



Free Questions for DEA-C01 by certscare

Shared by Roberson on 29-01-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You as Data engineer might want to consider disabling auto-suspend for a warehouse if?

Options:

- A- You have a low, fluctuating workload for the warehouse.
- B- You have a heavy, steady workload for the warehouse.
- C- You require the warehouse to be available with delay.
- D- You require the warehouse to be available with no delay or lag time.

Answer:

B, D

Explanation:

Automating Warehouse Suspension

Data Engineer might want to consider disabling auto-suspend for a warehouse if:

He/She have a heavy, steady workload for the warehouse.

He/She require the warehouse to be available with no delay or lag time. Warehouse provisioning is generally very fast (e.g. 1 or 2 seconds); however, depending on the size of the warehouse and the availability of compute resources to provision, it can take longer.

If he/she chose to disable auto-suspend, He/she must carefully consider the costs associated with running a warehouse continually, even when the warehouse is not processing queries. The costs can be significant, especially for larger warehouses (X-Large, 2X-Large, etc.).

To disable auto-suspend, Engineer must explicitly select Never in the web interface, or specify 0 or NULL in SQL.

Question 2

Question Type: MultipleChoice

Jonas, a Lead Performance Engineer, identified that some of the operation of his query which functionally remove the duplicates from huge data set is spilling the data to remote disk. How can he alleviate spilling to a remote disk for better query performance?

Options:

A- Jonas can recommend using a large warehouse which effectively increase the available memory/local disk space for the operations.

B- He can Process data in smaller batches to manage workload.

C- Spilling do not have a profound effect on query performance (especially if remote disk is used for spilling).

D- Data Sharing can be helpful to improve query performance.

Answer:

A, B

Explanation:

For some operations (e.g. duplicate elimination for a huge data set), the amount of memory available for the compute resources used to execute the operation might not be sufficient to hold intermediate results. As a result, the query processing engine will start spilling the data to local disk. If the local disk space is not sufficient, the spilled data is then saved to remote disks.

This spilling can have a profound effect on query performance (especially if remote disk is used for spilling). To alleviate this, It is recommend that:

* Using a larger warehouse (effectively increasing the available memory/local disk space for the op-eration), and/or

Processing data in smaller batches.

Question 3

Question Type: MultipleChoice

Michael, a Data Engineer Running a Data query to achieve Union of Data sets coming from Multi-ple data sources, later he figured out that Data processing query is taking more time than expected. He started analyzing the Query performance using query profile interface. He discovered & realized that he used UNION when the UNION ALL semantics was sufficient.

Which Extra Data Processing Operator Michael figured out while doing query profile analysis in this case which helps him to identify this performance bottlenecks?

Options:

- A- Aggregate
- B- UNION ALL
- C- Flatten
- D- Join
- E- Filter

Answer:

A

Explanation:

In SQL, it is possible to combine two sets of data with either UNION or UNION ALL constructs. The difference between them is that UNION ALL simply concatenates inputs, while UNION does the same, but also performs duplicate elimination.

A common mistake is to use UNION when the UNION ALL semantics are sufficient. These queries show in Query Profile as a UnionAll operator with an extra Aggregate operator on top (which performs duplicate elimination).

To Know more about Data Processing Operators, please do refer:

<https://docs.snowflake.com/en/user-guide/ui-query-profile#operator-types>

Question 4

Question Type: MultipleChoice

What are Common Query Problems a Data Engineer can identified using Query Profiler?

Options:

- A- "Exploding" Joins i.e Joins resulting due to a "Cartesian product"
- B- Queries Too Large to Fit in Memory

C- Inefficient Pruning

D- Ineffective Data Sharing

Answer:

A, B, C

Explanation:

"Exploding" Joins

One of the common mistakes SQL users make is joining tables without providing a join condition (resulting in a "Cartesian product"), or providing a condition where records from one table match multiple records from another table. For such queries, the Join operator produces significantly (often by orders of magnitude) more tuples than it consumes.

This can be observed by looking at the number of records produced by a Join operator in the profile interface, and typically is also reflected in Join operator consuming a lot of time.

Queries Too Large to Fit in Memory

For some operations (e.g. duplicate elimination for a huge data set), the amount of memory available for the compute resources used to execute the operation might not be sufficient to hold intermediate results. As a result, the query processing engine will start spilling the data to local disk. If the local disk space is not sufficient, the spilled data is then saved to remote disks.

This spilling can have a profound effect on query performance (especially if remote disk is used for spilling).

Spilling statistics can be checked in Query Profile Interface.

Inefficient Pruning

Snowflake collects rich statistics on data allowing it not to read unnecessary parts of a table based on the query filters. However, for this to have an effect, the data storage order needs to be correlated with the query filter attributes.

The efficiency of pruning can be observed by comparing Partitions scanned and Partitions total statistics in the TableScan operators. If the former is a small fraction of the latter, pruning is efficient. If not, the pruning did not have an effect.

Of course, pruning can only help for queries that actually filter out a significant amount of data. If the pruning statistics do not show data reduction, but there is a Filter operator above TableScan which filters out a number of records, this might signal that a different data organization might be beneficial for this query.

Question 5

Question Type: MultipleChoice

Data Engineer wants to analyze query performance & looking out for profiling information, He went to Query/Operator Details also called Profile Overview of Query Profile Interface & searching for statistics attributes around I/O. Which of the following information he can't get from there?

Options:

- A- Percentage scanned from cache --- the percentage of data scanned from the local disk cache.
- B- Bytes written --- bytes written (e.g. when loading into a table).
- C- External bytes scanned --- bytes read from an external object, e.g. a stage.
- D- Bytes sent over the wireframe --- amount of data sent over the wireframe
- E- Bytes read from result --- bytes read from the result object.

Answer:

D

Explanation:

To help you analyze query performance, Query/Operator Details panel also called Profile overview panel provides two classes of profiling information:

* Execution time, broken down into categories

* Detailed statistics

Apart from Option Bytes sent over the wireframe --- amount of data sent over the wireframe , Rest of the Statistics Information provided by Query/Operator details in the Query Profile Inter-face.

To Know More about the Query/Operator Details options , please refer the link:

<https://docs.snowflake.com/en/user-guide/ui-query-profile#query-operator-details>

Question 6

Question Type: MultipleChoice

Data Engineer looking out for quick tool for understanding the mechanics of queries & need to know more about the performance or behaviour of a particular query.

He should go to which feature of snowflake which can help him to spot typical mistakes in SQL query expressions to identify potential performance bottlenecks and improvement opportunities?

Options:

A- Query Optimizer

B- Performance Metadata table

C- Query Profile

D- Query Designer

Answer:

C

Explanation:

Query Profile, available through the classic web interface, provides execution details for a query. For the selected query, it provides a graphical representation of the main components of the processing plan for the query, with statistics for each component, along with details and statistics for the overall query.

Query Profile is a powerful tool for understanding the mechanics of queries. It can be used whenever you want or need to know more about the performance or behavior of a particular query. It is designed to help you spot typical mistakes in SQL query expressions to identify potential performance bottlenecks and improvement opportunities.

Question 7

Question Type: MultipleChoice

Dominic, a Data Engineer wants to resume the pipe named stalepipe3 which got stale after 14 days. To do the same, he called the `SYSTEM$PIPE_FORCE_RESUME` function

```
select sys-tem$pipe_force_resume('snowmydb.mysnowschemastalepipe3','staleness_check_override');
```

Let's say If the pipe is resumed 16 days after it was paused, what will happened to the event notification that were received on the first and second days after the pipe was paused?

Options:

- A-** Snowpipe generally skips any event notifications that were received on the first and second days after the pipe was paused.
- B-** Pipe maintains Metadata history of files for 64 days, so in this scenarios Snowpipe processed all the event notifications that were received for 16 days or so.
- C-** Once the Pipe got stale, all the events got purged automatically & pipe needs to be re-created with modified properties.
- D-** All the events get processed from day 1 if the PURGE properties in the PIPE object definition set to be FALSE initially.

Answer:

A

Explanation:

When a pipe is paused, event messages received for the pipe enter a limited retention period. The period is 14 days by default. If a pipe is paused for longer than 14 days, it is considered stale.

To resume a stale pipe, a qualified role must call the `SYSTEM$PIPE_FORCE_RESUME` function and input the `STALENESS_CHECK_OVERRIDE` argument. This argument indicates an understanding that the role is resuming a stale pipe.

For example, resume the stale `stalepipe1` pipe in the `mydb.myschema` database and schema:

```
select sys-tem$pipe_force_resume('mydb.myschema.stalepipe3','staleness_check_override');
```

As an event notification received while a pipe is paused reaches the end of the limited retention period, Snowflake schedules it to be dropped from the internal metadata. If the pipe is later resumed, Snowflake processes these older notifications on a best effort basis. Snowflake cannot guarantee that they are processed.

For example, if a pipe is resumed 15 days after it was paused, Snowflake generally skips any event notifications that were received on the first day the pipe was paused (i.e. that are now more than 14 days old). If the pipe is resumed 16 days after it was paused, Snowflake generally skips any event notifications that were received on the first and second days after the pipe was paused. And so on.

Question 8

Question Type: MultipleChoice

Data Engineer is using existing pipe that automates data loads using event notifications, later he figured out the needs to modify pipe properties. For the same, He decided to recreate the pipe as best practice. He followed the below steps for the same.

1. Query the `SYSTEM$PIPE_STATUS` function and verify that the pipe execution state is `RUN-NING`.

2. Recreate the pipe (using CREATE OR REPLACE PIPE).
3. Query the SYSTEM\$PIPE_STATUS function and verify that the pipe execution state is RUN-NING.

Which are the Missing recommended steps while Recreating Pipes for Automated Data Loads?

Options:

- A-** CREATE OR REPLACE PIPE command will recreate the PIPE successfully.
- B-** Terminate the existing pipe (using ALTER PIPE ... SET PIPE_EXECUTION_TERMINATE = true) before recreation.
- C-** Pause the pipe (using ALTER PIPE ... SET PIPE_EXECUTION_PAUSED = true) Pre & Post recreation & Resume after recreation (using ALTER PIPE ... SET PIPE_EXECUTION_PAUSED = false).
- D-** Force the pipe to resume (using SYSTEM\$PIPE_FORCE_RESUME).

Answer:

C

Explanation:

Recreating a pipe (using a CREATE OR REPLACE PIPE statement) is necessary to modify most pipe properties.

Recreating Pipes for Automated Data Loads

When recreating a pipe that automates data loads using event notifications, it's recommended that Data Engineer complete the following steps:

1. Pause the pipe (using ALTER PIPE ... SET PIPE_EXECUTION_PAUSED = true).
2. Query the SYSTEM\$PIPE_STATUS function and verify that the pipe execution state is PAUSED.
3. Recreate the pipe (using CREATE OR REPLACE PIPE).
4. Pause the pipe again.
5. Review the configuration steps for your cloud messaging service to ensure the settings are still accurate.
6. Query the SYSTEM\$PIPE_STATUS function again and verify that the pipe execution state is RUNNING.

Question 9

Question Type: MultipleChoice

Data Engineer is looking out to delete staged files automatically/periodically when the data is successfully loaded into tables by the Snowpipe. For achieving the same, which options/command is best suited: [Select 2]

Options:

- A-** PURGE option can be set as True in the COPY INTO Command embedded in PIPE objects definition.
- B-** To remove staged files that no longer needed, periodically REMOVE command can be executed to delete the files.
- C-** To remove staged files that no longer needed, periodically DELETE command can be executed to delete the files.
- D-** REMOVE_STAGE_FILES option can be set as True in the COPY INTO Command embedded in PIPE objects definition.

Answer:

A, B

Explanation:

Deleting Staged Files After Snowpipe Loads the Data

Pipe objects do not support the PURGE copy option. Snowpipe cannot delete staged files automatically when the data is successfully loaded into tables.

To remove staged files that you no longer need, It is recommended to periodically executing the REMOVE command to delete the files.

Alternatively, configure any lifecycle management features provided by cloud storage service provider.

Question 10

Question Type: MultipleChoice

Data Engineer try to load data from external stage using Snowpipe & later find out that some Set of Files Not Loaded. To debug the issue, she used COPY_HISTORY function & cross verified that its output indicates a subset of files was not loaded. What is possible reason of arising this situation in both REST API call and Auto-Ingest methods? [Select 2]

Options:

- A-** External event-driven functionality is used to call the REST APIs, and a backlog of data files already existed in the external stage before the events were configured.
- B-** An event notification failure prevented a set of files from getting queued.
- C-** Files modified and staged again after 14 days and Snowpipe ignores modified files that are staged again.
- D-** A backlog of data files already existed in the external stage do not have any impact on Load failure, as this is well managed by serverless SnowPipe

Answer:

A, B

Explanation:

COPY_HISTORY Record Indicates Unloaded Subset of Files:

If the COPY_HISTORY function output indicates a subset of files was not loaded, you may try to "refresh" the pipe.

This situation can arise in any of the following situations:

- * The external stage was previously used to bulk load data using the COPY INTO table command.

- * REST API:

- o External event-driven functionality is used to call the REST APIs, and a backlog of data files already existed in the external stage before the events were configured.

- * Auto-ingest:

- o A backlog of data files already existed in the external stage before event notifications were configured.

- o An event notification failure prevented a set of files from getting queued.

To load the data files in your external stage using the configured pipe, execute an ALTER PIPE ... REFRESH statement.

Question 11

Question Type: MultipleChoice

The Snowpipe API provides REST endpoints for fetching load reports. One of the Endpoint named insertReport helps to retrieves a report of files submitted via insertFiles end point whose contents were recently ingested into a table. A success response (200) contains information about files that have recently been added to the table. Response Looks like below:

1. {
2. "pipe": "SNOWTESTDB.SFTESTSCHEMA.SFpipe",
3. "completeResult": true,
4. "nextBeginMark": "1_16",
5. "files": [- 6. {
- 7. "path": "data4859992083898.csv",
- 8. "stageLocation": "s3://mybucket/",
- 9. "fileSize": 89,
- 10. "timeReceived": "2022-01-31T04:47:41.453Z",
- 11. "lastInsertTime": "2022-01-31T04:48:28.575Z",
- 12. "rowsInserted": 1,
- 13. "rowsParsed": 1,

14. "errorsSeen": 0,

15. "errorLimit": 1,

16. "complete": true,

17. "status": "????"

18. }

19.]

20. }

Which one is the correct value of status string data in the Response Body?

Options:

A- LOADED

B- LOADED_SUCCESS

C- LOAD_SUCCESS

D- SUCCESS

Answer:

C

Explanation:

Permissible Load status for the file:

LOAD_IN_PROGRESS: Part of the file has been loaded into the table, but the load process has not completed yet.

LOADED: The entire file has been loaded successfully into the table.

LOAD_FAILED: The file load failed.

PARTIALLY_LOADED: Some rows from this file were loaded successfully, but others were not loaded due to errors. Processing of this file is completed.

Please note the different Response Codes available with their meaning.

200 --- Success. Report returned.

400 --- Failure. Invalid request due to an invalid format, or limit exceeded.

404 --- Failure. pipeName not recognized.

This error code can also be returned if the role used when calling the endpoint does not have sufficient privileges. For more information, see [Granting Access Privileges](#).

429 --- Failure. Request rate limit exceeded.

500 --- Failure. Internal error occurred.

As you could understand from the questions, there is 200 Success response returned, Status in the response body would be LOADED.

Question 12

Question Type: MultipleChoice

Snowpipe API provides a REST endpoint for defining the list of files to ingest that Informs Snow-flake about the files to be ingested into a table. A successful response from this endpoint means that Snowflake has recorded the list of files to add to the table. It does not necessarily mean the files have been ingested. What is name of this Endpoint?

Options:

- A- REST endpoints --> insertReport
- B- REST endpoints --> loadHistoryScan
- C- REST endpoints --> ingestfiles
- D- REST endpoints--> insertfiles

Answer:

D

Explanation:

The Snowpipe API provides a REST endpoint for defining the list of files to ingest.

Endpoint: insertFiles

Informs Snowflake about the files to be ingested into a table. A successful response from this end-point means that Snowflake has recorded the list of files to add to the table. It does not necessarily mean the files have been ingested. For more details, see the response codes below.

In most cases, Snowflake inserts fresh data into the target table within a few minutes.

To Know more about SnowFlake Rest API used for Data File ingestion, do refer:

<https://docs.snowflake.com/en/user-guide/data-load-snowpipe-rest-apis.html#data-file-ingestion>

To Get Premium Files for DEA-C01 Visit

<https://www.p2pexams.com/products/dea-c01>

For More Free Questions Visit

<https://www.p2pexams.com/snowflake/pdf/dea-c01>

