WEEK 7 - DBT PROJECT WITH SNOWFLAKE

Prepared By: MANSI DESHMUKH

Course: DATA WAREHOUSE

Instructor: Prof. Keeyong Han

GitHub:https://github.com/Mansi3107/DW7_DBT

Objective

The objective of this project was to implement and execute a complete dbt (Data Build Tool) workflow
integrated with the Snowflake data warehouse. The goal was to validate the setup, create models, perform testing,
apply snapshots, and confirm successful execution in Snowflake as part of the Week 10 lab exercises for the
Data Warehouse course.

Tools and Technologies Used

1. dbt (Data Build Tool) - for model creation, testing, and transformation management.
2. Snowflake - for cloud-based data warehousing and storage.
3. SQL - to create and query models within dbt and Snowflake.
4. Visual Studio Code / Command Line Interface - for executing dbt commands and managing configurations.

Implementation and Execution Steps

Step 1: dbt debug – Connection Test Successful

This step verifies the successful connection between dbt and Snowflake. The 'dbt debug' command was executed to check the configuration in the profiles.yml file. The output shows 'All checks passed!', confirming a proper setup and connectivity with Snowflake.

```
07:55:58  Connection:
07:55:58    account: sfedu02-lvb17920
07:55:58    user: LOBSTER
07:55:58    database: USER_DB_LOBSTER
07:55:58    warehouse: LOBSTER_QUERY_WH
07:55:58    role: TRAINING_ROLE
07:55:58    schema: ANALYTICS
07:55:58    authenticator: None
07:55:58    oauth_client_id: None
07:55:58    query_tag: None
07:55:58    client_session_keep_alive: False
07:55:58    host: None
07:55:58    port: None
07:55:58    proxy_host: None
07:55:58    proxy_port: None
07:55:58    protocol: None
07:55:58    connect_retries: 1
07:55:58    connect_timeout: None
07:55:58    retry_on_database_errors: False
07:55:58    retry_all: False
07:55:58    insecure_mode: False
07:55:58    reuse_connections: True
07:55:58    s3_stage_vpce_dns_name: None
07:55:58  Registered adapter: snowflake=1.10.2
07:55:59    Connection test: [OK connection ok]
```

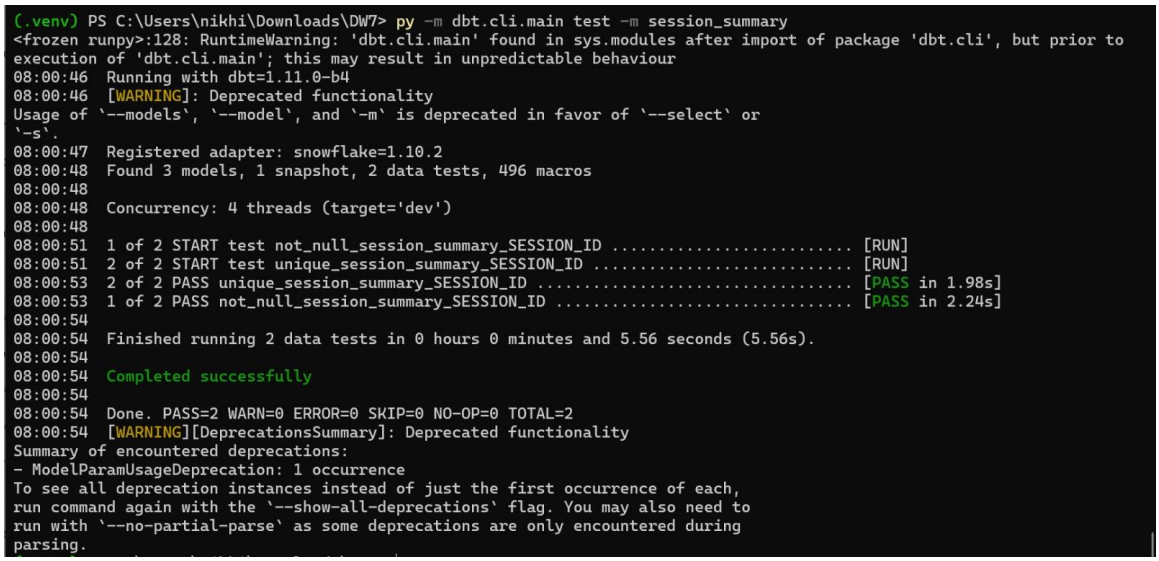
Step 2: dbt run – Model Building

The 'dbt run' command executes all models defined within the dbt project. This step successfully created two input models (views) and one output model (table) in the Snowflake database. The successful build message indicates that data transformation logic has been applied correctly.

```
(.venv) PS C:\Users\nikhi\Downloads\DW7> py -m dbt.cli.main run
<frozen runpy>:128: RuntimeWarning: 'dbt.cli.main' found in sys.modules after import of package 'dbt.cli', but prior to
execution of 'dbt.cli.main'; this may result in unpredictable behaviour
07:59:11  Running with dbt=1.11.0-b4
07:59:12  Registered adapter: snowflake=1.10.2
07:59:12  Unable to do partial parsing because profile has changed
07:59:14  Found 3 models, 1 snapshot, 2 data tests, 496 macros
07:59:14
07:59:14  Concurrency: 4 threads (target='dev')
07:59:14
07:59:15  1 of 3 START sql view model ANALYTICS.session_timestamp ....................... [RUN]
07:59:15  2 of 3 START sql view model ANALYTICS.user_session_channel ..................... [RUN]
07:59:16  1 of 3 OK created sql view model ANALYTICS.session_timestamp .................. [SUCCESS 1 in 0.55s]
07:59:16  2 of 3 OK created sql view model ANALYTICS.user_session_channel ................ [SUCCESS 1 in 0.74s]
07:59:16  3 of 3 START sql table model ANALYTICS.session_summary ......................... [RUN]
07:59:18  3 of 3 OK created sql table model ANALYTICS.session_summary .................... [SUCCESS 1 in 1.82s]
07:59:18
07:59:18  Finished running 1 table model, 2 view models in 0 hours 0 minutes and 4.29 seconds (4.29s).
07:59:18
07:59:18  Completed successfully
07:59:18
07:59:18  Done. PASS=3 WARN=0 ERROR=0 SKIP=0 NO-OP=0 TOTAL=3
(.venv) PS C:\Users\nikhi\Downloads\DW7>
```

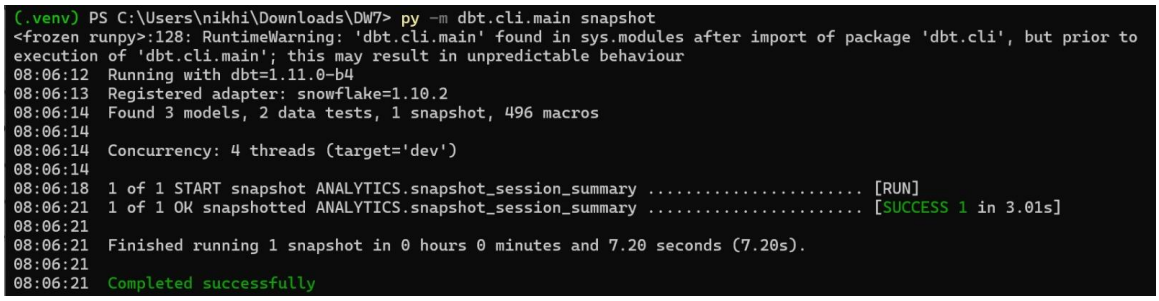
Step 3: dbt test – Schema Testing

This step validates data integrity within the 'session_summary' model using dbt's testing framework. Two tests were executed on the SESSION_ID field—NOT NULL and UNIQUE—and both passed successfully, confirming data accuracy and uniqueness.



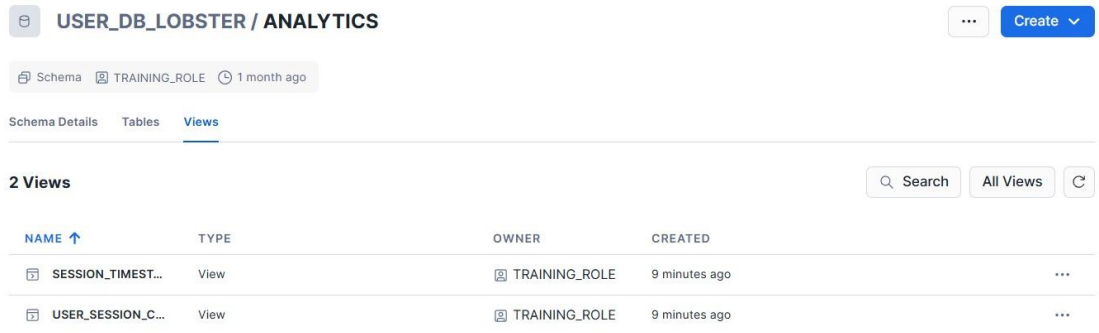
Step 4: dbt snapshot – Snapshot Execution

Snapshots track changes to records over time. The 'dbt snapshot' command was executed to capture the state of the 'session_summary' table. The snapshot completed successfully, ensuring historical data tracking in Snowflake.



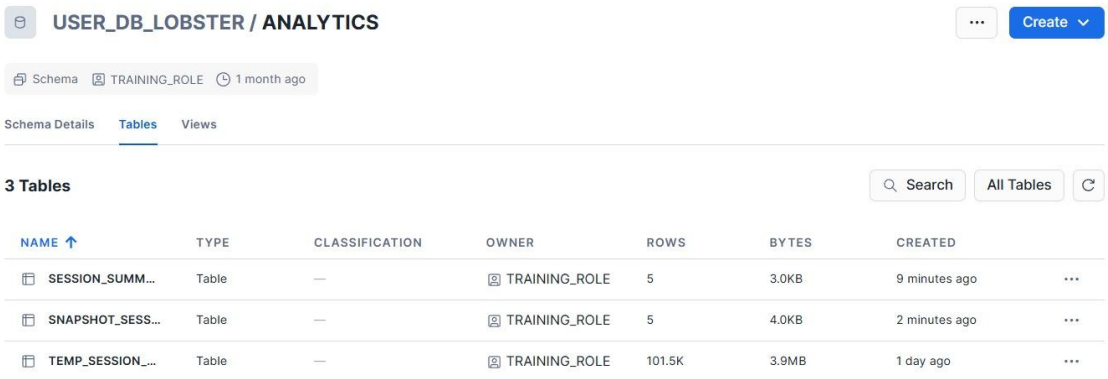
Step 5: Snowflake Verification – Tables

This screenshot displays the Snowflake user interface showing the tables created by dbt. The 'SESSION_SUMMARY' and 'SNAPSHOT_SESSION_SUMMARY' tables confirm that data has been materialized and the snapshot process was successful.



Step 6: Snowflake Verification – Views

This screenshot verifies that the input models—'USER_SESSION_CHANNEL' and 'SESSION_TIMESTAMP'—were created as views in Snowflake. It confirms that the models built by dbt were properly configured as views according to project settings.



Step 7: dbt init – Snowflake Profile Setup

This step illustrates the interactive dbt initialization process ('dbt init'). The setup defines connection parameters such as account, user, role, warehouse, and schema. The generated profiles.yml ensures dbt can connect securely to the Snowflake data warehouse.

```
Which database would you like to use?
[1] snowflake

(Don't see the one you want? https://docs.getdbt.com/docs/available-adapters)

Enter a number: 1
account (https://<this_value>.snowflakecomputing.com): sfedu02-lvb17920
user (dev username): LOBSTER
[1] password
[2] keypair
[3] sso
Desired authentication type option (enter a number): 1
password (dev password):
password (dev password):
role (dev role): TRAINING_ROLE
warehouse (warehouse name): LOBSTER_QUERY_WH
database (default database that dbt will build objects in): USER_DB_LOBSTER
schema (default schema that dbt will build objects in): ANALYTICS
threads (1 or more) [1]: 1
```

Conclusion

The dbt project was successfully integrated with Snowflake, covering all essential stages—configuration,
model creation, testing, and snapshot generation. The outputs verified in Snowflake confirm that the models
and transformations were correctly deployed. This project demonstrates proficiency in using dbt for data
transformation and Snowflake for cloud-based warehousing, fulfilling the learning objectives of the course.