DATA ENGINEERING ASSOCIATE WITH DATABRICKS PRACTICE EXAM (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://dataengineeringassociate-databricks.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 does the flatten function do in SQL?

A. Converts rows into columns

B. Combines multiple arrays into a single array

C. Removes duplicates from an array

D. Restructures data into a flat schema

2. How does Spark ensure fault tolerance?

A. By regularly backing up data to an external system

B. Through the use of Resilient Distributed Datasets (RDDs)

C. By utilizing high availability clusters

D. Through data replication across multiple nodes

3. What is the SQL command to vacuum a specified Delta table?

A. DELETE FROM table_name

B. TRUNCATE TABLE table_name

C. VACUUM table_name

D. OPTIMIZE table_name

4. What is the primary function of a storage account in Databricks?

A. To provide a temporary workspace for computations

B. To store intermediate results from Spark computations

C. To offer a scalable solution for storing data

D. To facilitate real-time data streaming

5. What is the main function of the Databricks Runtime?

A. To manage user permissions

B. To provide an optimized Spark environment

C. To create data visualizations

D. To write SQL queries

6. In Spark, what is considered a task?

A. A unit of resource management

B. A grouping of jobs into stages

C. A single operation on a partition of data

D. A method for scheduling jobs

SAMPLE

7. Which SQL command would be used to check for record matches in a merge operation?

A. WHEN MATCHED

B. IF EXISTS

C. IF MATCHED

D. ON MATCH FOUND

8. What is a key feature of Delta Lake in terms of data management?

A. In-memory data processing

B. Time travel capabilities

C. Complex query optimization

D. Secure data sharing

9. What distinguishes DEEP CLONE from SHALLOW CLONE in SQL?

A. DEEP CLONE only copies transaction logs

B. SHALLOW CLONE fully copies data and metadata

C. DEEP CLONE does not incrementally sync changes

D. DEEP CLONE fully copies both data and metadata

10. What is the purpose of the Input_file_name() function in SQL?

A. To get the name of the current table

B. To retrieve the name of the file being processed

C. To filter data based on file type

D. To check the size of the input data

SAMPLE

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. What does the flatten function do in SQL?

A. Converts rows into columns

B. Combines multiple arrays into a single array

C. Removes duplicates from an array

D. Restructures data into a flat schema

The flatten function in SQL is primarily used to combine multiple arrays into a single array. This function is particularly useful in scenarios where you are dealing with nested data structures, such as arrays of arrays, and you want to simplify that structure by consolidating the elements into one array. Utilizing the flatten function allows data engineers to streamline complex datasets, making them easier to query and analyze. When you flatten an array, it reduces the complexity of working with nested structures and enables straightforward operations and functions to be applied to the data. In this context, while other options touch on relevant operations, only combining multiple arrays into a single array accurately defines the specific behavior of the flatten function. For instance, converting rows into columns typically pertains to pivot operations, removing duplicates is related to the distinct functions, and restructuring data into a flat schema aligns more with the concepts of normalization or data modeling rather than the specific functionality of flattening arrays.

2. How does Spark ensure fault tolerance?

A. By regularly backing up data to an external system

B. Through the use of Resilient Distributed Datasets (RDDs)

C. By utilizing high availability clusters

D. Through data replication across multiple nodes

Spark ensures fault tolerance primarily through the use of Resilient Distributed Datasets (RDDs). RDDs are a core data structure in Spark that provide a robust way to manage distributed datasets. Each RDD is immutable and can be created from existing datasets through transformations. The fundamental feature that enables fault tolerance is the lineage information that RDDs maintain. This lineage allows Spark to reconstruct lost data due to node failures by re-computing the data from original sources instead of restoring it from backups. This computation-centric approach means that even if a partition of data is lost, it can be rebuilt based on the transformations applied to the original dataset. In contrast, options such as backing up data to an external system do contribute to data recovery strategies but are not inherent to Spark's operation. High availability clusters refer to the infrastructure setup aimed at minimizing downtime but do not address the specifics of how Spark processes and handles data faults. Similarly, while data replication across multiple nodes can be part of a fault tolerance strategy in distributed systems, it is not the primary mechanism employed by Spark for achieving fault tolerance. Instead, the use of RDDs with their lineage tracking effectively allows Spark to maintain high availability and recover from failures seamlessly.

SAMPLE

3. What is the SQL command to vacuum a specified Delta table?

A. DELETE FROM table_name

B. TRUNCATE TABLE table_name

C. VACUUM table_name

D. OPTIMIZE table_name

The command to vacuum a specified Delta table is accurately identified as VACUUM table_name. This command is essential for managing the data retention and performance of Delta tables. When data is deleted or updated in a Delta Lake table, the old files are not immediately removed but are marked for deletion. The VACUUM command is used to physically delete these files that are no longer needed, which helps reclaim storage space and improve query performance by reducing the amount of data that needs to be scanned during operations. Using VACUUM is particularly important because Delta Lake maintains a transactional log of operations, and if older versions of the data are not vacuumed, they will continue to occupy space. This process can help manage the storage footprint and maintain the efficiency of the data lake over time. The other options, while they are valid SQL commands, do not serve the purpose of cleaning up the files and managing the storage for a Delta table. For instance, deleting data does not remove older versions of it from storage, truncating a table removes all rows but does not manage historical files, and optimizing a table reorganizes its files for better performance but does not delete old data. Thus, the VACUUM command is specifically tailored for the maintenance of Delta tables.

4. What is the primary function of a storage account in Databricks?

A. To provide a temporary workspace for computations

B. To store intermediate results from Spark computations

C. To offer a scalable solution for storing data

D. To facilitate real-time data streaming

The primary function of a storage account in Databricks is to offer a scalable solution for storing data. Databricks integrates seamlessly with cloud storage systems, which enables users to leverage the robust scalability and accessibility provided by these platforms. This allows teams to store vast amounts of structured and unstructured data efficiently. In the context of a data engineering workflow, having a scalable storage solution is critical as it supports data growth over time and provides the necessary infrastructure to help organizations efficiently manage and analyze their data assets. The ability to scale storage as needed ensures that Databricks can handle big data workloads without leading to performance bottlenecks or data access issues. Other options like providing a temporary workspace for computations or storing intermediate results are more aligned with specific functionalities within the Databricks environment, but they do not capture the overarching purpose of the storage account itself. Facilitating real-time data streaming, while important in data processing workflows, is not the core function of a storage account within Databricks either, as it focuses on data at rest rather than data in motion.

SAMPLE

5. What is the main function of the Databricks Runtime?

A. To manage user permissions

B. To provide an optimized Spark environment

C. To create data visualizations

D. To write SQL queries

The main function of the Databricks Runtime is to provide an optimized Spark environment. This runtime is built specifically to improve the performance of Apache Spark applications and offers various enhancements that are not available in the open-source version of Spark. It includes optimizations in areas like execution speed, integration with Delta Lake, and support for advanced analytics and machine learning workloads. The Databricks Runtime streamlines processes such as data processing and analysis, making it easier for data engineers and data scientists to perform complex data transformations and calculations efficiently. By leveraging this specialized environment, users can take advantage of features that enhance performance, scalability, and reliability in their big data workflows. Creating data visualizations and writing SQL queries are important tasks typically performed within the Databricks environment; however, they are not the primary function of the runtime itself. User permissions are managed through separate configurations and roles, while the primary focus of the Databricks Runtime is on delivering a high-performance execution engine for data processing.

6. In Spark, what is considered a task?

A. A unit of resource management

B. A grouping of jobs into stages

C. A single operation on a partition of data

D. A method for scheduling jobs

A task in Apache Spark refers to a single operation that is performed on a partition of data. When processing data, Spark breaks down jobs into smaller manageable chunks, and each partition of data can be processed independently. This parallel processing is what allows Spark to efficiently handle large datasets. The task is the fundamental unit of work in Spark's execution model. Each task operates on a specific partition of the data and executes computations defined in the job. Because tasks can be executed in parallel across different nodes in a cluster, they play a crucial role in maximizing resource utilization and minimizing execution time. Understanding this concept highlights how Spark's architecture enables scalability and performance, attributes that are essential for big data processing. This clarity on tasks is key for anyone looking to efficiently use Spark for data engineering tasks. The other options provided, while related to different aspects of Spark's execution environment, do not define a task correctly. Resource management pertains to the allocation of resources across jobs, grouping of jobs into stages refers to how Spark organizes tasks for execution, and scheduling jobs relates to how Spark determines when and where to run various jobs based on resource availability. None of these define what constitutes a task in Spark, making the definition of a task as a single operation on a partition the most accurate

SAMPLE

7. Which SQL command would be used to check for record matches in a merge operation?

A. WHEN MATCHED

B. IF EXISTS

C. IF MATCHED

D. ON MATCH FOUND

In a merge operation, the SQL command used to check for record matches is "WHEN MATCHED." This command is part of the MERGE statement in SQL, which allows for conditional updates, inserts, and deletions based on whether a specified condition is met for the records being merged. The "WHEN MATCHED" clause specifies what action should be taken when a record from the target table matches a record from the source table based on the defined criteria. This is essential for streamlining and optimizing data updates or changes, ensuring that the right records are handled appropriately. The other options do not accurately represent the syntax used in a merge operation. Utilizing "IF EXISTS," "IF MATCHED," or "ON MATCH FOUND" does not align with the standard SQL syntax for merging records or checking for matches. Thus, "WHEN MATCHED" is the correct and necessary command to facilitate the logic of a merge operation effectively.

8. What is a key feature of Delta Lake in terms of data management?

A. In-memory data processing

B. Time travel capabilities

C. Complex query optimization

D. Secure data sharing

A key feature of Delta Lake is its time travel capabilities. This allows users to access and revert to previous versions of their data within the Delta Lake tables. Time travel is made possible by Delta Lake's transaction log, which records changes made to the data, effectively enabling users to query historical states. By using this functionality, data engineering teams can analyze and understand data changes over time, rollback errors, and conduct audits effortlessly. The ability to perform time travel not only enhances data management practices but also supports operational tasks such as debugging and compliance with data governance policies. This feature is particularly valuable in scenarios where data is frequently updated or deleted, providing a robust mechanism to maintain data integrity and accessibility across different versions.

9. What distinguishes DEEP CLONE from SHALLOW CLONE in SQL?

A. DEEP CLONE only copies transaction logs

B. SHALLOW CLONE fully copies data and metadata

C. DEEP CLONE does not incrementally sync changes

D. DEEP CLONE fully copies both data and metadata

DEEP CLONE stands out from SHALLOW CLONE primarily due to its capability to fully duplicate both data and metadata. When a DEEP CLONE is performed, it creates a new table or data object that is a complete and independent copy of the original, ensuring that not only the data itself is replicated, but also all of the metadata associated with that data—including its schema, permissions, and other relevant characteristics. This makes DEEP CLONE a robust solution for scenarios where a true, standalone copy of a dataset is required for analysis or further processing. In contrast, SHALLOW CLONE is typically more limited, as it usually references the existing data without making a full copy. This distinction allows for more efficient storage and potential performance improvements, but doesn't allow the flexibility that a DEEP CLONE offers in situations where a fully independent version of the dataset is necessary. This feature of DEEP CLONE is essential in contexts requiring data isolation or when you need to ensure that subsequent changes to the original data do not impact the cloned version.

EXAMPLE

10. What is the purpose of the Input_file_name() function in SQL?

A. To get the name of the current table

B. To retrieve the name of the file being processed

C. To filter data based on file type

D. To check the size of the input data

The Input_file_name() function in SQL plays a crucial role in data processing, particularly in environments like Databricks where data is often ingested from various files. The primary purpose of this function is to retrieve the name of the file that is currently being processed. This capability is particularly useful when working with large datasets that are distributed across multiple files, as it allows users to track the origin of the data they are analyzing. By obtaining the filename, users can implement various data processing strategies, such as filtering data, troubleshooting, or conducting audits based on the source files. This information can also be vital for debugging purposes, as it enables data engineers to trace back the data to its original file for verification or correction. In contrast, the other options do not accurately reflect the function's purpose. For instance, there is no inherent functionality in Input_file_name() to determine table names, filter data types, or check input data sizes directly. These tasks require different SQL functions or methods, indicating that the focus of Input_file_name() is strictly on file identification during processing workflows.

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://dataengineeringassociate-databricks.examzify.com

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

SAMPLE