Snowflake Data Engineer Practice Exam (Sample)

Study Guide

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. **The amount of metadata stored in the RECORD_METADATA column is configurable using optional Kafka configuration properties.**

   A. True

   B. False

   C. Not configurable

   D. Not applicable

2. **Which stage types does Snowpipe support for loading data?**

   A. Named internal or external stages

   B. Only named internal stages

   C. Only external stages

   D. Table stages

3. **Which statement accurately describes the scope of Snowflake's Search Optimization Service?**

   A. It is a database-level feature.

   B. It is a per-column feature.

   C. It is a table-level property and applies to all columns with supported data types.

   D. It applies only to string columns.

4. **Task_history returns both completed and running tasks.**

   A. TRUE

   B. FALSE

   C. Only completed

   D. Only running

5. **Version 2.2.0+ of the Snowflake Spark Connector uses a default staging for data exchange.**

   A. Temporary Snowflake internal stage

   B. Amazon S3 bucket

   C. Azure Blob storage container

   D. Local storage

6. To filter by city inside a VARIANT column when querying weather data, which syntax is used?

A. V:city::STRING = 'HARTFORD'.

B. city:V = 'HARTFORD'.

C. TEMPVAL:city = 'HARTFORD'.

D. CITY:city = 'HARTFORD'.

7. Which technique leverages performance efficiencies by enabling large and complex Spark logical plans to be processed in Snowflake, thus using Snowflake to do most of the actual work?

A. Query Plan Pushdown

B. Query Pushdown

C. Query Optimization

D. Pushdown is not possible in Spark UDFs

8. If a data transfer is expected to take 48 hours, which Spark connector mode is recommended?

A. Internal Transfer

B. External Transfer

C. Both can be used

D. Neither can be used

9. What is the default state of query pushdown in the Snowflake Spark Connector?

A. Enabled

B. Disabled

C. Auto

D. Not supported

10. What is an API integration in Snowflake?

A. It provides an interface between Snowflake and third-party services.

B. It stores data in external stages.

C. It is used to manage user authentication in Snowflake.

D. It is a type of data warehouse.

Answers

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

Explanations

1. The amount of metadata stored in the RECORD_METADATA column is configurable using optional Kafka configuration properties.

A. True

B. False

C. Not configurable

D. Not applicable

The amount of metadata stored in the RECORD_METADATA column can be adjusted through the Kafka configuration properties used by the Snowflake Kafka integration. This means you can tailor how much context is captured for each record—such as topic, partition, offset, timestamp, key, headers, and other fields—by enabling or limiting specific metadata via those optional properties. Because Snowflake provides these configuration knobs, the statement is true: you can configure how much metadata ends up in RECORD_METADATA rather than it being fixed. If someone suggests it isn't configurable or isn't applicable, that would contradict the documented capability of the connector to control metadata granularity through its Kafka settings.

2. Which stage types does Snowpipe support for loading data?

A. Named internal or external stages

B. Only named internal stages

C. Only external stages

D. Table stages

Snowpipe loads data from stages, which are locations where files are prepared before loading into Snowflake. These stages can be internal (stored inside Snowflake) or external (pointing to cloud storage like S3, Azure, or GCS). Snowpipe is designed to work with either type, using a pipe to automatically ingest new files as they appear in the stage. Because both named internal stages and named external stages are valid sources for Snowpipe, the correct statement is that Snowpipe supports loading data from both. The other options limit the sources or refer to a concept that isn't used as a separate stage type for Snowpipe.

3. Which statement accurately describes the scope of Snowflake's Search Optimization Service?

A. It is a database-level feature.

B. It is a per-column feature.

C. It is a table-level property and applies to all columns with supported data types.

D. It applies only to string columns.

The main idea being tested is where Snowflake's Search Optimization Service is applied and how broadly it affects data. This service is configured at the table level, meaning you enable it for a specific table and it governs how that table's data is stored and scanned. It isn't a database-wide setting, and it isn't limited to a single column. Instead, once enabled on a table, the optimization applies across all columns in that table that have supported data types. That's why the statement describing it as a table-level property that applies to all columns with supported data types is the best description. The feature accelerates queries by speeding up how the table's data is pruned and scanned, regardless of which non-string or string columns are involved, as long as they are of a supported type.

4. Task_history returns both completed and running tasks.

A. TRUE

B. FALSE

C. Only completed

D. Only running

Task history is a log of every execution a task has run, including runs that are still in progress. When you query it, you get entries for each run with fields like start time, end time, and status. If a run is currently happening, end time is typically NULL and the status shows RUNNING. Once that run finishes, its status updates to SUCCESS or FAILED and the end time is filled in. Because it captures both completed and in-flight executions, task history can return both completed and running tasks, depending on when you query and how many results you pull.

5. Version 2.2.0+ of the Snowflake Spark Connector uses a default staging for data exchange.

A. Temporary Snowflake internal stage

B. Amazon S3 bucket

C. Azure Blob storage container

D. Local storage

The concept being tested is how data is staged when moving data from Spark into Snowflake using the Spark Connector. In version 2.2.0 and newer, the connector uses a default staging area that is a temporary Snowflake internal stage. This means Snowflake itself provides and manages the storage location for the data during the transfer, without requiring you to configure external cloud storage or local disks. It streamlines the process, avoids extra credential setup, and the staged data is handled by Snowflake and cleaned up after loading. Using an external storage option like S3 or Azure would require explicit stage configuration and credentials, and staging to local storage isn't feasible for distributed Spark workloads that need to share data across workers.

6. To filter by city inside a VARIANT column when querying weather data, which syntax is used?

A. V:city::STRING = 'HARTFORD'.

B. city:V = 'HARTFORD'.

C. TEMPVAL:city = 'HARTFORD'.

D. CITY:city = 'HARTFORD'.

Accessing data inside a VARIANT uses the path notation: column:field. When the VARIANT column contains an object with a city field, you pull that value with V:city and convert it to the expected type before comparing. So filtering with V:city::STRING = 'HARTFORD' reads the city value from the VARIANT, casts it to STRING, and checks equality to 'HARTFORD'. The other forms try to reference a path in the wrong order or structure (city:V, TEMPVAL:city, CITY:city), which Snowflake won't interpret as the city field inside the VARIANT. Casting to STRING is a safe, explicit way to ensure a proper comparison.

7. Which technique leverages performance efficiencies by enabling large and complex Spark logical plans to be processed in Snowflake, thus using Snowflake to do most of the actual work?

A. Query Plan Pushdown

B. Query Pushdown

C. Query Optimization

D. Pushdown is not possible in Spark UDFs

The technique is Query Pushdown. It works by offloading as much of the Spark plan as possible to Snowflake so Snowflake handles the heavy lifting—filters, projections, joins, and even some aggregations—while Spark just orchestrates and receives the results. This leverages Snowflake's optimized execution engine, reduces data movement across the network, and lowers the workload on Spark workers, leading to faster overall performance for large and complex plans. Other terms aren't as precise for this scenario: "Query Plan Pushdown" isn't the standard naming, "Query Optimization" is a broader process that can occur inside either engine and doesn't specifically describe pushing work to Snowflake, and the notion that pushdown isn't possible with Spark UDFs is not a defining characteristic of this technique.

8. If a data transfer is expected to take 48 hours, which Spark connector mode is recommended?

A. Internal Transfer

B. External Transfer

C. Both can be used

D. Neither can be used

For long-running data moves, external transfer is the best fit. This mode uses a cloud storage staging area to bulk-load data into Snowflake, which lets the transfer run in parallel and decouples the Spark process from Snowflake ingestion. It reduces memory and resource pressure on the Spark cluster and avoids prolonged streaming of data through Spark, making multi-day transfers more reliable and scalable. Internal transfer, by contrast, streams data directly between Spark and Snowflake, which can become a bottleneck and is more sensitive to driver/executor memory and network constraints. So for a 48-hour transfer, External Transfer is the recommended option.

9. What is the default state of query pushdown in the Snowflake Spark Connector?

A. Enabled

B. Disabled

C. Auto

D. Not supported

Query pushdown means having the Snowflake engine handle as much of the work as possible—such as filters, projections, and some aggregations—so Spark transfers less data and Snowflake does the heavy lifting. In the Snowflake Spark Connector, this capability exists but is off by default. Keeping pushdown disabled avoids potential issues when Spark expressions can't be translated into valid Snowflake SQL, ensuring correctness across a wide range of queries. If you want to leverage Snowflake to do more work, you can enable pushdown with the appropriate configuration option, allowing the connector to translate supported operations into Snowflake SQL and push them down. Not supported would imply there's no pushdown capability at all, which isn't the case, and Auto would suggest a per-query automatic decision, which isn't how the default behavior is described.

10. What is an API integration in Snowflake?

A. It provides an interface between Snowflake and third-party services.

B. It stores data in external stages.

C. It is used to manage user authentication in Snowflake.

D. It is a type of data warehouse.

An API integration in Snowflake is a configuration that creates a secure interface between Snowflake and external services, enabling programmatic access and communication between Snowflake and third-party applications. It defines how Snowflake authenticates to those services and which endpoints can be used, so external tools can interact with Snowflake or Snowflake can call out to other systems without embedding credentials in code. That's why this option best captures the idea of an API integration. The other ideas describe different concepts: storing data in external stages points to data in external storage, not an integration interface; managing user authentication is handled by security integrations or authentication methods rather than API integrations; and Snowflake being a data warehouse is the overall system, not the feature described.

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://snowflakedataengineer.examzify.com

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