EXAMTOPICS

-Expert Verified, Online, **Free**.



CERTIFICATION TEST

-CertificationTest.net - Cheap & Quality Resources With Best Support

Question #1 *Topic 1*

A table is loaded using Snowpipe and truncated afterwards. Later, a Data Engineer finds that the table needs to be reloaded, but the metadata of the pipe will not allow the same files to be loaded again.

How can this issue be solved using the LEAST amount of operational overhead?

A. Wait until the metadata expires and then reload the file using Snowpipe.

B. Modify the file by adding a blank row to the bottom and re-stage the file.

C. Set the FORCE=TRUE option in the Snowpipe COPY INTO command.

D. Recreate the pipe by using the CREATE OR REPLACE PIPE command.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #2 *Topic 1*

A company using Snowflake works with a third party that provides economic data. The third-party provider recently moved their data to Snowflake.

How can the company access the third-party data in Snowflake with the LEAST amount of operational overhead?

A. Have the data provider enable database replication so the company can create a replica of the provider's database.

B. Have the data provider create a share object and add the company's Snowflake account to the share.

C. Have the data provider create a listing on the Snowflake Marketplace so the company can request access to the data.

D. Have the data provider create a new user and role specifically for the company so the company can log into the data provider's Snowflake account to access the data.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #3 *Topic 1*

A Data Engineer made modifications to staged files and re-staged them. A pipe is running but the changed data is not appearing in the table.

What step should the Engineer take in order to see the changed data without duplicating data?

A. Run the CREATE or REPLACE pipe syntax.

B. Re-run the pipe by using the RELOAD PIPE syntax.

C. Re-run the pipe by using the ONLY_CHANGES syntax.

D. Create new files with the changed data.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #4 *Topic 1*

How can a Data Engineer return all the errors encountered during a previous file upload that used the COPY command?

A. Query the VALIDATE information schema view.

B. Call the VALIDATE table function.

C. Call the VALIDATE_PIPE_LOAD table function.

D. Query the VALIDATE account usage schema view.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #5 *Topic 1*

A company called IOT Corporation has subsidiary companies in Germany and Japan.

The German subsidiary has a Snowflake account called IOTGER in AWS region EU (Frankfurt) (Region ID: eu-central-1).

The Japanese subsidiary has a Snowflake account called IOTJPN in Azure region Japan East (Tokyo) (Region ID: japaneast).

The subsidiaries are totally independent of one another, and their Snowflake accounts belong to different Snowflake organizations.

A Data Engineer needs to share data in a database called IOT PROD from the German account to the Japanese account.

What steps need to be taken by the German subsidiary so that the Japanese subsidiary can use the shared data?

A. Create share S1 into IOTGER. Add the objects to be shared from IOT_PROD in IOTGER to the share S1, then add account IOTJPN to the share S1.

B. Replicate database IOT_PROD from IOTGER to IOTJPN with the same database name. Create share S1 into IOTJPN, add the objects to be shared from IOTPROD in IOTJPN to the share S1, and add account IOTJPN to the share S1.

C. Create a clone of the database IOT_PROD into IOTJPN. Create secure views into the cloned database that read data from the objects to be shared. Create share S1 into IOTJPN. Add the secure views to the share S1, and add account IOTJPN to the share S1.

D. Create an additional Snowflake account IOTGER2 into region Japan East. Replicate database IOT_PROD from IOTGER to IOTGER2 with the same database name. Create share S1 into IOTGER2, and add the objects to be shared from IOT_PROD in IOTGER2 to the share S1. Then add account IOTJPN to the share S1.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #6 *Topic 1*

When loading data using the Snowflake Connector for Kafka, the Kafka log file contained this information:

org.apache.kafka.clients.consumer.CommitFailedException: Commit cannot be completed since the group has already rebalanced and assigned the partitions to another member

What combination of tasks should be used to address this issue? (Choose two.)

A. Decrease the number of records loaded with each operation to 50.

B. Disable automatic offset migration by setting it to = FALSE.

C. Increase the execution timeout to 900000.

D. Reduce the session.timeout.ms setting to enable faster rebalancing.

E. Add more partitions to the Kafka topic to prevent rebalancing.

**Suggested Answer:** *A, C*

Currently there are no comments in this discussion, be the first to comment!

Question #7 *Topic 1*

If data was loaded into a table that had no row limitations, which object can be used to troubleshoot the data loading history?

A. LOAD_HISTORY view

B. COPY_HISTORY table function

C. STAGE_DIRECTORY_FILE_REGISTRATION_HISTORY view

D. SYSTEM$PIPE_STATUS function

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #8 *Topic 1*

A Data Engineer is working on a continuous data pipeline which receives data from Amazon Kinesis Data Firehose and loads the data into a staging table which will later be used in the data transformation process. The average file size is 300-500 MB.

How can the Engineer ensure that Snowpipe is performant while MINIMIZING costs?

A. Increase the size of the virtual warehouse used by Snowpipe.

B. Split the files before loading them and set the SIZE_LIMIT option to 250 MB.

C. Change the file compression size and increase the frequency of the Snowpipe loads.

D. Decrease the buffer size to trigger delivery of files sized between 100 to 250 MB in Kinesis Data Firehose.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #9 *Topic 1*

A Data Engineer is implementing a near real-time ingestion pipeline to load data into Snowflake using the Snowflake Kafka connector. There will be three Kafka topics created.

Which Snowflake objects are created automatically when the Kafka connector starts? (Choose three.)

A. Tables

B. Tasks

C. Pipes

D. Internal stages

E. External stages

F. Materialized views

**Suggested Answer:** *A, C, E*

Currently there are no comments in this discussion, be the first to comment!

Question #10 *Topic 1*

Which file format option is used to escape the '\' character while unloading data to a stage from a table if the FIELD_DELIMITER = ','?

A. ESCAPE_UNENCLOSED_FIELD=NONE

B. ESCAPE_UNENCLOSED_FIELD='\\'

C. ESCAPE_UNENCLOSED_FIELD='\'

D. ESCAPE_UNENCLOSED_FIELD=','

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #11 *Topic 1*

A Data Engineer needs to create a stage that will be used to ingest files from a Customer Relationship Management (CRM) system.

These are the requirements for the stage:

1. Use of an internal named stage with different folders that will be the repositories for the files

2. The files must be encrypted by the cloud service hosting the Snowflake account when they arrive on the stage

3. The files will be queried later using pre-signed URLs

4. Ability to query a metadata list of all the unstructured files on the stage

Which statement will meet these requirements?

A.

B.

C.

D.

```
CREATE or REPLACE STAGE crm_files
ENCRYPTION = (TYPE = 'SNOWFLAKE_SSE')
DIRECTORY = ( ENABLE = TRUE REFRESH_ON_CREATE = TRUE)
FILE_FORMAT = (TYPE = JSON );
```

```
CREATE or REPLACE STAGE crm_files
ENCRYPTION = (TYPE = 'SNOWFLAKE_FULL' )
DIRECTORY = ( ENABLE = TRUE REFRESH_ON_CREATE = TRUE)
FILE_FORMAT = ( TYPE = CSV );
```

```
CREATE or REPLACE STAGE crm_files
URL = 's3://load/crm/files/'
STORAGE_INTEGRATION = crm_int
ENCRYPTION = (TYPE = 'AWS_SSE_S3' )
DIRECTORY = ( ENABLE = TRUE REFRESH_ON_CREATE = TRUE AUTO_REFRESH = TRUE)
FILE_FORMAT = ( TYPE = CSV );
```

```
CREATE or REPLACE STAGE crm_files
ENCRYPTION = (TYPE = 'SNOWFLAKE_FULL' )
DIRECTORY = ( ENABLE = FALSE )
FILE_FORMAT = (TYPE = JSON )
COMPRESSION = AUTO;
```

**Suggested Answer:**
Correct choice: A. An internal named stage (no URL) with ENCRYPTION = (TYPE = 'SNOWFLAKE_SSE') gives server-side encryption by the cloud service on arrival, which Snowflake requires when files will be accessed via pre-signed URLs; DIRECTORY = (ENABLE = TRUE ...) provides the directory table to query metadata of the unstructured files. B uses SNOWFLAKE_FULL (client-side), C is an external S3 stage, and D disables the directory table.

Currently there are no comments in this discussion, be the first to comment!

Question #12 *Topic 1*

Which command will load all data successfully into the table named finance_dept_location?

A.

```
copy into finance_dept_location(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$3, t.$4 from @finstage/finance.csv.gz t)
file_format = (format_name = fincsvformat);
```

B.

```
copy into finance_dept_location(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$3, t.$4 from @finstage/finance.csv.gz t)
file_format = (format_name = fincsvformat)
validation_mode=return_all_errors;
```

C.

```
copy into finance_dept_location(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$3, t.$4 from @finstage/finance.csv.gz t limit 100)
file_format = (format_name = fincsvformat);
```

D.

```
copy into finance_dept_location(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$3, t.$4 from @finstage/finance.csv.gz t where t.$1 like '%austin%')
file_format = (format_name = fincsvformat);
```

**Suggested Answer:**
Correct choice: A. It loads every row of finance.csv.gz with a simple column-reordering transformation. B adds VALIDATION_MODE, which only validates (and is not supported with transformations), so nothing is loaded; C limits the load to 100 rows; D filters rows with a WHERE clause, so not all data is loaded (and filtering is not supported in COPY transformations).

Currently there are no comments in this discussion, be the first to comment!

Question #13 *Topic 1*

A Data Engineer is designing a solution that will send report files generated and stored in a Snowflake stage to a customer, using email. The files can only be available for download for 60 minutes as new files will be generated and the old ones become obsolete.

What is the recommended way to meet these requirements?

A. Share data using a scoped URL and the SYSTEM$SEND_EMAIL function.

B. Share data using pre-signed URL and the SYSTEM$SEND_EMAIL function.

C. Share data using a stage file URL and the SYSTEM$SEND_EMAIL function.

D. Share data using a scoped URL and write a custom Python script to send the email.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #14 *Topic 1*

A Data Engineer is working to optimize the performance of end-users' queries about automobile shipment data for a portal web application that generates dashboards for different clients.

An internal team queries the information using order_date as a WHERE condition; the ingestion and natural clustering are performed in this column. However, the application recovers data using the client_id, warehouse, product_id, and destination_city. The Engineer needs to create materialized views with different cluster keys, and must choose two columns combined with client_id to optimize the first cluster key.

Which statements will provide all the information needed to select the MOST effective combination of columns to cluster? (Choose two.)

A. SELECT SYSTEM$CLUSTERING_DEPTH('shipments');

B. SELECT SYSTEM$CLUSTERING_DEPTH('shipments', '(client_id, warehouse, product_id)');

C. SELECT SYSTEM$CLUSTERING_DEPTH('shipments', ' (client_id, destination_city, product_id)')

D. SELECT SYSTEM$CLUSTERING_DEPTH('shipments', '(warehouse, product_id, destination_city)')

E. SELECT SYSTEM$CLUSTERING_INFORMATION('shipments', 3);

**Suggested Answer:** *B, C*

Currently there are no comments in this discussion, be the first to comment!

Question #15 *Topic 1*

A user ran a one-time AI_SUMMARIZE_AGG query on a large log table using a Cortex AI model. The query consumed more than 1,000 credits, even though it ran on a size Small virtual warehouse.

A Data Engineer has been asked to implement a Snowflake-native, automated alert that triggers whenever a single Cortex AI SQL query consumes 300 credits, so that the query text and model choice can be reviewed.

How should these requirements be met?

A. Create a scheduled task that queries SNOWFLAKE.ACCOUNT_USAGE.CORTEX_AISQL_USAGE_HISTORY for recent rows where CREDITS_USED > 300. Join those rows to

SNOWFLAKE.ACCOUNT_USAGE.QUERY_HISTORY to capture the SQL text, and use a notification integration in the task body to send an alert with the query and model details.

B. Attach a resource monitor with a 300-credit limit to the virtual warehouse used to run the Cortex AI SQL. Configure the monitor to send a notification when the limit is reached, and use the resource monitor alerts to identify any Cortex query that exceeds 300 credits.

C. Create a scheduled task that checks SNOWFLAKE.ACCOUNT_USAGE.METERING_DAILY_HISTORY for SERVICE_TYPE = 'AI_SERVICES ' and TOTAL_CREDITS > 300 on the current date, and send a notification whenever the daily AI_SERVICES usage exceeds the threshold so the team can review the queries that ran that day.

D. Set a low STATEMENT_TIMEOUT_IN_SECONDS value for sessions that use Cortex AI SQL, based on how long a typical query would take to reach 300 credits, and use the statement timeout threshold to prevent any single Cortex query from exceeding the 300-credit threshold.

**Suggested Answer:** *A*

Currently there are no comments in this discussion, be the first to comment!

Question #16 *Topic 1*

How can a clustering key definition be used to reduce the number of distinct values in a column that has high cardinality?

A. Define the key as an expression on the column.

B. Define the key by using the CONCAT function on the column.

C. Define the key by using the SCAN function on the column.

D. Define the key by using the TRIM function on the column.

**Suggested Answer:** *A*

Currently there are no comments in this discussion, be the first to comment!

Question #17 *Topic 1*

A Data Engineer ran this command:

ALTER WAREHOUSE my_wh SET

ENABLE_QUERY_ACCELERATION = true

QUERY_ACCELERATION_MAX_SCALE_FACTOR

The warehouse is now consuming more credits than expected.

What could be causing this to occur, and what action will reduce the credit consumption?

A. The scale factor is switched off (set to 0). A value must be set to initiate the query acceleration service.

B. The query acceleration service is being applied to all queries in my_wh. Configure load sharing with a virtual warehouse that is not using the query acceleration service.

C. The queries running against my_wh are consuming as many resources as necessary. Set a scale factor to reduce credit consumption.

D. The query acceleration service costs are based on the entire time the warehouse is active. Identify additional queries to run against my_wh to reduce credit consumption.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #18 *Topic 1*

Assuming that the session parameter USE_CACHED_RESULT is set to false, what are characteristics of Snowflake virtual warehouses in terms of the use of Snowpark?

A. Creating a DataFrame from a table will start a virtual warehouse.

B. Creating a DataFrame from a staged file with the read() method will start a virtual warehouse.

C. Transforming a DataFrame with methods like replace() will start a virtual warehouse.

D. Calling a Snowpark stored procedure to query the database with session.call() will start a virtual warehouse.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #19 *Topic 1*

A Data Engineer needs to monitor data usage patterns, performance metrics, and query efficiency to optimize the data workloads in a Snowflake account. The Engineer needs insights into long-running queries, frequently accessed tables, and warehouse resource utilization while also tracking user activities and ensuring compliance with governance policies.

Which combination of views should the Engineer use to meet these requirements? (Choose three.)

A. SNOWFLAKE.ACCOUNT_USAGE.OBJECT_DEPENDENCIES view to track dependencies between objects.

B. Use the SNOWFLAKE.ACCOUNT_USAGE.QUERY_HISTORY view to analyze query performance.

C. Use the SNOWFLAKE.ACCOUNT_USAGE.SCHEMA_PRLVILEGES view to ensure compliance with data governance policies.

D. Use the SNOWFLAKE.ACCOUNT_USAGE.DATABASE_STORAGE_USAGE_HISTORY view to analyze historical storage usage across databases.

E. Use the SNOWFLAKE.ACCOUNT_USAGE.ACCESS_HISTORY view to track access patterns.

F. Use the SNOWFLAKE.ACCOUNT_USAGE.WAREHOUSE_METERING_HISTORY view to monitor warehouse resource consumption.

**Suggested Answer:** *B, E, F*

Currently there are no comments in this discussion, be the first to comment!

Question #20 *Topic 1*

Which query will show a list of the 20 most recent executions of a specified task, MYTASK, that have been scheduled within the last hour that have ended or are still running?

A.

B.

C.

D.

```
select * from table(information_schema.task_history(scheduled_time_range_start
=>dateadd('hour',-1,current_timestamp()), result_limit => 20,
task_name=>'MYTASK'));
```

```
select * from table(information_schema.task_history(scheduled_time_range_start
=>dateadd('hour',-1,current_timestamp()), result_limit => 20,
task_name=>'MYTASK')) where query_id IS NOT NULL;
```

```
select * from table(information_schema.task_history(scheduled_time_range_start
=>dateadd('hour',-1,current_timestamp()), result_limit => 20,
task_name=>'MYTASK')) where STATE IN ('EXECUTING', 'SUCCEEDED', 'FAILED');
```

```
select * from table(information_schema.task_history(scheduled_time_range_end =>dateadd('hour',-1,current_timestamp()), result_limit => 10,
task_name=>'MYTASK'))
where STATE IN ('EXECUTING', 'SUCCEEDED');
```

**Suggested Answer:**
Correct choice: B (WHERE query_id IS NOT NULL). TASK_HISTORY with scheduled_time_range_start => dateadd('hour',-1,current_timestamp()), result_limit => 20 and task_name => 'MYTASK' returns the 20 most recent runs scheduled in the last hour; Snowflake documents filtering on query_id IS NOT NULL to keep only runs that have completed or are still running (future scheduled runs have no query ID). A also returns future scheduled runs, C omits ended states such as CANCELLED/SKIPPED, and D uses the wrong time range and result_limit 10.

Currently there are no comments in this discussion, be the first to comment!

Question #21 *Topic 1*

A Query Profile shows that Snowflake is scanning 80% of the partitions on a very large standard table when a specific filter is used. The query is running for 30 minutes.

To investigate, the Engineer runs: SELECT SYSTEM$CLUSTERING_INFORMATION ('LARGE_TABLE', '(COL_A)');

The JSON output reveals a total_partition_count of 1,000, but the partition_depth_histogram shows that 900 partitions have an average_overlap and average_depth of 1.

What conclusion can be drawn from this metadata?

A. The table should be converted to a dynamic table with incremental refreshes to address micro-partition fragmentation.

B. Automatic Clustering should be enabled to force a reorganization of 100 overlapping partitions.

C. The total_partition_count is below the recommended threshold for the system function to return a statistically significant result.

D. The table is well-clustered, but the specific filter used in the query is likely not on COL_A, or the filter value exists in highly-overlapped partitions.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #22 *Topic 1*

How can a Data Engineer track credit consumption related to the use of Data Metric Functions (DMFs)?

A. Query the DATABASE_STORAGE_USAGE_HISTORY table function.

B. Query the DATA_QUALITY_MONITORING_USAGE_HISTORY view.

C. Query the DATABASE_STORAGE_USAGE_HISTORY view.

D. Query the WAREHOUSE_METERING_HISTORY table function.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #23 *Topic 1*

What Snowflake expenses may be incurred when using external functions? (Choose two.)

A. Data storage

B. Data transfer

C. Cloud services compute

D. Snowflake warehouse compute

E. External function serverless compute

**Suggested Answer:** *B, E*

Currently there are no comments in this discussion, be the first to comment!

Question #24 *Topic 1*

A Data Engineer has developed a dashboard that will issue the same SQL select clause to Snowflake every 12 hours.

How long will Snowflake use the persisted query results from the result cache, provided that the underlying data has not changed?

A. 12 hours

B. 24 hours

C. 14 days

D. 31 days

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!

Question #25 *Topic 1*

A company built a sales reporting system with Python, connecting to Snowflake using the Python Connector. Based on the user's selections, the system generates the SQL queries needed to fetch the data for the report. First it gets the customers that meet the given query parameters (on average 1000 customer records for each report run), and then it loops the customer records sequentially. Inside that loop it runs the generated SQL clause for the current customer to get the detailed data for that customer number from the sales data table.

When the Data Engineer tested the individual SQL clauses, they were fast enough (1 second to get the customers, 0.5 second to get the sales data for one customer), but the total runtime of the report is too long.

How can this situation be improved?

A. Increase the size of the virtual warehouse.

B. Increase the number of maximum clusters of the virtual warehouse.

C. Define a clustering key for the sales data table.

D. Rewrite the report to eliminate the use of the loop construct.

**Suggested Answer:** *D*

Currently there are no comments in this discussion, be the first to comment!

Question #26 *Topic 1*

A Data Engineer needs to optimize the performance and minimize the costs of a dynamic table pipeline in Snowflake. The Engineer wants to monitor and separate the dynamic table costs prior to transferring the table refresh processing to a shared virtual warehouse.

How should the Engineer meet these requirements?

A. Use a shared virtual warehouse to test the performance of the dynamic table refresh process and monitor the costs for all workloads.

B. Before moving the table, test the process on a dedicated virtual warehouse to establish a cost baseline.

C. Execute the dynamic tables on multiple virtual warehouses simultaneously to distribute the load equally.

D. Set up the dynamic tables directly on a production virtual warehouse to obtain real-time cost insights.

**Suggested Answer:** *B*

Currently there are no comments in this discussion, be the first to comment!