SQA Higher Computing Science Practice Exam (Sample)

Study Guide

BY EXAMZIFY

Everything you need from our exam experts!

Table of Contents

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

1. What is an essential characteristic of relational databases?

A. They can only contain text data

B. They can be accessed through tables with relationships

C. They are always stored remotely

D. They require complex algorithms

2. What is the term for a system that helps people make decisions by utilizing a database of facts and rules?

A. Expert system

B. Decision tree

C. Neural network

D. Data warehouse

3. What do you call a detailed documentation of data structures and their relationships in a project?

A. Data schema

B. Data model

C. Data dictionary

D. Data repository

4. What is a reference parameter in a subprogram definition?

A. A variable that cannot be modified

B. A parameter passed by value

C. A parameter whose value is changed by the function

D. A default parameter

5. What is the primary purpose of a data dictionary in programming projects?

A. To store user preferences

B. To define data structures and relationships

C. To back up data

D. To display user interfaces

6. Which term refers to the procedure to increase a variable's value by a fixed amount?

A. Decrement

B. Multiply

C. Increment

D. Assign

7. Which of the following best describes an operating system?

A. A type of application software

B. The system software that manages hardware and software resources

C. A programming language

D. A hardware component

8. What type of encryption uses the same key for both encrypting and decrypting data?

A. Asymmetric key encryption

B. Hash encryption

C. Symmetric key encryption

D. Public key encryption

9. Which of the following best describes a semantic error in programming?

A. The program crashes at runtime

B. The code contains grammatical mistakes

C. The logic of the code produces unintended outcomes

D. The code runs without issues but returns incorrect results

10. Which term describes the characteristic of objects in object-oriented programming where objects cannot be altered from outside?

A. Inheritance

B. Encapsulation

C. Abstraction

D. Polymorphism

Answers

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

Explanations

1. What is an essential characteristic of relational databases?

A. They can only contain text data

B. They can be accessed through tables with relationships

C. They are always stored remotely

D. They require complex algorithms

Relational databases are fundamentally designed to store data in structured formats using tables. Each table consists of rows and columns, where each row represents a unique record and each column represents a particular attribute of that record. A key aspect of relational databases is the use of relationships between these tables, which allows for the organization and retrieval of data in a meaningful way. This characteristic supports various operations, such as joins, which enable the combination of data from different tables based on related fields. The capability to access and manipulate data through these relationships is a core principle of relational databases, following the relational model proposed by E.F. Codd. This model emphasizes the importance of relationships in data organization and allows for greater flexibility and efficiency in data management, enabling developers to maintain data integrity and reduce redundancy. In contrast, other options do not accurately describe essential characteristics of relational databases. They are not limited to only text data; they can handle various data types, including integers, dates, and more. Additionally, relational databases can be stored locally or remotely, depending on the application's design and requirements. Finally, while some algorithms may be used for complex queries or data processing, the fundamental characteristics of relational databases do not inherently require complex algorithms to function. Thus, the ability to access

2. What is the term for a system that helps people make decisions by utilizing a database of facts and rules?

A. Expert system

B. Decision tree

C. Neural network

D. Data warehouse

The term that describes a system designed to assist people in making decisions by leveraging a database of facts and rules is an expert system. Expert systems are designed to mimic the decision-making ability of a human expert in a specific domain. They utilize a knowledge base, which contains both factual information and heuristic rules that guide the decision-making process. Expert systems work by applying logical reasoning to the set of rules and information they possess to arrive at conclusions or recommendations. This approach provides users with solutions based on structured knowledge, enhancing the decision-making process, especially in complex issues where human expertise is required. The other options, while related to data processing and analysis, serve different purposes. Decision trees are graphical representations used for decision-making, but they do not function as standalone systems with a knowledge base. Neural networks are a form of machine learning that imitates the way human brains operate to recognize patterns, rather than providing rule-based decisions. A data warehouse, on the other hand, is a centralized repository for storing large amounts of data but is not directly involved in decision-making. These distinctions highlight why the concept of an expert system is directly aligned with the goal of decision-support using established facts and rules.

3. What do you call a detailed documentation of data structures and their relationships in a project?

A. Data schema

B. Data model

C. Data dictionary

D. Data repository

A detailed documentation of data structures and their relationships in a project is referred to as a data dictionary. This is a comprehensive resource that describes the various components of the data used within the project, including definitions, characteristics, and data types for each element. It serves as a reference for developers, database designers, and stakeholders by providing clarity on how data is structured, how it interconnects, and the constraints applied to it. In addition to defining specific data attributes, the data dictionary often encompasses relationships among different data entities, documenting how they interact and relate within the larger framework of the project. This can help ensure consistent use of terminology and understanding across different teams involved in the project. While a data schema specifies the structure of a database and defines tables, fields, and their types, it does not provide the same level of detail in documentation as a data dictionary. A data model is more about the conceptual representation of data and often relies on a higher abstraction level, which may also not include the depth of descriptive detail found in a data dictionary. A data repository refers to a storage location for data rather than documentation. Thus, the specific emphasis on detailed documentation and the relationships between data structures makes the identification of a data dictionary the most fitting choice.

4. What is a reference parameter in a subprogram definition?

A. A variable that cannot be modified

B. A parameter passed by value

C. A parameter whose value is changed by the function

D. A default parameter

A reference parameter in a subprogram definition allows the function or procedure to modify the value of the argument passed to it directly. When a reference parameter is used, the function can access and change the value of the variable that was passed in. This means that any changes made to the parameter within the subprogram will affect the original variable outside of the subprogram, reflecting the changes when control returns to the caller. This behavior distinguishes reference parameters from parameters passed by value, where a copy of the variable is made and any changes are local to the subprogram. Therefore, when using reference parameters, it is crucial to understand that the original data can be altered, enabling the subprogram to return updated values to the caller without needing to pass return values explicitly. Other choices do not accurately represent what a reference parameter is and how it operates within a subprogram context. For instance, a variable that cannot be modified would suggest immutability, which contrasts with the very purpose of a reference parameter, while a default parameter refers to a parameter that has a predefined value if none is provided by the caller.

5. What is the primary purpose of a data dictionary in programming projects?

A. To store user preferences

B. To define data structures and relationships

C. To back up data

D. To display user interfaces

The primary purpose of a data dictionary in programming projects is to define data structures and relationships. A data dictionary serves as a centralized repository that provides comprehensive details about the data elements used within a system. It outlines the attributes, data types, constraints, and relationships between different data entities, which helps to ensure consistency and clarity throughout the development process. By having a structured definition of the data, developers can understand how various components interact and how data is organized within the application, which is vital for effective coding, maintenance, and database management. This understanding aids in data validation, helps with database design, and ensures that all team members have a clear reference point, reducing misunderstandings and errors as the project progresses.

6. Which term refers to the procedure to increase a variable's value by a fixed amount?

A. Decrement

B. Multiply

C. Increment

D. Assign

The term that refers to the procedure of increasing a variable's value by a fixed amount is "increment." Incrementing a variable typically means adding a specified value, which is often one, to the current value of that variable. This operation is frequently used in programming when you want to progress through a sequence, count occurrences, or adjust values in loops. For example, in a loop that counts from 1 to 10, you might increment the counter variable by 1 in each iteration to move from one number to the next. The concept of incrementing is foundational in many algorithms and coding practices, making it a critical operation in programming. The other terms provided represent different operations: decrement refers to reducing a variable's value; multiply involves scaling a variable's value by a factor; and assign relates to setting a variable to a specific value, rather than changing it incrementally.

7. Which of the following best describes an operating system?

A. A type of application software

B. The system software that manages hardware and software resources

C. A programming language

D. A hardware component

The best description of an operating system is that it is the system software that manages hardware and software resources. An operating system serves as an intermediary between users and the computer hardware. Its primary functions include managing the computer's hardware components such as the CPU, memory, and storage devices, while also providing services for application software. This enabling environment allows users to run programs, manage files, and interact with the system effectively. In contrast, application software refers to programs designed to perform specific tasks for users, such as word processing or spreadsheet applications, and is not responsible for managing the overall system. A programming language is a formal set of instructions that can be used to create software applications but is not itself an operating system. Lastly, hardware components refer to the physical parts of a computer system, such as the keyboard, monitor, and internal circuits, which work in conjunction with the operating system but do not encompass its functions. Thus, option B clearly captures the essential role and definition of an operating system.

8. What type of encryption uses the same key for both encrypting and decrypting data?

A. Asymmetric key encryption

B. Hash encryption

C. Symmetric key encryption

D. Public key encryption

Symmetric key encryption utilizes a single key for both the encryption and decryption processes. This means that the same key that transforms plaintext into ciphertext is also used to revert it back to its original form. This method is efficient and allows for quick encryption and decryption which can be crucial for performance in many applications. In symmetric key encryption, both the sender and receiver must have access to the same key and must keep it secret from potential eavesdroppers. The primary benefit is that it generally requires less computational power compared to its asymmetric counterpart, making it faster and suitable for encrypting large volumes of data. Asymmetric key encryption, in contrast, involves a pair of keys: a public key for encryption and a private key for decryption, thereby differentiating the processes. Hash encryption is a different concept altogether, focusing on transforming data into a fixed-size string of characters (a hash) that cannot be reversed to obtain the original data. Lastly, public key encryption is another term for asymmetric encryption which is characterized by its use of two different keys.

9. Which of the following best describes a semantic error in programming?

A. The program crashes at runtime

B. The code contains grammatical mistakes

C. The logic of the code produces unintended outcomes

D. The code runs without issues but returns incorrect results

A semantic error in programming refers to a situation where the code is syntactically correct—meaning that it follows the grammatical rules of the programming language—but does not produce the intended result due to a flaw in the logic. Option D accurately captures this definition; the code runs successfully without generating any syntax errors, but it ultimately fails to deliver the expected outcomes or results. This type of error often arises from misinterpretation of requirements, incorrect assumptions, or logical flaws that lead the program to behave in unintended ways. Identifying and fixing semantic errors can be challenging because the program's structure appears correct, and it executes without crashing, yet it may yield results that are logically incorrect based on the intended functionality.

10. Which term describes the characteristic of objects in object-oriented programming where objects cannot be altered from outside?

A. Inheritance

B. Encapsulation

C. Abstraction

D. Polymorphism

Encapsulation is the term that describes the characteristic of objects in object-oriented programming where the internal state of an object is protected from being directly modified from outside its class. This is achieved by restricting access to the object's internal data and allowing it to be modified only through well-defined methods, often referred to as getters and setters. By encapsulating data, the object's integrity is maintained, ensuring that any changes to its state adhere to the rules defined within the class itself. This characteristic promotes data hiding, which enhances security and makes the code easier to maintain and understand. The other terms reflect different concepts within object-oriented programming: inheritance relates to deriving new classes from existing ones, abstraction focuses on exposing only the essential features while hiding the complexities, and polymorphism allows methods to do different things based on the object it is acting upon.

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://sqahighercomputingscience.examzify.com

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