Snowflake CLI – Homework / Play Tasks

1. Basic CLI Usage

SnowCLI is a command-line tool used to interact with Snowflake — run queries, manage objects, stages, warehouses, etc.

Check the Snowflake CLI version.

snow –version

snow connection add

```
C:\Users\Admin>snow --version
Snowflake CLI version: 3.13.0

C:\Users\Admin>snow connection add
Enter connection name: snow_con
Enter account: FSBBSTN-IH05083
Enter user: ROSHINI
Enter password:
Enter role:
Enter warehouse:
Enter database:
Enter schema:
Enter host:
Enter port:
Enter region:
Enter authenticator:
Enter workload identity provider:
Enter private key file:
Enter token file path:
Wrote new connection snow_con to C:\Users\Admin\AppData\Local\snowflake\config.toml
```

Test your connection to Snowflake using the CLI.

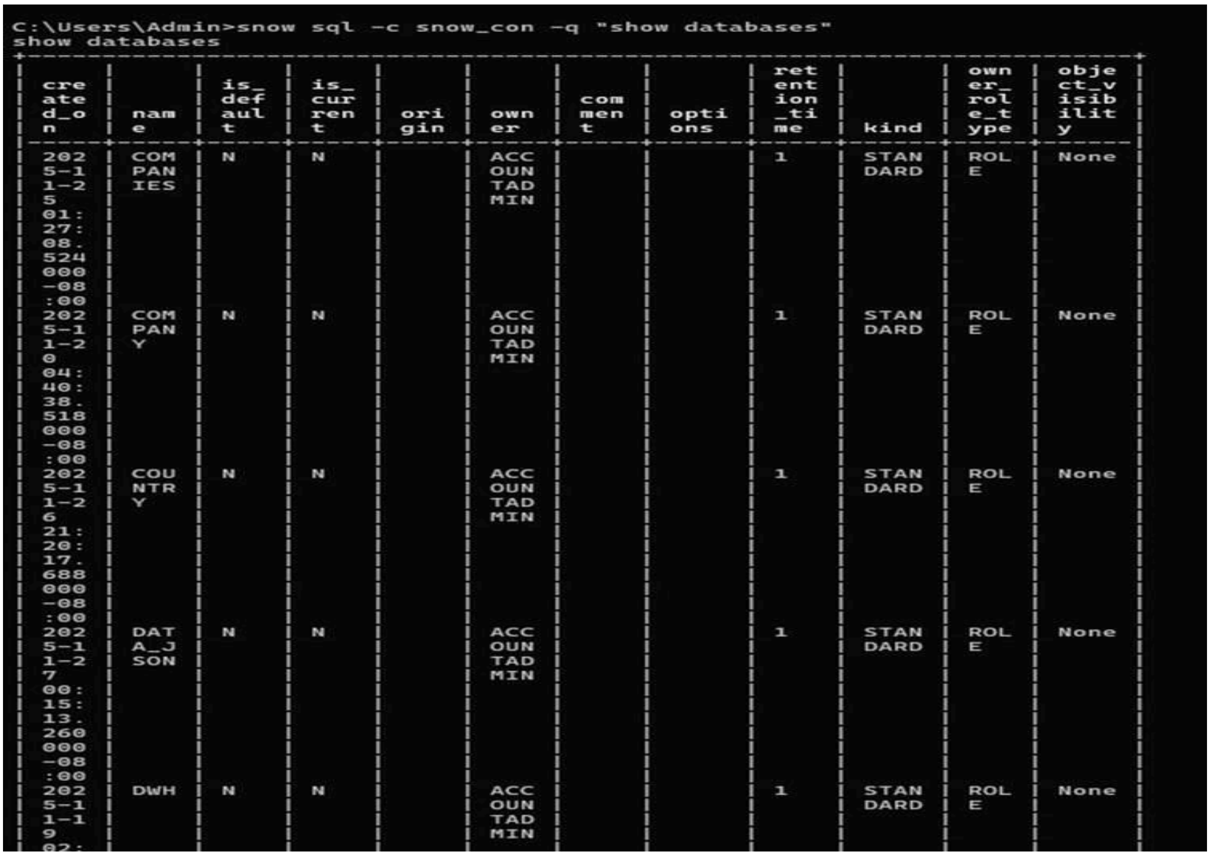
snow connection test

```
C:\Users\Admin>snow connection test --connection snow_con
+-----------------------------------------------------------+
| key             | value                                   |
|-----------------+-----------------------------------------|
| Connection name | snow_con                                |
| Status          | OK                                      |
| Host            | FSBBSTN-IH05083.snowflakecomputing.com  |
| Account         | FSBBSTN-IH05083                         |
| User            | ROSHINI                                 |
| Role            | ACCOUNTADMIN                            |
| Database        | not set                                 |
| Warehouse       | COMPUTE_WH                              |
+-----------------------------------------------------------+
```

2. Running Simple SQL

Run a SQL command to show all databases.

snow sql -q "SHOW DATABASES;"



C:\Users\Admin>snow sql -c snow_con -q "show databases"
show databases

created_on	name	is_default	is_current	origin	owner	comment	options	retention_time	kind	owner_role_type	object_visibility
2025-11-25 01:27:08.524000-08:00	COMPANIES	N	N		ACCOUNTADMIN			1	STANDARD	ROLE	None
2025-11-20 04:40:38.518000-08:00	COMPANY	N	N		ACCOUNTADMIN			1	STANDARD	ROLE	None
2025-11-26 21:20:17.688000-08:00	COUNTRY	N	N		ACCOUNTADMIN			1	STANDARD	ROLE	None
2025-11-27 00:15:13.260000-08:00	DATA_JSON	N	N		ACCOUNTADMIN			1	STANDARD	ROLE	None
2025-11-19 02:	DWH	N	N		ACCOUNTADMIN			1	STANDARD	ROLE	None

Run a SQL file using the Snowflake CLI.
snow sql -f C:\Users\Admin\Downloads\test_script.sql

```
C:\Users\Admin>snow sql -f C:\Users\Admin\Downloads\test_script.sql
SELECT CURRENT_USER();
+-----------------+
| CURRENT_USER() |
|-----------------|
| SHRIMATHI       |
+-----------------+

SELECT CURRENT_ROLE();
+-----------------+
| CURRENT_ROLE() |
|-----------------|
| ACCOUNTADMIN    |
+-----------------+

SHOW DATABASES;
+--------------------------------------------------------------------------------------------------------------------------------------+
|             |            |            |            |        |            |         |         | retention_ |          | owner_role | object_visi |
| created_on  | name       | is_default | is_current | origin | owner      | comment | options | time       | kind     | _type      | bility      |
|-------------+------------+------------+------------+--------+------------+---------+---------+------------+----------+------------+-------------|
| 2025-12-02  | APPLICATIO | N          | N          |        | ACCOUNTADM |         |         | 1          | STANDARD | ROLE       | None        |
| 23:56:49.1  | N          |            |            |        | IN         |         |         |            |          |            |             |
| 12000-08:0  |            |            |            |        |            |         |         |            |          |            |             |
| 0           |            |            |            |        |            |         |         |            |          |            |             |
| 2025-12-02  | ASSESSMENT | N          | N          |        | ACCOUNTADM |         |         | 1          | STANDARD | ROLE       | None        |
| 22:33:33.8  |            |            |            |        | IN         |         |         |            |          |            |             |
| 35000-08:0  |            |            |            |        |            |         |         |            |          |            |             |
| 0           |            |            |            |        |            |         |         |            |          |            |             |
| 2025-11-25  | CRICKET    | N          | N          |        | ACCOUNTADM |         |         | 1          | STANDARD | ROLE       | None        |
| 00:31:22.3  |            |            |            |        | IN         |         |         |            |          |            |             |
| 60000-08:0  |            |            |            |        |            |         |         |            |          |            |             |
| 0           |            |            |            |        |            |         |         |            |          |            |             |
| 2025-11-25  | CRICKET_PL | N          | N          |        | ACCOUNTADM |         |         | 1          | STANDARD | ROLE       | None        |
| 01:25:14.5  | AY         |            |            |        | IN         |         |         |            |          |            |             |
| 11000-08:0  |            |            |            |        |            |         |         |            |          |            |             |
| 0           |            |            |            |        |            |         |         |            |          |            |             |
| 2025-11-19  | DWH        | N          | N          |        | ACCOUNTADM |         |         | 1          | STANDARD | ROLE       | None        |
```

3. Creating Objects

Create a database using the CLI.
snow sql -q "CREATE OR REPLACE DATABASE my_database;"

Create a schema inside that database.
snow sql -q "CREATE OR REPLACE SCHEMA my_database.my_schema;"

Create a table using a SQL file and run it through the CLI.
C:\Users\Admin\Downloads>snow sql -f create_table.sql

```
CREATE OR REPLACE TABLE CLI_TEST_DB.PUBLIC.EMPLOYEE (
        ID INT,
        NAME STRING,
        SALARY NUMBER(10,2)
);
```

```
C:\Users\Admin>snow sql -q "CREATE OR REPLACE DATABASE my_database;"
CREATE OR REPLACE DATABASE my_database;
+------------------------------------------+
| status                                   |
|------------------------------------------|
| Database MY_DATABASE successfully created. |
+------------------------------------------+


C:\Users\Admin>snow sql -q "CREATE OR REPLACE SCHEMA my_database.my_schema;"
CREATE OR REPLACE SCHEMA my_database.my_schema;
+--------------------------------------+
| status                               |
|--------------------------------------|
| Schema MY_SCHEMA successfully created. |
+--------------------------------------+


C:\Users\Admin>cd C:\Users\Admin\Downloads\create_table.sql
The directory name is invalid.

C:\Users\Admin>cd C:\Users\Admin\Downloads

C:\Users\Admin\Downloads>snow sql -f create_table.sql
CREATE OR REPLACE TABLE CLI_TEST_DB.PUBLIC.EMPLOYEE (
    ID INT,
    NAME STRING,
    SALARY NUMBER(10,2)
);
+-------------------------------------+
| status                              |
|-------------------------------------|
| Table EMPLOYEE successfully created. |
+-------------------------------------+
```

4. Working With Stages

Create a table using a SQL file and run it through the CLI.

```
C:\Users\Admin\Downloads>snow sql -f create_table.sql
CREATE OR REPLACE TABLE CLI_TEST_DB.PUBLIC.EMPLOYEE (
        ID INT,
        NAME STRING,
        SALARY NUMBER(10,2)
);
```

List the files inside the stage.

```
snow sql -q "LIST @mystage;"
```

```
C:\Users\Admin\Downloads>snow sql -q "LIST @mystage;"
LIST @mystage;
+-----------------------------------------------------------------------------------------------------+
| name                 | size | md5                              | last_modified                 |
|----------------------+------+----------------------------------+-------------------------------|
| mystage/employee.csv | 208  | 89c5c661bf57ae7b3fd848daa432dd75 | Fri, 5 Dec 2025 09:50:16 GMT |
+-----------------------------------------------------------------------------------------------------+
```

Download the uploaded file back to your computer.

```
snow stage copy @mystage/employee.csv
"C:\Users\Admin\Downloads\employee_downloaded.csv"
```

```
C:\Users\Admin\Downloads>snow stage copy @mystage/employee.csv "C:\Users\Admin\Downloads\employee_downloaded.csv"
Use `--recursive` flag, which copy files recursively with directory structure. This will be the default behavior in the future.
+-----------------------------------------+
| file         | size | status     | message |
|--------------+------+------------+---------|
| employee.csv | 199  | DOWNLOADED |         |
+-----------------------------------------+
```

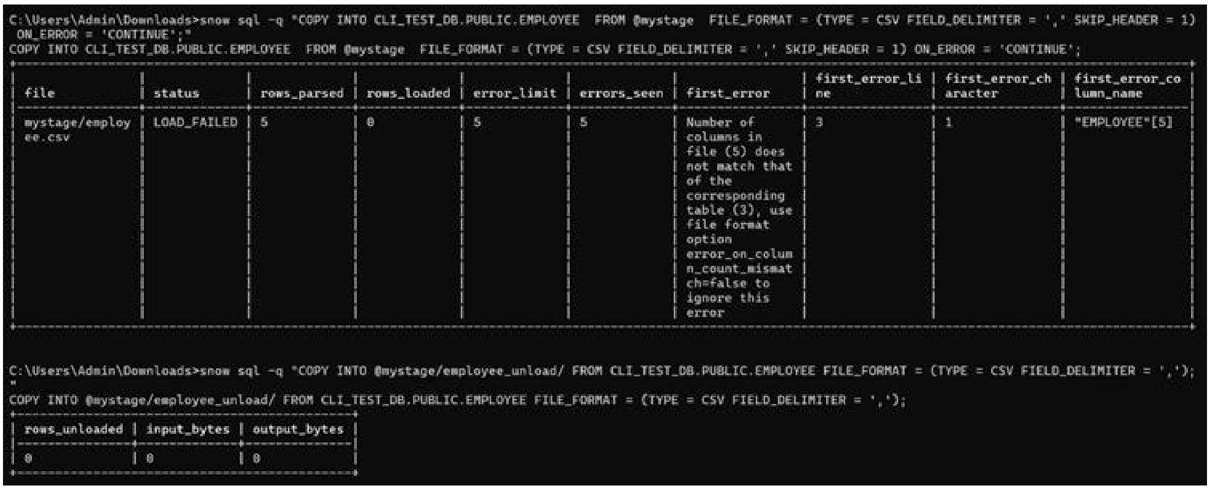
5. Loading and Unloading Data

Load CSV data into a table using COPY INTO via the CLI.

```
snow sql -q "COPY INTO CLI_TEST_DB.PUBLIC.EMPLOYEE  FROM @mystage
FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = ',' SKIP_HEADER = 1) ON_ERROR =
'CONTINUE';"
```

Unload data from a table into a stage.

```
snow sql -q "COPY INTO @mystage/employee_unload/ FROM
CLI_TEST_DB.PUBLIC.EMPLOYEE FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER =
',');
```



```
C:\Users\Admin\Downloads>snow sql -q "COPY INTO CLI_TEST_DB.PUBLIC.EMPLOYEE  FROM @mystage  FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = ',' SKIP_HEADER = 1)
 ON_ERROR = 'CONTINUE';"
COPY INTO CLI_TEST_DB.PUBLIC.EMPLOYEE  FROM @mystage  FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = ',' SKIP_HEADER = 1) ON_ERROR = 'CONTINUE';
```

file	status	rows_parsed	rows_loaded	error_limit	errors_seen	first_error	first_error_line	first_error_character	first_error_column_name
mystage/employee.csv	LOAD_FAILED	5	0	5	5	Number of columns in file (5) does not match that of the corresponding table (3), use file format option error_on_column_count_mismatch=false to ignore this error	3	1	"EMPLOYEE"[5]

```
C:\Users\Admin\Downloads>snow sql -q "COPY INTO @mystage/employee_unload/ FROM CLI_TEST_DB.PUBLIC.EMPLOYEE FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = ',');
"
COPY INTO @mystage/employee_unload/ FROM CLI_TEST_DB.PUBLIC.EMPLOYEE FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = ',');
```

rows_unloaded	input_bytes	output_bytes
0	0	0

6. Deploying a Snowflake Project

Initialize a Snowflake project using snow init.

Deploy the project using snow deploy.

7. Query Output Practice

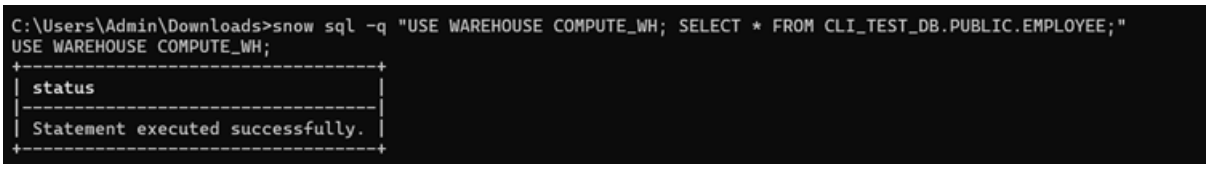
Run a SELECT query and save the results to a CSV file using the CLI.

snow sql -q "SELECT * FROM CLI_TEST_DB.PUBLIC.EMPLOYEE;" > employee_data.csv

8. Playing With Context

Change the warehouse while running a query.

snow sql -q "USE WAREHOUSE COMPUTE_WH; SELECT * FROM CLI_TEST_DB.PUBLIC.EMPLOYEE;"



Change the database and schema using CLI options.

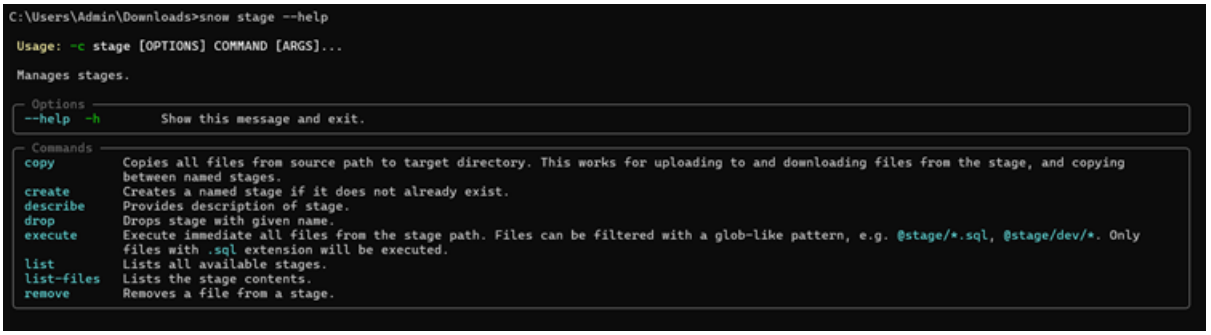
snow sql --dbname CLI_TEST_DB --schemaname PUBLIC -q "SELECT * FROM EMPLOYEE;" SELECT * FROM EMPLOYEE;

snow sql --dbname CLI_TEST_DB --schemaname PUBLIC -q "SELECT CURRENT_DATABASE(), CURRENT_SCHEMA();"
SELECT CURRENT_DATABASE(), CURRENT_SCHEMA();

9. Troubleshooting and Help

Use the help command to list all available CLI commands.

snow stage –help



Run a command in verbose mode to view detailed logs.
set SNOWFLAKE_CLI_LOG_LEVEL=DEBUG
snow sql -q "SELECT CURRENT_TIMESTAMP();"