

ARA-C01^{Q&As}

SnowPro Advanced: Architect Certification Exam

Pass Snowflake ARA-C01 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass2lead.com/ara-c01.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Snowflake
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



QUESTION 1

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.

Correct 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 and Snowflake | Snowflake Documentation, Azure Private Link and Snowflake | Snowflake Documentation, Google Cloud Private Service Connect and 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

QUESTION 2

What is a valid object hierarchy when building a Snowflake environment?

- A. Account --> Database --> Schema --> Warehouse
- B. Organization --> Account --> Database --> Schema --> Stage
- C. Account --> Schema > Table --> Stage
- D. Organization --> Account --> Stage --> Table --> View

Correct Answer: B

Explanation: This is the valid object hierarchy when building a Snowflake environment, according to the Snowflake documentation and the web search results. Snowflake is a cloud data platform that supports various types of objects, such as

databases, schemas, tables, views, stages, warehouses, and more. These objects are organized in a hierarchical structure, as follows:

Organization: An organization is the top-level entity that represents a group of Snowflake accounts that are related by business needs or ownership. An organization can have one or more accounts, and can enable features such as cross-

account data sharing, billing and usage reporting, and single sign-on across accounts¹².

Account: An account is the primary entity that represents a Snowflake customer. An account can have one or more databases, schemas, stages, warehouses, and other objects. An account can also have one or more users, roles, and

security integrations. An account is associated with a specific cloud platform, region, and Snowflake edition³⁴.

Database: A database is a logical grouping of schemas. A database can have one or more schemas, and can store structured, semi-structured, or unstructured data. A database can also have properties such as retention time, encryption,

and ownership⁵⁶.

Schema: A schema is a logical grouping of tables, views, stages, and other objects. A schema can have one or more objects, and can define the namespace and access control for the objects. A schema can also have properties such as

ownership and default warehouse .

Stage: A stage is a named location that references the files in external or internal storage. A stage can be used to load data into Snowflake tables using the COPY INTO command, or to unload data from Snowflake tables using the COPY

INTO LOCATION command. A stage can be created at the account, database, or schema level, and can have properties such as file format, encryption, and credentials .

The other options listed are not valid object hierarchies, because they either omit or misplace some objects in the structure. For example, option A omits the organization level and places the warehouse under the schema level, which is

incorrect. Option C omits the organization, account, and stage levels, and places the table under the schema level, which is incorrect. Option D omits the database level and places the stage and table under the account level, which is

incorrect.

References:

Snowflake Documentation: Organizations

Snowflake Blog: Introducing Organizations in Snowflake Snowflake Documentation: Accounts

Snowflake Blog: Understanding Snowflake Account Structures Snowflake Documentation: Databases

Snowflake Blog: How to Create a Database in Snowflake [Snowflake Documentation: Schemas]

[Snowflake Blog: How to Create a Schema in Snowflake] [Snowflake Documentation: Stages]

[Snowflake Blog: How to Use Stages in Snowflake]

QUESTION 3

An Architect would like to save quarter-end financial results for the previous six years.

Which Snowflake feature can the Architect use to accomplish this?

- A. Search optimization service
- B. Materialized view
- C. Time Travel
- D. Zero-copy cloning
- E. Secure views

Correct Answer: D

Explanation: Zero-copy cloning is a Snowflake feature that can be used to save quarter- end financial results for the previous six years. Zero-copy cloning allows creating a copy of a database, schema, table, or view without duplicating the data or metadata. The clone shares the same data files as the original object, but tracks any changes made to the clone or the original separately. Zero-copy cloning can be used to create snapshots of data at different points in time, such as quarter-end financial results, and preserve them for future analysis or comparison. Zero-copy cloning is fast, efficient, and does not consume any additional storage space unless the data is modified¹. References: Zero-Copy Cloning | Snowflake Documentation

QUESTION 4

A user has the appropriate privilege to see unmasked data in a column.

If the user loads this column data into another column that does not have a masking policy, what will occur?

- A. Unmasked data will be loaded in the new column.
- B. Masked data will be loaded into the new column.
- C. Unmasked data will be loaded into the new column but only users with the appropriate privileges will be able to see the unmasked data.
- D. Unmasked data will be loaded into the new column and no users will be able to see the unmasked data.

Correct Answer: A

Explanation: According to the SnowPro Advanced: Architect documents and learning resources, column masking policies are applied at query time based on the privileges of the user who runs the query. Therefore, if a user has the privilege to see unmasked data in a column, they will see the original data when they query that column. If they load this column data into another column that does not have a masking policy, the unmasked data will be loaded in the new column, and any user who can query the new column will see the unmasked data as well. The masking policy does not affect the underlying data in the column, only the query results. References: Snowflake Documentation: Column Masking Snowflake Learning: Column Masking

QUESTION 5

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.

Correct 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