

Snowflake.ARA-R01.v2024-05-06.q49

□□□□:	ARA-R01
□□□□:	SnowPro Advanced: Architect Recertification Exam
□□□:	Snowflake
□□ □□ □□□:	49
□□:	v2024-05-06
# □□ □:	257
# □□ □□□:	490
https://www.krdump.com/Snowflake.ARA-R01.v2024-05-06.q49.html	

NEW QUESTION: 1

□□ SQL □□□□ □□ □□□ □□□□□? (3□ □□).

□□□ □□□...□□□□□...FROM(...)

A. □□□ where □□ □□□□ □□□□ □□□□ □ □□□□.

B. □□ □□ □□□ □ □□□□.

C. □□ □□□ □ □□□□.

D. □□ □□□□ □□□□□.

E. □□□□ □□□□ □□ □□□□ □□□ □ □□□□.

F. ON ERROR - ABORT □ □□□ □□□ □ □□□□.

Answer: A,B,C ([LEAVE A REPLY](#))

SQL □□ Snowflake□□ □□□□ □□□□ □□ □□□□, Snowpipe□ □□ □□□□ □□□□
tables1□ □□□□ □ □□□□ COPY INTO <table> □□ □□□□ □□□□□. □ □□ FROM
□□ □□ □□□ □□□□ □□□ □□□ □□□□ table2□ □□□□ □□ □□□□□.

□□ □□□□ □□□□ □□□ □□□ □□□□2:

□□ □□□□ □□□ □□ □□□□ □□ □□□ □□□□ □□□ WHERE □□ □□□□ □□□
□ □□□□ □ □□□□. □□ □□:

SQLAI □□ □□. □□ □□ □□□□ □□□□□. FAQ□ □□ □□ □□.

createpipe mypipeas

□□□□ □□□

□□(

□□*from@mystage

□□□col1='A'andcol2>10

);

□□ □□ □□□ □ □□□□. □, □□ □□□ □ □□□ □□ □□□□ □ □□□ □□□□□ □
□□□ □□□□. □□ □□:

SQLAI □□ □□. □□ □□ □□□□ □□□□□. FAQ□ □□ □□ □□.

createpipe mypipeas

[illegible]

1: □□□ □□ | □□□ □□

NEW QUESTION: 2

```

-- ON_ERROR=SKIP_FILE -- COPY INTO -- -- --
-- CSV -- TABLE -- -- file5.csv -- --
-- Architect -- -- -- -- --
-- --

```

□□□□□□ file5.csv □□□ □□□□□ Architect□ □□ □□□ □□□□ □□□? (2□□ □□ □□□.)

- A.** `@%tablea□□ □□□a□ □□ RETURN_FAILED_ONLY = TRUE;`
- B.** `@%tablea□□ tablea□ □□□□□.`
- C.** `@%tablea□□ tablea□ □□ FILES = ('file5.csv');`
- D.** `@%tablea□□ tablea□ □□ FORCE = TRUE;`
- E.** `@%tablea□□ □□□□ □□ NEW_FILES_ONLY = TRUE;`
- F.** `@%tablea□□ tablea□ □□ MERGE = TRUE;`

Answer: ([SHOW ANSWER](#))

□□ A(RETURN_FAILED_ONLY)□ □□□ □□□ □□□ □□□□□. file5.csv□ □□□
□□□ □□□□□ □□ □□□□□ □ □□□ □□□□ □□□ □□□□ □□□□.

□□ D(FORCE)□ □□□□ □□ □□□□ □□□□□. file5.csv□□□ □□□□ □□□□□ □
□□ □□ □□□□□ □□□□□.

□□ E(NEW FILES ONLY)□ □□□ COPY □□ □□ □□□ □□□ □□□ □□□□□.

file5.csv □ □□□ □□ □□ □□□ □□□ □□□ □ □□□ □□□□ □□□□.

□□ F(MERGE) □ □□ □□□ □ □ □□□ □□□ □ □ □□□ □ □ □□□ □ □

□ □ □ □ □ □ □ □ □ □ □ . □ □ □ □ □ □ □ file5.csv □ □ □ □ □ □ □ □ □ □ □
□ □ □ □ □ □ □ □ □ □ □ .

```

COPY INTO tablea FROM @%tablea COPY INTO tablea FROM
.

```

```
@%tablea FILES = ('file5.csv') □□□□□□ file5.csv□ □□□□□. □ □□ □□ □□ □□□□  
□□□□□ □□ □□□ □□ □□ □□□□ □□□□ □□□□□.
```

NEW QUESTION: 3

CREDITCARDINFO ☐☐☐☐ CREDICARDND ☐☐ ☐☐ ☐☐ ☐☐ ☐☐ ☐☐ ☐☐
☐☐☐☐. ☐☐☐ ☐☐ ☐☐ ☐☐ ☐☐.

```
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;
```

CREDITCARDINFO □□□□ □□ □□□□ □□□□ □□□□.

□ □ □ □ □ □ □ □ □ □ □ □

2022-07-23 4321 5678 9012 1234

Snowflake `CREDITCARDNO` column is masked with the following mask pattern?

- A. `CREDITCARDNO` is masked with the mask pattern `PI_ANALYTICS`.
- B. `CREDITCARDNO` is masked with the mask pattern `CREDITCARDNO`.
- C. `PI_ANALYTICS` is masked with the mask pattern `CREDITCARDNO` and the mask pattern `4` is used to mask the last 4 digits of the credit card number.
- D. `PI_ANALYTICS` is masked with the mask pattern `CREDITCARDNO` and the mask pattern `'MASKED'` is used to mask the last 4 digits of the credit card number.

Answer: D (LEAVE A REPLY)

The `CREDITCARDNO` column in the `PI_ANALYTICS` table is masked with the mask pattern `CREDITCARDNO` and the mask pattern `4` is used to mask the last 4 digits of the credit card number. The `'MASKED'` mask pattern is used to mask the last 4 digits of the credit card number. Snowflake uses the `PI_ANALYTICS` table to store the credit card information.

Snowflake uses the `ALTER TABLE ... ALTER COLUMN` statement to alter the mask pattern of the `CREDITCARDNO` column. The `CREDITCARDINFO` table is used to store the credit card information. The `CREDITCARDNO` column is masked with the mask pattern `Creditcardno_mask` and the mask pattern `4` is used to mask the last 4 digits of the credit card number.

`ALTER TABLE CREDITCARDINFO ALTER COLUMN CREDITCARDNO MASKED WITH MASKING POLICY CREDITCARDNO; Snowflake uses the ALTER TABLE ... ALTER COLUMN statement to alter the mask pattern of the CREDITCARDNO column.`

`CREATE MASKING POLICY: Snowflake uses the CREATE MASKING POLICY statement to create a masking policy.`

`Snowflake uses the ALTER MASKING POLICY statement to alter the mask pattern of the CREDITCARDNO column.`

`ALTER MASKING POLICY: Snowflake uses the ALTER MASKING POLICY statement to alter the mask pattern of the CREDITCARDNO column.`

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: 4

The `CURRENT_TIME()` function returns the current time in the local time zone. The `CURRENT_TIMESTAMP()` function returns the current time in the UTC time zone.

- A. `COPY` is used to copy data from a table to a file.
- B. `COPY` is used to copy data from a file to a table.
- C. `COPY` is used to copy data from a table to a table.

□□□□ □□□□ □□□□ □□□□ □□□□□□□□ □□□ □ □□□ □□□□ □□□.
□ □□ □□□ □□□□□ □□□ □□□ □□ □□□ □□□ □□□? (2□□ □□□□□.)

- A.** □□□□ □□;
- B.** □□□□ □□;
- C.** □□□□□□ □□;
- D.** □□ □□;
- E.** □□□□□□ □□□□;

Answer: C,E (LEAVE A REPLY)

```
Snowflake
CREATE DATABASE account1;
IMPORT DATABASE account2;
IMPORT SHARE account3;
IMPORT PRIVILEGES account2;
CREATE SHARE account4;
--
```

NEW QUESTION: 8

□□□□ □□□□ VARIANT □□ JSON□□ □□□□ □□□□. □□ □□□ □□□□□ □□□
 □ □□ □□□ □□ □□□ □□□ □□□□□.
 □ □□□ □□□ □□□□□?
A. □□ □□□ □□□□□ JSON null□ □□□□□.
B. JSON□ □ □□□ □□□□□□□.
C. □□ □□□ □□□□□□ □□□□ □□ □□ □□□ □□□□□□□.
D. □□ □□□ □□□□□□ JSON □□ □□□ □□□ □□□ □□□□□.

Answer: B,D (LEAVE A REPLY)

□□□□ □□□□ VARIANT □□ JSON□□ □□□□ □□□□. □□ □□□ □□□□□ □□□
 □ □□ □□□ □□ □□ □□□ □□□□□□□. □□ □□ □□□□ □□ □□□ □ □□□□.
 JSON□ □ □□□ □□□□□□□. Snowflake□ □□ □□□□ □□□ □□ □□ □□ □□□ □
 □□□□ □□□□□ □□□□ □□□□ □□□□ □□ □□ □□ □□ □□□□□. JSON□ □
 □□□ Snowflake□ □□ □□□ □□□□ □□□ □□ □□□ □□□□□ □□□ □□□ □□□
 □. JSON□ □ □□□ □□□ □□ Snowflake□ □□□□ □□ □□ □□□□ □□ □□□□ □□
 □□□□ □ □□□□ □□ □□ □□ □□□ □□□ □ □□□□□.

□□□□ □□□□□. □□□□□ □□□□ □□ □□□□, □ □□□ □, □ □□ □□□ □, □□ □□ □, □□ □□1□ □□□□□.

SYSTEM\$CLUSTERING_DEPTH□ □□□□ □□ □□ □□□ □□ □□□□ □□□□ □□□ □□ □□□ □□□□ □□□ □□□□□. □ □□□ □□□ □□□ □□□ □ □□ □□ □□□□ □□□ □□□□ □□□□□ □□□ □□ □□ □□ □□□□□. □□□□□ □□□ □□□□ □ □□□□ □□□□ □□ □□□ □□□□ □□□□ □□ □□□□. □□□□□ □□□ □□□ □ □□□□□ □ □□□□ □□□□□2.

□□□□:

□□□\$CLUSTERING_INFORMATION | □□□ □□

□□□\$CLUSTERING_DEPTH | □□□ □□

NEW QUESTION: 10

□□□ □□ □□□□□□ ID □□□□□ Snowflake□□ □□ □□□ □□□□□ □□□.

□□ □□□ □□□□□□□ Snowflake□□ □□□ □□□□□□□□. □□□ □□□□ □□□ □

Snowflake□ □□□□ □□□□ □□□ □□ □□□□ □□□ □□□ □□□□ □□□□.

□□□□ □□ □□□□□ □□□ □□□ □□□ □□□□ □ □□□□?

A. ID □□□□□ Snowflake □□ OAuth□ □□□□ □□□. □□ □□ □□ □□ □□□ □□□□ □□□ □□□ □□□□ □□□□□ □□□.

B. ID □□□□□ Snowflake □□ OAuth□ □□□□ □□□. □□ □□ □□ □□ □□□ □□□ □ □□ □□□□ □□□□□ □□, □□□ □□□ □□□ □□ □□ □□□□ □□□□□ □□□.

C. ID □□□□□ Snowflake □□ SCIM□ □□□□□ □□□. SCIM□ □□ □ □ □□□□□ □□ □□□□ Snowflake□□ □□ □□□□ □□□□ □□□ □□□ □□□ □ □□□□.

D. ID □□□□□ Snowflake □□ SCIM□ □□□□□ □□□. □□ □ □ SCIM□ □□ □□□□□ □. □□□□ □□□□ □□□□ □□ □□ □□□□□ Snowflake□ □□□ □□□□□.

Answer: D (LEAVE A REPLY)

□□□□□□ ID □□□□□ □□ □□□ □□□□□ Snowflake□□ □□ □□□ □□ □ □□ □□□ □□□□□ □□ □□ □□□ SCIM(System for Cross-domain Identity Management)□ □□□□ □□□□□. SCIM□ □□□□□ Snowflake□ ID □□□□□ □□□□□ ID □□□□□ □□□ □□□ □ □□□ □□□ □ □□□ □□ □ □□□□. □□□□ Snowflake□ □□□□ □□□□□ □□□□□□ □ □□□□ □□□ □□□ □□ □□□□□. □□□□ □□□□ ID □□□□□ □□□ □ □□ □□□ □ □□ □□□ □□□□□ Snowflake□ □□ □□ □□ □□□□ □□ □□□ □□□□□. □□ □ □□ □□□□ □□□ □□□□□□ □□ □□□ □□ □□□□□ OAuth□ □□□□ □□ □ □□ □ □□□ □□ □□□ □□□□ □□□ □□□□ □□□ □□□□□.

NEW QUESTION: 11

Snowflake □□□□□ □□ □□ □□□ □□□□ □□ □□□□□□ □□□ □□□□ □□□□. □ □□□□□□□□□ □□ □□□□ □□□□.

□□□□□□□□ □□□ □□□□ □□□?

A. □□ □□□□□□□ □□□□ □□□ □□ □□□□□□□ □□□□□.

B. □□ □□□□ □□□□ □□ □□□□□□□□ □□□ □□ □□ □□□□□□□ □□□□□.

□□□ □□ □□ | □□□□ □□

Snowflake□□ □□ □□□ □□□□□□□ □□□□ □□ □□□ □□

NEW QUESTION: 16

□□□ □□ □□□ □□ □□ □□□□ CSV □□□ □□□□ Snowflake □□□□ □□□□ □□
COPY □□□ □□□ □□□.

```
copy into home_sales(city, zip, sale_date, price)
from (select t.$1, t.$2, t.$6, t.$7 from @mystage/sales.csv.qz t)
file_format =
(
  format_name = mycsvformat
  empty_field_as_null = true
  field_optionally_enclosed_by = ''
)
validation_mode - return_all_errors
;
```

□ □□□ □□□□ □□ □□□ □□□□□.

SQL □□□ □□: □□□ □□□□ 'validation_mode'

□□□ □□□□□ □□□□ □ □□□ □□□ □□□□□?

- A. VALIDATION_MODE □□□□□ □□ □□□□□ □□□□ □□□□ COPY □□ □□□□□.
- B. VALIDATION_MODE □□□□□ CSV □□ □□□ COPY □□ □□□□ □□□□.
- C. VALIDATION_MODE □□□□□ □□ □□ □□□□ □□□□ COPY □□ □□□□ □□□□.
- D. VALIDATION_MODE □□□ return_all_errors □□□ □□ □□□ □□□ □□□□□□.

Answer: C (LEAVE A REPLY)

VALIDATION_MODE □□□□□ □□□□ □□□□ □□□ □ COPY □□ □□□ □□□□ □□□□□. □□□ □ □ □□□□ □□□□ □□ □□ COPY □□ □□□ □□□□ □□□ □□ □□ □□ □□□□ □□□ □□□ □□□□ □ □□□□□. VALIDATION_MODE □□□ □□ □□ □□□□□□ □□□□ □□□□ COPY □□□ □□□□□1.

□□□ □□□ □□□ □□ □□□ □□□□ □□ □□□□ □□□□ □□□□□. □□□ □□ □□ JSON □□ XML □□□ □□ □□, □□ □□, □□ □□□□ □□ □ □□ □□□□ □□ □□□□ □□□□□2.

□□□□ □□□ VALIDATION_MODE□ □□ □□ □□□□ □□□□ COPY □□ □□□□ □□□□. □□□□□ □□□ □□ COPY □□ error1□ □□□□□.

□□□ □□ C□ □□□□□.

□□: : COPY INTO <table> : □□ □ □□□ □□

ARA-R01 □□ □□□ □□□□□ □□ DumpTop □□ □□□□ □□□ ARA-R01 □□!

DumpTop □ □□ **ARA-R01** □□ □□□ □□□□□□, DumpTop ARA-R01 □□ □□□ □□ □□□□□□ □□□ □□□□□□□. □□□□ □□□ □□□□ □□ DumpTop ARA-R01 □

NEW QUESTION: 17

□□□ □□□ □□□ □ □□ Snowflake □□□ □□□□□? (2□ □□).

- A. □□ □□
- B. □□ □□
- C. □□ □□□□
- D. □□ □□□
- E. □□□

Answer: A,B ([LEAVE A REPLY](#))

□□□ □□□ □□ □□□ □□□□□□□□ □□□ □□□ □□ Snowflake □□□ □□□ □ □
□ □□□□□. □□ □□□, □□ □□□, □□ □□, □□ □□□□ □□, □□ UDF □ □□□□
□□ Snowflake □□□□□□ □□□ □□□ □ □□□□. □□□ □□□ □□ □ □□□ □□□
□□□ □□□ □□ □□□□. □□□ □□□ □□□□□□□ □□□□ □ □□□ □□□ □□□
□□ □□□ □□□□□. □□□□□□ □□□ □□ □□ □□□ □□ □□□ □□ □□□ □□□
□□□ □ □□□□. □□□ □□ □□□ □□□ □ □□ □□□ □□ □□, □□ □□, □□ UDF
□ □□□□□□.

□□□ □□□ A□ B□□.

□□ □□□ □□□ □□ □□□ □□□ □ □□ □□□ □□□□ □□□□. □□ □□□, □□ □
□□ □ □□□□ □□□□□□ □□□ □□□□ □□□ □ □□□□. □□ □□□□□ □□ □□
□ □ □□□□. □□□□:

[□□□ □□□ □□ □□] 1

[□□ □□] 2

[□□□□□□ □□ □□ □□ □□] 3

NEW QUESTION: 18

□□□ □□□ Snowflake □□□ □□□□ □□□□□ □□ □ □□ □□□□ □□□□ □□□□
□□□.

□□□ AWS eu-west 2 □□(□□) □□□ Snowflake □□□ □□□ □□□□. □□□□□
Snowflake □□□ □□□ □□□ Azure □□□□ □□□□. □□□ □□□□ □□□ □□□ □□
□ □□□□ □□□?

- A.** 1. □□□ □□□□□.
2. □□□ □□□ □□□□□.
3. □□□□□ □□□□ □ □□□ □□□ □□□ □□□ □□□□□.

- B.** 1. □□□ □□□□□.
2. □□□ □□□ □□ □□□ □□□□□.
3. □□ □□□ □□□ □□□□□.

- C.** 1. db_share□□ □ □□□ □□□□□.
2. □□ □□□□□□ □ □□□□□ □□□□ □□ □ □□ db_share □□ □□□ □□□□□.

3. □□□□□ □□ □□□□ □□□□□.
4. □□□□ □□□□□ ds_share □□□ □□□□□.
- D.**
1. □□ □□ □□□ □□ □□□□□□□ □□ □□□□□□□ □□□□□.
2. □□□ □□□ Azure on Snowflake□ □□□□□□□ □□□□□.
3. □□□ □□□□ □□□ □□□ □□□□□.
4. □□□□□ □□□□ □ □□□ □□□ □□□ □□□ □□□□□.

B. SECURITY LOGS □□□□ ADD SEARCH OPTIMIZATION□ □□□□ □□ □□□ □□ □
□□ □□□□.

C. VPN_ACCESS_LOGS ☐ OWNERSHIP ☐ ☐ ☐ SECURITY_LOGS ☐ ☐ ☐ ☐
SEARCH OPTIMIZATION ☐ ☐ ☐ ☐ ☐.

D. VPN_ACCESS_LOGS ☐ ALL PRIVILEGES ☐ ☐ SECURITY_LOGS ☐ ☐ ☐
ADD SEARCH OPTIMIZATION ☐ ☐ ☐ ☐ ☐.

Answer: C (LEAVE A REPLY)

SnowPro Advanced: Architect Exam Study Guide

```

000 000000 000000 00000000 000 00 00 00 0000 000 0 000
0. 0000 00 OWNERSHIP 000 ADD SEARCH OPTIMIZATION 000 000000 00
00 000 00000 00 00000 00 00 000 00000 00000 0 00000. 000 0
00 VPN_ACCESS_LOGS00 OWNERSHIP 000 00 SECURITY_LOGS 00000 ADD
SEARCH OPTIMIZATION0 00000 00000. 00 00 00000 VPN_ACCESS_LOGS 00
00 SECURITY_LOGS 00000 0000 00 00 000000 00 000 00000 00000
0 00000. 00 0000 0000 0000 0000000 0000 00 00000 0000 0000 0000
0 00 0000 000000 00000. 00000:

```

SnowPro Advanced: 000 00 00 000, 11000, 00 2.3.1

Snowflake □□□: □□ □□□ □□□ □□□

NEW QUESTION: 21

☐☐☐☐ ☐☐☐☐☐ ☐☐☐☐☐☐ ☐☐ ☐☐☐☐☐☐ ☐☐☐☐☐☐

[illegible]

A. □□□ □ □□□□ □□□ Snowflake □□□ □□□ □□ □□□□ □□□□ □□□ □ □□□
□.

B. □□□ Snowflake □□ □□ □□□□ □□□□ □□□.

C. ☐☐☐☐ ☐ ☐☐☐☐☐ ☐☐ ☐☐☐☐☐☐ ☐☐ ☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐.

D. □□□ □□ □□□ □□□□ □□□ □□□□ □□□.

Answer: (SHOW ANSWER)

[illegible]

□ □ □ □ :

Snowflake ☐☐: ☐☐☐☐☐ ☐☐

Snowflake 00: 00 00

NEW QUESTION: 22

物联网 IoT 数据 JSON 数据 数据 数据. JSON 数据
 数据 数据 数据 数据 数据. 数据 数据 数据 数据 数据 数据
 数据 数据 数据 数据. 数据 数据 create_date 数据 数据 数据 数据
 数据 数据 数据.

☐☐☐ ☐☐☐☐☐☐ ☐☐ ☐☐☐☐ ☐ ☐ ☐☐☐☐

A. JSON □□□□□ □□□ □□ □□□ □□□□□ □□ □□□□ □□□□□. □□□□□ □□□
□□□ □□□□□□ create_date □□□ □□□□□. □ □□□ □□□ □□□□ □□□ □□□
□□□□□.

B. JSON □□□□□ □□□ □□ □□□ □□□□□ □□ □□□□ □□□□□. □□□□ □□□ □□□ varchar□ create_date □□□ □□□□□. □ □□□ □□□ □□□□ □□□ □□□ □□ □□□.

C. □□ □□ □□□ □□□ □□□□□. □□□ □□ □□ □□ □□□□□ XL □□□ □ □□ □
□□□ □□□□ □ □□□ □□ □□□ □□□.

D. □□ □□□ □□□ □□ □□□ □□□□□. □□□□ □□□□□ □□ □□ □□□ □
□□□ □□ □□ □□□ □□ □□□□ □□□□□ □□□□□.

Answer: A (LEAVE A REPLY)

```

CREATE TABLE IF NOT EXISTS users (
  id INT PRIMARY KEY,
  name VARCHAR(50) NOT NULL,
  email VARCHAR(100) NOT NULL,
  create_date TIMESTAMP NOT NULL,
  metadata JSON
);

```

[illegible][illegible]

00 D0 0000 00 0 00 000000 00000 00000 000000 00000 00000 0
 0000. 00 0000 0000 00 00000 0000 0000000 Snowflake 00 0000 0000
 0 0000 00 00000 000000 000000 00 00 0 00000. 0000 00 000000 00 0
 0000 00 0 00 00000, 0000 000000 0000 000000 000000 000000, 00 00 0
 00 00 00 0000 00 00000 000000 0000. 00000:

```
Snowflake >>: Snowpipe> >>>> >>> >>: > >>>>> Snowpipe> >>>> >> >
>> >>>> Snowflake >>>> >>>>> >>>> >>> >>>>>. >> ON_ERROR,
PURGE > SKIP_FILE> >> >> >>> >>>> >>> >>> >>>>> >>>> COPY
INTO >>> >>> >>>> >> >>>>>.
```


A. Kafka ☐ Snowflake ☐ ☐

Snowflake `CREATE OR REPLACE`, `ALTER`, `DROP` or `GRANT`
`REVOKE` or `USE ROLE`. `SHOW` or `DESCRIBE` or `EXPLAIN`
or `SET`. `WAIT FOR`, `EXECUTE IMMEDIATE` `AUTO_REFRESH`
or `MODIFY` or `ACCOUNTADMIN` or
or `SECURITYADMIN` or `SYSADMIN`

Which of the following is a valid Snowflake role name?
 A. ROLE
 B. role
 C. ACCOUNTADMIN, USER with PRIVILEGE
 D. ROLE:
 E. ROLE | ROLE
 F. ROLE | ROLE
 G. ROLE | ROLE

NEW QUESTION: 29

Which of the following is a valid Snowflake role name?
 CREATE DATABASE SANDBOX;
 CREATE ROLE INTERN;
 CREATE TABLE SANDBOX.PUBLIC.AGENDA (ID INT, ITEMS STRING);
 GRANT SELECT ON ALL TABLES IN SCHEMA SANDBOX.PUBLIC TO ROLE INTERN;
 GRANT ROLE INTERN TO USER USER1;
 USER1
 Which of the following is a valid Snowflake role name?
 A. ROLE
 B. role
 C. ACCOUNTADMIN, USER with PRIVILEGE
 D. ROLE:
 E. ROLE | ROLE
 F. ROLE | ROLE
 G. ROLE | ROLE

Answer: B,C (LEAVE A REPLY)

Which of the following is a valid Snowflake role name?
 A. ROLE
 B. role
 C. ACCOUNTADMIN, USER with PRIVILEGE
 D. ROLE:
 E. ROLE | ROLE
 F. ROLE | ROLE
 G. ROLE | ROLE

NEW QUESTION: 30

Which of the following is a valid Snowflake role name?
 A. ROLE
 B. role
 C. ACCOUNTADMIN, USER with PRIVILEGE
 D. ROLE:
 E. ROLE | ROLE
 F. ROLE | ROLE
 G. ROLE | ROLE

Snowflake □□□□ □□□ □□□ □□□ □□□□□ □□□□ □□□ □□ □□□ □□ □□□
 □□ □□□□ □ □□□ □□□□□ □□□.

☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ?

A. □□□ □□ □□ □□□ □□□□ □□ □□□□ □□□□□ □□□□ □□□□□.

```

□□ □□ □□ reader_acctl admin_name = userl , adroin_password 'Sdfed43da!44T , type =
reader:

```

B. ☐☐☐ ☐☐ ☐ ☐☐☐ ☐☐☐ ☐☐☐☐ ☐☐☐ ☐☐☐☐.

```

0000 00 rap_acct (acct_id varchar) 00 00 0000 00 -> 00 000 0000 0
0000.

```

```
'acctl role' = current role()□□ true, □□□ □□□ false end;
```

C. □□□ □□□ □□□□ □□ □□□ □□ SIMULATED_DATA_SHARING_CONSUMER□□
□□ □□□□□□ □□□□□.

□□ □□ Simulated data sharing consumer □□ - '□□□ Acctl*

D. □□ □□ □□□ □□□□□ □□□ □□ □□ □□□ □□□□□.

```
□□ □□ □□ □□ □□ □□ = 'Consumer!' □□ □□ = true
```

Answer: C (LEAVE A REPLY)

SIMULATED_DATA_SHARING_CONSUMER □□ □□□□□ □□□□ □□□ □□□□ □□
 □□□□ □□□□□ □□□ □□ □□□□ □□□□□ □□□ □□□ □□□ □□□ □□□□ □□
 □□□□ □ □□□□. □ □□□□□ □□ □□ □□□□ □□□ □□□□ □□□□ □□□□□
 □□ □□ □□ □□ UDF□ □□□ □ □□□ □□□□ □□□□ □□□ □□□□ □□□□ □ □
 □□ □ □□□□. □ □□□□□ □□□ □□□ □□□□ □□□□ □□□ □□□□ □□□□ □
 □ □□□□□□□ □□□ □ □□ □□ □□□ □□□□ □ □ □□□□. □□ □□□ □□ □□□
 □□□□□ □□□ □□□□ □□□□□ □□□□□ □□□□ □□□□ □□□□□. □□□□□
 □□□□:

SIMULATED DATA SHARING CONSUMER □□ □□□□ □□

SnowPro Advanced: □□□ □□ □□ □□□

ARA-R01 00 000 00000 00 DumpTop 00 0000 000 ARA-R01 00!

DumpTop   **ARA-R01**                           

Dumps, **30%OFF** Special Discount: **KrDump**)

NEW QUESTION: 32

[illegible]

A. □□□□ □□□ □□□□ □□□ □□□ □□ □□□□ □□ □□ □□□ □□□ □□□□□.

B. 在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

C. 在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

D. 在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

E. 在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

Answer: A,C ([LEAVE A REPLY](#))

在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。在 Snowflake 中，使用 COPY INTO 命令时，如果指定了 ON_ERROR 选项，那么当遇到错误时，默认的行为是继续执行后续的数据加载，而不会中断整个操作。

000000 00000 0000 00 0000 000000 00 0000 0000 00 0000 00 00
 00 0000000 00 0000 00 0000 0000 0000 0000 0 00000. Zero-Copy 00
 0 0 00000 00 00 0000 000000000 000000 0000 0 0000, 00000 00 00
 000000 0000 0000 0 00000000 00 0 00000. 00000 00 00 0 00 0000
 0 0000 0000 0 0000, 0000 00 00 0000 00 000000000 00 0000 0000 0
 00 00000.7 0000 000000000 00 00 0 0000 000000 000000 00 0000 00
 00 00000 000000 0000000. 00000 0000 00 0000000 000000 0000000 00
 0000. 0 000000000 000000 0000 00 0000 0000000 00 0000 0000 00 0
 00 00 00000 00000000 00 0000 00 0000 0000 0000 0000 0 00000. 0
 0 00 0000 00000 0 000000000 00000 0000000000 00000 0000 0 00
 0, 00000 00 000000000 0000 0000 0 00000000 00 0 000000. 00 0 00
 00000 0000 00000 00 00 0 0000 000000 00 0000 00000 0 0000, 0000
 00 00 0000 00 000000000 00 0080 0000 0000 00000.

- 1: SnowPro 00: 0000 | 00 0000 9
 - 2: 0000 00 | 00 00
 - 3: 0000 00 | COPY0 00000 00000 0000 00
 - 4: 0000 00 | 00 0 0000 00
 - 5: 0000 00 | 0000 00 00
 - 6: 0000 00 | 00 00
 - 7: 0000 00 | 00000000, 0000 0 0000 00
 - 8: 0000 00 | 0000 0 0000 00 00
- : SnowPro 00: 0000 | 00 0000
 : 00 00
 : COPY0 00000 00000 0000 00
 : 00 0 0000 00
 : 0000 00 00
 : 00 00
 : 00000000, 0000, 0000 00
 : 0000 0 0000 00 00

NEW QUESTION: 33

00000 00 0000 00000 00 0000 00 60 10 0000 00 800 STAGING 00000
 0000000 0000 0000000.
 STAGING 00000 00 0000 5000000.
 Architect0 00 00000 0000000.
 (timestamp => '2021-06-01 08:00:00')0 CREATE SCHEMA STAGING_CLONE 00 00000;
 Architect0 00 0000 0000000. 0000 STAGING00 00 00 00000 0000 0 0000
 0. 0000 0000 0000 00 00 0000 0000000 00 00 00 0000000.
 00 00 00000 0000 0000 000000 0000 0000000.
 CREATED_ON|00|DROPPED_ON

☐ ☐☐☐ ☐☐☐ ☐☐☐ ☐ ☐☐☐

_____ Snowflake _____
_____. _____, _____.

_____?

A. _____ Snowflake _____.

B. _____ Snowflake _____.

C. Snowflake _____.

D. _____ Snowflake _____.

Answer: D ([LEAVE A REPLY](#))

_____ Snowflake _____.

Snowflake _____, _____.

_____: _____.

_____: _____.

_____: _____.

_____, Snowflake _____: _____.

_____]

NEW QUESTION: 37

_____ IOT _____ 20 _____.

```
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)
)
```



Which column is the primary key?

1. DeviceId, IOT_timestamp, CustomerId
2. City, DeviceManufacturer
3. UniqueId

What is the data type of the primary key?

- A. IOT_timestamp
- B. VARCHAR
- C. DeviceId, CustomerId
- D. NUMBER

Answer: (SHOW ANSWER)

The correct answer is D. NUMBER. In the provided SQL code, the column UniqueId is defined as a NUMBER type and is the only column without a data type specified in parentheses, which typically indicates a primary key. The other columns (DeviceId, DeviceManufacturer, CustomerId, IOT_timestamp, City, Location) are all VARCHAR or timestamp_ntz types. The question asks for the primary key, and the correct answer is UniqueId, which is a NUMBER type.

NEW QUESTION: 38

Which of the following is a valid Snowflake identifier?

- A. 'SNOWFL%'
- B. SNOWFLAKE MAX(FLAKE_ID)
- C. SNOWFLAKE (*)
- D. FLAKE_ID SNOWFLAKE (FLAKE_ID)

□□□□ □ □□ □□ Snowflake □□□ □□□□ Snowflake □□□□ □ QA □□□ □□□□□ □□□□□□.

QA □□□ □□ □□□ □□□□ □□□ □□□□ □□ □□□ □ □□□□□□ □□□ □□ □□ □□□ □□□□ □□□□□ □□ □□□□. QA □□□ □□ □□□□□□ □□□ □□□□ □□ □□□□ Production □□□ □□□ □□□□ □□□□ □□□□□□ □□□ □□□ □□□□□ □□□□ □□ □□□ □□□□.

□□ □□□ □□□□ □□□ □□□ □ □□□□□□ □□□ QA □□□ □□□ □ □□□ □ □ □□ □ □□□ □□ □□□ □□□□□?

A. 1) □ □□□□□□□ □□ □□□□ □□□ □□□ □□□□□.

2) □□□□□ QA □□□ □□ □□□□ □□□□□.

3) QA □□□ □ □□□□ □□ □□□□□□□ □□□□□.

4) □□ □ □□ □□□□□□□ □□□□ □□□□□.

5) □□□ □□□□□□□□ □□ □□□□ □□□□□.

B. 1) □□□□ □□□□ □□□ □□□□□.

2) □□□ □□ □□ □□ □□□ □□□□ QA □□□ □□□ □□□□.

3) □□□□ □□□ □ □□□□ □□□□□ □□□□□ □□ □□□ □□□□ □□□ □□□□.

4) Snowpipe□ □□□□ QA □□ □□□

C. 1) □□□□ □□□ □ □□□□□□□ □□ □□□ □□□□□□.

2) QA □□□ □□□ □□□□□□ □□

3) □□□ □□□ □□□□□□□ □□□□ □□□□□.

4) □□□ □□□□□□□□ □□ □□□□ □□□□□.

D. 1) Production □□□□ QA □□□ □□□□ □□ □□□ □□□ □□ □□ □□□□ □□□□ □□ □□□ □□□□.

2) Production □□□ □ □□□□ □□□□ QA □□□ □ □□□□ □□□ □□ □□□□□ □□ □□ □□□ □□□□□.

Answer: C ([LEAVE A REPLY](#))

□ □□ □□□ Snowflake□ □□ □□ □□ □□□ □□□□ Production □□□ □□□ □ □□□ □□□ □□□ QA □□□□ □□□□□ □□ □ □□□□□. □□□□ □□, □□ □ □□□□ □□ □ □□□ □□ □□□□ □□□□□ □□□ □□□□ □□□□□. □□ □□□ □□□ □□□ □ □□□□□ □□□□□. □□□ □□□□□□□ □□□□ □□□□□□ QA □□□ □□ □□□□ □□□ □□ □□ □□□ □□□□ □□ □□□□ □□□ □ □□□□. □□□ □□ □□□ □□□ □ □□□□. □, □□□□ □□□□ □□ □ □□ □□□□ □□□ □□□□ □□□□. □ □□ □ □□□ □□□ □□ □□□□□ □□□□ □□□□□ □□□ □ □□ □□ □□, □□, Snowpipe □□ □□ □□□ □□□□ □□□□.

□□□□:

□□ □ □□ □□ □□

□□ □□□ □□ □□□□□□ □□

□□ □□ □□

NEW QUESTION: 41

□□ □□□□ □□ □□□□ □□ □□□ □□□□□□ □□□ □□□□□?

- A.** □□ □□□□ □□□□□.
- B.** □□ □□□ □□□ □□□□.
- C.** □□ □□□ □□ □□□□ UTF-8□ □□□□□.
- D.** □□ □□□□ □□ □□ □□□□□ □□□□ □□ □□□ □□□□□.

Answer: ([SHOW ANSWER](#))

00 000 000 0000 000 00 00 0000 00 0000 00 000 00000
 00000. 00 0000 00000 0000 00 00 00 0000 000 00 0000
 00000 0000 000 0000 00 000 000 00 0000 000. 000 00 0
 0 00000(0: 00 00, 00, 00 00 00 00) 00 00 000(0: 0 0 00 JSON 0
 0)00 000 0 0000. 00 0000 0000 00 000 00000 00 0000 00
 00 00 000 0000 0000 000 00 0 000 000 0 0000.2 00 000 0
 0 0000 00 0000 00 000 00000 0000 000 0000.

□□ □□□ □□□ □□□□. □□ □□□ □□ □□□ □□□□ □□□□ □ □□□ □□ □□□
 □□□ □□□ □□□□. □□ □□□ □□ □□□□ □□□□ □□ □□□ □□□□ □□□ □□
 □□□.3 □□ □□□ □□ □□□□ UTF-8□ □□□□□. Snowflake□ UTF-8, UTF-16, UTF-32,
 ISO-8859-1 □ Windows-1252□ □□ □□ □□□ □□□ □□ □□□ □□ □□□□ □□□□□
 □ □□□ □□ □□□ □□□ □□ □□□□. Snowflake□ □□□ □□ □□□□ □□□□ □□□
 □ □□ □□□ □□ □□□□□ UTF-8□ □□□□□.4 □□ □□ □□ □□ □□□ □□□□ □
 □ □□□ □□□□□. □□ □□□□ Amazon S3, Google Cloud Storage □□ Azure Blob
 Storage□ □□ □□ □□□ □□□ □□□ □□□ □ □□□□ □ □□□ □□□□ □□□□. □
 □ □□□ □□ □□□□ □□ □□ □□□□ □□□□ □□□□ □ □□□□□5

1: SnowPro □□: □□□ | □□ □□□

2: □□□ □□ | □□ □□□ □□□□

3: |

4: □□□ □□ | □□□ □□□ □□□ □□□ □□□ □□ □□ □□

5: |

: SnowPro ☐☐: ☐☐☐ | ☐☐ ☐☐☐

: □ □ □ □ □ □ □ □ □

. □ □ □ □ □ □ □ □ □ □

: □ □ □ □ □ □ □

:

: □□□ □□□ □□□ □□ □□□□ □□ □□ □□

:

NEW QUESTION: 42

□□□□ □□□□ □□□ □□□□ □ □□ DDL □□□ □□□□□□□.

```

TASK ts_insert_new_customers
HOUSE = MY_WH
rule = '5 minute'

m$STREAM_HAS_DATA('MYSTREAM')

T INTO new_customers(id, name) SE
stream WHERE METADATA$ACTION = 'I

```

A. MY_WH □□□□□□ □□□□ □□□□ □□ 5□□□ □□□□□□.

B. MY_WH □□□ □□□□ □□□ □□ □□ □□□□□□.

C. MY_WH □□□ □□ □□□□ □□□□.

D. MY WH □□□□□□ □□□□ □□□ □□ □□□□ □□□ □□□□□□.

```
MY_WH 000 0000 000 00 00 0000000. 00 000 0000 0000 000
0 00000. 0, 0000 0 0000 00 00 000 00000. 00 000000
auto_suspension - 6000 00000. 00 0000000 600 00 000 000 0000 0
0 0000 00000. 000 000 0000 000 00 00 0000000. 0000:
[00 000 | 000 00]
[000 0 00 00 | 000 00]
[00 000 | 000 00]
```

[illegible]

- C. 00 0000 00000 00000 00 00000 00000 00 0000 00 00 0000 0000
0 0000 00000 0000 00000 0000.
- D. 00 000000 00000 00000 0000 0000 00000 SnowSQL 0000 00000 00
0 00 00 0000 00000 0000 00000 00000 0000.
- E. 00 0000 00 0000 000000 00 000000 0000 0000 0000 00 0000 00
00 0000.

Answer: B,C ([LEAVE A REPLY](#))

00000 000000 00000 0000 00 0000 00000 00 Architect 00 0000 0000
0 00000.

POS 00 00 0000 0000 00000 0000 0000 00000 00 0000 00000. 00 0
00 00 0000 00000 00 00 0 0000 0000 00000 0000. 2 00 0000 0000
Snowflake 00 00000 00000 Snowpipe 0000000. Snowpipe 00 AUTO_INGEST =
true 0000000 0000. 0, 0 0000 00 0000 00000 00000 000000 000000. 0
0 Snowpipe 00 0000 00000 00 0 00 00000 0000 00000 00 0000 0000
0 0000.3 Snowpipe 00 00 INSERTS 00000 00 00000 00000 0000000. 00
0000 00 00, 00, 00 0 0000 00 0000 00 0000 00000 000000 00 00
0000 0000. 000000 0000 00 00 0000 00000 00 0000 0000 0000.4 00
0000 00000 00000 00 000000 0000 00000 0000 000000. 0000 0000
0000000 00000 00 0000 00000 00 0000 0000000 00000, 00 0000
00000 00, 00 0 00 0000 00000 0000. 00 0000 00000 00000 00000 0
00 00 000000 00 0000 00000 0000. 0000 10000 00 500000 00 0000
00 00 0000 00000 0000 0000000 000000 0000.

00 0000 00 0000 0000 00000 0 00000 0000 00 00000 00000.

00000 0000 000000 Snowflake 00000 00 00 0000 00000 0000. 0 0000
0000 00000000 00 00 0000 00000 000000 00000 00000.

0000 0000000 00000 0000 0000 00000000 00 00 0000 00000 00 0000
0 00 00000 0000000. 00 00 Snowflake 0 0000 00000 00 00000 0000
00 00 00 0000 0000000. 00 Snowpipe 0000 0 0000 0000 0000 00 00
0000 0000 00000 00000 0000 000000 00000.

00 0000000 00000 00000 0000 0000 00000 SnowSQL 0000 00000 0000
00 00 0000 00 0000 00000 00000 0000. Snowpipe 00 00000 00000 0
0 00 00000 0 0000 00000 0000 0 00000 0 0000 000000 00000. 00 00
0000 00000 0000 00000000 0 00 000000 00000 00000, 00 0000 00
000000 00000 00 0000000 00 0000 0000 000000 00000.

00 0000 00 0000 0000000 00 0000000 0000 0000 00 0000 00000 00
0. Snowflake 0000 0000 00 0000000 0000 0 00 0000 0 0000 00 00000 0
0000. 0000 00 0000 10000, 0000 00 00 0 0000 0000 0000000 0 0000 0
0000 000000. 00 0000 00 copy into 0000 000000 00 00 00, 00 0000, 00
00 0000 00000 00 Snowpipe 0000 000000 000000. 000000:

1: SnowPro 00: 0000 | 00 0000

[Snowflake □□: □□□□ □□ GRANT □□]

NEW QUESTION: 45

□□□□ ORDER_ADMIN □□□ □□□□ □□□□ □□ □□□ □□□□ □□ ORDERS □□
 □□ □□□ □□□□ □□(5□□ □□ □□ □□□ □□)□ □ □□ □□□ □□□□ □□□. □□
 □□□(ORDER_MANAGER)□ □□□□ □□ □□ DELETE □□□ □□□ □□□□.
 ORDER_MANAGER □□□ □□□ DELETE □□ □□ ORDER_ADMIN □□□ □□□□□ □
 □□□ □□□ □□□□□ □□□ □□ □□□?

- A.** `CREATE TABLE IF NOT EXISTS ORDER_MANAGER (`
 `ORDER_ID INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,`
 `ORDER_NO VARCHAR(20) NOT NULL,`
 `ORDER_STATUS VARCHAR(20) NOT NULL,`
 `ORDER_TIME VARCHAR(20) NOT NULL,`
 `ORDER_MANAGER VARCHAR(20) NOT NULL,`
 `PRIMARY KEY (ORDER_ID)`
`) ENGINE=InnoDB;`
- B.** `CREATE TABLE IF NOT EXISTS ORDER_MANAGER (`
 `ORDER_ID INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,`
 `ORDER_NO VARCHAR(20) NOT NULL,`
 `ORDER_STATUS VARCHAR(20) NOT NULL,`
 `ORDER_TIME VARCHAR(20) NOT NULL,`
 `ORDER_MANAGER VARCHAR(20) NOT NULL,`
 `PRIMARY KEY (ORDER_ID)`
`) ENGINE=InnoDB;`
- C.** `CREATE TABLE IF NOT EXISTS ORDER_MANAGER (`
 `ORDER_ID INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,`
 `ORDER_NO VARCHAR(20) NOT NULL,`
 `ORDER_STATUS VARCHAR(20) NOT NULL,`
 `ORDER_TIME VARCHAR(20) NOT NULL,`
 `ORDER_MANAGER VARCHAR(20) NOT NULL,`
 `PRIMARY KEY (ORDER_ID)`
`) ENGINE=InnoDB;`
- D.** `CREATE TABLE IF NOT EXISTS ORDER_MANAGER (`
 `ORDER_ID INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,`
 `ORDER_NO VARCHAR(20) NOT NULL,`
 `ORDER_STATUS VARCHAR(20) NOT NULL,`
 `ORDER_TIME VARCHAR(20) NOT NULL,`
 `ORDER_MANAGER VARCHAR(20) NOT NULL,`
 `PRIMARY KEY (ORDER_ID)`
`) ENGINE=InnoDB;`

Answer: ([SHOW ANSWER](#))

[illegible]

```
ORDER_MANAGER □□□ □□□ □□□□ □□□□ □□ □□□□□ □□□□□□
ORDER_ADMIN □□□ ORDERS □□□□ □□ □□ □□□□ □□□ □□□□ □□□ □□□
□□□□ □□□□ □□ □□□ ORDER_ADMIN □□□ □□□ □ □□□□. □□ □□□□□ 5
□□ □□ □□□□ □□□□ □□□ □□□□ □□□ □□□□ □□, ORDER_MANAGER □□
□ □□ □□□□□ □□ USAGE □□□ ORDER_ADMIN □□□ □□□□ □□□. □□ □□
ORDER ADMIN □□□ □□ □□□□□ □□□□ □□□ □□12□ □□□ □ □□□□.
```

□ □ □ □ :

Snowflake □□□: □□ □□□□

Snowflake ☐☐☐: ☐☐☐☐ ☐☐ ☐☐☐☐ ☐☐ ☐☐☐☐ ☐☐

NEW QUESTION: 46

☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐.

□□ □□ □□□ □□□ □□□□ □□□ □ □□ □□□ □□□□ 6□□□ □□ □□□□
□□□ □□□ □□□□□ □□□.

Which of the following is a valid Snowflake table name? (Select all that apply.)

- A. `event_timestamp` is a valid Snowflake table name.
- B. `user_events` is a valid Snowflake table name.
- C. `event_timestamp` is a valid Snowflake table name.
- D. `Case` is a valid Snowflake table name.

Answer: C (LEAVE A REPLY)

Which of the following is a valid Snowflake table name? (Select all that apply.)

event_timestamp is a valid Snowflake table name. Case is a valid Snowflake table name. user_events is a valid Snowflake table name. table2 is a valid Snowflake table name.

Which of the following is a valid Snowflake table name? (Select all that apply.)

event_timestamp is a valid Snowflake table name. Case is a valid Snowflake table name. user_events is a valid Snowflake table name. table2 is a valid Snowflake table name.

Which of the following is a valid Snowflake table name? (Select all that apply.)

event_timestamp is a valid Snowflake table name. Case is a valid Snowflake table name. user_events is a valid Snowflake table name. table2 is a valid Snowflake table name.

Which of the following is a valid Snowflake table name? (Select all that apply.)

- 1: `event_timestamp` is a valid Snowflake table name.
- 2: `Case` is a valid Snowflake table name.
- 3: `user_events` is a valid Snowflake table name.

ARA-R01 is a valid Snowflake table name. DumpTop is a valid Snowflake table name. ARA-R01 is a valid Snowflake table name.

DumpTop is a valid Snowflake table name. ARA-R01 is a valid Snowflake table name. DumpTop ARA-R01 is a valid Snowflake table name.

□□□ □□□□ □□ □□□□□ □□□ □□□□ □□□□□. □□ □□□
MAX_CONCURRENCY_LEVEL □□ □□□ □□□□ □□□ 16□□ □□ □□□ □□□□ □□
□□□□□ □□ □□□ □ □□□ □□□□ □□□ □□□ □□□□ □□□□. □□□□:
[□□□□□ □□□□ □□] 1
[Snowpark Python□ □□□ □□□□ □□ □□] 2
[□□□ □□: □□□□□ □□□ □□] 3

NEW QUESTION: 49

Snowflake □□ □□ □□□ □□ Snowflake □□□ □□ □□□ □□ □□□ □□□□□?
(3□ □□).

- A. □□□□□□
- B. □□□
- C. □□□
- D. □□□□
- E. □□
- F. □□

Answer: A,E,F ([LEAVE A REPLY](#))

□□ □□□ Snowflake□□ □□□ □□□ □□□ □ □□ □□□□□□. □□ □□□□ □□□□
□□, □□□, □□□, □□, □□, □□□, □□, □□□□, □□□, □□, □□□, □□, □□□□□
□ □□□ □□□□□1.

Snowflake □□ □□ □□□ □□ □□□ □□□ □□□ □□□□ □□□□ □□□ □□□□□. □
□□ □□□□□ □□ □□□ □□ □□□□□□, □□ □ □□□□□□ □□□□ □□□□□. □
□□□□□□□□□ □□□, □, □□□□, □□□, □□, □□□□, □□□, □□ □ □□□□ □□□
□□□□ □□□□ □□□□. □ □□□□ □□ □□□□ □□ □□□ □□ □□□ □□□ □ □□
□□. □ □□□□□□ □□ □□□ □□ □□□ □□□□ □ □□□ □ □□□□2.

Snowflake □□ □□ □□□ □□ Snowflake □□□ □□ □□□ □□ □□□ □□□□□□, □□
□ □□□□□□□□□. □□□□ □□□ □□ □□□□ □□ □ □□□□ □□ □□ □□□ □□□□
□□□□. □□ □□□ □□□ □□ □□□ □□□□ □□□□.

□□□□ □□□ □□ □□□□□□□ □□□□. □□□□ □□ □□□□□□ □□ □□□□□ □
□□3.

□□□□ □□□ □□ □□□□ □□□□. □□□□ □□ □□□ □□ □□□□□ □□□4.
□□□ □□□ □□ □□□□ □□□□ □□□□. □□ □□□ □□ □□□ □□ □□□ □□□□
□□□.

□□□□:

- 1: □□□ □□ □□ | □□□ □□
- 2: □□ □□ | □□□ □□
- 3: □□□ □□ | □□□ □□
- 4: □□□ □□ | □□□ □□
- [5]: □□□□ □□ | □□□ □□

ARA-R01 ☐☐ ☐☐☐ ☐☐☐☐☐ ☐☐ DumpTop ☐☐ ☐☐☐☐ ☐☐☐ ARA-R01 ☐☐!
DumpTop ☐ ☐☐ **ARA-R01** ☐☐ ☐☐☐ ☐☐☐☐☐☐, DumpTop ARA-R01 ☐☐ ☐☐☐ ☐☐
☐☐☐☐☐☐ ☐☐☐ ☐☐☐☐☐☐☐. ☐☐☐☐☐ ☐☐☐☐☐ ☐☐☐☐☐ ☐☐ DumpTop ARA-R01 ☐
☐☐ ☐☐☐☐☐. <https://www.dumptop.com/Snowflake/ARA-R01-dump.html> (**163** Q&As
Dumps, **30%OFF** Special Discount: **KrDump**)