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.pass4itsure.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

A company is storing large numbers of small JSON files (ranging from 1-4 bytes) that are received from IoT devices and sent to a cloud provider. In any given hour, 100,000 files are added to the cloud provider.

What is the MOST cost-effective way to bring this data into a Snowflake table?

A. An external table

B. A pipe

C. A stream

D. A copy command at regular intervals

Correct Answer: B

A pipe is a Snowflake object that continuously loads data from files in a stage (internal or external) into a table. A pipe can be configured to use auto-ingest, which means that Snowflake automatically detects new or modified files in the stage and loads them into the table without any manual intervention1. A pipe is the most cost-effective way to bring large numbers of small JSON files into a Snowflake table, because it minimizes the number of COPY commands executed and the number of micro-partitions created. A pipe can use file aggregation, which means that it can combine multiple small files into a single larger file before loading them into the table. This reduces the load time and the storage cost of the data2. An external table is a Snowflake object that references data files stored in an external location, such as Amazon S3, Google Cloud Storage, or Microsoft Azure Blob Storage. An external table does not store the data in Snowflake, but only provides a view of the data for querying. An external table is not a cost-effective way to bring data into a Snowflake table, because it does not support file aggregation, and it requires additional network bandwidth and compute resources to

query the external data3.

A stream is a Snowflake object that records the history of changes (inserts, updates, and deletes) made to a table. A stream can be used to consume the changes from a table and apply them to another table or a task. A stream is not a way

to bring data into a Snowflake table, but a way to process the data after it is loaded into a table4.

A copy command is a Snowflake command that loads data from files in a stage into a table. A copy command can be executed manually or scheduled using a task. A copy command is not a cost-effective way to bring large numbers of small

JSON files into a Snowflake table, because it does not support file aggregation, and it may create many micro-partitions that increase the storage cost of the data5.

References: : Pipes : Loading Data Using Snowpipe : External Tables : Streams : COPY INTO

QUESTION 2

Which feature provides the capability to define an alternate cluster key for a table with an existing cluster key?

A. External table

B. Materialized view

C. Search optimization

D. Result cache

Correct Answer: B

Explanation: A materialized view is a feature that provides the capability to define an alternate cluster key for a table with an existing cluster key. A materialized view is a pre- computed result set that is stored in Snowflake and can be queried like a regular table. A materialized view can have a different cluster key than the base table, which can improve the performance and efficiency of queries on the materialized view. A materialized view can also support aggregations, joins, and filters on the base table data. A materialized view is automatically refreshed when the underlying data in the base table changes, as long as the AUTO_REFRESH parameter is set to true1. References: Materialized Views | Snowflake Documentation

QUESTION 3

A company has a table with that has corrupted data, named Data. The company wants to recover the data as it was 5 minutes ago using cloning and Time Travel.

What command will accomplish this?

A. CREATE CLONE TABLE Recover_Data FROM Data AT(OFFSET => -60*5);

B. CREATE CLONE Recover_Data FROM Data AT(OFFSET => -60*5);

C. CREATE TABLE Recover_Data CLONE Data AT(OFFSET => -60*5);

D. CREATE TABLE Recover Data CLONE Data AT(TIME => -60*5);

Correct Answer: C

Explanation: This is the correct command to create a clone of the table Data as it was 5 minutes ago using cloning and Time Travel. Cloning is a feature that allows creating a copy of a database, schema, table, or view without duplicating the

data or metadata. Time Travel is a feature that enables accessing historical data (i.e. data that has been changed or deleted) at any point within a defined period. To create a clone of a table at a point in time in the past, the syntax is:

CREATE TABLE CLONE AT (OFFSET =>);

The OFFSET parameter specifies the time difference in seconds from the present time. A negative value indicates a point in the past. For example, -60*5 means 5 minutes ago. Alternatively, the TIMESTAMP parameter can be used to specify

an exact timestamp in the past. The clone will contain the data as it existed in the source table at the specified point in time12.

References:

Snowflake Documentation: Cloning Objects

Snowflake Documentation: Cloning Objects at a Point in Time in the Past

QUESTION 4

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 5

A company has an inbound share set up with eight tables and five secure views. The company plans to make the share part of its production data pipelines.

Which actions can the company take with the inbound share? (Choose two.)

A. Clone a table from a share.

B. Grant modify permissions on the share.

C. Create a table from the shared database.

D. Create additional views inside the shared database.

E. Create a table stream on the shared table.

Correct Answer: AD

Explanation: These two actions are possible with an inbound share, according to the Snowflake documentation and the web search results. An inbound share is a share that is created by another Snowflake account (the provider) and imported into your account (the consumer). An inbound share allows you to access the data shared by the provider, but not to modify or delete it. However, you can perform some actions with the inbound share, such as: Clone a table from a share. You can create a copy of a table from an inbound share using the CREATE TABLE ... CLONE statement. The clone will contain the same data and metadata as the original table, but it will be independent of the share. You can modify or delete the clone as you wish, but it will not reflect any changes made to the original table by the provider1. Create additional views inside the shared database. You can create views on the tables or views from an inbound share using the CREATE VIEW statement. The views will be stored in the shared database, but they will be owned by your account. You can query the views as you would query any other view in your account, but you cannot modify or delete the underlying objects from the share2. The other actions listed are not possible with an inbound share, because they

would require modifying the share or the shared objects, which are read-only for the consumer. You cannot grant modify permissions on the share, create a table from the shared database, or create a table stream on the shared table34. References: Cloning Objects from a Share | Snowflake Documentation Creating Views on Shared Data | Snowflake Documentation Importing Data from a Share | Snowflake Documentation Streams on Shared Tables | Snowflake Documentation