Assignment 2- Team 6

Post-Upload Testing Report

1. Steps Taken to Ensure Successful Data Upload

Raw Storage (As-Is)

- Verified that the Extracted files from the SEC site existed on S3(`sec_extracted_tsv/).`
- Verified the connections for AWS, Snowflake and Airflow were properly configured.
- Used `LIST @sec_txt_stage` to confirm that the extracted files were staged
- Checked if the tables (sec_numbers, sec_presentation, sec_submissions and sec_tags were created before uploading the data.
- Checked the row count of the tables using :
  SELECT COUNT(*) FROM raw_data.sec_numbers;

JSON Transformation

- Verified JSON files exist in S3 (`sec_json_data/`).
- Used `LIST @sec_json_stage` to confirm JSON files were staged.

Checked row count in JSON table:
SELECT COUNT(*) FROM raw_data.sec_financial_json;

Denormalized Fact Tables

- Verified the existence of JSON files used for staging on S3(`sec_json_data/`).
- Verified the default connections for snowflake and aws are configured correctly on airflow
- Verified the variables used to extract the JSON file and AWS Bucket are configured properly on airflow
- Used `LIST @sec_json_stage` to confirm JSON files were staged.
- Checked if the fact tables (income_statement, cash_flow and balance_sheet are created on snowflake before inserting the data.
- Once the tables are created, checked the row count using the query
  SELECT COUNT(*) FROM raw_data.balance_sheet;

2. Verification of Data Integrity in Snowflake

To ensure the integrity of uploaded data, the following checks were performed:

Raw Data Integrity Checks

– Ensured that the Raw Data tables are populated by running the query:

```
SELECT COUNT(*) FROM raw_data.sec_numbers;
SELECT COUNT(*) FROM raw_data.sec_presentation;
SELECT COUNT(*) FROM raw_data.sec_submissions;
SELECT COUNT(*) FROM raw_data.sec_tags;
```

– Ensured that the DBT Validations passed.

JSON Data Integrity Checks

```
-- Ensure JSON table is populated
SELECT COUNT(*) FROM raw_data.sec_financial_json;
```

Fact Table Integrity Checks

-- Ensured the fact_tables are populated. Verified the count using queries

```
SELECT COUNT(*) FROM raw_data.balance_sheet
SELECT COUNT(*) FROM raw_data.income_statement
SELECT COUNT(*) FROM raw_data.cash_flow
```

-- Ensured all the dbt tests are passed. Checked the logs on Airflow to see if all the given tests are passed

– Checked if the Primary Key Constraint holds up for all the tables,by using the queries

```
SELECT company_name,fiscal_year, fiscal_period,COUNT(*) from
raw_data.balance_sheet group by
 (company_name,fiscal_year, fiscal_period) having COUNT(*)>1;
```

3. Methods Used to Confirm Pipeline Execution

Airflow Execution Validation

- Verified Airflow UI logs **show successful DAG runs**.
- Checked that no tasks failed in **JSON S3 to Snowflake DAG**.
- Checked that no tasks failed in **Create_fact_tables_to_snowflake DAG**
- Confirmed execution timestamps in **Airflow Logs**.

Snowflake Load History Validation

Checked `information_schema.load_history` for successful loads:
SELECT COUNT(*) FROM information_schema.load_history WHERE table_name = 'SEC_FINANCIAL_JSON';

4. Running Tests for the Pipeline

DBT Tests (Automated Validations)

Test Type	Model/Table	Expected Outcome
`integer`	`stg_data_json.fiscal_year`	Year is always an integer
`accepted_values`	`stg_data_json.fiscal_period`	Values only `Q1, Q2, Q3, Q4`
`not null`	`stg_sec_num.sec_numbers`	reported_value not null
`date`	`stg_sec_num.sec_numbers`	`date` is in correct format

The following factors confirm that the **data upload and transformation process was successful**:

1. **Airflow logs confirm the successful execution** of all DAGs without failures.
2. **Snowflake query results confirm expected row counts**, ensuring no data loss.
3. **DBT tests validate the correctness of transformed JSON data** before insertion into fact tables.
4. **The final dataset in Snowflake is fully queryable**, meaning data integrity is maintained.