Free Questions for ADA-C01 by actualtestdumps

Shared by Slater on 03-01-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: **MultipleChoice**

The ACCOUNTADMIN of Account 123 works with Snowflake Support to set up a Data Exchange. After the exchange is populated with listings from other Snowflake accounts,

what roles in Account 123 are allowed to request and get data?

Options:

A- Only the ACCOUNTADMIN role, and no other roles

B- Any role with USAGE privilege on the Data Exchange

C- Any role with IMPORT SHARE and CREATE DATABASE privileges

D- Any role that the listing provider has designated as authorized

Answer:

B

Explanation:

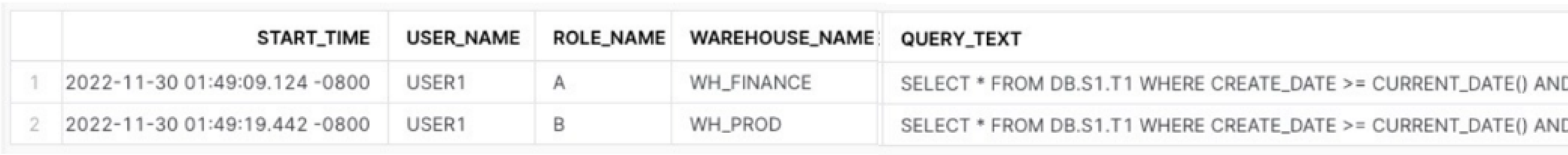
To request and get data from a Data Exchange, the role in Account 123 must have the USAGE privilege on the Data Exchange object. This privilege allows the role to view the listings and request access to the data. According to theSnowflake documentation, "To view the listings in a data exchange, a role must have the USAGE privilege on the data exchange object. To request access to a listing, a role must have the USAGE privilege on the data exchange object and the IMPORT SHARE privilege on the account." The other options are either incorrect or not sufficient to request and get data from a Data Exchange. Option A is incorrect, as the ACCOUNTADMIN role is not the only role that can request and get data, as long as other roles have the necessary privileges. Option C is incorrect, as the IMPORT SHARE and CREATE DATABASE privileges are not required to request and get data, but only to create a database from a share after the access is granted. Option D is incorrect, as the listing provider does not designate the authorized roles in Account 123, but only approves or denies the requests from Account 123.

Question 2

Question Type: **MultipleChoice**

A Snowflake Administrator is investigating why a query is not re-using the persisted result cache.

The Administrator found the two relevant queries from the SNOWFLAKE. ACCOUNT_USAGE. QUERY_HISTORY view:



	START_TIME	USER_NAME	ROLE_NAME	WAREHOUSE_NAME	QUERY_TEXT
1	2022-11-30 01:49:09.124 -0800	USER1	A	WH_FINANCE	SELECT * FROM DB.S1.T1 WHERE CREATE_DATE >= CURRENT_DATE() AND
2	2022-11-30 01:49:19.442 -0800	USER1	B	WH_PROD	SELECT * FROM DB.S1.T1 WHERE CREATE_DATE >= CURRENT_DATE() AND

Why is the second query re-scanning micro-partitions instead of using the first query's persisted result cache?

Options:

A- The second query includes a CURRENT_TIMESTAMP () function.

B- The second query includes a CURRENT_DATE () function.

C- The queries are executed with two different virtual warehouses.

D- The queries are executed with two different roles.

Answer:

A

Explanation:

The inclusion of the CURRENT_TIMESTAMP() function in the second query prevents it from re-using the first query's persisted result cache because this function makes each execution unique due to the constantly changing timestamp. According to theSnowflake documentation, "The query does not include non-reusable functions, which return different results for successive runs of the same query. UUID_STRING, RANDOM, and RANDSTR are good examples of non-reusable functions." The CURRENT_TIMESTAMP() function is another example of a non-reusable function, as it returns the current date and time at the start of query execution, which varies for each run. Therefore, the second query is not identical to the first query, and the result cache is not reused. The other options are either incorrect or irrelevant to the question. Option B is incorrect, as the CURRENT_DATE() function is a reusable function, as it returns the same value for all queries executed within the same day. Option C is irrelevant, as the virtual warehouse used to execute the query does not affect the result cache reuse. Option D is also irrelevant, as the role used to execute the query does not affect the result cache reuse, as long as the role has the necessary access privileges for all the tables used in the query.

Question 3

Question Type: MultipleChoice

How should an Administrator configure a Snowflake account to use AWS PrivateLink?

Options:

A- Create CNAME records in the DNS.

B- Contact Snowflake Support.

C- Block public access to Snowflake.

D- Use SnowCD to evaluate the network connection.

Answer:

A

Explanation:

To configure a Snowflake account to use AWS PrivateLink, the Administrator needs to create CNAME records in the DNS that point to the private endpoints provided by Snowflake. This allows the clients to connect to Snowflake using the same URL as before, but with private connectivity. According to theSnowflake documentation, "After you have created the VPC endpoints, Snowflake provides you with a list of private endpoints for your account. You must create CNAME records in your DNS that point to these private endpoints. The CNAME records must use the same hostnames as the original Snowflake URLs for your account." The other options are either incorrect or not sufficient to configure AWS PrivateLink.Option B is not necessary, as the Administrator can enable AWS PrivateLink using the SYSTEM$AUTHORIZE_PRIVATELINK function1.Option C is not recommended, as it may prevent some data traffic from reaching Snowflake, such as large result sets stored on AWS S32.Option D is not related to AWS PrivateLink, but to Snowflake Connectivity Diagnostic (SnowCD), which is a tool for diagnosing network issues between clients and Snowflake3.

Question 4

Question Type: MultipleChoice

An Administrator needs to create a sample of the table LINEITEM. The sample should not be repeatable and the sampling function should take the data by blocks of rows.

What select command will generate a sample of 20% of the table?

Options:

A- select * from LINEITEM sample bernoulli (20);

B- select * from LINEITEM sample system (20);

C- select * from LINEITEM tablesample block (20 rows);

D- select * from LINEITEM tablesample system (20) seed (1);

Answer:

B

Explanation:

This command will generate a sample of 20% of the table by using the SYSTEM (or BLOCK) sampling method, which selects each block of rows with a probability of 20/100. This method is suitable for taking data by blocks of rows, as the question requires. According to theSnowflake documentation, "SYSTEM (or BLOCK): Includes each block of rows with a probability of p/100. Similar to flipping a weighted coin for each block of rows. This method does not support fixed-size sampling." The other options are either incorrect or do not meet the requirements of the question. Option A uses the BERNOULLI (or ROW) sampling method, which selects each row with a probability of 20/100, but does not take data by blocks of rows. Option C uses the BLOCK sampling method, but specifies a fixed number of rows (20) instead of a percentage (20%). Option D uses the SYSTEM sampling method, but specifies a seed value (1), which makes the sampling repeatable, contrary to the question.

Question 5

Question Type: MultipleChoice

A company's Snowflake account has multiple roles. Each role should have access only to data that resides in the given role's specific region.

When creating a row access policy, which code snippet below will provide privileges to the role ALL_ACCESS_ROLE to see all rows regardless of region, while the other

roles can only see rows for their own regions?

Options:

A- create or replace row access policy region policy as (region_value varchar) returns boolean ->

```
'ALL ACCESS_ROLE' = current_role ()
and exists (
select 1 from entitlement_table
where role = current_role ()
and region = region_value
)
```

B- create or replace row access policy region policy as (region_value varchar) returns boolean ->

```
exists (
select 1 from entitlement_table
where role = current_role ()
and region = region_value
)
```

C-
```
create or replace row access policy region policy as (region_value varchar) returns boolean ->
'ALL_ACCESS_ROLE' = current_role ()
or exists (
select 1 from entitlement_table
where role = current_role ()
and region = region_value
)
```

D-
```
create or replace row access policy region policy as (region_value varchar) returns boolean ->
'ALL ACCESS ROLE' = current_role ()
)
```

Answer:

C

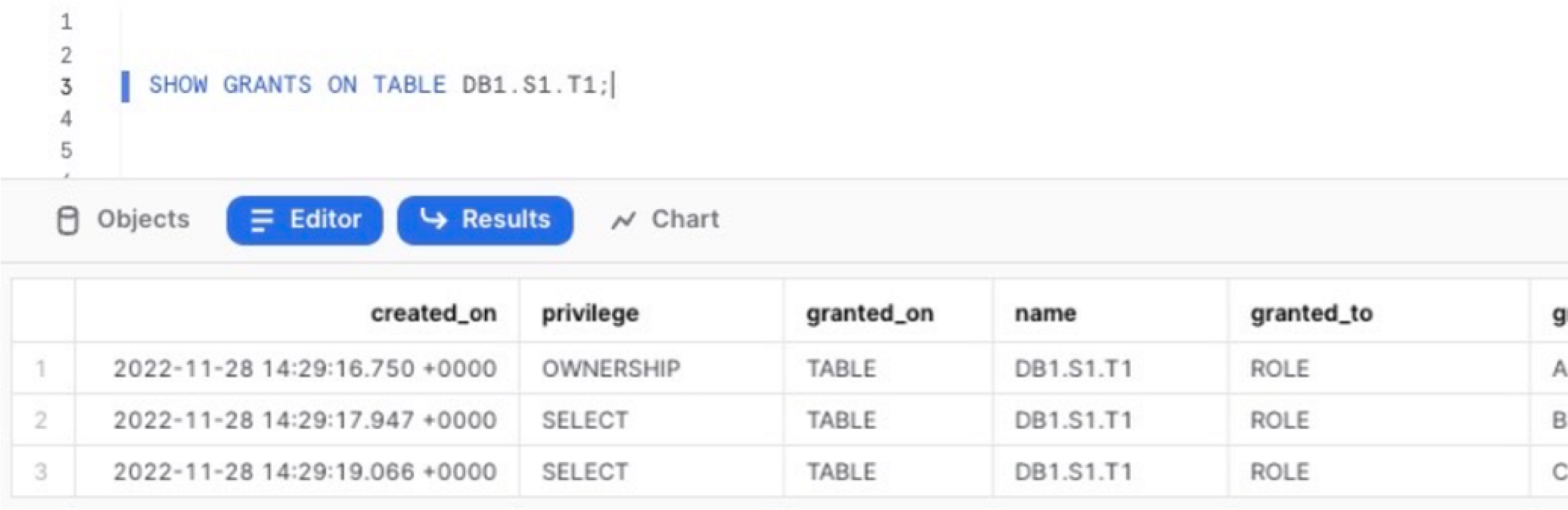
Explanation:

This code snippet will create a row access policy that returns true if the current role is ALL_ACCESS_ROLE or if the current role matches the region value in the entitlement_table. This means that the ALL_ACCESS_ROLE can see all rows regardless of region, while the other roles can only see rows for their own regions. According to theSnowflake documentation, the CURRENT_ROLE context function returns the name of the current role for the session. The EXISTS function returns true if the subquery returns any rows. The OR operator returns true if either operand is true. Therefore, this code snippet satisfies the requirements of the question.

Question 6

Question Type: **MultipleChoice**

Review the output of the SHOW statement below which displays the current grants on the table DB1. S1. T1:



```
SHOW GRANTS ON TABLE DB1.S1.T1;
```

	created_on	privilege	granted_on	name	granted_to	g
1	2022-11-28 14:29:16.750 +0000	OWNERSHIP	TABLE	DB1.S1.T1	ROLE	A
2	2022-11-28 14:29:17.947 +0000	SELECT	TABLE	DB1.S1.T1	ROLE	B
3	2022-11-28 14:29:19.066 +0000	SELECT	TABLE	DB1.S1.T1	ROLE	C

This statement is executed:

USE ROLE ACCOUNTADMIN;

DROP ROLE A;

What will occur?

Options:

A- The table object DB1. S1. T1 will be dropped.

B- The OWNERSHIP privilege on table DB1. S1. T1 will be transferred to the ACCOUNTADMIN role.

C- The SELECT privilege on table DB1. S1. T1 to role B will be shown as GRANTED_BY the role ACCOUNTADMIN.

D- The SELECT privileges for roles B and C will remain.

Answer:

D

Explanation:

Dropping role A does not affect the SELECT privileges granted to roles B and C on the table DB1.S1.T1. According to theSnowflake documentation, dropping a role revokes all privileges granted to the role, but does not revoke any privileges granted by the role. Therefore, the OWNERSHIP privilege on the table DB1.S1.T1 will be revoked from role A, but the SELECT privileges granted by role A to role B and by role B to role C will remain. The GRANTED_BY column will still show the original grantor of the privilege, not the ACCOUNTADMIN role.

Question 7

Question Type: MultipleChoice

Which statement allows this user to access this Snowflake account from a specific IP address (192.168.1.100) while blocking their access from anywhere else?

Options:

A- CREATE NETWORK POLICY ADMIN_POLICY
ALLOWED_IP_LIST = ('192.168.1.100');
ALTER USER ABC SET NETWORK_POLICY = 'ADMIN_POLICY';
User ABC is the only user with an ACCOUNTADMIN role.

B- CREATE NETWORK POLICY ADMIN POLICY
ALLOWED_IP_LIST = ('192.168.1.100');
ALTER ROLE ACCOUNTADMIN SET NETWORK_POLICY = 'ADMIN_POLICY';

C- CREATE NETWORK POLICY ADMIN_POLICY
ALLOWED IP LIST = ('192.168.1.100')
BLOCKED_IP_LIST = ('0.0.0.0/0');
ALTER USER ABC SET NETWORK_POLICY = 'ADMIN_POLICY';

D- CREATE OR REPLACE NETWORK POLICY ADMIN_POLICY
ALLOWED_IP_LIST = ('192.168. 1. 100/0') ;
ALTER USER ABC SET NETWORK_POLICY = 'ADMIN_POLICY';

Answer:

C

Explanation:

Option C creates a network policy that allows only the IP address 192.168.1.100 and blocks all other IP addresses using the CIDR notation 0.0.0.0/01. It then applies the network policy to the user ABC, who has the ACCOUNTADMIN role. This ensures that only this user can access the Snowflake account from the specified IP address, while blocking their access from anywhere else. Option A does not block any other IP addresses, option B applies the network policy to the role instead of the user, and option D uses an invalid CIDR notation.

Question 8

Question Type: MultipleChoice

In general, the monthly billing for database replication is proportional to which variables? (Select TWO).

Options:

A- The frequency of changes to the primary database as a result of data loading or DML operations

B- The amount of table data in the primary database that changes as a result of data loading or DML operations

C- The frequency of the secondary database refreshes from the primary database

D- The number of times data moves across regions and/or cloud service providers between the primary and secondary database accounts

E- The number and size of warehouses defined in the primary account

Answer:

A, B

Explanation:

Snowflake charges for database replication based on two categories: data transfer and compute resources1. Data transfer costs depend on the amount of data that is transferred from the primary database to the secondary database across regions and/or cloud service providers2. Compute resource costs depend on the use of Snowflake-provided compute resources to copy data between accounts across regions1. Both data transfer and compute resource costs are proportional to the frequency and amount of changes to the primary database as a result of data loading or DML operations3. Therefore, the answer is A and B. The other options are not directly related to the replication billing, as the frequency of secondary database refreshes does not affect the amount of data transferred or copied4, and the number and size of warehouses defined in the primary account do not affect the replication process5.

Question 9

Question Type: **MultipleChoice**

What roles can be used to create network policies within Snowflake accounts? (Select THREE).

Options:

A- SYSADMIN

B- SECURITYADMIN

C- ACCOUNTADMIN

D- ORGADMIN

E- Any role with the global permission of CREATE NETWORK POLICY

F- Any role that owns the database where the network policy is created

Answer:

B, C, E

Explanation:

Network policies are used to restrict access to the Snowflake service and internal stages based on user IP address1. To create network policies, a role must have the global permission of CREATE NETWORK POLICY2. By default, the system-defined roles of SECURITYADMIN and ACCOUNTADMIN have this permission3. However, any other role can be granted this permission by an administrator4. Therefore, the answer is B, C, and E. The other options are incorrect because SYSADMIN and ORGADMIN do not have the CREATE NETWORK POLICY permission by default3, and network policies are not tied to specific databases5.

20%
DISCOUNT