Number: DEA-C01
Passing Score: 800
Time Limit: 120
File Version: 5.0

# Exam Code: DEA-C01

**Exam A**

**QUESTION 1**
In one of your created Schema, you have been required to create Internal Stages, what are the Incorrect considerations you can noticed from the below options? [Select All that Apply]

A. User stages can be altered or dropped just like Table Stage.
B. Table stage type is designed to store files that are staged and managed by one or more users but only loaded into a single table.
C. A named internal stage type can store files that are staged and managed by one or more users and loaded into one or more tables.
D. A table stage is available for each table created in Snowflake.

**Correct Answer: A**
**Section:**
**Explanation:**
A stage specifies where data files are stored (i.e. "staged") so that the data in the files can be loaded into a table.
Types of Internal Stages
· User Stages
· Table Stages
· Named Stages
By default, each user and table in Snowflake is automatically allocated an internal stage for staging data files to be loaded. In addition, you can create named internal stages.
File staging information is required during both steps in the data loading process:
You must specify an internal stage in the PUT command when uploading files to Snowflake.
You must specify the same stage in the COPY INTO <table> command when loading data into a table from the staged files.
Consider the best type of stage for specific data files. Each option provides benefits and potential drawbacks.
User Stages
Each user has a Snowflake stage allocated to them by default for storing files. This stage is a convenient option if your files will only be accessed by a single user, but need to be copied into multi-ple tables.
User stages have the following characteristics and limitations:
User stages are referenced using @~; e.g. use LIST @~ to list the files in a user stage.
Unlike named stages, user stages cannot be altered or dropped.
User stages do not support setting file format options. Instead, you must specify file format and copy options as part of the COPY INTO <table> command.
This option is not appropriate if:
Multiple users require access to the files.
The current user does not have INSERT privileges on the tables the data will be loaded into.
Table Stages
Each table has a Snowflake stage allocated to it by default for storing files. This stage is a conven-ient option if your files need to be accessible to multiple users and only need to be copied into a sin-gle table.
Table stages have the following characteristics and limitations:
Table stages have the same name as the table; e.g. a table named mytable has a stage referenced as @%mytable.
Unlike named stages, table stages cannot be altered or dropped.
Table stages do not support transforming data while loading it (i.e. using a query as the source for the COPY command).
Note that a table stage is not a separate database object; rather, it is an implicit stage tied to the table itself. A table stage has no grantable privileges of its own. To stage files to a table stage, list the files, query them on the stage, or drop them, you must be the table owner (have the role with the OWNERSHIP privilege on the table).
This option is not appropriate if you need to copy the data in the files into multiple tables.
Named Stages
Named stages are database objects that provide the greatest degree of flexibility for data loading:
Users with the appropriate privileges on the stage can load data into any table.
Because the stage is a database object, the security/access rules that apply to all objects apply. The privileges to use a stage can be granted or revoked from roles. In addition, ownership of the stage can be transferred to another role.

If you plan to stage data files that will be loaded only by you, or will be loaded only into a single table, then you may prefer to simply use either your user stage or the stage for the table into which you will be loading data. Named stages are optional but recommended when you plan regular data loads that could involve multiple users and/or tables.

**QUESTION 2**
As a Data Engineer, you have requirement to query most recent data from the Large Dataset that reside in the external cloud storage, how would you design your data pipelines keeping in mind fastest time to delivery?

A. Data pipelines would be created to first load data into internal stages & then into Per-manent table with SCD Type 2 transformation.
B. Direct Querying External tables on top of existing data stored in external cloud storage for analysis without first loading it into Snowflake.
C. Unload data into SnowFlake Internal data storage using PUT command.
D. Snowpipe can be leveraged with streams to load data in micro batch fashion with CDC streams that capture most recent data only.
E. External tables with Materialized views can be created in Snowflake.

**Correct Answer: E**
**Section:**
**Explanation:**
In a typical table, the data is stored in the database; however, in an external table, the data is stored in files in an external stage. External tables store file-level metadata about the data files, such as the filename, a version identifier and related properties. This enables querying data stored in files in an external stage as if it were inside a database. External tables can access data stored in any format supported by COPY INTO <table> statements.
External tables are read-only, therefore no DML operations can be performed on them; however, external tables can be used for query and join operations. Views can be created against external tables.
Querying data stored external to the database is likely to be slower than querying native database tables; however, materialized views based on external tables can improve query performance.
Creating External tables enable user for querying existing data stored in external cloud storage for analysis without first loading it into Snowflake. The source of truth for the data remains in the external cloud storage. Data sets materialized in Snowflake via materialized views are read-only.
This solution is especially beneficial to accounts that have a large amount of data stored in external cloud storage and only want to query a portion of the data; for example, the most recent data. Users can create materialized views on subsets of this data for improved query performance.

**QUESTION 3**
Mark the incorrect statement in case Data engineer using the COPY INTO <table> command to load data from files into Snowflake tables?

A. For Data loading of files with semi-structured file formats (JSON, Avro, etc.), the only supported character set is UTF-16.
B. For loading data from all semi-structured supported file formats (JSON, Avro, etc.), as well as unloading data, UTF-8 is the only supported character set.
C. For Local environment, Files are first copied ("staged") to an internal (Snowflake) stage, then loaded into a table.
D. UTF-32 & UTF-16 both encoding character sets supported for loading data from de-limited files (CSV, TSV, etc.)

**Correct Answer: A**
**Section:**
**Explanation:**
For Data Loading of delimited files (CSV, TSV, etc.), the default character set is UTF-8. To use any other characters sets, you must explicitly specify the encoding to use for loading.
For semi-structured file formats (JSON, Avro, etc.), the only supported character set is UTF-8.
Rest of the statements are correct.

**QUESTION 4**
Let us say you have List of 50 Source files, which needs to be loaded into Snowflake internal stage. All these Source system files are already Brotli-compressed files. Which statement is correct with respect to Compression of Staged Files?

A. Even though Source files are already compressed, Snowflake do apply default gzip2 Compression to optimize the storage cost.
B. Snowflake automatically detect Brotli Compression, will skip further compression of all 50 files.
C. Auto-detection is not yet supported for Brotli-compressed files; when staging or loading Brotlicompressed files, you must explicitly specify the compression method that was used.
D. When staging 50 compressed files in a Snowflake stage, the files are automatically com-pressed using gzip.

**Correct Answer: C**
**Section:**
**Explanation:**
Auto-detection is not yet supported for Brotli-compressed files; when staging or loading Brotlicompressed files, you must explicitly specify the compression method that was used.
To Know more about Compression of Staged Files, please refer the link:
https://docs.snowflake.com/en/user-guide/intro-summary-loading.html#compression-of-stagedfiles

**QUESTION 5**
For the most efficient and cost-effective Data load experience, Data Engineer needs to inconsider-ate which of the following considerations?

A. Split larger files into a greater number of smaller files, maximize the processing over-head for each file.
(Correct)
B. Enabling the STRIP_OUTER_ARRAY file format option for the COPY INTO <ta-ble> command to remove the outer array structure and load the records into separate table rows.
C. Amazon Kinesis Firehose can be convenient way to aggregate and batch data files which also allows defining both the desired file size, called the buffer size, and the wait interval after which a new file is sent, called the buffer interval.
D. When preparing your delimited text (CSV) files for loading, the number of columns in each row should be consistent.
E. if the "null" values in your files indicate missing values and have no other special mean-ing, Snowflake recommend setting the file format option STRIP_NULL_VALUES to TRUE when loading the semi-structured data file.

**Correct Answer: A**
**Section:**
**Explanation:**
Split larger files into a greater number of smaller files to distribute the load among the compute resources in an active warehouse. This would minimize the processing overhead rather than maximize it.
Rest is recommended Data loading considerations.

**QUESTION 6**
The COPY command supports several options for loading data files from a stage i.e.

A. By path
II. Specifying a list of specific files to load.
III. Using pattern matching to identify specific files by pattern.
IV. Organize files into logical paths that reflect a scheduling pattern.
Of the aforesaid options for identifying/specifying data files to load from a stage, which option in general is the fastest & best considerate?
B. I
C. II
D. III
E. IV

**Correct Answer: B**
**Section:**
**Explanation:**
Of the above options for identifying/specifying data files to load from a stage, providing a discrete list of files is generally the fastest; however, the FILES parameter supports a maximum of 1,000 files, meaning a COPY command executed with the FILES parameter can only load up to 1,000 files.
For example:
copy into load1 from @%load1/Snow1/ files=('mydata1.csv', 'mydata2.csv', 'mydata3.csv')

**QUESTION 7**
As Data Engineer, you have requirement to Load set of New Product Files containing Product relevant information into the Snowflake internal tables, Later you analyzed that some of the Source files are already loaded in one of the historical batch & for that you have prechecked Metadata col-umn LAST_MODIFIED date for a staged data file & found out that LAST_MODIFIED date is older than 64 days for few files and the initial set of data was

loaded into the table more than 64 days earlier, Which one is the best approach to Load Source data files with expired load metadata along with set of files whose metadata might be available to avoid data duplication?

A. Since the initial set of data for the table (i.e. the first batch after the table was created) was loaded, we can simply use the COPY INTO command to load all the product files with the known load status irrespective of their column LAST_MODIFIED date values.
B. The COPY command cannot definitively determine whether a file has been loaded al-ready if the LAST_MODIFIED date is older than 64 days and the initial set of data was loaded into the table more than 64 days earlier (and if the file was loaded into the table, that also occurred more than 64 days earlier). In this case, to prevent accidental reload, the command skips the product files by default.
C. Set the FORCE option to load all files, ignoring load metadata if it exists.
D. To load files whose metadata has expired, set the LOAD_UNCERTAIN_FILES copy option to true.

**Correct Answer: D**
**Section:**
**Explanation:**
To load files whose metadata has expired, set the LOAD_UNCERTAIN_FILES copy option to true. The copy option references load metadata, if available, to avoid data duplication, but also at-tempts to load files with expired load metadata.
Alternatively, set the FORCE option to load all files, ignoring load metadata if it exists. Note that this option reloads files, potentially duplicating data in a table.
Please refer the Example as mentioned in the link below:
https://docs.snowflake.com/en/user-guide/data-load-considerations-load.html#loading-older-files

**QUESTION 8**
If external software i.e. TIBCO, exports Data fields enclosed in quotes but inserts a leading space before the opening quotation character for each field, How Snowflake handle it? [Select 2]

A. Snowflake automatically handles leading spaces by trimming implicitly & removes the quotation marks enclosing each field.
B. field_optionally_enclosed_by option along with TRIM_IF function in COPY INTO statement can be used to handle this scenario successfully.
C. Snowflake reads the leading space rather than the opening quotation character as the beginning of the field and the quotation characters are interpreted as string data. (Correct)
D. COPY command trims the leading space and removes the quotation marks enclosing each field
E. copy into SFtable
F. from @%SFtable
G. file_format = (type = csv trim_space=true field_optionally_enclosed_by = '0x22');

**Correct Answer: D**
**Section:**
**Explanation:**
If your external software exports fields enclosed in quotes but inserts a leading space before the opening quotation character for each field, Snowflake reads the leading space rather than the opening quotation character as the beginning of the field. The quotation characters are interpreted as string data.
Use the TRIM_SPACE file format option to remove undesirable spaces during the data load.

**QUESTION 9**
Data Engineer Loading File named snowdata.tsv in the /datadir directory from his local machine to Snowflake stage and try to prefix the file with a folder named tablestage, please mark the correct command which helps him to load the files data into snowflake internal Table stage?

A. put file://c:\datadir\snowdata.tsv @~/tablestage;
B. put file://c:\datadir\snowdata.tsv @%tablestage;
C. put file://c:\datadir\snowdata.tsv @tablestage;
D. put file:///datadir/snowdata.tsv @%tablestage;

**Correct Answer: B**
**Section:**
**Explanation:**
Execute PUT to upload (stage) local data files into an internal stage.
@% character combination identifies a table stage.

**QUESTION 10**
Mark the Correct Statements for the VALIDATION_MODE option used by Data Engineer for Da-ta loading operations in his/her COPY INTO <table> command:

A.
VALIDATION_MODE instructs the COPY command to validate the data files instead of loading them into the specified table; i.e., the COPY command tests the files for er-rors but does not load them.

B.
VALIDATION_MODE option supported these values:
RETURN_n_ROWS,
RETURN_ERRORS,
RETURN_ALL_ERRORS

C.
VALIDATION_MODE does not support COPY statements that transform data during a load. If the parameter is specified, the COPY statement returns an error.

D.
VALIDATION_MODE only support Data loading operation i.e., do not work while da-ta unloading.

**Correct Answer: A, B, C**
**Section:**
**Explanation:**
All the Statements are correct except the statement saying VALIDATION_MODE only support Data loading operation.
VALIDATION_MODE can be used with COPY INTO <location> command as well i.e for data unloading operation.
VALIDATION_MODE = RETURN_ROWS can be used at the time of Data unloading.
This option instructs the COPY command to return the results of the query in the SQL statement instead of unloading the results to the specified cloud storage location. The only supported validation option is RETURN_ROWS.
This option returns all rows produced by the query.
When you have validated the query, you can remove the VALIDATION_MODE to perform the unload operation.

**QUESTION 11**
To troubleshoot data load failure in one of your Copy Statement, Data Engineer have Executed a COPY statement with the VALIDATION_MODE copy option set to RETURN_ALL_ERRORS with reference to the set of files he had attempted to load. Which below function can facilitate analysis of the problematic records on top of the Results produced? [Select 2]

A. RESULT_SCAN
B. LAST_QUERY_ID
C. Rejected_record
D. LOAD_ERROR

**Correct Answer: A, B**
**Section:**
**Explanation:**
LAST_QUERY_ID() Function
Returns the ID of a specified query in the current session. If no query is specified, the most recently executed query is returned.
RESULT_SCAN() Function
Returns the result set of a previous command (within 24 hours of when you executed the query) as if the result was a table.
The following example validates a set of files (SFfile.csv.gz) that contain errors. To facilitate analy-sis of the errors, a COPY INTO <location> statement then unloads the problematic records into a text file so they could be analyzed and fixed in the original data files. The statement queries the RESULT_SCAN table.

1. #copy into Snowtable
2. from @SFstage/SFfile.csv.gz
3. validation_mode=return_all_errors;
4. #set qid=last_query_id();
5. #copy into @SFstage/errors/load_errors.txt from (select rejected_record from table( result_scan($qid))); Note: Other options are not valid functions.

**QUESTION 12**
As part of Table Designing, Data Engineer added a timestamp column that inserts the current timestamp as the default value as records are loaded into a table. The intent is to capture the time when each record was loaded into the table; however, the timestamps are earlier than the LOAD_TIME column values returned by COPY_HISTORY view (Account Usage). What could be reason of this issue?

A. LOAD_TIME column values returned by COPY_HISTORY view (Account Usage) gives the same time as returned by CURRENT_TIMESTAMP.
B. CURRENT_TIMESTAMP values might be different due to query gets executed in warehouse located in different region.
C. It might be possible that Cloud Provider hosted on Snowflake belongs to region having server time zone lagging Cluster time zone of warehouse where queries get processed & committed.
D. The reason is, CURRENT_TIMESTAMP is evaluated when the load operation is com-piled in cloud services rather than when the record is inserted into the table (i.e. when the transaction for the load operation is committed).

**Correct Answer: D**
**Section:**
**Explanation:**
The reason timestamps are earlier than the LOAD_TIME column values which is returned by COPY_HISTORY view (Account Usage) is that CURRENT_TIMESTAMP is evaluated when the load operation is compiled in cloud services rather than when the record is inserted into the table (i.e.
when the transaction for the load operation is committed).

**QUESTION 13**
Snowpipe loads data from files as soon as they are available in a stage. Automated data loads leverage event notifications for cloud storage to inform Snowpipe of the arrival of new data files to load.
Which Cloud hosted platform provides cross cloud support for automated data loading via Snowpipe?

A. GCP
B. AWS
C. AZURE
D. None of the Above currently provide cross cloud support for Snowpipe.

**Correct Answer: B**
**Section:**
**Explanation:**
Cross-cloud support only available to accounts hosted on Amazon Web Services currently.

**QUESTION 14**
Find out the odd one out:

A.
B. Bulk Data Load: Loads are always performed in a single transaction.
C. SnowPipe: Loads are combined or split into a single or multiple transactions based on the number and size of the rows in each data file.
D.
E. Bulk Data Load: Requires a user-specified warehouse to execute COPY statements.
F. SnowPipe: Uses Snowflake-supplied compute resources.
G.

H. Bulk Data Load: Billed for the amount of time each virtual warehouse is active.
I. SnowPipe: Billed according to the compute resources used in the Snowpipe ware-house while loading the files.
J.
K. Bulk Data Load: Load history Stored in the metadata of the target table for 365 days.
L. SnowPipe: Load history Stored in the metadata of the pipe for 64 days.

**Correct Answer: D**
**Section:**
**Explanation:**
Bulk data load
Load History Stored in the metadata of the target table for 64 days. Available upon completion of the COPY statement as the statement output.
Snowpipe
Load History Stored in the metadata of the pipe for 14 days. Must be requested from Snowflake via a REST endpoint, SQL table function, or ACCOUNT_USAGE view.
Rest are correct statements.

**QUESTION 15**
Data engineer designed the data pipelines using Snowpipe to load data files into Snowflake tables, what will happen in case few files with same name but modified data are queued for reloading?

A. Data will be reloaded as files are modified & its associated metadata also changed. But Snowflake handle implicitly deduplication.
B. eTAG is changed for Files even they are having same name, so data will be duplicated in SnowFlake tables.
C. Snowpipe uses file loading metadata associated with each table object, so no metadata available to prevent duplication.
D. Snowpipe uses file loading metadata associated with each pipe object to prevent reload-ing the same files (and duplicating data) in a table.

**Correct Answer: D**
**Section:**
**Explanation:**
Snowflake uses file loading metadata to prevent reloading the same files (and duplicating data) in a table. Snowpipe prevents loading files with the same name even if they were later modified (i.e. have a different eTag).
The file loading metadata is associated with the pipe object rather than the table. As a result:
· Staged files with the same name as files that were already loaded are ignored, even if they have been modified, e.g. if new rows were added or errors in the file were corrected.
· Truncating the table using the TRUNCATE TABLE command does not delete the Snowpipe file loading metadata.

**QUESTION 16**
Data Engineer decided to call the public REST endpoints to load data and retrieve load history reports.
Which of the following REST endpoints and a Snowflake Information Schema table function for viewing your load history can be used by her? [Select All that apply]

A. REST endpoints: loadinsertReport or HistoryScan
B. Information Schema table function: COPY_HISTORY
C. Account Usage view:COPY_HISTORY
D. REST endpoints: insertReport or loadHistoryScan
E. Information Schema table function: LOAD_HISTORY

**Correct Answer: B, C, D**
**Section:**
**Explanation:**
Snowflake provides REST endpoints and an Snowflake Information Schema table function for viewing your load history:
REST endpoints:
insertReport

loadHistoryScan
Information Schema table function:
COPY_HISTORY
Account Usage view:
COPY_HISTORY

**QUESTION 17**
Mark the incorrect statement when Data Engineer implement Automating Continuous Data Loading Using Cloud Messaging?

A. Automated Snowpipe uses event notifications to determine when new files arrive in monitored cloud storage and are ready to load.
B. 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.
C. Notifications identify the cloud storage event and include a list of the file names. They do not include the actual data in the files.
D. Triggering automated Snowpipe data loads using S3 event messages is supported by Snowflake accounts hosted on Cloud Platform like AWS, GCP or AZURE.

**Correct Answer: D**
**Section:**
**Explanation:**
Triggering automated Snowpipe data loads using S3 event messages is supported by Snowflake accounts hosted on Amazon Web Services (AWS) only.
Rest is correct statements.

**QUESTION 18**
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?

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

**Correct Answer: D**
**Section:**
**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 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. 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

**QUESTION 19**
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:

A. {
B. "pipe": "SNOWTESTDB.SFTESTSCHEMA.SFpipe",
C. "completeResult": true,
D. "nextBeginMark": "1_16",

E. "files": [
F. {
G. "path": "data4859992083898.csv",
H. "stageLocation": "s3://mybucket/",
I. "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?

J. LOADED
K. LOADED_SUCCESS
L. LOAD_SUCCESS
M. SUCCESS

**Correct Answer: C**
**Section:**
**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 not 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 20**
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]

A. External event-driven functionality is used to call the REST APIs, and a backlog of da-ta 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

**Correct Answer: A, B**
**Section:**
**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 al-ready 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 21**
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]

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.

**Correct Answer: A, B**
**Section:**
**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 22**
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.

A. Query the SYSTEM$PIPE_STATUS function and verify that the pipe execution state is RUN-NING.
B. Recreate the pipe (using CREATE OR REPLACE PIPE).
C. Query the SYSTEM$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?
D. CREATE OR REPLACE PIPE command will recreate the PIPE successfully.
E. Terminate the existing pipe (using ALTER PIPE … SET PIPE_EXECUTION_TERMINATE = true) before recreation.
F. 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).
G. Force the pipe to resume (using SYSTEM$PIPE_FORCE_RESUME).

**Correct Answer: C**
**Section:**
**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 23**
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$PIPE_FORCE_RESUME function select system$ pipe_force_resume('snowmydb.mysnowschema.stalepipe3','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?

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 pro-cessed 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 rec-reated 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.

**Correct Answer: A**
**Section:**
**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$PIPE_FORCE_RESUME function and input the STALENESS_CHECK_OVERRIDE argument. This argument indicates an under-standing that the role is resuming a stale pipe.
For example, resume the stale stalepipe1 pipe in the mydb.myschema database and schema:
select sys-tem$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, Snowpipe 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, Snowpipe 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, Snowpipe generally skips any event notifications that were received on the first and second days after the pipe was paused. And so on.

**QUESTION 24**
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?

A. Query Optimizer
B. Performance Metadata table
C. Query Profile
D. Query Designer

**Correct Answer: C**
**Section:**
**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 pro-cessing 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 25**
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?

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.

**Correct Answer: D**
**Section:**
**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 à 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 26**
What are Common Query Problems a Data Engineer can identified using Query Profiler?

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

**Correct Answer: A, B, C**
**Section:**
**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 correlat-ed 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 27**

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?

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

**Correct Answer: A**
**Section:**
**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 que-ries 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 28**
Jonas, a Lead Performance Engineer,identifed 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?

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.

**Correct Answer: A, B**
**Section:**
**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 operation), and/or Processing data in smaller batches.

**QUESTION 29**
You as Data engineer might want to consider disabling auto-suspend for a warehouse if?

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.

**Correct Answer: B, D**
**Section:**
**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 30**
Harry using Snowflake Enterprise Edition & decided to scale in/out the Cluster in automatic mode.
He needs to configure some warehouses as multi cluster mode and some among them in Standard mode as per needs.
If Harry is using Snowflake Enterprise Edition (or a higher edition), all his warehouses should be configured as multi-cluster warehouses only.

A. TRUE
B. FALSE

**Correct Answer: A**
**Section:**
**Explanation:**
If you are using Snowflake Enterprise Edition (or a higher edition), all your warehouses should be configured as multi-cluster warehouses.

**QUESTION 31**
Ryan, a Data Engineer, wants to improve the performance of large, complex queries against large data sets. He decided to Scale up underlying warehouse/cluster. What is correct Snowflake consideration while scaling up so that he can achieve better performance results? [Select all that apply]

A. Resizing can help reduce the queuing that occurs if a warehouse does not have enough compute resources to process all the queries that are submitted concurrently.
B. Scaling up is not intended for handling concurrency issues; instead, use additional warehouses to handle the workload or use a multi-cluster warehouse (if this feature is available for your account).
C. Snowflake supports resizing a warehouse at any time, even while running.
D. Resizing a running warehouse does not impact queries that are already being processed by the warehouse; the additional compute resources, once fully provisioned, are only used for queued and new queries.
E. Resizing between a 5XL or 6XL warehouse to a 4XL or smaller warehouse results in a brief period during which the customer is charged for both the new warehouse and the old warehouse while the old warehouse is quiesced.

**Correct Answer: A, B, C, D, E**
**Section:**
**Explanation:**
Resizing a warehouse generally improves query performance, particularly for larger, more complex queries. It can also help reduce the queuing that occurs if a warehouse does not have enough compute resources to process all the queries that are submitted concurrently. Note that warehouse resizing is not intended for handling concurrency issues; instead, use additional warehouses to handle the workload or use a multi-cluster warehouse (if this feature is available for your account).
Snowflake supports resizing a warehouse at any time, even while running. If a query is running slowly and you have additional queries of similar size and complexity that you want to run on the same warehouse, you might choose to resize the warehouse while it is running; however, note the following:
· Larger warehouse size is not necessarily faster; for smaller, basic queries that are already executing quickly, you may not see any significant improvement after resizing.
· Resizing a running warehouse does not impact queries that are already being processed by the warehouse; the additional compute resources, once fully provisioned, are only used for queued and new queries.
Resizing between a 5XL or 6XL warehouse to a 4XL or smaller warehouse results in a brief period during which the customer is charged for both the new warehouse and the old warehouse while the old warehouse is quiesced.

**QUESTION 32**
While working with Multi Cluster Warehouses, Select the incorrect understanding of Data Engineer about its usage?

A. Multi-cluster warehouses are designed specifically for handling queuing and perfor-mance issues related to large numbers of concurrent users and/or queries.
B. Unless you have a specific requirement for running in Maximized mode, multi-cluster warehouses should be configured to run in Auto-scale mode, which enables Snowflake to automatically start and stop clusters as needed.
C. When choosing the minimum number of clusters for a multi-cluster warehouse keep the default value as 1.
D. Multi-cluster warehouses generally improve query performance, particularly for larger, more complex queries.
E. When choosing the maximum number of clusters for a multi-cluster warehouse set its value as large as possible.

**Correct Answer: D**
**Section:**
**Explanation:**
Resizing a warehouse generally improves query performance, particularly for larger, more complex queries.
Multi-cluster warehouses are designed specifically for handling queuing and performance issues related to large numbers of concurrent users and/or queries. In addition, multi-cluster warehouses can help automate this process if your number of users/queries tend to fluctuate.
Rest all are correct understanding.

**QUESTION 33**
Select the incorrect statement while working with warehouses?

A. Compute resources waiting to shut down are considered to be in "quiesce" mode.
B. Resizing a warehouse to a larger size is useful while loading and unloading significant amounts of data.
C. Resizing a warehouse will have any immediate impact on statements that are currently being executed by the warehouse.
D. Resizing a suspended warehouse does not provision any new compute resources for the warehouse.

**Correct Answer: C**
**Section:**
**Explanation:**
Resizing a warehouse doesn't have any impact on statements that are currently being executed by the warehouse. When resizing to a larger size, the new compute resources, once fully provisioned, are used only to execute statements that are already in the warehouse queue, as well as all future statements submitted to the warehouse.

**QUESTION 34**
Ira a Data Engineer with TESLA IT systems, looking out to Compare Traditional Partitioning vs Snowflake micro-partitions for one of the Snowflake Project implementations. Which one of the following is incorrect understanding of Ira about Micro Partitioning?

A. All data in Snowflake tables is automatically divided into micro-partitions, which are contiguous units of storage compared to traditional partitioning where specialized DDL required.
B. All DML operations (e.g. DELETE, UPDATE, MERGE) take advantage of the under-lying micropartition metadata to facilitate and simplify table maintenance.
C. Snowflake stores metadata about all rows stored in a micro-partition, including number of distinct columns.
D. The micro-partition metadata maintained by Snowflake enables precise pruning of col-umns in micro-partitions at query run-time, including columns containing semi-structured data.
E. In Snowflake, as data is inserted/loaded into a table, clustering metadata is collected and recorded for each micro-partition created during the process.

**Correct Answer: C**
**Section:**
**Explanation:**
What are Micro-partitions?
All data in Snowflake tables is automatically divided into micro-partitions, which are contiguous units of storage. Each micro-partition contains between 50 MB and 500 MB of uncompressed data (note that the actual size in Snowflake is smaller because data is always stored compressed). Groups of rows in tables are mapped into individual micro-partitions, organized in a columnar fashion. This size and structure allow for extremely granular pruning of very large tables, which can be comprised of millions, or even hundreds of millions, of micro-partitions.
Snowflake stores metadata about all rows stored in a micro-partition, including:
· The range of values for each of the columns in the micro-partition.

· The number of distinct values.
· Additional properties used for both optimization and efficient query processing.
It Never stores number of columns as part of Metadata.
Rest of the statements are correct.

**QUESTION 35**
Which one is not the Core benefits of micro-partitioning

A. Snowflake micro-partitions are derived automatically they do not need to be explicitly defined upfront or maintained by users.
B. Enables extremely efficient DML and fine-grained pruning for faster queries.
C. Micro-partitions can overlap in their range of values, helps data skewing.
D. Columns are stored independently within micro-partitions, often referred to as colum-nar storage.
E. Columns are also compressed individually within micro-partitions.

**Correct Answer: C**
**Section:**
**Explanation:**
The benefits of Snowflake's approach to partitioning table data include:
· In contrast to traditional static partitioning, Snowflake micro-partitions are derived automatically; they don't need to be explicitly defined up-front or maintained by users.
· As the name suggests, micro-partitions are small in size (50 to 500 MB, before compression), which enables extremely efficient DML and fine-grained pruning for faster queries.
· Micro-partitions can overlap in their range of values, which, combined with their uniformly small size, helps prevent skew.
· Columns are stored independently within micro-partitions, often referred to as columnar storage.
This enables efficient scanning of individual columns; only the columns referenced by a query are scanned.
· Columns are also compressed individually within micro-partitions. Snowflake automatically determines the most efficient compression algorithm for the columns in each micro-partition.

**QUESTION 36**
Which Role that is dedicated to user and role management only?

A. ORGADMIN
B. SECURITYADMIN
C. USERADMIN
D. SYSADMIN
E. PUBLIC

**Correct Answer: C**
**Section:**

**QUESTION 37**
By default, a newly-created Custom role is not assigned to any user, nor granted to any other role?

A. TRUE
B. FALSE

**Correct Answer: A**
**Section:**

**QUESTION 38**
UDTFs also called a table function, returns zero, one, or multiple rows for each input row?

A. YES
B. NO

**Correct Answer: A**
**Section:**
**Explanation:**
UDFs may be scalar or tabular.
A scalar function returns one output row for each input row. The returned row consists of a single column/value.
A tabular function, also called a table function, returns zero, one, or multiple rows for each input row.
A tabular UDF is defined by specifying a return clause that contains the TABLE keyword and specifies the names and data types of the columns in the table results. Tabular UDFs are often called UDTFs (user-defined table functions) or table UDFs.

**QUESTION 39**
Which are supported Programming Languages for Creating UDTFs?

A. Python
B. Node.javascript
C. Javascript
D. Java
E. Perl

**Correct Answer: A, C, D**
**Section:**

**QUESTION 40**
Which UDF programming language is not supported with Snowflake Secure Data Sharing feature?

A. SQL
B. JAVA
C. JAVASCRIPT
D. PYTHON

**Correct Answer: C**
**Section:**

**QUESTION 41**
While creating even Secure UDF, snowflake recommended to use randomized identifiers (e.g. generated by UUID_STRING) instead of sequence-generated values?

A. TRUE (Correct)
B. FALSE

**Correct Answer: A**
**Section:**

**QUESTION 42**
When using the CURRENT_ROLE and CURRENT_USER functions with secure UDFs that will be shared with Snowflake accounts, Snowflake returns a NULL value for these functions?

A. TRUE
B. FALSE

**Correct Answer: A**
**Section:**
**Explanation:**
When using the CURRENT_ROLE and CURRENT_USER functions with secure UDFs that will be shared with Snowflake accounts, Snowflake returns a NULL value for these functions. The rea-son is that the owner of the data being shared does not typically control the users or roles in the ac-count with which the UDF is being shared.

**QUESTION 43**
Data Engineer, ran the below clustering depth analysis function:
select system$clustering_depth('TPCH_CUSTOMERS', '(C1, C6)', 'C9 = 30'); on TPCH_CUSTOMERS table, will return which of the following?

A. An error: this function does not accept lists of columns as a third parameter.
B. An error: this function does not accept predicates ('C9 = 30') as parameter.
C. Calculate the clustering depth for a table using mentioned columns in the table.
D. Calculate the clustering depth for a table using the clustering key defined for the table.

**Correct Answer: C**
**Section:**

**QUESTION 44**
Mark the Correct Statements:
Statement 1. Snowflake's zero-copy cloning feature provides a convenient way to quickly take a "snapshot" of any table, schema, or database.
Statement 2. Data Engineer can use zero-copy cloning feature for creating instant backups that do not incur any additional costs (until changes are made to the cloned object).

A. Statement 1
B. Statement 2
C. Both are False.
D. Statement 1 & 2 are correct.

**Correct Answer: C**
**Section:**
**Explanation:**
Snowflake's zero-copy cloning feature provides a convenient way to quickly take a "snapshot" of any table, schema, or database and create a derived copy of that object which initially shares the underlying storage. This can be extremely useful for creating instant backups that do not incur any additional costs (until changes are made to the cloned object).
For example, when a clone is created of a table, the clone utilizes no data storage because it shares all the existing micro-partitions of the original table at the time it was cloned; however, rows can then be added, deleted, or updated in the clone independently from the original table. Each change to the clone results in new micro-partitions that are owned exclusively by the clone and are protected through CDP.

**QUESTION 45**
Clones can be cloned, with no limitations on the number or iterations of clones that can be created (e.g. you can create a clone of a clone of a clone, and so on), which results in a n-level hierarchy of cloned objects, each with their own portion of shared and independent data storage?

A. TRUE
B. FALSE

**Correct Answer: A**
**Section:**

**QUESTION 46**
Mark the Correct Statements:
Statement 1. Enable failover for a primary database to one or more accounts in your organization using an ALTER DATABASE … ENABLE FAILOVER TO ACCOUNTS statement.
Statement 2. Enabling failover for a primary database can be done by Data Engineer either before or after a replica of the primary database has been created in a specified account.

A. Statement 1
B. Statement 2
C. Both are False.
D. Both are Correct.

**Correct Answer: D**
**Section:**

**QUESTION 47**
Which Role inherits the privileges of the USERADMIN role via the system role hierarchy?

A. SYSADMIN
B. SECURITYADMIN
C. PUBLIC
D. CUSTOM ROLE

**Correct Answer: B**
**Section:**

**QUESTION 48**
Which privilege are required on an object (i.e. user or role) with USERADMIN Role can modify the object properties?

A. OPEARTE
B. MANAGE GRANTS
C. OWNERSHIP
D. MODIFY

**Correct Answer: C**
**Section:**

**QUESTION 49**
Does sensitive data in Snowflake is modified in an existing table while applying Masking policies?

A. YES
B. NO

**Correct Answer: B**
**Section:**
**Explanation:**
Snowflake supports masking policies as a schema-level object to protect sensitive data from unauthorized access while allowing authorized users to access sensitive data at query runtime. This means that sensitive data in Snowflake is not modified in an existing table (i.e. no static masking).
Rather, when users execute a query in which a masking policy applies, the masking policy condi-tions determine whether unauthorized users see masked, partially masked, obfuscated, or tokenized data.

**QUESTION 50**
Can Masking policies be applied to virtual columns?

A. TRUE
B. FALSE

**Correct Answer: B**
**Section:**