors. This concept is key in OOP as it allows for encapsulation, making code more modular, organized, and easier to manage. Objects can interact with one another and can also maintain their own states, which is central to the object-oriented paradigm.

3. What is the primary function of a browser?

- A. To edit documents
- B. To connect devices in a local network
- C. To view HTML pages/websites**
- D. To manage operating system files

The primary function of a browser is to view HTML pages and websites. Browsers act as intermediaries that interpret and render the codes and content of web pages, allowing users to access and navigate the vast information available on the internet. By translating HTML, CSS, and JavaScript, browsers enable users to interact with websites, displaying text, images, videos, and other multimedia content in a user-friendly format. This functionality is essential for accessing online information, services, and applications, making browsers a pivotal element of web experience. In contrast, editing documents pertains to word processors or other specialized software applications designed for document creation and modification. Connecting devices in a local network involves networking hardware and protocols rather than browser functionality. Managing operating system files is a task performed by the operating system itself, not by a web browser. Thus, viewing HTML pages and websites is the distinct and fundamental role that browsers serve in computing.

4. What is a fundamental concept that aids in the organization and management of related data and functions in programming?

- A. Class
- B. Encapsulation**
- C. Object
- D. Structure

The concept that aids in the organization and management of related data and functions in programming is encapsulation. Encapsulation is a fundamental principle of object-oriented programming that combines data (attributes) and the methods (functions) that manipulate that data into a single unit known as a class. This organization not only helps in grouping related functionalities but also provides a means to protect the integrity of the data. By encapsulating data and functions, programming promotes modularity, making code easier to understand and maintain. Encapsulation allows for the implementation of access controls (like private or public visibility in a class), thereby shielding the internal state of the object from unintended interference and misuse. This organized structure is crucial in managing complexity in software design. While other options relate to structuring data and functions—such as classes, which are blueprints for creating objects, and data structures, which are ways to organize data—encapsulation specifically emphasizes the bundling of data with the methods that operate on it. This focus on protecting data and promoting modular design sets encapsulation apart as the key concept in this context.

5. Which process relates to the automatic exchange of information between computing devices?

- A. Networking**
- B. Computing
- C. Interfacing
- D. Interconnectivity

The process that relates to the automatic exchange of information between computing devices is networking. Networking involves the establishment of interconnected systems that allow devices to communicate and share data seamlessly without manual intervention. This communication can happen over various mediums such as wired connections, wireless signals, or the internet, enabling devices to send and receive data autonomously. Networking encompasses a range of technologies and protocols designed to facilitate this communication, ensuring that information can flow efficiently and securely between different devices. In many systems, this automatic exchange is fundamental to maintaining the functionality of applications, databases, and services that rely on real-time data sharing. The other options, while related to computing in general, do not specifically capture the essence of automatic information exchange to the same degree. Computing focuses more broadly on the processing of information, interfacing deals with how users or systems interact with devices, and interconnectivity refers to the state of being connected but does not specifically emphasize the automatic exchange aspect. Networking distinctly characterizes the automated communication processes that facilitate information flow between devices, making it the most appropriate answer.

6. What is the main advantage of using XML over HTML?

- A. Faster page loading
- B. Greater flexibility in data representation**
- C. More advanced styling options
- D. Easier error checking

The main advantage of using XML over HTML lies in XML's greater flexibility in data representation. XML, which stands for eXtensible Markup Language, is designed to store and transport data. It allows for the creation of custom tags that can be tailored to fit specific data types and structures, making it highly adaptable for various applications. This flexibility enables developers to define their own data structure, creating a semantic representation of information that is both human-readable and machine-readable. In contrast, HTML (HyperText Markup Language) is primarily focused on the presentation of web content rather than the data structure itself. HTML provides a fixed set of tags that are primarily concerned with how information is displayed in a web browser. While HTML is powerful for structuring content on the web, it does not offer the same level of customizability and data interchange capabilities that XML provides. This flexibility makes XML a preferred choice in scenarios where complex data needs to be shared across different systems, such as in web services, data feeds, and APIs. By using XML, developers can create clear and robust systems for data exchange, accommodating changes in data structure without the need for restructuring entire applications.

7. What does computer literacy encompass?

- A. Understanding basic math
- B. Knowledge of how to write code only
- C. In-depth knowledge of how computers operate and solve problems**
- D. Expertise in operating systems

Computer literacy is a comprehensive understanding of how computers function and the various tasks they can perform to solve problems. It includes not only the ability to operate computers and software but also knowledge about the fundamental principles of technology, the internet, and the impact of digital tools on society. The correct choice highlights that computer literacy goes beyond just basic user skills or coding knowledge; it encompasses a broader spectrum of understanding the interactions between hardware and software, how data is processed, and how to effectively utilize technology to achieve specific outcomes. This foundational knowledge is crucial in navigating the increasingly digital world and is applicable in numerous contexts, from everyday applications to complex problem-solving scenarios. In contrast, other options focus on either too narrow of skills or domains. For instance, simply understanding basic math does not directly correlate with computer literacy. Similarly, while knowing how to write code is valuable, it represents only a subset of the knowledge required for comprehensive computer literacy. Expertise in operating systems, while important, is also a specialized area that does not encompass the entire breadth of computer literacy, which includes user experience with various applications and tools, as well as understanding the ethical implications of technology use.

8. Which methodology involves a collaborative iterative software development process including sprints?

- A. Agile methodology**
- B. Algorithm
- C. API
- D. Application

Agile methodology is a framework for software development that emphasizes flexibility, collaboration, and customer feedback throughout the development process. It involves breaking down the project into small, manageable units called sprints, which typically last for a few weeks. During these sprints, cross-functional teams work collaboratively to design, develop, and test features, allowing them to adapt to changes quickly and efficiently. This iterative approach enables teams to continuously improve their product by integrating user feedback at the end of each sprint, making it possible to respond to changing requirements and market conditions effectively. The emphasis on collaboration fosters communication among team members and stakeholders, ensuring that everyone involved is aligned and contributing towards the shared goals throughout the development cycle. The other options do not encompass this collaborative iterative aspect. An algorithm is a specific set of instructions for solving a problem, whereas an API (Application Programming Interface) provides a set of rules for how different software components should interact. An application refers to software designed to perform specific tasks for users and does not describe a development methodology.

9. Which model of software development was developed by Winston W Royce?

- A. Agile
- B. Spiral
- C. Waterfall**
- D. Incremental

Winston W. Royce is primarily known for introducing the Waterfall model of software development in his 1970 paper. This model is characterized by a linear and sequential approach, where each phase of the development process must be completed before moving on to the next. The phases typically include requirements analysis, design, implementation, testing, deployment, and maintenance. The Waterfall model emphasizes thorough documentation and planning at each stage. This structured approach was influential in increasingly complex software projects, as it provided a clear framework for how development should proceed. It's particularly suitable for projects with well-understood requirements where changes are unlikely throughout the development process. In comparison, Agile focuses on iterative development and flexibility, while the Spiral model combines iterative design with systematic risk analysis, and the Incremental model breaks the project into smaller segments that build upon one another. These models reflect different philosophies and strategies that evolved after the Waterfall model was established, highlighting the adaptability needed in an ever-changing software landscape.

10. What type of program is designed to detect and remove malware from a computer?

- A. Anti-virus**
- B. Adware
- C. Algorithm
- D. Array

The correct answer is that an anti-virus program is specifically designed to detect and remove malware from a computer. Anti-virus software acts as a security measure that scans the system for malicious software, including viruses, worms, trojans, malware, and spyware. These programs use various techniques to identify malware, such as signature-based detection, which looks for known patterns of malicious code, and heuristic-based detection, which analyzes behavior and characteristics of programs to identify potential threats. In addition to detecting and removing malware, anti-virus programs often include features for real-time protection, scheduled scans, and updates that keep the software effective against new and evolving threats. This makes them essential tools for maintaining the integrity and security of computer systems. The other options do not fulfill the same purpose. Adware is a type of malware itself that displays advertisements on a computer, an algorithm is a step-by-step procedure used for calculations or problem-solving, and an array is a data structure that stores a collection of items. These options do not provide the malware detection and removal capabilities that an anti-virus program does.

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://pimajtedsoftwareappdesign.examzify.com>

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

SAMPLE