Exam Questions ARA-C01

SnowPro Advanced: Architect Certification Exam

https://www.2passeasy.com/dumps/ARA-C01/

NEW QUESTION 1
An Architect is troubleshooting a query with poor performance using the QUERY function. The Architect observes that the COMPILATION_TIME Is greater than the EXECUTION_TIME.
What is the reason for this?

A. The query is processing a very large dataset.
B. The query has overly complex logic.
C. The query Is queued for execution.
D. The query Is reading from remote storage

Answer: B

Explanation:
? The correct answer is B because the compilation time is the time it takes for the optimizer to create an optimal query plan for the efficient execution of the query. The compilation time depends on the complexity of the query, such as the number of tables, columns, joins, filters, aggregations, subqueries, etc. The more complex the query, the longer it takes to compile.
? Option A is incorrect because the query processing time is not affected by the size of the dataset, but by the size of the virtual warehouse. Snowflake automatically scales the compute resources to match the data volume and parallelizes the query execution. The size of the dataset may affect the execution time, but not the compilation time.
? Option C is incorrect because the query queue time is not part of the compilation time or the execution time. It is a separate metric that indicates how long the query waits for a warehouse slot before it starts running. The query queue time depends on the warehouse load, concurrency, and priority settings.
? Option D is incorrect because the query remote IO time is not part of the compilation time or the execution time. It is a separate metric that indicates how long the query spends reading data from remote storage, such as S3 or Azure Blob Storage. The query remote IO time depends on the network latency, bandwidth, and caching efficiency. References:
? Understanding Why Compilation Time in Snowflake Can Be Higher than Execution Time: This article explains why the total duration (compilation + execution) time is an essential metric to measure query performance in Snowflake. It discusses the reasons for the long compilation time, including query complexity and the number of tables and columns.
? Exploring Execution Times: This document explains how to examine the past performance of queries and tasks using Snowsight or by writing queries against views in the ACCOUNT_USAGE schema. It also describes the different metrics and dimensions that affect query performance, such as duration, compilation, execution, queue, and remote IO time.
? What is the ??compilation time?? and how to optimize it?: This community post
provides some tips and best practices on how to reduce the compilation time, such as simplifying the query logic, using views or common table expressions, and avoiding unnecessary columns or joins.

NEW QUESTION 2
Is it possible for a data provider account with a Snowflake Business Critical edition to share data with an Enterprise edition data consumer account?

A. A Business Critical account cannot be a data sharing provider to an Enterprise consume
B. Any consumer accounts must also be Business Critical.
C. If a user in the provider account with role authority to create or alter share adds an Enterprise account as a consumer, it can import the share.
D. If a user in the provider account with a share owning role sets share_restrictions to False when adding an Enterprise consumer account, it can import the share.
E. If a user in the provider account with a share owning role which also has override share restrictions privilege share_restrictions set to False when adding an Enterprise consumer account, it can import the share.

Answer: B

Explanation:
In Snowflake, data sharing capabilities allow a Business Critical edition account to share data with an Enterprise edition consumer account. The ability to share data is contingent upon the role permissions within the provider account. If a user has the necessary role authority (like ACCOUNTADMIN or a role with similar privileges to create or manage shares), they can add an Enterprise edition account as a consumer. This feature enables flexibility in data sharing across different Snowflake account editions, facilitating broader data collaboration and accessibility.References: Snowflake's data sharing documentation and the specifics of edition-based capabilities discussed in SnowPro Advanced: Architect certification materials.

NEW QUESTION 3
A company??s client application supports multiple authentication methods, and is using Okta.
What is the best practice recommendation for the order of priority when applications authenticate to Snowflake?
A.
1) OAuth (either Snowflake OAuth or External OAuth)
2) External browser
3) Okta native authentication
4) Key Pair Authentication, mostly used for service account users
5) Password
B.
1) External browser, SSO
2) Key Pair Authentication, mostly used for development environment users
3) Okta native authentication
4) OAuth (ether Snowflake OAuth or External OAuth)
5) Password
C.
1) Okta native authentication
2) Key Pair Authentication, mostly used for production environment users
3) Password
4) OAuth (either Snowflake OAuth or External OAuth)
5) External browser, SSO
D.
1) Password
2) Key Pair Authentication, mostly used for production environment users
3) Okta native authentication
4) OAuth (either Snowflake OAuth or External OAuth)

5) External browser, SSO

A.

Answer: A

Explanation:
This is the best practice recommendation for the order of priority when applications authenticate to Snowflake, according to the Snowflake documentation and the web search results. Authentication is the process of verifying the identity of a user or application that connects to Snowflake. Snowflake supports multiple authentication methods, each with different advantages and disadvantages. The recommended order of priority is based on the following factors:
? Security: The authentication method should provide a high level of security and protection against unauthorized access or data breaches. The authentication method should also support multi-factor authentication (MFA) or single sign-on (SSO) for additional security.
? Convenience: The authentication method should provide a smooth and easy user experience, without requiring complex or manual steps. The authentication method should also support seamless integration with external identity providers or applications.
? Flexibility: The authentication method should provide a range of options and features to suit different use cases and scenarios. The authentication method should also support customization and configuration to meet specific requirements.
Based on these factors, the recommended order of priority is:
? OAuth (either Snowflake OAuth or External OAuth): OAuth is an open standard for authorization that allows applications to access Snowflake resources on behalf of a user, without exposing the user??s credentials. OAuth provides a high level of security, convenience, and flexibility, as it supports MFA, SSO, token-based authentication, and various grant types and scopes. OAuth can be implemented using either Snowflake OAuth or External OAuth, depending on the identity provider and the application12.
? External browser: External browser is an authentication method that allows users to log in to Snowflake using a web browser and an external identity provider, such as Okta, Azure AD, or Ping Identity. External browser provides a high level of security and convenience, as it supports MFA, SSO, and federated authentication. External browser also provides a consistent user interface and experience across different platforms and devices34.
? Okta native authentication: Okta native authentication is an authentication method that allows users to log in to Snowflake using Okta as the identity provider, without using a web browser. Okta native authentication provides a high level of security and convenience, as it supports MFA, SSO, and federated authentication. Okta native authentication also provides a native user interface and experience for Okta users, and supports various Okta features, such as password policies and user management56.
? Key Pair Authentication: Key Pair Authentication is an authentication method that
allows users to log in to Snowflake using a public-private key pair, without using a password. Key Pair Authentication provides a high level of security, as it relies on asymmetric encryption and digital signatures. Key Pair Authentication also provides a flexible and customizable authentication option, as it supports various key formats, algorithms, and expiration times. Key Pair Authentication is mostly used for service account users, such as applications or scripts that connect to Snowflake programmatically7 .
? Password: Password is the simplest and most basic authentication method that allows users to log in to Snowflake using a username and password. Password provides a low level of security, as it relies on symmetric encryption and is vulnerable to brute force attacks or phishing. Password also provides a low level of convenience and flexibility, as it requires manual input and management, and does not support MFA or SSO. Password is the least recommended authentication method, and should be used only as a last resort or for testing purposes .
References:
? Snowflake Documentation: Snowflake OAuth
? Snowflake Documentation: External OAuth
? Snowflake Documentation: External Browser Authentication
? Snowflake Blog: How to Use External Browser Authentication with Snowflake
? Snowflake Documentation: Okta Native Authentication
? Snowflake Blog: How to Use Okta Native Authentication with Snowflake
? Snowflake Documentation: Key Pair Authentication
? [Snowflake Blog: How to Use Key Pair Authentication with Snowflake]
? [Snowflake Documentation: Password Authentication]
? [Snowflake Blog: How to Use Password Authentication with Snowflake]

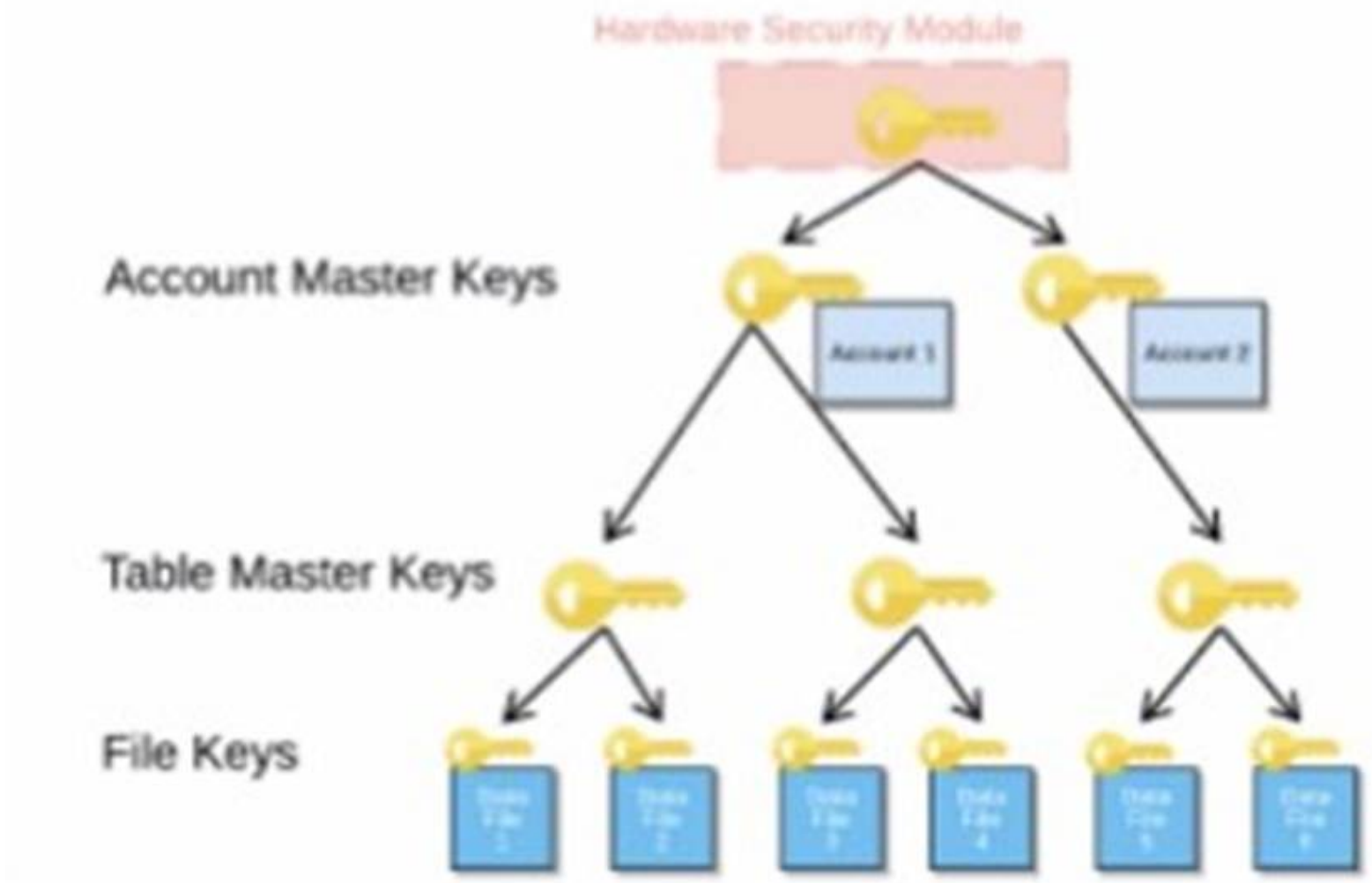
NEW QUESTION 4
A Data Engineer is designing a near real-time ingestion pipeline for a retail company to ingest event logs into Snowflake to derive insights. A Snowflake Architect is asked to define security best practices to configure access control privileges for the data load for auto- ingest to Snowpipe.
What are the MINIMUM object privileges required for the Snowpipe user to execute Snowpipe?

A. OWNERSHIP on the named pipe, USAGE on the named stage, target database, and schema, and INSERT and SELECT on the target table
B. OWNERSHIP on the named pipe, USAGE and READ on the named stage, USAGE on the target database and schema, and INSERT end SELECT on the target table
C. CREATE on the named pipe, USAGE and READ on the named stage, USAGE on the target database and schema, and INSERT end SELECT on the target table
D. USAGE on the named pipe, named stage, target database, and schema, and INSERT and SELECT on the target table

Answer: B

Explanation:
According to the SnowPro Advanced: Architect documents and learning resources, the minimum object privileges required for the Snowpipe user to execute Snowpipe are:
? OWNERSHIP on the named pipe. This privilege allows the Snowpipe user to
create, modify, and drop the pipe object that defines the COPY statement for loading data from the stage to the table1.
? USAGE and READ on the named stage. These privileges allow the Snowpipe user
to access and read the data files from the stage that are loaded by Snowpipe2.
? USAGE on the target database and schema. These privileges allow the Snowpipe user to access the database and schema that contain the target table3.
? INSERT and SELECT on the target table. These privileges allow the Snowpipe user to insert data into the table and select data from the table4.
The other options are incorrect because they do not specify the minimum object privileges required for the Snowpipe user to execute Snowpipe. Option A is incorrect because it does not include the READ privilege on the named stage, which is required for the Snowpipe user to read the data files from the stage. Option C is incorrect because it does not include the OWNERSHIP privilege on the named pipe, which is required for the Snowpipe user to create, modify, and drop the pipe object. Option D is incorrect because it does not include the OWNERSHIP privilege on the named pipe or the READ privilege on the named stage, which are both required for the Snowpipe user to execute Snowpipe. References
: CREATE PIPE | Snowflake Documentation, CREATE STAGE | Snowflake Documentation, CREATE DATABASE | Snowflake Documentation, CREATE TABLE | Snowflake Documentation

NEW QUESTION 5
When activating Tri-Secret Secure in a hierarchical encryption model in a Snowflake account, at what level is the customer-managed key used?



A. At the root level (HSM)
B. At the account level (AMK)
C. At the table level (TMK)
D. At the micro-partition level

Answer: B

Explanation:
Tri-Secret Secure is a feature that allows customers to use their own key, called the customer-managed key (CMK), in addition to the Snowflake-managed key, to create a composite master key that encrypts the data in Snowflake. The composite master key is also known as the account master key (AMK), as it is unique for each account and encrypts the table master keys (TMKs) that encrypt the file keys that encrypt the data files. The customer-managed key is used at the account level, not at the root level, the table level, or the micro-partition level. The root level is protected by a hardware security module (HSM), the table level is protected by the TMKs, and the micro-partition level is protected
by the file keys12. References:
? Understanding Encryption Key Management in Snowflake
? Tri-Secret Secure FAQ for Snowflake on AWS

NEW QUESTION 6
A retail company has over 3000 stores all using the same Point of Sale (POS) system. The company wants to deliver near real-time sales results to category managers. The stores operate in a variety of time zones and exhibit a dynamic range of transactions each minute, with some stores having higher sales volumes than others.
Sales results are provided in a uniform fashion using data engineered fields that will be calculated in a complex data pipeline. Calculations include exceptions, aggregations, and scoring using external functions interfaced to scoring algorithms. The source data for aggregations has over 100M rows.
Every minute, the POS sends all sales transactions files to a cloud storage location with a naming convention that includes store numbers and timestamps to identify the set of transactions contained in the files. The files are typically less than 10MB in size.
How can the near real-time results be provided to the category managers? (Select TWO).

A. All files should be concatenated before ingestion into Snowflake to avoid micro- ingestion.
B. A Snowpipe should be created and configured with AUTO_INGEST = tru
C. A stream should be created to process INSERTS into a single target table using the stream metadata to inform the store number and timestamps.
D. A stream should be created to accumulate the near real-time data and a task should be created that runs at a frequency that matches the real-time analytics needs.
E. An external scheduler should examine the contents of the cloud storage location and issue SnowSQL commands to process the data at a frequency that matches the real-time analytics needs.
F. The copy into command with a task scheduled to run every second should be used to achieve the near-real time requirement.

Answer: BC

Explanation:
To provide near real-time sales results to category managers, the Architect can use the following steps:
? Create an external stage that references the cloud storage location where the POS
sends the sales transactions files. The external stage should use the file format and encryption settings that match the source files2
? Create a Snowpipe that loads the files from the external stage into a target table in
Snowflake. The Snowpipe should be configured with AUTO_INGEST = true, which means that it will automatically detect and ingest new files as they arrive in the external stage. The Snowpipe should also use a copy option to purge the files from the external stage after loading, to avoid duplicate ingestion3
? Create a stream on the target table that captures the INSERTS made by the
Snowpipe. The stream should include the metadata columns that provide information about the file name, path, size, and last modified time. The stream should also have a retention period that matches the real-time analytics needs4
? Create a task that runs a query on the stream to process the near real-time data.

The query should use the stream metadata to extract the store number and timestamps from the file name and path, and perform the calculations for exceptions, aggregations, and scoring using external functions. The query should also output the results to another table or view that can be accessed by the category managers. The task should be scheduled to run at a frequency that matches the real-time analytics needs, such as every minute or every 5 minutes.
The other options are not optimal or feasible for providing near real-time results:
? All files should be concatenated before ingestion into Snowflake to avoid micro-
ingestion. This option is not recommended because it would introduce additional latency and complexity in the data pipeline. Concatenating files would require an external process or service that monitors the cloud storage location and performs the file merging operation. This would delay the ingestion of new files into Snowflake and increase the risk of data loss or corruption. Moreover, concatenating files would not avoid micro-ingestion, as Snowpipe would still ingest each concatenated file as a separate load.
? An external scheduler should examine the contents of the cloud storage location
and issue SnowSQL commands to process the data at a frequency that matches the real-time analytics needs. This option is not necessary because Snowpipe can automatically ingest new files from the external stage without requiring an external trigger or scheduler. Using an external scheduler would add more overhead and dependency to the data pipeline, and it would not guarantee near real-time ingestion, as it would depend on the polling interval and the availability of the external scheduler.
? The copy into command with a task scheduled to run every second should be used
to achieve the near-real time requirement. This option is not feasible because tasks cannot be scheduled to run every second in Snowflake. The minimum interval for tasks is one minute, and even that is not guaranteed, as tasks are subject to scheduling delays and concurrency limits. Moreover, using the copy into command with a task would not leverage the benefits of Snowpipe, such as automatic file detection, load balancing, and micro-partition
optimization. References:
? 1: SnowPro Advanced: Architect | Study Guide
? 2: Snowflake Documentation | Creating Stages
? 3: Snowflake Documentation | Loading Data Using Snowpipe
? 4: Snowflake Documentation | Using Streams and Tasks for ELT
? : Snowflake Documentation | Creating Tasks
? : Snowflake Documentation | Best Practices for Loading Data
? : Snowflake Documentation | Using the Snowpipe REST API
? : Snowflake Documentation | Scheduling Tasks
? : SnowPro Advanced: Architect | Study Guide
? : Creating Stages
? : Loading Data Using Snowpipe
? : Using Streams and Tasks for ELT
? : [Creating Tasks]
? : [Best Practices for Loading Data]
? : [Using the Snowpipe REST API]
? : [Scheduling Tasks]

NEW QUESTION 7
A table for IOT devices that measures water usage is created. The table quickly becomes large and contains more than 2 billion rows.

```
create table water_iot (
UniqueId number,
DeviceId varchar(20),
DeviceManufacturer varchar(50)
CustomerId varchar(20),
IOT_timestamp timestamp_ntz,
City varchar(80),
Location varchar(50)
)
```

The general query patterns for the table are:
* 1. DeviceId, lOT_timestamp and Customerld are frequently used in the filter predicate for the select statement
* 2. The columns City and DeviceManuf acturer are often retrieved
* 3. There is often a count on Uniqueld
Which field(s) should be used for the clustering key?

A. lOT_timestamp
B. City and DeviceManuf acturer
C. Deviceld and Customerld
D. Uniqueld

Answer: C

Explanation:
A clustering key is a subset of columns or expressions that are used to co- locate the data in the same micro-partitions, which are the units of storage in Snowflake. Clustering can improve the performance of queries that filter on the clustering key columns, as it reduces the amount of data that needs to be scanned. The best choice for a clustering key depends on the query patterns and the data distribution in the table. In this case, the columns DeviceId, IOT_timestamp, and CustomerId are frequently used in the filter predicate for the select statement, which means they are good candidates for the clustering key. The columns City and DeviceManufacturer are often retrieved, but not filtered on, so they are not as important for the clustering key. The column UniqueId is used for counting, but it is not a good choice for the clustering key, as it is likely to have a high cardinality and a uniform distribution, which means it will not help to co-locate the data. Therefore, the best option is to use DeviceId and CustomerId as the clustering key, as they can help to prune the micro-partitions and speed up the queries.
References: Clustering Keys & Clustered Tables, Micro-partitions & Data Clustering, A Complete Guide to Snowflake Clustering

NEW QUESTION 8
An Architect needs to grant a group of ORDER_ADMIN users the ability to clean old data in an ORDERS table (deleting all records older than 5 years), without granting any privileges on the table. The group??s manager (ORDER_MANAGER) has full DELETE privileges on the table.
How can the ORDER_ADMIN role be enabled to perform this data cleanup, without needing the DELETE privilege held by the ORDER_MANAGER role?

A. Create a stored procedure that runs with caller??s rights, including the appropriate "> 5 years" business logic, and grant USAGE on this procedure to ORDER_ADMI
B. The ORDER_MANAGER role owns the procedure.
C. Create a stored procedure that can be run using both caller??s and owner??s rights (allowing the user to specify which rights are used during execution), and grant USAGE on this procedure to ORDER_ADMI
D. The ORDER_MANAGER role owns the procedure.
E. Create a stored procedure that runs with owner??s rights, including the appropriate "> 5 years" business logic, and grant USAGE on this procedure to ORDER_ADMI
F. The ORDER_MANAGER role owns the procedure.
G. This scenario would actually not be possible in Snowflake – any user performing a DELETE on a table requires the DELETE privilege to be granted to the role they are using.

Answer: C

Explanation:
This is the correct answer because it allows the ORDER_ADMIN role to perform the data cleanup without needing the DELETE privilege on the ORDERS table. A stored procedure is a feature that allows scheduling and executing SQL statements or
stored procedures in Snowflake. A stored procedure can run with either the caller??s rights or the owner??s rights. A caller??s rights stored procedure runs with the privileges of the role that called the stored procedure, while an owner??s rights stored procedure runs with the privileges of the role that created the stored procedure. By creating a stored procedure that runs with owner??s rights, the ORDER_MANAGER role can delegate the specific task of deleting old data to the ORDER_ADMIN role, without granting the ORDER_ADMIN role more general privileges on the ORDERS table. The stored procedure must include the appropriate business logic to delete only the records older than 5 years, and the ORDER_MANAGER role must grant the USAGE privilege on the stored procedure to the ORDER_ADMIN role. The ORDER_ADMIN role can then execute the stored procedure to perform the data cleanup12.
References:
? Snowflake Documentation: Stored Procedures
? Snowflake Documentation: Understanding Caller??s Rights and Owner??s Rights Stored Procedures

NEW QUESTION 9
A Snowflake Architect is designing a multi-tenant application strategy for an organization in the Snowflake Data Cloud and is considering using an Account Per Tenant strategy.
Which requirements will be addressed with this approach? (Choose two.)

A. There needs to be fewer objects per tenant.
B. Security and Role-Based Access Control (RBAC) policies must be simple to configure.
C. Compute costs must be optimized.
D. Tenant data shape may be unique per tenant.
E. Storage costs must be optimized.

Answer: BD

Explanation:
The Account Per Tenant strategy involves creating separate Snowflake accounts for each tenant within the multi-tenant application. This approach offers a number of advantages.
Option B: With separate accounts, each tenant's environment is isolated, making security and RBAC policies simpler to configure and maintain. This is because each account can have its own set of roles and privileges without the risk of cross-tenant access or the complexity of maintaining a highly granular permission model within a shared environment. Option D: This approach also allows for each tenant to have a unique data shape, meaning that the database schema can be tailored to the specific needs of each tenant without affecting others. This can be essential when tenants have different data models,
usage patterns, or application customizations.

NEW QUESTION 10
What transformations are supported in the below SQL statement? (Select THREE). CREATE PIPE ... AS COPY ... FROM (...)

A. Data can be filtered by an optional where clause.
B. Columns can be reordered.
C. Columns can be omitted.
D. Type casts are supported.
E. Incoming data can be joined with other tables.
F. The ON ERROR - ABORT statement command can be used.

Answer: ABC

Explanation:
? The SQL statement is a command for creating a pipe in Snowflake, which is an object that defines the COPY INTO <table> statement used by Snowpipe to load

data from an ingestion queue into tables1. The statement uses a subquery in the FROM clause to transform the data from the staged files before loading it into the table2.
? The transformations supported in the subquery are as follows2: SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable from (
select * from @mystage where col1 = 'A' and col2 > 10
);
? uk.co.certification.simulator.questionpool.PList@1a11f5a0 SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable (col1, col2, col3) from (
select col3, col1, col2 from @mystage
);
? uk.co.certification.simulator.questionpool.PList@1a11f720 SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable (col1, col2) from (
select col1, col2 from @mystage
);
? The other options are not supported in the subquery because2: SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable (col1, col2) from (
select col1::date, col2 from @mystage
);
? uk.co.certification.simulator.questionpool.PList@1a11fa70 SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable (col1, col2, col3) from (
select s.col1, s.col2, t.col3 from @mystage s join othertable t on s.col1 = t.col1
);
? uk.co.certification.simulator.questionpool.PList@1a11fc60 SQLAI-generated code. Review and use carefully. More info on FAQ. create pipe mypipe as
copy into mytable from (
select * from @mystage on error abort
);
References:
? 1: CREATE PIPE | Snowflake Documentation
? 2: Transforming Data During a Load | Snowflake Documentation

NEW QUESTION 10
An Architect is designing a file ingestion recovery solution. The project will use an internal named stage for file storage. Currently, in the case of an ingestion failure, the Operations team must manually download the failed file and check for errors.
Which downloading method should the Architect recommend that requires the LEAST amount of operational overhead?

A. Use the Snowflake Connector for Python, connect to remote storage and download the file.
B. Use the get command in SnowSQL to retrieve the file.
C. Use the get command in Snowsight to retrieve the file.
D. Use the Snowflake API endpoint and download the file.

Answer: B

Explanation:
The get command in SnowSQL is a convenient way to download files from
an internal stage to a local directory. The get command can be used in interactive mode or in a script, and it supports wildcards and parallel downloads. The get command also allows specifying the overwrite option, which determines how to handle existing files with the
same name2
The Snowflake Connector for Python, the Snowflake API endpoint, and the get command in Snowsight are not recommended methods for downloading files from an internal stage, because they require more operational overhead than the get command in SnowSQL. The Snowflake Connector for Python and the Snowflake API endpoint require writing and maintaining code to handle the connection, authentication, and file transfer. The get command in Snowsight requires using the web interface and manually selecting the files to download34 References:
? 1: SnowPro Advanced: Architect | Study Guide
? 2: Snowflake Documentation | Using the GET Command
? 3: Snowflake Documentation | Using the Snowflake Connector for Python
? 4: Snowflake Documentation | Using the Snowflake API
? : Snowflake Documentation | Using the GET Command in Snowsight
? : SnowPro Advanced: Architect | Study Guide
? : Using the GET Command
? : Using the Snowflake Connector for Python
? : Using the Snowflake API
? : [Using the GET Command in Snowsight]

NEW QUESTION 11
Which of the below commands will use warehouse credits?

A. SHOW TABLES LIKE 'SNOWFL%';
B. SELECT MAX(FLAKE_ID) FROM SNOWFLAKE;
C. SELECT COUNT(*) FROM SNOWFLAKE;
D. SELECT COUNT(FLAKE_ID) FROM SNOWFLAKE GROUP BY FLAKE_ID;

Answer: BCD

Explanation:
? Warehouse credits are used to pay for the processing time used by each virtual warehouse in Snowflake. A virtual warehouse is a cluster of compute resources that enables executing queries, loading data, and performing other DML operations. Warehouse credits are charged based on the number of virtual warehouses you use, how long they run, and their size1.
? Among the commands listed in the question, the following ones will use warehouse credits:
? The command that will not use warehouse credits is:
References: : Understanding Compute Cost : MAX Function : COUNT Function : GROUP BY Clause : SHOW TABLES

NEW QUESTION 13
A company wants to deploy its Snowflake accounts inside its corporate network with no visibility on the internet. The company is using a VPN infrastructure and Virtual Desktop Infrastructure (VDI) for its Snowflake users. The company also wants to re-use the login credentials set up for the VDI to eliminate redundancy when managing logins.
What Snowflake functionality should be used to meet these requirements? (Choose two.)

A. Set up replication to allow users to connect from outside the company VPN.
B. Provision a unique company Tri-Secret Secure key.
C. Use private connectivity from a cloud provider.
D. Set up SSO for federated authentication.
E. Use a proxy Snowflake account outside the VPN, enabling client redirect for user logins.

Answer: CD

Explanation:
According to the SnowPro Advanced: Architect documents and learning resources, the Snowflake functionality that should be used to meet these requirements are:
? Use private connectivity from a cloud provider. This feature allows customers to
connect to Snowflake from their own private network without exposing their data to the public Internet. Snowflake integrates with AWS PrivateLink, Azure Private Link,
and Google Cloud Private Service Connect to offer private connectivity from customers?? VPCs or VNets to Snowflake endpoints. Customers can control how traffic reaches the Snowflake endpoint and avoid the need for proxies or public IP addresses123.
? Set up SSO for federated authentication. This feature allows customers to use
their existing identity provider (IdP) to authenticate users for SSO access to Snowflake. Snowflake supports most SAML 2.0-compliant vendors as an IdP, including Okta, Microsoft AD FS, Google G Suite, Microsoft Azure Active Directory, OneLogin, Ping Identity, and PingOne. By setting up SSO for federated authentication, customers can leverage their existing user credentials and profile information, and provide stronger security than username/password authentication4.
The other options are incorrect because they do not meet the requirements or are not feasible. Option A is incorrect because setting up replication does not allow users to connect from outside the company VPN. Replication is a feature of Snowflake that enables copying databases across accounts in different regions and cloud platforms. Replication does not affect the connectivity or visibility of the accounts5. Option B is incorrect because provisioning a unique company Tri-Secret Secure key does not affect the network or authentication requirements. Tri-Secret Secure is a feature of Snowflake that allows customers to manage their own encryption keys for data at rest in Snowflake, using a combination of three secrets: a master key, a service key, and a security password. Tri- Secret Secure provides an additional layer of security and control over the data encryption and decryption process, but it does not enable private connectivity or SSO6. Option E is incorrect because using a proxy Snowflake account outside the VPN, enabling client redirect for user logins, is not a supported or recommended way of meeting the requirements. Client redirect is a feature of Snowflake that allows customers to connect to a different Snowflake account than the one specified in the connection string. This feature is useful for scenarios such as cross-region failover, data sharing, and account migration, but it does not provide private connectivity or SSO7. References: AWS PrivateLink & Snowflake | Snowflake Documentation, Azure Private Link & Snowflake | Snowflake Documentation, Google Cloud Private Service Connect & Snowflake | Snowflake Documentation, Overview of Federated Authentication and SSO | Snowflake Documentation, Replicating Databases Across Multiple Accounts | Snowflake Documentation, Tri-Secret Secure | Snowflake Documentation, Redirecting Client Connections | Snowflake Documentation

NEW QUESTION 18
What actions are permitted when using the Snowflake SQL REST API? (Select TWO).

A. The use of a GET command
B. The use of a PUT command
C. The use of a ROLLBACK command
D. The use of a CALL command to a stored procedure which returns a table
E. Submitting multiple SQL statements in a single call

Answer: AD

Explanation:
A. The Snowflake SQL REST API does support the use of a GET command, which can be used to retrieve the status of a previously submitted query or to fetch the results of a query once it has been executed.D. The use of a CALL command to a stored procedure is supported, which can return a result set, including a table. This allows the invocation of stored procedures within Snowflake through the SQL REST API.

NEW QUESTION 23
When using the Snowflake Connector for Kafka, what data formats are supported for the messages? (Choose two.)

A. CSV
B. XML
C. Avro
D. JSON
E. Parquet

Answer: CD

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

NEW QUESTION 28
Which Snowflake data modeling approach is designed for BI queries?

A. 3 NF

B. Star schema
C. Data Vault
D. Snowflake schema

Answer: B

Explanation:
In the context of business intelligence (BI) queries, which are typically focused on data analysis and reporting, the star schema is the most suitable data modeling approach.
Option B: Star Schema - The star schema is a type of relational database schema that is widely used for developing data warehouses and data marts for BI purposes. It consists of a central fact table surrounded by dimension tables. The fact table contains the core data metrics, and the dimension tables contain descriptive attributes related to the fact data. The simplicity of the star schema allows for efficient querying and aggregation, which are common operations in BI reporting.

NEW QUESTION 33
What Snowflake features should be leveraged when modeling using Data Vault?

A. Snowflake??s support of multi-table inserts into the data model??s Data Vault tables
B. Data needs to be pre-partitioned to obtain a superior data access performance
C. Scaling up the virtual warehouses will support parallel processing of new source loads
D. Snowflake??s ability to hash keys so that hash key joins can run faster than integer joins

Answer: A

Explanation:
These two features are relevant for modeling using Data Vault on Snowflake. Data Vault is a data modeling approach that organizes data into hubs, links, and satellites. Data Vault is designed to enable high scalability, flexibility, and performance for data integration and analytics. Snowflake is a cloud data platform that supports various data modeling techniques, including Data Vault. Snowflake provides some features that can enhance the Data Vault modeling, such as:
? Snowflake??s support of multi-table inserts into the data model??s Data Vault tables.
Multi-table inserts (MTI) are a feature that allows inserting data from a single query into multiple tables in a single DML statement. MTI can improve the performance and efficiency of loading data into Data Vault tables, especially for real-time or near-real-time data integration. MTI can also reduce the complexity and maintenance of the loading code, as well as the data duplication and latency12.
? Scaling up the virtual warehouses will support parallel processing of new source
loads. Virtual warehouses are a feature that allows provisioning compute resources on demand for data processing. Virtual warehouses can be scaled up or down by changing the size of the warehouse, which determines the number of servers in the warehouse. Scaling up the virtual warehouses can improve the performance and concurrency of processing new source loads into Data Vault
tables, especially for large or complex data sets. Scaling up the virtual warehouses can also leverage the parallelism and distribution of Snowflake??s architecture, which can optimize the data loading and querying34.
References:
? Snowflake Documentation: Multi-table Inserts
? Snowflake Blog: Tips for Optimizing the Data Vault Architecture on Snowflake
? Snowflake Documentation: Virtual Warehouses
? Snowflake Blog: Building a Real-Time Data Vault in Snowflake

NEW QUESTION 36
How can the Snowpipe REST API be used to keep a log of data load history?

A. Call insertReport every 20 minutes, fetching the last 10,000 entries.
B. Call loadHistoryScan every minute for the maximum time range.
C. Call insertReport every 8 minutes for a 10-minute time range.
D. Call loadHistoryScan every 10 minutes for a 15-minute time range.

Answer: D

Explanation:
? Snowpipe is a service that automates and optimizes the loading of data from external stages into Snowflake tables. Snowpipe uses a queue to ingest files as they become available in the stage. Snowpipe also provides REST endpoints to load data and retrieve load history reports1.
? The loadHistoryScan endpoint returns the history of files that have been ingested by Snowpipe within a specified time range. The endpoint accepts the following parameters2:
? The loadHistoryScan endpoint can be used to keep a log of data load history by calling it periodically with a suitable time range. The best option among the choices is D, which is to call loadHistoryScan every 10 minutes for a 15-minute time range. This option ensures that the endpoint is called frequently enough to capture the latest files that have been ingested, and that the time range is wide enough to avoid missing any files that may have been delayed or retried by Snowpipe. The other options are either too infrequent, too narrow, or use the wrong endpoint3.
References:
? 1: Introduction to Snowpipe | Snowflake Documentation
? 2: loadHistoryScan | Snowflake Documentation
? 3: Monitoring Snowpipe Load History | Snowflake Documentation

NEW QUESTION 40
An Architect is designing a data lake with Snowflake. The company has structured, semi- structured, and unstructured data. The company wants to save the data inside the data lake within the Snowflake
system. The company is planning on sharing data among its corporate branches using Snowflake data sharing.
What should be considered when sharing the unstructured data within Snowflake?

A. A pre-signed URL should be used to save the unstructured data into Snowflake in order to share data over secure views, with no time limit for the URL.
B. A scoped URL should be used to save the unstructured data into Snowflake in order to share data over secure views, with a 24-hour time limit for the URL.
C. A file URL should be used to save the unstructured data into Snowflake in order to share data over secure views, with a 7-day time limit for the URL.
D. A file URL should be used to save the unstructured data into Snowflake in order to share data over secure views, with the "expiration_time" argument defined for the URL time limit.

Answer: B

Explanation:
When sharing unstructured data within Snowflake, using a scoped URL is recommended. Scoped URLs provide temporary access to staged files without granting privileges to the stage itself, enhancing security. The URL expires when the persisted query result period ends, which is currently set to 24 hours. This approach is suitable for sharing unstructured data over secure views within Snowflake??s data sharing framework. References: The answer is based on Snowflake??s official documentation regarding the sharing of unstructured data and the use of scoped URLs1.

NEW QUESTION 41
A company has an inbound share set up with eight tables and five secure views. The company plans to make the share part of its production data pipelines.
Which actions can the company take with the inbound share? (Choose two.)

A. Clone a table from a share.
B. Grant modify permissions on the share.
C. Create a table from the shared database.
D. Create additional views inside the shared database.
E. Create a table stream on the shared table.

Answer: AD

Explanation:
These two actions are possible with an inbound share, according to the Snowflake documentation and the web search results. An inbound share is a share that is created by another Snowflake account (the provider) and imported into your account (the consumer). An inbound share allows you to access the data shared by the provider, but not to modify or delete it. However, you can perform some actions with the inbound share, such as:
? Clone a table from a share. You can create a copy of a table from an inbound share using the CREATE TABLE ?? CLONE statement. The clone will contain the same data and metadata as the original table, but it will be independent of the share. You can modify or delete the clone as you wish, but it will not reflect any changes made to the original table by the provider1.
? Create additional views inside the shared database. You can create views on the tables or views from an inbound share using the CREATE VIEW statement. The views will be stored in the shared database, but they will be owned by your account. You can query the views as you would query any other view in your account, but you cannot modify or delete the underlying objects from the share2.
The other actions listed are not possible with an inbound share, because they would require modifying the share or the shared objects, which are read-only for the consumer. You cannot grant modify permissions on the share, create a table from the shared database, or create a table stream on the shared table34.
References:
? Cloning Objects from a Share | Snowflake Documentation
? Creating Views on Shared Data | Snowflake Documentation
? Importing Data from a Share | Snowflake Documentation
? Streams on Shared Tables | Snowflake Documentation

NEW QUESTION 43
A company has a Snowflake account named ACCOUNTA in AWS us-east-1 region. The company stores its marketing data in a Snowflake database named MARKET_DB. One of the company??s business partners has an account named PARTNERB in Azure East US 2 region. For marketing purposes the company has agreed to share the database MARKET_DB with the partner account.
Which of the following steps MUST be performed for the account PARTNERB to consume
data from the MARKET_DB database?

A. Create a new account (called AZABC123) in Azure East US 2 regio
B. From account ACCOUNTA create a share of database MARKET_DB, create a new database out of this share locally in AWS us-east-1 region, and replicate this new database to AZABC123 accoun
C. Then set up data sharing to the PARTNERB account.
D. From account ACCOUNTA create a share of database MARKET_DB, and create a new database out of this share locally in AWS us-east-1 regio
E. Then make this database the provider and share it with the PARTNERB account.
F. Create a new account (called AZABC123) in Azure East US 2 regio
G. From account ACCOUNTA replicate the database MARKET_DB to AZABC123 and from this account set up the data sharing to the PARTNERB account.
H. Create a share of database MARKET_DB, and create a new database out of this share locally in AWS us-east-1 regio
I. Then replicate this database to the partner??s account PARTNERB.

Answer: C

Explanation:
? Snowflake supports data sharing across regions and cloud platforms using account replication and share replication features. Account replication enables the replication of objects from a source account to one or more target accounts in the same organization. Share replication enables the replication of shares from a source account to one or more target accounts in the same organization1.
? To share data from the MARKET_DB database in the ACCOUNTA account in AWS us-east-1 region with the PARTNERB account in Azure East US 2 region, the following steps must be performed:
? Therefore, option C is the correct answer.
References: : Replicating Shares Across Regions and Cloud Platforms : Working with Organizations and Accounts : Replicating Databases Across Multiple Accounts : Replicating Shares Across Multiple Accounts

NEW QUESTION 48
What built-in Snowflake features make use of the change tracking metadata for a table? (Choose two.)

A. The MERGE command
B. The UPSERT command
C. The CHANGES clause
D. A STREAM object
E. The CHANGE_DATA_CAPTURE command

Answer: AD

Explanation:

In Snowflake, the change tracking metadata for a table is utilized by the MERGE command and the STREAM object. The MERGE command uses change tracking to determine how to apply updates and inserts efficiently based on differences between source and target tables. STREAM objects, on the other hand, specifically capture and store change data, enabling incremental processing based on changes made to a table since the last stream offset was committed.References: Snowflake Documentation on MERGE and STREAM Objects.

NEW QUESTION 53
A user can change object parameters using which of the following roles?

A. ACCOUNTADMIN, SECURITYADMIN
B. SYSADMIN, SECURITYADMIN
C. ACCOUNTADMIN, USER with PRIVILEGE
D. SECURITYADMIN, USER with PRIVILEGE

Answer: C

Explanation:
According to the Snowflake documentation, object parameters are parameters that can be set on individual objects such as databases, schemas, tables, and stages. Object parameters can be set by users with the appropriate privileges on the objects. For example, to set the object parameter AUTO_REFRESH on a table, the user must have the MODIFY privilege on the table. The ACCOUNTADMIN role has the highest level of privileges on all objects in the account, so it can set any object parameter on any
object. However, other roles, such as SECURITYADMIN or SYSADMIN, do not have the same level of privileges on all objects, so they cannot set object parameters on objects they do not own or have the required privileges on. Therefore, the correct answer is C. ACCOUNTADMIN, USER with PRIVILEGE.
References:
? Parameters | Snowflake Documentation
? Object Parameters | Snowflake Documentation
? Object Privileges | Snowflake Documentation

NEW QUESTION 57
An Architect has chosen to separate their Snowflake Production and QA environments using two separate Snowflake accounts.
The QA account is intended to run and test changes on data and database objects before pushing those changes to the Production account. It is a requirement that all database objects and data in the QA account need to be an exact copy of the database objects, including privileges and data in the Production account on at least a nightly basis.
Which is the LEAST complex approach to use to populate the QA account with the Production account??s data and database objects on a nightly basis?
A.
1) Create a share in the Production account for each database
2) Share access to the QA account as a Consumer
3) The QA account creates a database directly from each share
4) Create clones of those databases on a nightly basis
5) Run tests directly on those cloned databases
B.
1) Create a stage in the Production account
2) Create a stage in the QA account that points to the same external object-storage
location
3) Create a task that runs nightly to unload each table in the Production account into the stage
4) Use Snowpipe to populate the QA account
C.
1) Enable replication for each database in the Production account
2) Create replica databases in the QA account
3) Create clones of the replica databases on a nightly basis
4) Run tests directly on those cloned databases
D.
1) In the Production account, create an external function that connects into the QA account and returns all the data for one specific table
2) Run the external function as part of a stored procedure that loops through each table in the Production account and populates each table in the QA account
A.

Answer: C

Explanation:
This approach is the least complex because it uses Snowflake??s built-in replication feature to copy the data and database objects from the Production account to the QA account. Replication is a fast and efficient way to synchronize data across accounts, regions, and cloud platforms. It also preserves the privileges and metadata of the replicated objects. By creating clones of the replica databases, the QA account can run tests on the cloned data without affecting the original data. Clones are also zero-copy, meaning they do not consume any additional storage space unless the data is modified. This approach does not require any external stages, tasks, Snowpipe, or external functions, which can add complexity and overhead to the data transfer process. References:
? Introduction to Replication and Failover
? Replicating Databases Across Multiple Accounts
? Cloning Considerations

NEW QUESTION 59
A company is designing its serving layer for data that is in cloud storage. Multiple terabytes of the data will be used for reporting. Some data does not have a clear use case but could be useful for experimental analysis. This experimentation data changes frequently and is sometimes wiped out and replaced completely in a few days.
The company wants to centralize access control, provide a single point of connection for the end-users, and maintain data governance.
What solution meets these requirements while MINIMIZING costs, administrative effort, and development overhead?

A. Import the data used for reporting into a Snowflake schema with native table
B. Then create external tables pointing to the cloud storage folders used for the experimentation dat
C. Then create two different roles with grants to the different datasets to match the different user personas, and grant these roles to the corresponding users.
D. Import all the data in cloud storage to be used for reporting into a Snowflake schema with native table
E. Then create a role that has access to this schema and manage access to the data through that role.

F. Import all the data in cloud storage to be used for reporting into a Snowflake schema with native table
G. Then create two different roles with grants to the different datasets to match the different user personas, and grant these roles to the corresponding users.
H. Import the data used for reporting into a Snowflake schema with native table
I. Then create views that have SELECT commands pointing to the cloud storage files for the experimentation dat
J. Then create two different roles to match the different user personas, and grant these roles to the corresponding users.

Answer: A

Explanation:
The most cost-effective and administratively efficient solution is to use a combination of native and external tables. Native tables for reporting data ensure performance and governance, while external tables allow for flexibility with frequently changing experimental data. Creating roles with specific grants to datasets aligns with the principle of least privilege, centralizing access control and simplifying user management12. References
•Snowflake Documentation on Optimizing Cost1.
•Snowflake Documentation on Controlling Cost2.

NEW QUESTION 63
A user has activated primary and secondary roles for a session.
What operation is the user prohibited from using as part of SQL actions in Snowflake using the secondary role?

A. Insert
B. Create
C. Delete
D. Truncate

Answer: B

Explanation:
In Snowflake, when a user activates a secondary role during a session, certain privileges associated with DDL (Data Definition Language) operations are restricted. The CREATE statement, which falls under DDL operations, cannot be executed using a secondary role. This limitation is designed to enforce role-based access control and ensure that schema modifications are managed carefully, typically reserved for primary roles that have explicit permissions to modify database structures.References: Snowflake's security and access control documentation specifying the limitations and capabilities of primary versus secondary roles in session management.

NEW QUESTION 65
Which statements describe characteristics of the use of materialized views in Snowflake? (Choose two.)

A. They can include ORDER BY clauses.
B. They cannot include nested subqueries.
C. They can include context functions, such as CURRENT_TIME().
D. They can support MIN and MAX aggregates.
E. They can support inner joins, but not outer joins.

Answer: BD

Explanation:
According to the Snowflake documentation, materialized views have some limitations on the query specification that defines them. One of these limitations is that they cannot include nested subqueries, such as subqueries in the FROM clause or scalar subqueries in the SELECT list. Another limitation is that they cannot include ORDER BY clauses, context functions (such as CURRENT_TIME()), or outer joins. However, materialized views can support MIN and MAX aggregates, as well as other aggregate functions, such as SUM, COUNT, and AVG.
References:
? Limitations on Creating Materialized Views | Snowflake Documentation
? Working with Materialized Views | Snowflake Documentation

NEW QUESTION 70
How is the change of local time due to daylight savings time handled in Snowflake tasks? (Choose two.)

A. A task scheduled in a UTC-based schedule will have no issues with the time changes.
B. Task schedules can be designed to follow specified or local time zones to accommodate the time changes.
C. A task will move to a suspended state during the daylight savings time change.
D. A frequent task execution schedule like minutes may not cause a problem, but will affect the task history.
E. A task schedule will follow only the specified time and will fail to handle lost or duplicated hours.

Answer: AB

Explanation:
According to the Snowflake documentation1 and the web search results2, these two statements are true about how the change of local time due to daylight savings time is handled in Snowflake tasks. A task is a feature that allows scheduling and executing
SQL statements or stored procedures in Snowflake. A task can be scheduled using a cron expression that specifies the frequency and time zone of the task execution.
? A task scheduled in a UTC-based schedule will have no issues with the time
changes. UTC is a universal time standard that does not observe daylight savings time. Therefore, a task that uses UTC as the time zone will run at the same time throughout the year, regardless of the local time changes1.
? Task schedules can be designed to follow specified or local time zones to
accommodate the time changes. Snowflake supports using any valid IANA time zone identifier in the cron expression for a task. This allows the task to run according to the local time of the specified time zone, which may include daylight savings time adjustments. For example, a task that uses Europe/London as the time zone will run one hour earlier or later when the local time switches between GMT and BST12.
References:
? Snowflake Documentation: Scheduling Tasks
? Snowflake Community: Do the timezones used in scheduling tasks in Snowflake adhere to daylight savings?

NEW QUESTION 74
A user is executing the following command sequentially within a timeframe of 10 minutes from start to finish:

```
use role sysadmin;
use warehouse compute_wh;
use schema sales.public;
create table t_sales (numeric integer) data_retention_time_in_days=1;
create or replace table t_sales_clone clone t_sales at(offset => -60*30);
```

What would be the output of this query?

A. Table T_SALES_CLONE successfully created.
B. Time Travel data is not available for table T_SALES.
C. The offset -> is not a valid clause in the clone operation.
D. Syntax error line 1 at position 58 unexpected 'at??.

Answer: A

Explanation:
The query is executing a clone operation on an existing table t_sales with an offset to account for the retention time. The syntax used is correct for cloning a table in Snowflake, and the use of the at(offset => -60*30) clause is valid. This specifies that the clone should be based on the state of the table 30 minutes prior (60 seconds * 30).
Assuming the table t_sales exists and has been modified within the last 30 minutes, and considering the data_retention_time_in_days is set to 1 day (which enables time travel queries for the past 24 hours), the table t_sales_clone would be successfully created based on the state of t_sales 30 minutes before the clone command was issued.

NEW QUESTION 76
Which command will create a schema without Fail-safe and will restrict object owners from passing on access to other users?

A. create schema EDW.ACCOUNTING WITH MANAGED ACCESS;
B. create schema EDW.ACCOUNTING WITH MANAGED ACCESS DATA_RETENTION_TIME_IN_DAYS - 7;
C. create TRANSIENT schema EDW.ACCOUNTING WITH MANAGED ACCESS DATA_RETENTION_TIME_IN_DAYS = 1;
D. create TRANSIENT schema EDW.ACCOUNTING WITH MANAGED ACCESS DATA_RETENTION_TIME_IN_DAYS = 7;

Answer: D

Explanation:
A transient schema in Snowflake is designed without a Fail-safe period, meaning it does not incur additional storage costs once it leaves Time Travel, and it is not protected by Fail-safe in the event of a data loss. The WITH MANAGED ACCESS option ensures that all privilege grants, including future grants on objects within the schema, are managed by the schema owner, thus restricting object owners from passing on access to other users1.
References =
•Snowflake Documentation on creating schemas1
•Snowflake Documentation on configuring access control2
•Snowflake Documentation on understanding and viewing Fail-safe3

NEW QUESTION 77
A company has a source system that provides JSON records for various loT operations. The JSON Is loading directly into a persistent table with a variant field. The data Is quickly growing to 100s of millions of records and performance to becoming an issue. There is a generic access pattern that Is used to filter on the create_date key within the variant field.
What can be done to improve performance?

A. Alter the target table to Include additional fields pulled from the JSON record
B. This would Include a create_date field with a datatype of time stam
C. When this field Is used in the filter, partition pruning will occur.
D. Alter the target table to include additional fields pulled from the JSON record
E. This would include a create_date field with a datatype of varcha
F. When this field is used in the filter, partition pruning will occur.
G. Validate the size of the warehouse being use
H. If the record count is approaching 100s of millions, size XL will be the minimum size required to process this amount of data.
I. Incorporate the use of multiple tables partitioned by date range
J. When a user or process needs to query a particular date range, ensure the appropriate base table Is used.

Answer: A

Explanation:
? The correct answer is A because it improves the performance of queries by reducing the amount of data scanned and processed. By adding a create_date field with a timestamp data type, Snowflake can automatically cluster the table based on this field and prune the micro-partitions that do not match the filter condition. This avoids the need to parse the JSON data and access the variant field for every record.
? Option B is incorrect because it does not improve the performance of queries. By adding a create_date field with a varchar data type, Snowflake cannot automatically cluster the table based on this field and prune the micro-partitions that do not match the filter condition. This still requires parsing the JSON data and accessing the variant field for every record.
? Option C is incorrect because it does not address the root cause of the performance issue. By validating the size of the warehouse being used, Snowflake can adjust the compute resources to match the data volume and parallelize the query execution. However, this does not reduce the amount of data scanned and processed, which is the main bottleneck for queries on JSON data.
? Option D is incorrect because it adds unnecessary complexity and overhead to the data loading and querying process. By incorporating the use of multiple tables partitioned by date ranges, Snowflake can reduce the amount of data scanned and processed for queries that specify a date range. However, this requires creating and maintaining multiple tables, loading data into the appropriate table based on
the date, and joining the tables for queries that span multiple date ranges. References:

? Snowflake Documentation: Loading Data Using Snowpipe: This document explains how to use Snowpipe to continuously load data from external sources into Snowflake tables. It also describes the syntax and usage of the COPY INTO command, which supports various options and parameters to control the loading behavior, such as ON_ERROR, PURGE, and SKIP_FILE.
? Snowflake Documentation: Date and Time Data Types and Functions: This document explains the different data types and functions for working with date and time values in Snowflake. It also describes how to set and change the session timezone and the system timezone.
? Snowflake Documentation: Querying Metadata: This document explains how to query the metadata of the objects and operations in Snowflake using various functions, views, and tables. It also describes how to access the copy history information using the COPY_HISTORY function or the COPY_HISTORY view.
? Snowflake Documentation: Loading JSON Data: This document explains how to load JSON data into Snowflake tables using various methods, such as the COPY INTO command, the INSERT command, or the PUT command. It also describes how to access and query JSON data using the dot notation, the FLATTEN function, or the LATERAL join.
? Snowflake Documentation: Optimizing Storage for Performance: This document explains how to optimize the storage of data in Snowflake tables to improve the performance of queries. It also describes the concepts and benefits of automatic clustering, search optimization service, and materialized views.

NEW QUESTION 79
What step will im the performance of queries executed against an external table?

A. Partition the external table.
B. Shorten the names of the source files.
C. Convert the source files' character encoding to UTF-8.
D. Use an internal stage instead of an external stage to store the source files.

Answer: A

Explanation:
Partitioning an external table is a technique that improves the performance of queries executed against the table by reducing the amount of data scanned. Partitioning an external table involves creating one or more partition columns that define how the table is logically divided into subsets of data based on the values in those columns. The partition columns can be derived from the file metadata (such as file name, path, size, or modification time) or from the file content (such as a column value or a JSON attribute). Partitioning an external table allows the query optimizer to prune the files that do not match the query predicates, thus avoiding unnecessary data scanning and processing2 The other options are not effective steps for improving the performance of queries executed against an external table:
? Shorten the names of the source files. This option does not have any impact on the query performance, as the file names are not used for query processing. The file names are only used for creating the external table and displaying the query results3
? Convert the source files?? character encoding to UTF-8. This option does not affect the query performance, as Snowflake supports various character encodings for external table files, such as UTF-8, UTF-16, UTF-32, ISO-8859-1, and Windows- 1252. Snowflake automatically detects the character encoding of the files and converts them to UTF-8 internally for query processing4
? Use an internal stage instead of an external stage to store the source files. This option is not applicable, as external tables can only reference files stored in external stages, such as Amazon S3, Google Cloud Storage, or Azure Blob Storage. Internal stages are used for loading data into internal tables, not external tables5 References:
? 1: SnowPro Advanced: Architect | Study Guide
? 2: Snowflake Documentation | Partitioning External Tables
? 3: Snowflake Documentation | Creating External Tables
? 4: Snowflake Documentation | Supported File Formats and Compression for Staged Data Files
? 5: Snowflake Documentation | Overview of Stages
? : SnowPro Advanced: Architect | Study Guide
? : Partitioning External Tables
? : Creating External Tables
? : Supported File Formats and Compression for Staged Data Files
? : Overview of Stages

NEW QUESTION 82
Which feature provides the capability to define an alternate cluster key for a table with an existing cluster key?

A. External table
B. Materialized view
C. Search optimization
D. Result cache

Answer: B

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

NEW QUESTION 86
A company needs to have the following features available in its Snowflake account:
* 1. Support for Multi-Factor Authentication (MFA)
* 2. A minimum of 2 months of Time Travel availability
* 3. Database replication in between different regions
* 4. Native support for JDBC and ODBC
* 5. Customer-managed encryption keys using Tri-Secret Secure
* 6. Support for Payment Card Industry Data Security Standards (PCI DSS)

In order to provide all the listed services, what is the MINIMUM Snowflake edition that should be selected during account creation?

A. Standard
B. Enterprise
C. Business Critical
D. Virtual Private Snowflake (VPS)

Answer: C

Explanation:
According to the Snowflake documentation1, the Business Critical edition offers the following features that are relevant to the question:
? Support for Multi-Factor Authentication (MFA): This is a standard feature available in all Snowflake editions1.
? A minimum of 2 months of Time Travel availability: This is an enterprise feature that allows users to access historical data for up to 90 days1.
? Database replication in between different regions: This is an enterprise feature that enables users to replicate databases across different regions or cloud platforms1.
? Native support for JDBC and ODBC: This is a standard feature available in all Snowflake editions1.
? Customer-managed encryption keys using Tri-Secret Secure: This is a business critical feature that provides enhanced security and data protection by allowing customers to manage their own encryption keys1.
? Support for Payment Card Industry Data Security Standards (PCI DSS): This is a business critical feature that ensures compliance with PCI DSS regulations for handling sensitive cardholder data1.
Therefore, the minimum Snowflake edition that should be selected during account creation to provide all the listed services is the Business Critical edition.
References:
? Snowflake Editions | Snowflake Documentation

NEW QUESTION 90
A company is designing high availability and disaster recovery plans and needs to maximize redundancy and minimize recovery time objectives for their critical application processes. Cost is not a concern as long as the solution is the best available. The plan so far consists of the following steps:
* 1. Deployment of Snowflake accounts on two different cloud providers.
* 2. Selection of cloud provider regions that are geographically far apart.
* 3. The Snowflake deployment will replicate the databases and account data between both cloud provider accounts.
* 4. Implementation of Snowflake client redirect.
What is the MOST cost-effective way to provide the HIGHEST uptime and LEAST application disruption if there is a service event?

A. Connect the applications using the <organization_name>-<connection_name> UR
B. Use the Business Critical Snowflake edition.
C. Connect the applications using the <organization_name>-<connection_name> UR
D. Use the Virtual Private Snowflake (VPS) edition.
E. Connect the applications using the <organization_name>-<accountLocator> UR
F. Use the Enterprise Snowflake edition.
G. Connect the applications using the <organization_name>-<accountLocator> UR
H. Use the Business Critical Snowflake edition.

Answer: D

Explanation:
To provide the highest uptime and least application disruption in case of a service event, the best option is to use the Business Critical Snowflake edition and connect the applications using the <organization_name>-<accountLocator> URL. The Business Critical Snowflake edition offers the highest level of security, performance, and availability for Snowflake accounts. It includes features such as customer-managed encryption keys, HIPAA compliance, and 4-hour RPO and RTO SLAs. It also supports account replication and failover across regions and cloud platforms, which enables business continuity and disaster recovery. By using the <organization_name>-<accountLocator> URL, the applications can leverage the Snowflake Client Redirect feature, which automatically redirects the client connections to the secondary account in case of a failover. This way, the applications can seamlessly switch to the backup account without any manual intervention or configuration changes. The other options are less cost-effective or less reliable because they either use a lower edition of Snowflake, which does not support account replication and failover, or they use the <organization_name>-<connection_name> URL, which does not support client redirect and requires manual updates to the connection string in case of a failover. References:
? [Snowflake Editions] 1
? [Replication and Failover/Failback] 2
? [Client Redirect] 3
? [Snowflake Account Identifiers] 4

NEW QUESTION 92
Assuming all Snowflake accounts are using an Enterprise edition or higher, in which development and testing scenarios would be copying of data be required, and zero-copy cloning not be suitable? (Select TWO).

A. Developers create their own datasets to work against transformed versions of the live data.
B. Production and development run in different databases in the same account, and Developers need to see production-like data but with specific columns masked.
C. Data is in a production Snowflake account that needs to be provided to Developers in a separate development/testing Snowflake account in the same cloud region.
D. Developers create their own copies of a standard test database previously created for them in the development account, for their initial development and unit testing.
E. The release process requires pre-production testing of changes with data of production scale and complexit
F. For security reasons, pre-production also runs in the production account.

Answer: AC

Explanation:
Zero-copy cloning is a feature that allows creating a clone of a table, schema, or database without physically copying the data. Zero-copy cloning is suitable for scenarios where the cloned object needs to have the same data and metadata as the original object, and where the cloned object does not need to be modified or updated frequently. Zero-copy cloning is also suitable for scenarios where the cloned object needs to be shared within the same Snowflake account or across different accounts in the same cloud region2

However, zero-copy cloning is not suitable for scenarios where the cloned object needs to have different data or metadata than the original object, or where the cloned object needs to be modified or updated frequently. Zero-copy cloning is also not suitable for scenarios where the cloned object needs to be shared across different accounts in different cloud regions. In these scenarios, copying of data would be required, either by using the COPY INTO command or by using data sharing with secure views3
The following are examples of development and testing scenarios where copying of data would be required, and zero-copy cloning would not be suitable:
? Developers create their own datasets to work against transformed versions of the
live data. This scenario requires copying of data because the developers need to modify the data or metadata of the cloned object to perform transformations, such as adding, deleting, or updating columns, rows, or values. Zero-copy cloning would not be suitable because it would create a read-only clone that shares the same data and metadata as the original object, and any changes made to the clone would affect the original object as well4
? Data is in a production Snowflake account that needs to be provided to Developers
in a separate development/testing Snowflake account in the same cloud region. This scenario requires copying of data because the data needs to be shared across different accounts in the same cloud region. Zero-copy cloning would not be suitable because it would create a clone within the same account as the original object, and it would not allow sharing the clone with another account. To share data across different accounts in the same cloud region, data sharing with secure views or COPY INTO command can be used5
The following are examples of development and testing scenarios where zero-copy cloning would be suitable, and copying of data would not be required:
? Production and development run in different databases in the same account, and
Developers need to see production-like data but with specific columns masked. This scenario can use zero-copy cloning because the data needs to be shared within the same account, and the cloned object does not need to have different data or metadata than the original object. Zero-copy cloning can create a clone of the production database in the development database, and the clone can have the same data and metadata as the original database. To mask specific columns, secure views can be created on top of the clone, and the developers can access the secure views instead of the clone directly6
? Developers create their own copies of a standard test database previously created
for them in the development account, for their initial development and unit testing. This scenario can use zero-copy cloning because the data needs to be shared within the same account, and the cloned object does not need to have different data or metadata than the original object. Zero-copy cloning can create a clone of the standard test database for each developer, and the clone can have the same data and metadata as the original database. The developers can use the clone for their initial development and unit testing, and any changes made to the clone
would not affect the original database or other clones7
? The release process requires pre-production testing of changes with data of production scale and complexity. For security reasons, pre-production also runs in the production account. This scenario can use zero-copy cloning because the data needs to be shared within the same account, and the cloned object does not need to have different data or metadata than the original object. Zero-copy cloning can create a clone of the production database in the pre-production database, and the clone can have the same data and metadata as the original database. The pre- production testing can use the clone to test the changes with data of production scale and complexity, and any changes made to the clone would not affect the original database or the production environment8 References:
? 1: SnowPro Advanced: Architect | Study Guide 9
? 2: Snowflake Documentation | Cloning Overview
? 3: Snowflake Documentation | Loading Data Using COPY into a Table
? 4: Snowflake Documentation | Transforming Data During a Load
? 5: Snowflake Documentation | Data Sharing Overview
? 6: Snowflake Documentation | Secure Views
? 7: Snowflake Documentation | Cloning Databases, Schemas, and Tables
? 8: Snowflake Documentation | Cloning for Testing and Development
? : SnowPro Advanced: Architect | Study Guide
? : Cloning Overview
? : Loading Data Using COPY into a Table
? : Transforming Data During a Load
? : Data Sharing Overview
? : Secure Views
? : Cloning Databases, Schemas, and Tables
? : Cloning for Testing and Development

NEW QUESTION 97
A Snowflake Architect is designing a multiple-account design strategy.
This strategy will be MOST cost-effective with which scenarios? (Select TWO).

A. The company wants to clone a production database that resides on AWS to a development database that resides on Azure.
B. The company needs to share data between two databases, where one must support Payment Card Industry Data Security Standard (PCI DSS) compliance but the other one does not.
C. The company needs to support different role-based access control features for the development, test, and production environments.
D. The company security policy mandates the use of different Active Directory instances for the development, test, and production environments.
E. The company must use a specific network policy for certain users to allow and block given IP addresses.

Answer: BD

Explanation:
B. When dealing with PCI DSS compliance, having separate accounts can be beneficial because it enables strong isolation of environments that handle sensitive data from those that do not. By segregating the compliant from non-compliant resources, an organization can limit the scope of compliance, thus making it a cost-effective strategy.D. Different Active Directory instances can be managed more effectively and securely when separated into different accounts. This approach allows for distinct identity and access management policies, which can enforce security requirements and minimize the risk of access policy errors between environments.

NEW QUESTION 98
A company has a table with that has corrupted data, named Data. The company wants to recover the data as it was 5 minutes ago using cloning and Time Travel.
What command will accomplish this?

A. CREATE CLONE TABLE Recover_Data FROM Data AT(OFFSET => -60*5);
B. CREATE CLONE Recover_Data FROM Data AT(OFFSET => -60*5);
C. CREATE TABLE Recover_Data CLONE Data AT(OFFSET => -60*5);
D. CREATE TABLE Recover Data CLONE Data AT(TIME => -60*5);

Answer: C

Explanation:
This is the correct command to create a clone of the table Data as it was 5 minutes ago using cloning and Time Travel. Cloning is a feature that allows creating a

copy of a database, schema, table, or view without duplicating the data or metadata. Time Travel is a feature that enables accessing historical data (i.e. data that has been changed or deleted) at any point within a defined period. To create a clone of a table at a point in time in the past, the syntax is:
CREATE TABLE <clone_name> CLONE <source_table> AT (OFFSET =>
<offset_in_seconds>);
The OFFSET parameter specifies the time difference in seconds from the present time. A negative value indicates a point in the past. For example, -60*5 means 5 minutes ago. Alternatively, the TIMESTAMP parameter can be used to specify an exact timestamp in the past. The clone will contain the data as it existed in the source table at the specified point in time12.
References:
? Snowflake Documentation: Cloning Objects
? Snowflake Documentation: Cloning Objects at a Point in Time in the Past

NEW QUESTION 101
A company has an external vendor who puts data into Google Cloud Storage. The company's Snowflake account is set up in Azure.
What would be the MOST efficient way to load data from the vendor into Snowflake?

A. Ask the vendor to create a Snowflake account, load the data into Snowflake and create a data share.
B. Create an external stage on Google Cloud Storage and use the external table to load the data into Snowflake.
C. Copy the data from Google Cloud Storage to Azure Blob storage using external toolsand load data from Blob storage to Snowflake.
D. Create a Snowflake Account in the Google Cloud Platform (GCP), ingest data into this account and use data replication to move the data from GCP to Azure.

Answer: B

Explanation:
The most efficient way to load data from the vendor into Snowflake is to create an external stage on Google Cloud Storage and use the external table to load the data into Snowflake (Option B). This way, you can avoid copying or moving the data across different cloud platforms, which can incur additional costs and latency. You can also leverage the external table feature to query the data directly from Google Cloud Storage without loading it into Snowflake tables, which can save storage space and improve performance. Option A is not efficient because it requires the vendor to create a Snowflake account and a data share, which can be complicated and costly. Option C is not efficient because it involves copying the data from Google Cloud Storage to Azure Blob storage using external tools, which can be slow and expensive. Option D is not efficient because it requires creating a Snowflake account in the Google Cloud Platform (GCP), ingesting data into this account, and using data replication to move the data from GCP to Azure, which can be complex and time-consuming. References: The answer can be verified from Snowflake??s official documentation on external stages and external tables available on their website. Here are some relevant links:
? Using External Stages | Snowflake Documentation
? Using External Tables | Snowflake Documentation
? Loading Data from a Stage | Snowflake Documentation

NEW QUESTION 106
How can the Snowflake context functions be used to help determine whether a user is authorized to see data that has column-level security enforced? (Select TWO).

A. Set masking policy conditions using current_role targeting the role in use for the current session.
B. Set masking policy conditions using is_role_in_session targeting the role in use for the current account.
C. Set masking policy conditions using invoker_role targeting the executing role in a SQL statement.
D. Determine if there are ownership privileges on the masking policy that would allow the use of any function.
E. Assign the accountadmin role to the user who is executing the object.

Answer: AC

Explanation:
Snowflake context functions are functions that return information about the current session, user, role, warehouse, database, schema, or object. They can be used to help determine whether a user is authorized to see data that has column-level security enforced by setting masking policy conditions based on the context functions. The following context functions are relevant for column-level security:
? current_role: This function returns the name of the role in use for the current
session. It can be used to set masking policy conditions that target the current session and are not affected by the execution context of the SQL statement. For example, a masking policy condition using current_role can allow or deny access to a column based on the role that the user activated in the session.
? invoker_role: This function returns the name of the executing role in a SQL
statement. It can be used to set masking policy conditions that target the executing role and are affected by the execution context of the SQL statement. For example, a masking policy condition using invoker_role can allow or deny access to a column based on the role that the user specified in the SQL statement, such as using the AS ROLE clause or a stored procedure.
? is_role_in_session: This function returns TRUE if the user??s current role in the
session (i.e. the role returned by current_role) inherits the privileges of the specified role. It can be used to set masking policy conditions that involve role hierarchy and privilege inheritance. For example, a masking policy condition using is_role_in_session can allow or deny access to a column based on whether the user??s current role is a lower privilege role in the specified role hierarchy.
The other options are not valid ways to use the Snowflake context functions for column- level security:
? Set masking policy conditions using is_role_in_session targeting the role in use for
the current account. This option is incorrect because is_role_in_session does not target the role in use for the current account, but rather the role in use for the current session. Also, the current account is not a role, but rather a logical entity that contains users, roles, warehouses, databases, and other objects.
? Determine if there are ownership privileges on the masking policy that would allow
the use of any function. This option is incorrect because ownership privileges on the masking policy do not affect the use of any function, but rather the ability to create, alter, or drop the masking policy. Also, this is not a way to use the Snowflake context functions, but rather a way to check the privileges on the masking policy object.
? Assign the accountadmin role to the user who is executing the object. This option
is incorrect because assigning the accountadmin role to the user who is executing the object does not involve using the Snowflake context functions, but rather granting the highest-level role to the user. Also, this is not a recommended practice for column-level security, as it would give the user full access to all objects and data in the account, which could compromise data security and governance.
References:
? Context Functions
? Advanced Column-level Security topics
? Snowflake Data Governance: Column Level Security Overview
? Data Security Snowflake Part 2 - Column Level Security

NEW QUESTION 108
Which data models can be used when modeling tables in a Snowflake environment? (Select THREE).

A. Graph model
B. Dimensional/Kimball
C. Data lake
D. Inmon/3NF
E. Bayesian hierarchical model
F. Data vault

Answer: BDF

Explanation:
Snowflake is a cloud data platform that supports various data models for modeling tables in a Snowflake environment. The data models can be classified into two categories: dimensional and normalized. Dimensional data models are designed to optimize query performance and ease of use for business intelligence and analytics. Normalized data models are designed to reduce data redundancy and ensure data integrity for transactional and operational systems. The following are some of the data models that can be used in Snowflake:
? Dimensional/Kimball: This is a popular dimensional data model that uses a star or
snowflake schema to organize data into fact and dimension tables. Fact tables store quantitative measures and foreign keys to dimension tables. Dimension tables store descriptive attributes and hierarchies. A star schema has a single denormalized dimension table for each dimension, while a snowflake schema has multiple normalized dimension tables for each dimension. Snowflake supports both star and snowflake schemas, and allows users to create views and joins to simplify queries.
? Inmon/3NF: This is a common normalized data model that uses a third normal
form (3NF) schema to organize data into entities and relationships. 3NF schema eliminates data duplication and ensures data consistency by applying three rules: 1) every column in a table must depend on the primary key, 2) every column in a table must depend on the whole primary key, not a part of it, and 3) every column in a table must depend only on the primary key, not on other columns. Snowflake supports 3NF schema and allows users to create referential integrity constraints and foreign key relationships to enforce data quality.
? Data vault: This is a hybrid data model that combines the best practices of
dimensional and normalized data models to create a scalable, flexible, and resilient data warehouse. Data vault schema consists of three types of tables: hubs, links, and satellites. Hubs store business keys and metadata for each entity. Links store associations and relationships between entities. Satellites store descriptive attributes and historical changes for each entity or relationship. Snowflake supports data vault schema and allows users to leverage its features such as time travel, zero-copy cloning, and secure data sharing to implement data
vault methodology.
References: What is Data Modeling? | Snowflake, Snowflake Schema in Data Warehouse Model - GeeksforGeeks, [Data Vault 2.0 Modeling with Snowflake]

NEW QUESTION 109
An Architect needs to design a data unloading strategy for Snowflake, that will be used with the COPY INTO <location> command.
Which configuration is valid?

A. Location of files: Snowflake internal locatio
B. File formats: CSV, XM
C. File encoding: UTF-8. Encryption: 128-bit
D. Location of files: Amazon S3. File formats: CSV, JSO
E. File encoding: Latin-1 (ISO-8859). Encryption: 128-bit
F. Location of files: Google Cloud Storag
G. File formats: Parque
H. File encoding: UTF-8· Compression: gzip
I. Location of files: Azure ADL
J. File formats: JSON, XML, Avro, Parquet, OR
K. Compression: bzip2. Encryption: User-supplied key

Answer: C

Explanation:
For the configuration of data unloading in Snowflake, the valid option among the provided choices is "C." This is because Snowflake supports unloading data into Google Cloud Storage using the COPY INTO <location> command with specific configurations. The configurations listed in option C, such as Parquet file format with UTF-8 encoding and gzip compression, are all supported by Snowflake. Notably, Parquet is a columnar storage file format, which is optimal for high-performance data processing tasks in Snowflake. The UTF-8 file encoding and gzip compression are both standard and widely used settings that are compatible with Snowflake??s capabilities for data unloading to cloud storage platforms.References:
? Snowflake Documentation on COPY INTO command
? Snowflake Documentation on Supported File Formats
? Snowflake Documentation on Compression and Encoding Options

NEW QUESTION 110
Which columns can be included in an external table schema? (Select THREE).

A. VALUE
B. METADATASROW_ID
C. METADATASISUPDATE
D. METADAT A$ FILENAME
E. METADATAS FILE_ROW_NUMBER
F. METADATASEXTERNAL TABLE PARTITION

Answer: ADE

Explanation:
An external table schema defines the columns and data types of the data stored in an external stage. All external tables include the following columns by default:
? VALUE: A VARIANT type column that represents a single row in the external file.
? METADATA$FILENAME: A pseudocolumn that identifies the name of each staged data file included in the external table, including its path in the stage.
? METADATA$FILE_ROW_NUMBER: A pseudocolumn that shows the row number for each record in a staged data file.
You can also create additional virtual columns as expressions using the VALUE column and/or the pseudocolumns. However, the following columns are not valid

for external tables and cannot be included in the schema:
? METADATASROW_ID: This column is only available for internal tables and shows
the unique identifier for each row in the table.
? METADATASISUPDATE: This column is only available for internal tables and shows whether the row was inserted or updated by a merge operation.
? METADATASEXTERNAL TABLE PARTITION: This column is not a valid column name and does not exist in Snowflake.
References: Introduction to External Tables, CREATE EXTERNAL TABLE

NEW QUESTION 111
What is a characteristic of Role-Based Access Control (RBAC) as used in Snowflake?

A. Privileges can be granted at the database level and can be inherited by all underlying objects.
B. A user can use a "super-user" access along with securityadmin to bypass authorization checks and access all databases, schemas, and underlying objects.
C. A user can create managed access schemas to support future grants and ensure only schema owners can grant privileges to other roles.
D. A user can create managed access schemas to support current and future grants and ensure only object owners can grant privileges to other roles.

Answer: C

Explanation:
Role-Based Access Control (RBAC) is the Snowflake Access Control Framework that allows privileges to be granted by object owners to roles, and roles, in turn, can be assigned to users to restrict or allow actions to be performed on objects. A characteristic of RBAC as used in Snowflake is:
? Privileges can be granted at the database level and can be inherited by all
underlying objects. This means that a role that has a certain privilege on a database, such as CREATE SCHEMA or USAGE, can also perform the same action on any schema, table, view, or other object within that database, unless explicitly revoked. This simplifies the access control management and reduces the number of grants required.
? A user can create managed access schemas to support future grants and ensure
only schema owners can grant privileges to other roles. This means that a user can create a schema with the MANAGED ACCESS option, which changes the default behavior of object ownership and privilege granting within the schema. In a managed access schema, object owners lose the ability to grant privileges on their objects to other roles, and only the schema owner or a role with the MANAGE GRANTS privilege can do so. This enhances the security and governance of the schema and its objects.
The other options are not characteristics of RBAC as used in Snowflake:
? A user can use a ??super-user?? access along with securityadmin to bypass authorization checks and access all databases, schemas, and underlying objects. This is not true, as there is no such thing as a ??super-user?? access in Snowflake. The securityadmin role is a predefined role that can manage users and roles, but it does not have any privileges on any database objects by default. To access any object, the securityadmin role must be explicitly granted the appropriate privilege by the object owner or another role with the grant option.
? A user can create managed access schemas to support current and future grants and ensure only object owners can grant privileges to other roles. This is not true, as this contradicts the definition of a managed access schema. In a managed access schema, object owners cannot grant privileges on their objects to other roles, and only the schema owner or a role with the MANAGE GRANTS privilege can do so.
References:
? Overview of Access Control
? A Functional Approach For Snowflake??s Role-Based Access Controls
? Snowflake Role-Based Access Control simplified
? Snowflake RBAC security prefers role inheritance to role composition
? Overview of Snowflake Role Based Access Control

NEW QUESTION 112
Why might a Snowflake Architect use a star schema model rather than a 3NF model when designing a data architecture to run in Snowflake? (Select TWO).

A. Snowflake cannot handle the joins implied in a 3NF data model.
B. The Architect wants to remove data duplication from the data stored in Snowflake.
C. The Architect is designing a landing zone to receive raw data into Snowflake.
D. The Bl tool needs a data model that allows users to summarize facts across different dimensions, or to drill down from the summaries.
E. The Architect wants to present a simple flattened single view of the data to a particular group of end users.

Answer: DE

Explanation:
A star schema model is a type of dimensional data model that consists of a single fact table and multiple dimension tables. A 3NF model is a type of relational data model that follows the third normal form, which eliminates data redundancy and ensures referential integrity. A Snowflake Architect might use a star schema model rather than a 3NF model when designing a data architecture to run in Snowflake for the following reasons:
? A star schema model is more suitable for analytical queries that require
aggregating and slicing data across different dimensions, such as those performed by a BI tool. A 3NF model is more suitable for transactional queries that require inserting, updating, and deleting individual records.
? A star schema model is simpler and faster to query than a 3NF model, as it involves fewer joins and less complex SQL statements. A 3NF model is more complex and slower to query, as it involves more joins and more complex SQL statements.
? A star schema model can provide a simple flattened single view of the data to a
particular group of end users, such as business analysts or data scientists, who need to explore and visualize the data. A 3NF model can provide a more detailed and normalized view of the data to a different group of end users, such as application developers or data engineers, who need to maintain and update the data.
The other options are not valid reasons for choosing a star schema model over a 3NF model in Snowflake:
? Snowflake can handle the joins implied in a 3NF data model, as it supports ANSI
SQL and has a powerful query engine that can optimize and execute complex queries efficiently.
? The Architect can use both star schema and 3NF models to remove data
duplication from the data stored in Snowflake, as both models can enforce data integrity and avoid data anomalies. However, the trade-off is that a star schema model may have more data redundancy than a 3NF model, as it denormalizes the data for faster query performance, while a 3NF model may have less data redundancy than a star schema model, as it normalizes the data for easier data maintenance.
? The Architect can use both star schema and 3NF models to design a landing zone
to receive raw data into Snowflake, as both models can accommodate different types of data sources and formats. However, the choice of the model may depend on the purpose and scope of the landing zone, such as whether it is a temporary or permanent storage, whether it is a staging area or a data lake, and whether it is a single source or a multi-source integration.
References:
? Snowflake Architect Training
? Data Modeling: Understanding the Star and Snowflake Schemas

? Data Vault vs Star Schema vs Third Normal Form: Which Data Model to Use?
? Star Schema vs Snowflake Schema: 5 Key Differences
? Dimensional Data Modeling - Snowflake schema
? Star schema vs Snowflake Schema

NEW QUESTION 114
A company has built a data pipeline using Snowpipe to ingest files from an Amazon S3 bucket. Snowpipe is configured to load data into staging database tables. Then a task runs to load the data from the staging database tables into the reporting database tables.
The company is satisfied with the availability of the data in the reporting database tables,
but the reporting tables are not pruning effectively. Currently, a size 4X-Large virtual warehouse is being used to query all of the tables in the reporting database.
What step can be taken to improve the pruning of the reporting tables?

A. Eliminate the use of Snowpipe and load the files into internal stages using PUT commands.
B. Increase the size of the virtual warehouse to a size 5X-Large.
C. Use an ORDER BY <cluster_key (s) > command to load the reporting tables.
D. Create larger files for Snowpipe to ingest and ensure the staging frequency does not exceed 1 minute.

Answer: C

Explanation:
Effective pruning in Snowflake relies on the organization of data within micro- partitions. By using an ORDER BY clause with clustering keys when loading data into the reporting tables, Snowflake can better organize the data within micro-partitions. This organization allows Snowflake to skip over irrelevant micro-partitions during a query, thus improving query performance and reducing the amount of data scanned12.
References =
•Snowflake Documentation on micro-partitions and data clustering2
•Community article on recognizing unsatisfactory pruning and improving it1

NEW QUESTION 115
A new table and streams are created with the following commands: CREATE OR REPLACE TABLE LETTERS (ID INT, LETTER STRING) ;
CREATE OR REPLACE STREAM STREAM_1 ON TABLE LETTERS;
CREATE OR REPLACE STREAM STREAM_2 ON TABLE LETTERS APPEND_ONLY = TRUE;
The following operations are processed on the newly created table: INSERT INTO LETTERS VALUES (1, 'A');
INSERT INTO LETTERS VALUES (2, 'B'); INSERT INTO LETTERS VALUES (3, 'C');
TRUNCATE TABLE LETTERS;
INSERT INTO LETTERS VALUES (4, 'D'); INSERT INTO LETTERS VALUES (5, 'E'); INSERT INTO LETTERS VALUES (6, 'F'); DELETE FROM LETTERS WHERE ID = 6;
What would be the output of the following SQL commands, in order? SELECT COUNT (*) FROM STREAM_1;
SELECT COUNT (*) FROM STREAM_2;

A. 2 & 6
B. 2 & 3
C. 4 & 3
D. 4 & 6

Answer: C

Explanation:
In Snowflake, a stream records data manipulation language (DML) changes to its base table since the stream was created or last consumed. STREAM_1 will show all changes including the TRUNCATE operation, while STREAM_2, being APPEND_ONLY, will not show deletions like TRUNCATE. Therefore, STREAM_1 will count the three inserts, the TRUNCATE (counted as a single operation), and the subsequent two inserts before the delete, totaling 4. STREAM_2 will only count the three initial inserts and the two after the TRUNCATE, totaling 3, as it does not count the TRUNCATE or the delete operation. References: The explanation is based on the Snowflake documentation on streams, which details how streams track changes and the difference between standard and APPEND_ONLY streams12.

NEW QUESTION 116
An Architect uses COPY INTO with the ON_ERROR=SKIP_FILE option to bulk load CSV files into a table called TABLEA, using its table stage. One file named file5.csv fails to load. The Architect fixes the file and re-loads it to the stage with the exact same file name it had previously.
Which commands should the Architect use to load only file5.csv file from the stage? (Choose two.)

A. COPY INTO tablea FROM @%tablea RETURN_FAILED_ONLY = TRUE;
B. COPY INTO tablea FROM @%tablea;
C. COPY INTO tablea FROM @%tablea FILES = ('file5.csv');
D. COPY INTO tablea FROM @%tablea FORCE = TRUE;
E. COPY INTO tablea FROM @%tablea NEW_FILES_ONLY = TRUE;
F. COPY INTO tablea FROM @%tablea MERGE = TRUE;

Answer: BC

Explanation:
? Option A (RETURN_FAILED_ONLY) will only load files that previously failed to load. Since file5.csv already exists in the stage with the same name, it will not be considered a new file and will not be loaded.
? Option D (FORCE) will overwrite any existing data in the table. This is not desired as we only want to load the data from file5.csv.
? Option E (NEW_FILES_ONLY) will only load files that have been added to the stage since the last COPY command. This will not work because file5.csv was already in the stage before it was fixed.
? Option F (MERGE) is used to merge data from a stage into an existing
table, creating new rows for any data not already present. This is not needed in this case as we simply want to load the data from file5.csv.
Therefore, the architect can use either COPY INTO tablea FROM @%tablea or COPY INTO tablea FROM @%tablea FILES = ('file5.csv') to load only file5.csv from the stage. Both options will load the data from the specified file without overwriting any existing data or requiring additional configuration

NEW QUESTION 117
An Architect is troubleshooting a query with poor performance using the QUERY_HIST0RY function. The Architect observes that the COMPILATIONJHME is greater than the EXECUTIONJTIME.
What is the reason for this?

A. The query is processing a very large dataset.
B. The query has overly complex logic.
C. The query is queued for execution.
D. The query is reading from remote storage.

Answer: B

Explanation:
Compilation time is the time it takes for the optimizer to create an optimal query plan for the efficient execution of the query. It also involves some pruning of partition files, making the query execution efficient2
If the compilation time is greater than the execution time, it means that the optimizer spent more time analyzing the query than actually running it. This could indicate that the query has overly complex logic, such as multiple joins, subqueries, aggregations, or
expressions. The complexity of the query could also affect the size and quality of the query plan, which could impact the performance of the query3
To reduce the compilation time, the Architect can try to simplify the query logic, use views or common table expressions (CTEs) to break down the query into smaller parts, or use
hints to guide the optimizer. The Architect can also use the EXPLAIN command to examine the query plan and identify potential bottlenecks or inefficiencies4
References:
? 1: SnowPro Advanced: Architect | Study Guide 5
? 2: Snowflake Documentation | Query Profile Overview 6
? 3: Understanding Why Compilation Time in Snowflake Can Be Higher than Execution Time 7
? 4: Snowflake Documentation | Optimizing Query Performance 8
? : SnowPro Advanced: Architect | Study Guide
? : Query Profile Overview
? : Understanding Why Compilation Time in Snowflake Can Be Higher than Execution Time
? : Optimizing Query Performance

NEW QUESTION 120
Two queries are run on the customer_address table:
create or replace TABLE CUSTOMER_ADDRESS (CA_ADDRESS_SK NUMBER(38,0), CA_ADDRESS_ID VARCHAR(16), CA_STREET_NUMBER VARCHAR(IO) CA_STREET_NAME VARCHAR(60), CA_STREET_TYPE VARCHAR(15), CA_SUITE_NUMBER VARCHAR(10), CA_CITY VARCHAR(60), CA_COUNTY
VARCHAR(30), CA_STATE VARCHAR(2), CA_ZIP VARCHAR(10), CA_COUNTRY VARCHAR(20), CA_GMT_OFFSET NUMBER(5,2), CA_LOCATION_TYPE VARCHAR(20));
ALTER TABLE DEMO_DB.DEMO_SCH.CUSTOMER_ADDRESS ADD SEARCH OPTIMIZATION ON SUBSTRING(CA_ADDRESS_ID);
Which queries will benefit from the use of the search optimization service? (Select TWO).

A. select * from DEMO_DB.DEMO_SCH.CUSTOMER_ADDRESS Where substring(CA_ADDRESS_ID,1,8)= substring('AAAAAAAAPHPPLBAAASKDJHASLKDJHASKJD',1,8);
B. select * from DEMO_DB.DEMO_SCH.CUSTOMER_ADDRESS Where CA_ADDRESS_ID= substring('AAAAAAAAPHPPLBAAASKDJHASLKDJHASKJD',1,16);
C. select*fromDEMO_DB.DEMO_SCH.CUSTOMER_ADDRESSWhereCA_ADDRESS_IDLIKE ??%BAAASKD%';
D. select*fromDEMO_DB.DEMO_SCH.CUSTOMER_ADDRESSWhereCA_ADDRESS_IDLIK E '%PHPP%';
E. select*fromDEMO_DB.DEMO_SCH.CUSTOMER_ADDRESSWhereCA_ADDRESS_IDNO T LIKE '%AAAAAAAAPHPPL%';

Answer: AB

Explanation:
The use of the search optimization service in Snowflake is particularly effective when queries involve operations that match exact substrings or start from the beginning of a string. The ALTER TABLE command adding search optimization specifically for substrings on the CA_ADDRESS_ID field allows the service to create an optimized search path for queries using substring matches.
? Option A benefits because it directly matches a substring from the start of the
CA_ADDRESS_ID, aligning with the optimization's capability to quickly locate records based on the beginning segments of strings.
? Option B also benefits, despite performing a full equality check, because it
essentially compares the full length of CA_ADDRESS_ID to a substring, which can leverage the substring index for efficient retrieval.Options C, D, and E involve patterns that do not start from the beginning of the string or use negations, which are not optimized by the search optimization service configured for starting substring matches.References: Snowflake's documentation on the use of search optimization for substring matching in SQL queries.

NEW QUESTION 121
When loading data from stage using COPY INTO, what options can you specify for the ON_ERROR clause?

A. CONTINUE
B. SKIP_FILE
C. ABORT_STATEMENT
D. FAIL

Answer: ABC

Explanation:
? The ON_ERROR clause is an optional parameter for the COPY INTO command that specifies the behavior of the command when it encounters errors in the files. The ON_ERROR clause can have one of the following values1:
? Therefore, options A, B, and C are correct.
References: : COPY INTO <table>

NEW QUESTION 124
Which of the following are characteristics of how row access policies can be applied to external tables? (Choose three.)

A. An external table can be created with a row access policy, and the policy can be applied to the VALUE column.
B. A row access policy can be applied to the VALUE column of an existing external table.
C. A row access policy cannot be directly added to a virtual column of an external table.
D. External tables are supported as mapping tables in a row access policy.
E. While cloning a database, both the row access policy and the external table will be cloned.
F. A row access policy cannot be applied to a view created on top of an external table.

Answer: ABC

Explanation:
These three statements are true according to the Snowflake documentation and the web search results. A row access policy is a feature that allows filtering rows based on user-defined conditions. A row access policy can be applied to an external table, which is a table that reads data from external files in a stage. However, there are some limitations and considerations for using row access policies with external tables.
? An external table can be created with a row access policy by using the WITH ROW
ACCESS POLICY clause in the CREATE EXTERNAL TABLE statement. The policy can be applied to the VALUE column, which is the column that contains the raw data from the external files in a VARIANT data type1.
? A row access policy can also be applied to the VALUE column of an existing
external table by using the ALTER TABLE statement with the SET ROW ACCESS POLICY clause2.
? A row access policy cannot be directly added to a virtual column of an external
table. A virtual column is a column that is derived from the VALUE column using
an expression. To apply a row access policy to a virtual column, the policy must be applied to the VALUE column and the expression must be repeated in the policy definition3.
? External tables are not supported as mapping tables in a row access policy. A
mapping table is a table that is used to determine the access rights of users or roles based on some criteria. Snowflake does not support using an external table as a mapping table because it may cause performance issues or errors4.
? While cloning a database, Snowflake clones the row access policy, but not the
external table. Therefore, the policy in the cloned database refers to a table that is not present in the cloned database. To avoid this issue, the external table must be manually cloned or recreated in the cloned database4.
? A row access policy can be applied to a view created on top of an external table.
The policy can be applied to the view itself or to the underlying external
table. However, if the policy is applied to the view, the view must be a secure view, which is a view that hides the underlying data and the view definition from unauthorized users5.
References:
? CREATE EXTERNAL TABLE | Snowflake Documentation
? ALTER EXTERNAL TABLE | Snowflake Documentation
? Understanding Row Access Policies | Snowflake Documentation
? Snowflake Data Governance: Row Access Policy Overview
? Secure Views | Snowflake Documentation

NEW QUESTION 126
Consider the following scenario where a masking policy is applied on the CREDICARDND column of the CREDITCARDINFO table. The masking policy definition Is as follows:

```
create or replace masking policy creditcardno_mask as (val string) returns string ->
case
when is_role_in_session('PI_ANALYTICS') then
right(val,4)
else '***MASKED***'
end;
```

Sample data for the CREDITCARDINFO table is as follows: NAME EXPIRYDATE CREDITCARDNO
JOHN DOE 2022-07-23 4321 5678 9012 1234
if the Snowflake system rotes have not been granted any additional roles, what will be the result?

A. The sysadmin can see the CREDICARDND column data in clear text.
B. The owner of the table will see the CREDICARDND column data in clear text.
C. Anyone with the Pl_ANALYTICS role will see the last 4 characters of the CREDICARDND column data in dear text.
D. Anyone with the Pl_ANALYTICS role will see the CREDICARDND column as*** 'MASKED* **'.

Answer: D

Explanation:
? The masking policy defined in the image indicates that if a user has the PI_ANALYTICS role, they will be able to see the last 4 characters of the CREDITCARDNO column data in clear text. Otherwise, they will see ??MASKED??. Since Snowflake system roles have not been granted any additional roles, they won??t have the PI_ANALYTICS role and therefore cannot view the last 4 characters of credit card numbers.
? To apply a masking policy on a column in Snowflake, you need to use the ALTER TABLE ?? ALTER COLUMN command or the ALTER VIEW command and specify the policy name. For example, to apply the creditcardno_mask policy on the CREDITCARDNO column of the CREDITCARDINFO table, you can use the following command:
ALTER TABLE CREDITCARDINFO ALTER COLUMN CREDITCARDNO SET MASKING
POLICY creditcardno_mask;
? For more information on how to create and use masking policies in Snowflake, you can refer to the following resources:
CREATE MASKING POLICY: This document explains the syntax and usage of the CREATE MASKING POLICY command, which allows you to create a new masking policy or replace an existing one.
Using Dynamic Data Masking: This guide provides instructions on how to configure and use dynamic data masking in Snowflake, which is a feature that allows you to mask sensitive data based on the execution context of the user.
ALTER MASKING POLICY: This document explains the syntax and usage of the ALTER MASKING POLICY command, which allows you to modify the properties of an existing masking policy.
References: 1: https://docs.snowflake.com/en/sql-reference/sql/create-masking-policy 2:
https://docs.snowflake.com/en/user-guide/security-column-ddm-use 3: https://docs.snowflake.com/en/sql-reference/sql/alter-masking-policy

NEW QUESTION 129
Company A would like to share data in Snowflake with Company B. Company B is not on the same cloud platform as Company A.
What is required to allow data sharing between these two companies?

A. Create a pipeline to write shared data to a cloud storage location in the target cloud provider.
B. Ensure that all views are persisted, as views cannot be shared across cloud platforms.
C. Setup data replication to the region and cloud platform where the consumer resides.
D. Company A and Company B must agree to use a single cloud platform: Data sharing is only possible if the companies share the same cloud provider.

Answer: C

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

NEW QUESTION 133
You are a snowflake architect in an organization. The business team came to to deploy an use case which requires you to load some data which they can visualize through tableau. Everyday new data comes in and the old data is no longer required.
What type of table you will use in this case to optimize cost

A. TRANSIENT
B. TEMPORARY
C. PERMANENT

Answer: A

Explanation:
? A transient table is a type of table in Snowflake that does not have a Fail-safe period and can have a Time Travel retention period of either 0 or 1 day. Transient tables are suitable for temporary or intermediate data that can be easily reproduced or replicated1.
? A temporary table is a type of table in Snowflake that is automatically dropped when the session ends or the current user logs out. Temporary tables do not incur any storage costs, but they are not visible to other users or sessions2.
? A permanent table is a type of table in Snowflake that has a Fail-safe period and a Time Travel retention period of up to 90 days. Permanent tables are suitable for persistent and durable data that needs to be protected from accidental or malicious deletion3.
? In this case, the use case requires loading some data that can be visualized through Tableau. The data is updated every day and the old data is no longer required. Therefore, the best type of table to use in this case to optimize cost is a transient table, because it does not incur any Fail-safe costs and it can have a short Time Travel retention period of 0 or 1 day. This way, the data can be loaded and queried by Tableau, and then deleted or overwritten without incurring any unnecessary storage costs.
References: : Transient Tables : Temporary Tables : Understanding & Using Time Travel

NEW QUESTION 138
What Snowflake system functions are used to view and or monitor the clustering metadata for a table? (Select TWO).

A. SYSTEMSCLUSTERING
B. SYSTEMSTABLE_CLUSTERING
C. SYSTEMSCLUSTERING_DEPTH
D. SYSTEMSCLUSTERING_RATIO
E. SYSTEMSCLUSTERING_INFORMATION

Answer: CE

Explanation:
? The SYSTEM$CLUSTERING_INFORMATION function in Snowflake returns a variety of clustering information for a specified table. This information includes the average clustering depth, total number of micro-partitions, total constant partition count, average overlaps, average depth, and a partition depth histogram. This function allows you to specify either one or multiple columns for which the clustering information is returned, and it returns this data in JSON format.
? The SYSTEM$CLUSTERING_DEPTH function computes the average depth of a table based on specified columns or the clustering key defined for the table. A lower average depth indicates that the table is better clustered with respect to the specified columns. This function also allows specifying columns to calculate the depth, and the values need to be enclosed in single quotes.
References:
? SYSTEM$CLUSTERING_INFORMATION: Snowflake Documentation
? SYSTEM$CLUSTERING_DEPTH: Snowflake Documentation

NEW QUESTION 142
.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual ARA-C01 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the ARA-C01 Product From:

https://www.2passeasy.com/dumps/ARA-C01/

Money Back Guarantee

ARA-C01 Practice Exam Features:

* ARA-C01 Questions and Answers Updated Frequently

* ARA-C01 Practice Questions Verified by Expert Senior Certified Staff

* ARA-C01 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* ARA-C01 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year