ARA-C01 Q&As

SnowPro Advanced: Architect Certification Exam

Pass Snowflake ARA-C01 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.geekcert.com/ara-c01.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Snowflake Official Exam Center

QUESTION 1

How can an Architect enable optimal clustering to enhance performance for different access paths on a given table?

A. Create multiple clustering keys for a table.

B. Create multiple materialized views with different cluster keys.

C. Create super projections that will automatically create clustering.

D. Create a clustering key that contains all columns used in the access paths.

Correct Answer: B

Explanation: According to the SnowPro Advanced: Architect documents and learning resources, the best way to enable optimal clustering to enhance performance for different access paths on a given table is to create multiple materialized views with different cluster keys. A materialized view is a pre-computed result set that is derived from a query on one or more base tables. A materialized view can be clustered by specifying a clustering key, which is a subset of columns or expressions that determines how the data in the materialized view is co-located in micro-partitions. By creating multiple materialized views with different cluster keys, an Architect can optimize the performance of queries that use different access paths on the same base table. For example, if a base table has columns A, B, C, and D, and there are queries that filter on A and B, or on C and D, or on A and C, the Architect can create three materialized views, each with a different cluster key: (A, B), (C, D), and (A, C). This way, each query can leverage the optimal clustering of the corresponding materialized view and achieve faster scan efficiency and better compression. References: Snowflake Documentation: Materialized Views Snowflake Learning: Materialized Views https://www.snowflake.com/blog/using-materialized-views-to-solve-multi-clustering- performance-problems/

QUESTION 2

Consider the following COPY command which is loading data with CSV format into a Snowflake table from an internal stage through a data transformation query.

```
copy into home_sales(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$6, t.$7 from @mystage/sales.csv.qz t)
file_format –
(
format_name = mycsvformat
empty_field_as_null = true
field_optionally_enclosed_by = '"'
)
validation_mode – return_all_errors
;
```

This command results in the following error:

SQL compilation error: invalid parameter \\'validation_mode\\'

Assuming the syntax is correct, what is the cause of this error?

A. The VALIDATION_MODE parameter supports COPY statements that load data from external stages only.

B. The VALIDATION_MODE parameter does not support COPY statements with CSV file formats.

C. The VALIDATION_MODE parameter does not support COPY statements that transform data during a load.

D. The value return_all_errors of the option VALIDATION_MODE is causing a compilation error.

Correct Answer: C

The VALIDATION_MODE parameter is used to specify the behavior of the COPY statement when loading data into a table. It is used to specify whether the COPY statement should return an error if any of the rows in the file are invalid or if it should continue loading the valid rows. The VALIDATION_MODE parameter is only supported for COPY statements that load data from external stages1. The query in the question uses a data transformation query to load data from an internal stage. A data transformation query is a query that transforms the data during the load process, such as parsing JSON or XML data, applying functions, or joining with other tables2. According to the documentation, VALIDATION_MODE does not support COPY statements that transform data during a load. If the parameter is specified, the COPY statement returns an error1. Therefore, option C is the correct answer. References: : COPY INTO : Transforming Data During a Load

QUESTION 3

When using the Snowflake Connector for Kafka, what data formats are supported for the messages? (Choose two.)

A. CSV

B. XML

C. Avro

D. JSON

E. Parquet

Correct Answer: CD

Explanation: The data formats that are supported for the messages when using the Snowflake Connector for Kafka are Avro and JSON. These are the two formats that the connector can parse and convert into Snowflake table rows. The connector supports both schemaless and schematized JSON, as well as Avro with or without a schema registry1. The other options are incorrect because they are not supported data formats for the messages. CSV, XML, and Parquet are not formats that the connector can parse and convert into Snowflake table rows. If the messages are in these formats, the connector will load them as VARIANT data type and store them as raw strings in the table2. References: Snowflake Connector for Kafka | Snowflake Documentation, Loading Protobuf Data using the Snowflake Connector for Kafka | Snowflake Documentation

QUESTION 4

Company A would like to share data in Snowflake with Company B. Company B is not on the same cloud platform as Company A.

What is required to allow data sharing between these two companies?

A. Create a pipeline to write shared data to a cloud storage location in the target cloud provider.

B. Ensure that all views are persisted, as views cannot be shared across cloud platforms.

C. Setup data replication to the region and cloud platform where the consumer resides.

D. Company A and Company B must agree to use a single cloud platform: Data sharing is only possible if the companies share the same cloud provider.

Correct Answer: C

Explanation: According to the SnowPro Advanced: Architect documents and learning resources, the requirement to allow data sharing between two companies that are not on the same cloud platform is to set up data replication to the region and cloud platform where the consumer resides. Data replication is a feature of Snowflake that enables copying databases across accounts in different regions and cloud platforms. Data replication allows data providers to securely share data with data consumers across different regions and cloud platforms by creating a replica database in the consumer\\'s account. The replica database is read-only and automatically synchronized with the primary database in the provider\\'s account. Data replication is useful for scenarios where data sharing is not possible or desirable due to latency, compliance, or security reasons1. The other options are incorrect because they are not required or feasible to allow data sharing between two companies that are not on the same cloud platform. Option A is incorrect because creating a pipeline to write shared data to a cloud storage location in the target cloud provider is not a secure or efficient way of sharing data. It would require additional steps to load the data from the cloud storage to the consumer\\'s account, and it would not leverage the benefits of Snowflake\\'s data sharing features. Option B is incorrect because ensuring that all views are persisted is not relevant for data sharing across cloud platforms. Views can be shared across cloud platforms as long as they reference objects in the same database. Persisting views is an option to improve the performance of querying views, but it is not required for data sharing2. Option D is incorrect because Company A and Company B do not need to agree to use a single cloud platform. Data sharing is possible across different cloud platforms using data replication or other methods, such as listings or auto- fulfillment3. References: ReplicatingDatabases Across Multiple Accounts | Snowflake Documentation, Persisting Views | Snowflake Documentation, Sharing Data Across Regions and Cloud Platforms | Snowflake Documentation

QUESTION 5

The Data Engineering team at a large manufacturing company needs to engineer data coming from many sources to support a wide variety of use cases and data consumer requirements which include:

1) Finance and Vendor Management team members who require reporting and visualization

2) Data Science team members who require access to raw data for ML model development

3) Sales team members who require engineered and protected data for data monetization

What Snowflake data modeling approaches will meet these requirements? (Choose two.)

A. Consolidate data in the company\\'s data lake and use EXTERNAL TABLES.

B. Create a raw database for landing and persisting raw data entering the data pipelines.

C. Create a set of profile-specific databases that aligns data with usage patterns.

D. Create a single star schema in a single database to support all consumers\\' requirements.

E. Create a Data Vault as the sole data pipeline endpoint and have all consumers directly access the Vault.

Correct Answer: BC

Explanation: These two approaches are recommended by Snowflake for data modeling in a data lake scenario. Creating a raw database allows the data engineering team to ingest data from various sources without any transformation or cleansing, preserving the original data quality and format. This enables the data science team to access the raw data for

ML model development. Creating a set of profile-specific databases allows the data engineering team to apply different transformations and optimizations for different use cases and data consumer requirements. For example, the finance and vendor management team can access a dimensional database that supports reporting and visualization, while the sales team can access a secure database that supports data monetization. References: Snowflake Data Lake Architecture | Snowflake Documentation Snowflake Data Lake Best Practices | Snowflake Documentation