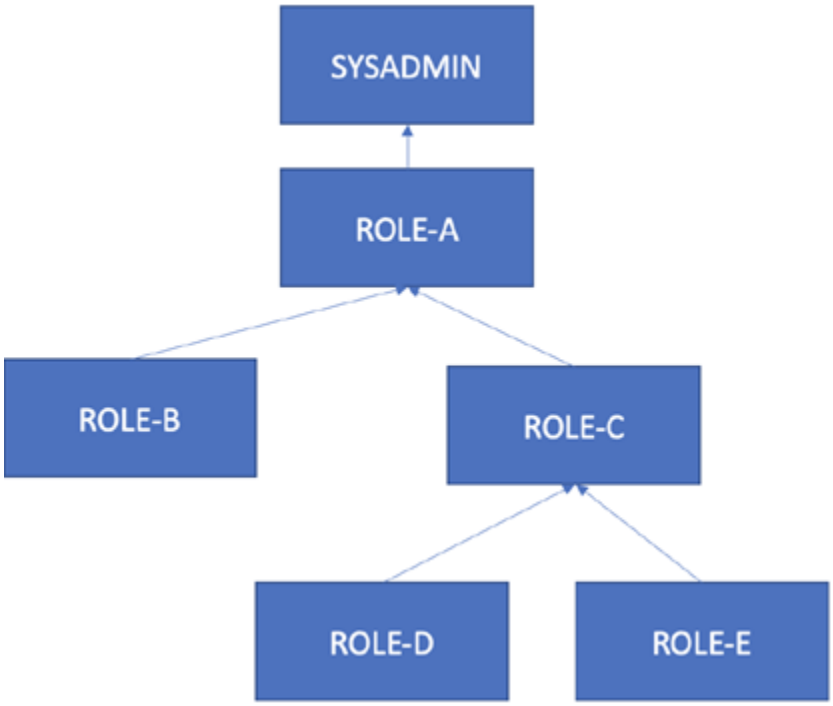
Certification Questions Practice Set (Security)

1. Consider the below scenario where the user is granted the ROLE A, Then in this case through Snowflake UI, what additional roles he can switch along with ROLE A ?



A) ROLE-A , ROLE-B, ROLE-C.

B) SYSADMIN, ROLE-A, ROLE-B, ROLE-C.

C) ROLE-A, ROLE-B, ROLE-C, ROLE-D, ROLE-E.

D) ROLE-A only.

Answer : C

Explanation : Follow the concept of Role hierarchy

---

2. Which of the system defined roles is dedicated to user and role management ONLY ?

   A) ACCOUNTADMIN

   B) SYSADMIN

   C) SECURITYADMIN

   D) USERADMIN

Answer : D

Explanation : Concepts of system defined roles

Mark the correct statements with respect to secure views and their creation in the Snowflake Account.

A. For a secure view, internal optimizations can indirectly expose data and the view definition is visible to other users.

B. Secure views should not be used for views that are defined solely for query convenience, such as views created to simplify queries for users who do not need to understand the underlying data representation.

C. To convert an existing view to a secure view and back to a regular view, set/unset the SECURE keyword in the ALTER VIEW or ALTER MATERIALIZED VIEW command.

D. For non-materialized views, the IS_SECURE column in the Information Schema and Account Usage views identifies whether a view is secure.

E. The internals of a secure view are not exposed in the Query Profile (in the web interface). This is the case even for the owner of the secure view because non-owners might have access to an owner's Query Profile.

Answers : B, C, E

Explanation : https://docs.snowflake.com/en/user-guide/views-secure

Why Should I Use Secure Views?

- For a non-secure view, internal optimizations can indirectly expose data.

  Some of the internal optimizations for views require access to the underlying data in the base tables for the view. This access might allow data that is hidden from users of the view to be exposed through user code, such as user-defined functions, or other programmatic methods. Secure views do not utilize these optimizations, ensuring that users have no access to the underlying data.

- For a non-secure view, the view definition is visible to other users.

  By default, the query expression used to create a standard view, also known as the view definition or text, is visible to users in various commands and interfaces. For details, see Interacting with Secure Views (in this topic).

  For security or privacy reasons, you might not wish to expose the underlying tables or internal structural details for a view. With secure views, the view definition and details are visible only to authorized users (i.e. users who are granted the role that owns the view).

- To create a secure view, specify the SECURE keyword in the CREATE VIEW or CREATE MATERIALIZED VIEW command.
- To convert an existing view to a secure view and back to a regular view, set/unset the SECURE keyword in the ALTER VIEW or ALTER MATERIALIZED VIEW command.

When Should I Use a Secure View?

Views should be defined as secure when they are specifically designated for data privacy (i.e. to limit access to sensitive data that should not be exposed to all users of the underlying table(s)).

Secure views should ***not*** be used for views that are defined solely for query convenience, such as views created to simplify queries for which users do not need to understand the underlying data representation. Secure views can execute more slowly than non-secure views.

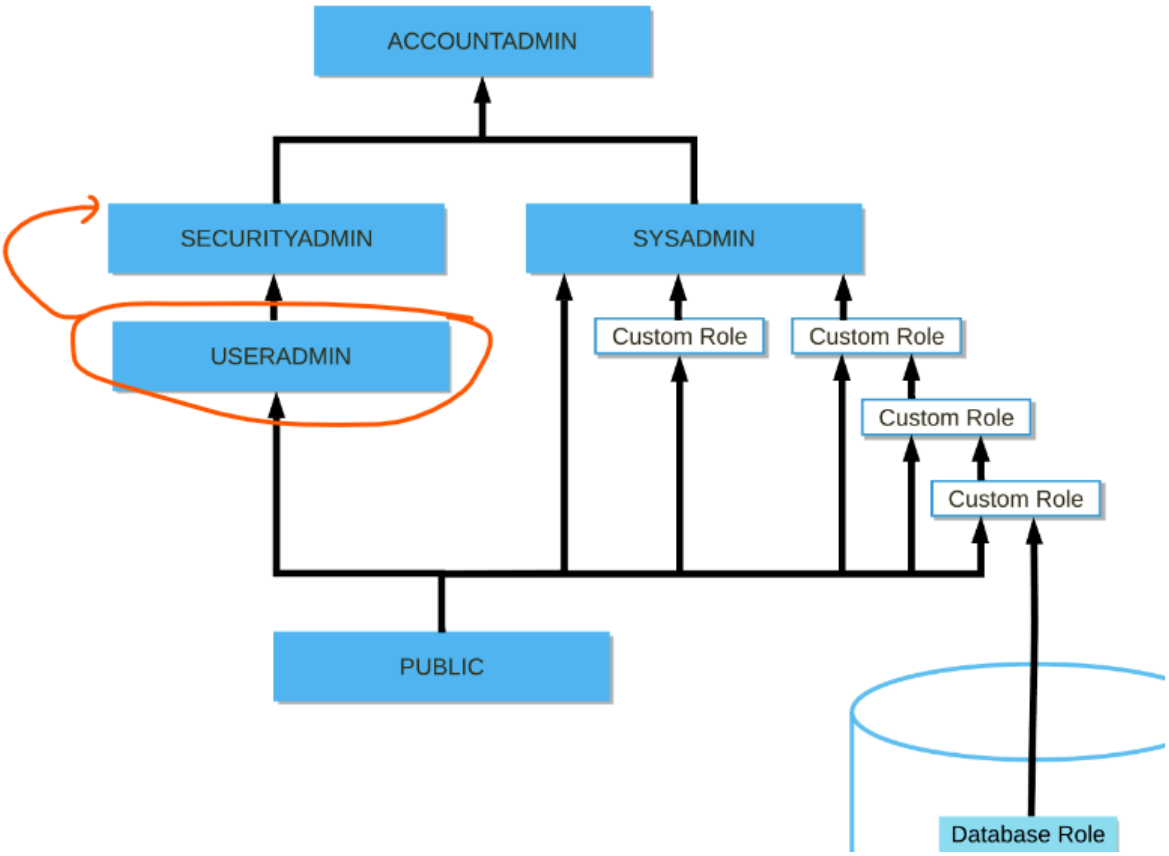
---

7. Which role inherits the privileges of the USERADMIN role via the system role hierarchy?

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

Answer : B

Explanation :



---

8. Does sensitive data in Snowflake get modified in an existing table while applying masking policies?

A. YES
B. NO

Answer - B

---

9. Can the same column be specified in both a dynamic data masking policy signature and a row access policy signature at the same time?

A. YES

B. NO

Answer : YES

---

10. Which privileges are required on an object (i.e., user or role) with the USERADMIN role to modify the object properties?

A. OPERATE

B. MANAGE GRANTS

C. OWNERSHIP

D. MODIFY

Answer : C

---

11. Select the correct statements with regard to using federated authentication/SSO?

A. Snowflake supports using MFA in conjunction with SSO to provide additional levels of security.

B. Snowflake supports multiple audience values (i.e., Audience or Audience Restriction Fields) in the SAML 2.0 assertion from the identity provider to Snowflake.

C. Snowflake supports SSO with private connectivity to the Snowflake Service for Snowflake accounts on Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform.

D. Snowflake supports using SSO with organizations, and you can use the corresponding URL in the SAML2 security integration

Answers : B, C

Explanation : https://docs.snowflake.com/en/user-guide/admin-security-fed-auth-use

Using SSO with multiple audience values

Snowflake supports multiple audience values (i.e. Audience or Audience Restriction Fields) in the SAML 2.0 assertion from the identity provider to Snowflake.

This functionality supports the URLs to access Snowflake as audience values. The URLs for multiple Snowflake accounts are supported because each account has a URL with a unique account identifier to access Snowflake. Additionally, Snowflake accepts the account domain names and the URLs to access

SSO with private connectivity

Snowflake supports SSO with private connectivity to the Snowflake service for Snowflake accounts on Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

Currently, for any given Snowflake account, SSO works with only one account URL at a time: either the public account URL or the URL associated with the private connectivity service on AWS, Microsoft Azure, or Google Cloud Platform.

---

12. Which of the following options are correct regarding the SECURITYADMIN and USERADMIN roles?

1. The USERADMIN role includes the privileges to create and manage users and roles, and it's a child of the SECURITYADMIN role.
2. The SECURITYADMIN role includes the privileges to create and manage users and roles, and it's a child of the USERADMIN role.
3. The SECURITYADMIN role includes the global MANAGE GRANTS privilege to grant or revoke privileges on objects in the account.
4. The USERADMIN role includes the global MANAGE GRANTS privilege to grant or revoke privileges on objects in the account.

Answer : 1

---

14. Rachel is managing data access in Snowflake and needs to ensure that only specific users can see certain columns in a table. What is the best way to enforce this?

A. Use row-level security to restrict access to specific columns.

B. Create secure views that expose only the required columns to the users.

C. Apply column-level encryption on the sensitive columns.

D. Use external tables to manage access to specific columns

Answer : A

---

15. A new CUSTOMER table is created by a data pipeline in a Snowflake schema where MANAGED ACCESS is enabled.
Which roles can grant access to the CUSTOMER table? (Choose three.)

A. The role that owns the schema

B. The role that owns the database

C. The role that owns the CUSTOMER table

D. The SYSADMIN role

E. The SECURITYADMIN role

F. The USERADMIN role with the MANAGE GRANTS privilege

Answers : A, E, F

Explanation : Check the below snapshot at this link :

The following table indicates which roles can manage object privileges in a regular or managed access schema:

Role	Can grant object privileges in a regular schema	Can grant object privileges in a managed access schema
SYSADMIN	No	No
SECURITYADMIN or higher	Yes	Yes
Database owner	No	No
Schema owner	***No***	***Yes***
Object owner	***Yes***	***No***
Any role with the MANAGE GRANTS privilege	Yes	Yes

A team of analysts needs the ability to create their own tables but should not have permission to grant access to those tables to other roles.

How should a Data Engineer handle this?

1. Create a new schema and grant `CREATE TABLE` in the schema to the analyst role.
2. Grant the analyst role the `CREATE SCHEMA` privilege, have the analyst role create the schema, and then remove manage grants from the new schema.
3. Create a new schema with `MANAGED ACCESS` and grant `CREATE TABLE` in the schema to the analyst role.
4. Create a new schema with `MANAGED ACCESS` and grant `OWNERSHIP` of the schema to the analyst role.

Answer :

Explanation :

A development environment is configured with the below settings:

1. `DATA_ENGINEER` role does not have privileges to delete rows from the `sys_logs` table
2. `OPS` role has privileges to delete rows from the `sys_logs` table
3. `OPS` role creates an owner's rights to the stored procedures that deletes rows from the `sys_logs` table
4. `OPS` role grants appropriate privileges on the stored procedure to the `DATA_ENGINEER` role

If a user with the role `DATA_ENGINEER` calls the stored procedure, what will occur?

1. The procedure will run with the privileges of `DATA_ENGINEER` and not the privileges of `OPS`
2. `The procedure will run with the privileges of OPS and not the privileges of DATA_ENGINEER`
3. `The procedure will error when deleting rows from the sys_logs table`
4. `The procedure will inherit the current virtual warehouse of the OPS role.`

`Answer : 2`

`Explanation :` here

`A Data Engineer is cloning a database for a new development environment.`

`What should the Engineer take into consideration when performing the cloning process?[1]  (Select TWO).`

1. Pipes that reference an external stage will not be cloned.
2. Tasks will be suspended by default when created.
3. Database tables will be locked during the cloning process.
4. Unconsumed records in the streams will be inaccessible.
5. The cloned database will retain any granted privileges from the source database.

Can we revisit the question please ? Below are the options that are not correct.

(A) --> False.

(E) --> False

(C) --> False

(D) --> False