DatafestAfrica Datathon 2023 - Data Engineering Project

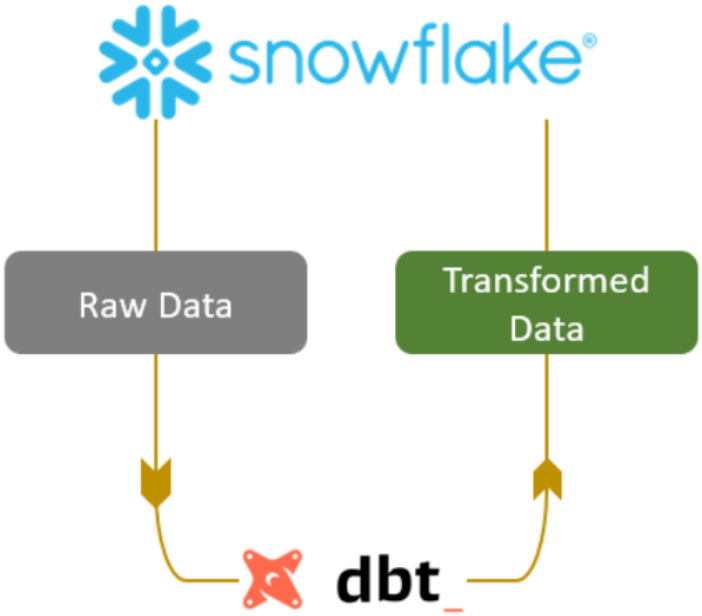
TEAM: BLACKBOX

GitHub Rep: link

https://github.com/uchiharon/DataFestAfrica-BlackBox/tree/main/DataEngineering

Snowflake worksheets can be found when you login.

Username: Blackbox
Password: V8pTqN2fG



NOTE:

- **All code can be found in the GitHub Repo: https://github.com/uchiharon/DataFestAfrica-BlackBox/tree/main/DataEngineering**
- **Data Dictionary for the final table is at the end of the report.**

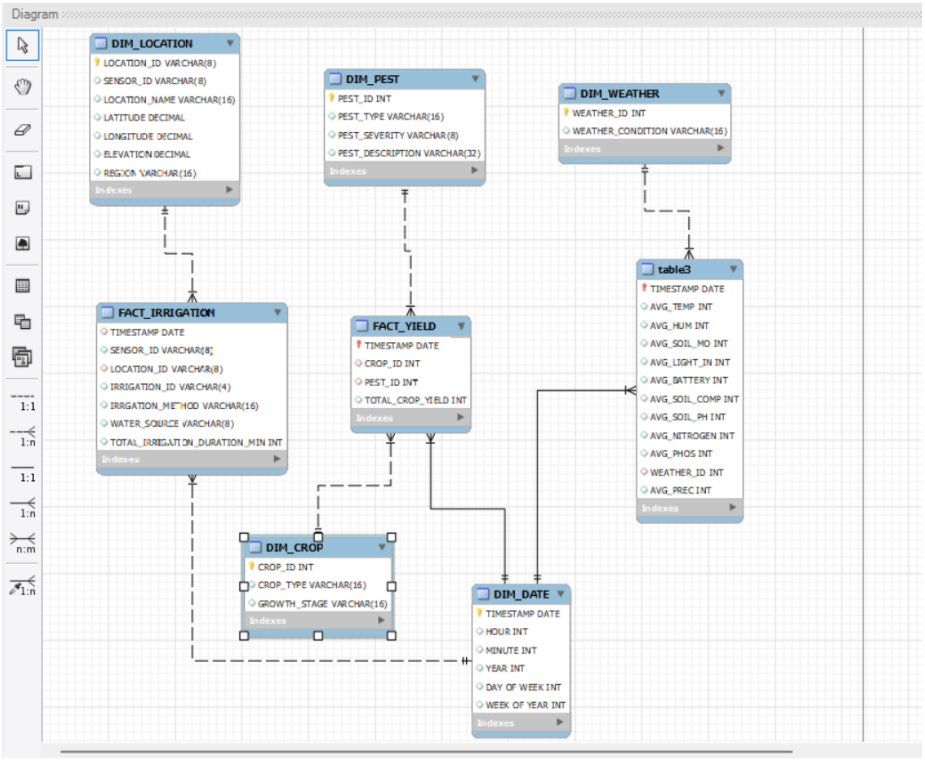
Documentation and Final Testing Report

Objective:

The objective of this report is to document the data modeling process, schema structure, table relationships, and data dictionary. Additionally, it highlights the final testing phase aimed at ensuring the accuracy and consistency of the data. This report also covers the preparation of the data warehouse for querying by data analysts and scientists.

Data Model Diagram:

This is the model diagram of the final table.



Project Overview:

Tools Used: Snowflake and DBT were employed for this project, providing a robust framework for data modeling and transformation.

Data Modeling Approach: The data modeling process commenced with the creation of staging tables. Initial data cleaning was conducted based on observations made during data exploration in Snowflake Workspace.

Data Cleaning:

Staging Tables:

Staging tables were designed to facilitate the initial cleaning and transformation of raw data. These tables served as an intermediary step in the data modeling process.

Initial Data Cleaning:

The cleaning process encompassed several steps:

- Removal of null and 'NA' values.
- Elimination of string whitespace.
- Correction of misspellings.
- Split column with two entries like the Irrigration and Location table Sensor ID

Index Key Creation:

To enhance the efficiency of dimension tables, index keys were introduced. These keys provided a structured and unique identifier for each record.

Intermediate Staging and Table Creation:

Table Joins:

The intermediate stage involved the joining of various tables required to generate the final tables for consumers. This process ensured that data from disparate sources was integrated effectively.

Main Table Creation:

The project culminated in the creation of **five-dimensional tables and three fact tables**. These tables play a pivotal role in the data model, providing essential context and metrics for analysis.

Order of Dimension Tables:

The order of dimension tables was meticulously maintained to follow a logical modeling sequence, facilitating a more intuitive understanding of the data.

Data Quality Assurance (Testing):

This was done in DBT for the core model, which was materialized as table (link to schema with test).

All tests used were DBT inbuilt test. For columns intended to be used as primary key, they were tested for nulls and uniqueness, and expected to break the pipeline if the test is not passed.

The other columns test was based on intuition. For column I felt were subjected to change, I made the severity warning, but for the others, error (the pipeline should break). The major testing, I carried out was:

- Null
- Unique
- Accepted Values

Which are the minimum tests required for the model to run successfully.

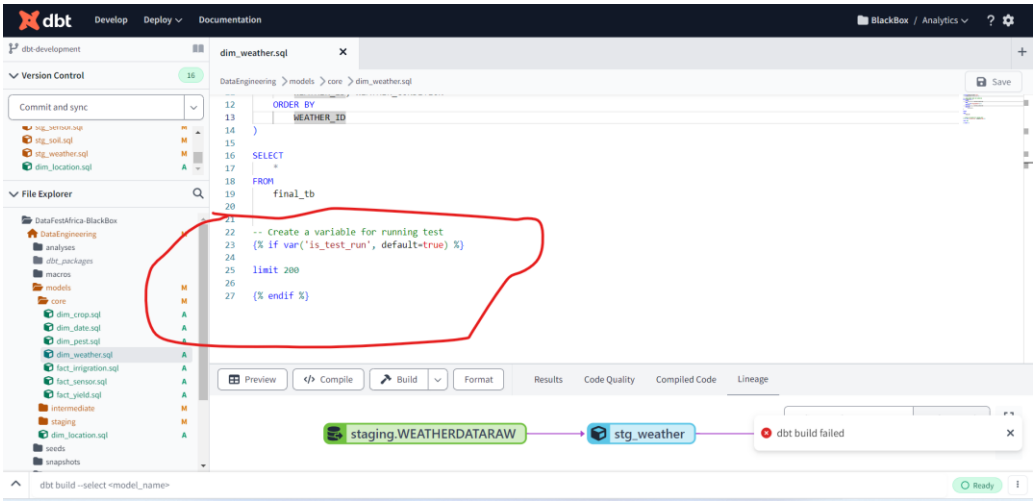
Optimization:

View Materialization:

To optimize query performance, view materialization was employed for staging and intermediate models. This strategy minimized computational overhead and expedited data retrieval.

Testing with Limited Rows:

A test variable was introduced to restrict the dataset to 200 rows during development. This controlled environment was proved instrumental in assessing data quality and performance.



Snowflake Caching:

Snowflake's caching feature was harnessed to streamline the data exploration process, enhancing the efficiency of both back-end and front-end operations.

Data Integrity and Quality:

DBT Tests:

To uphold data integrity and quality, a series of tests were implemented using DBT. These tests rigorously evaluated critical data components, ensuring that the data met predefined standards.

Monitoring and Optimization:

Query Performance Monitoring:

Regular monitoring of query performance was executed through Snowflake's query history and performance views. These insights were leveraged to fine-tune and optimize slower queries.

Best Practices:

Several best practices were adhered to in the optimization process:

Limiting SELECT: Specifying only the necessary columns was favored overusing SELECT * to minimize data transfer and query execution time.

Avoiding SELECT DISTINCT: The use of SELECT DISTINCT was discouraged, especially with large datasets, to improve query efficiency.

Using WHERE Clauses Wisely: Efficient use of WHERE clauses, coupled with indexed columns, was a key strategy to reduce data retrieval time.

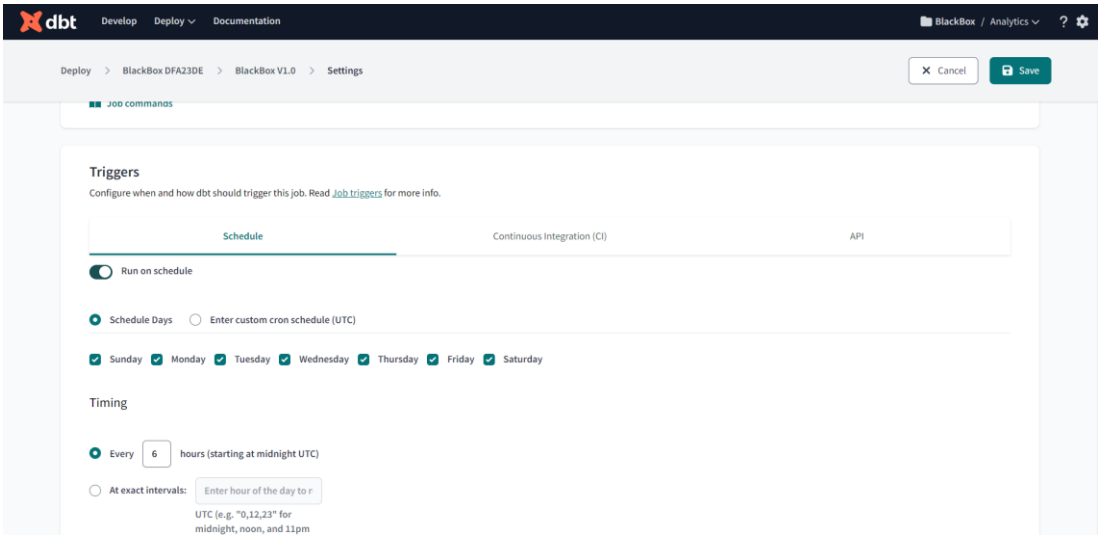
Optimizing JOINs: INNER JOINs were preferred over OUTER JOINs, and careful consideration was given to joining large tables to prevent performance bottlenecks.

Early Data Aggregation: The application of aggregate functions (e.g., SUM, AVG, COUNT) was encouraged for data reduction, ideally performed at the source to enhance performance.

Deployment:

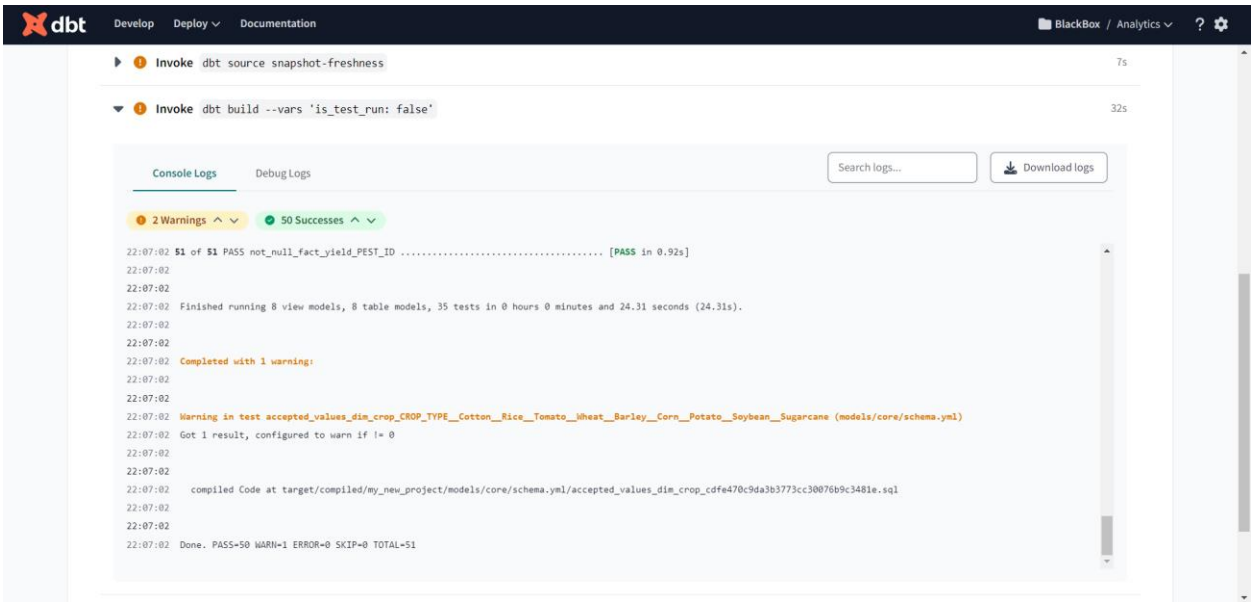
After successfully architecting and implementing a comprehensive data pipeline using dbt, I proceeded with its deployment. With a keen understanding of our stakeholders' needs, I determined that they required access to up-to-date reports at four distinct intervals throughout the day.

To meet this demand, I meticulously configured our scheduler to execute data processing and report generation tasks, commencing each cycle promptly at midnight (6 hours interval). This thoughtful scheduling approach ensures that our stakeholders receive timely and accurate insights to support their decision-making processes.



Note: This can always be changed to meet future business requirements.

Final Run:



Challenges and Opportunities for Enhancement:

Optimizing Data Refresh Strategy:

Consider enhancing the pipeline's efficiency through the implementation of incremental data refresh techniques. While the current approach serves as an initial solution, transitioning to incremental updates can lead to faster processing times and resource savings.

Data Quality Investigation:

A comprehensive investigation into certain tables has revealed potential data inconsistencies. A future initiative should prioritize data quality assurance to rectify missing or inaccurate values, thereby ensuring the integrity of the dataset.

Enhanced Data Governance:

Reevaluate the current Snowflake access configuration, aiming to separate the test and deployment tables into distinct schemas. This segregation strengthens data governance practices and streamlines access control, contributing to better data management.

Refining Assumptions Through Stakeholder Engagement:

Acknowledge that some assumptions were made during the project, such as the homogeneity of all locations within the farm. To mitigate potential inaccuracies, engage in further discussions with stakeholders to align assumptions with their specific needs and expectations.

Leveraging Machine Learning for Data Imputation:

Explore the possibility of employing machine learning models for predictive data imputation rather than resorting to the removal of missing values. This approach not only preserves valuable data but also offers a more sophisticated method of handling missing information.

Conclusion:

In conclusion, this report has detailed the comprehensive data modeling process, from initial cleaning and staging to the creation of dimension and fact tables. Optimization techniques, testing, and monitoring strategies have been described to ensure data accuracy and efficiency. The project has resulted in a well-structured data model and warehouse, prepared for the valuable insights and analyses of data analysts and scientists.

DATA DICTIONARY

Table: DIM_CROP

- Description: Contains information about different crop types and their growth stages.
- Columns:
    - **CROP_ID**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: Unique identifier for a crop.
        - Example Values: ["1", "2", "3"]
    - **CROP_TYPE**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: Type of crop.
        - Example Values: ["Cotton", "Rice", "Tomato"]
    - **GROWTH_STAGE**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: The current growth stage of the crop.
        - Example Values: ["Seedling", "Vegetative", "Flowering"]

Table: DIM_DATE

- Description: Contains date-related information.
- Columns:
    - **TIMESTAMP**
        - Data Type: TIMESTAMP_NTZ(9)
        - Unit of Measurement: None
        - Description: Date and time of the timestamp.
        - Example Values: ["2023-09-18 15:30:00", "2023-09-19 08:45:00", "2023-09-20 12:15:00"]
    - **HOUR**
        - Data Type: NUMBER(2,0)
        - Unit of Measurement: Hours
        - Description: Hour of the day.
        - Example Values: [15, 8, 12]
    - **MINUTE**
        - Data Type: NUMBER(2,0)
        - Unit of Measurement: Minutes
        - Description: Minute of the hour.
        - Example Values: [30, 45, 15]
    - **MONTH**
        - Data Type: NUMBER(2,0)

        - Unit of Measurement: Months
        - Description: Month of the year.
        - Example Values: [9, 10, 11]
    - **YEAR**
        - Data Type: NUMBER(4,0)
        - Unit of Measurement: Years
        - Description: Year.
        - Example Values: [2023, 2024, 2025]
    - **DAY_OF_WEEK**
        - Data Type: NUMBER(2,0)
        - Unit of Measurement: Days
        - Description: Day of the week.
        - Example Values: [1, 4, 7]
    - **WEEK_OF_YEAR**
        - Data Type: NUMBER(2,0)
        - Unit of Measurement: Weeks
        - Description: Week of the year.
        - Example Values: [38, 39, 40]

Table: DIM_PEST

- Description: Contains information about different pest types and their severity.
- Columns:
    - **PEST_ID**
        - Data Type: NUMBER(38,0)
        - Unit of Measurement: None
        - Description: Unique identifier for a pest.
        - Example Values: [101, 102, 103]
    - **PEST_TYPE**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: Type of pest.
        - Example Values: ["Aphids", "Caterpillars", "Whiteflies"]
    - **PEST_SEVERITY**
        - Data Type: VARCHAR(8)
        - Unit of Measurement: None
        - Description: Severity level of the pest infestation.
        - Example Values: ["Moderate", "High", "Low"]
    - **PEST_DESCRIPTION**
        - Data Type: VARCHAR(32)
        - Unit of Measurement: None
        - Description: Description of the pest.
        - Example Values: ["Small insects damaging leaves", "Green caterpillars on stems", "Whiteflies on undersides of leaves"]

Table: DIM_WEATHER

- Description: Contains information about weather conditions.
- Columns:
    - **WEATHER_ID**
        - Data Type: NUMBER(38,0)
        - Unit of Measurement: None
        - Description: Unique identifier for a weather condition.
        - Example Values: [501, 502, 503]
    - **WEATHER_CONDITION**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: Description of the weather condition.
        - Example Values: ["Sunny", "Partly Cloudy", "Rainy"]

Table: FACT_IRRIGATION

- Description: Records irrigation events, including method, water source, and duration.
- Columns:
    - **TIMESTAMP**
        - Data Type: TIMESTAMP_NTZ(9)
        - Unit of Measurement: None
        - Description: Date and time of the irrigation event.
        - Example Values: ["2023-09-18 08:45:00", "2023-09-19 14:30:00", "2023-09-20 10:15:00"]
    - **SENSOR_ID**
        - Data Type: VARCHAR(8)
        - Unit of Measurement: None
        - Description: Unique identifier for the sensor recording the irrigation event.
        - Example Values: ["S001", "S002", "S003"]
    - **LOCATION_ID**
        - Data Type: VARCHAR(8)
        - Unit of Measurement: None
        - Description: Unique identifier for the location of the irrigation event.
        - Example Values: ["L001", "L002", "L003"]
    - **IRRIGATION_ID**
        - Data Type: VARCHAR(4)
        - Unit of Measurement: None
        - Description: Unique identifier for an irrigation event.
        - Example Values: ["I001", "I002", "I003"]
    - **IRRIGATION_METHOD**
        - Data Type: VARCHAR(16)
        - Unit of Measurement: None
        - Description: Method used for irrigation.
        - Example Values: ["Drip", "Sprinkler", "Flood"]

- **WATER_SOURCE**
  - Data Type: VARCHAR(8)
  - Unit of Measurement: None
  - Description: Source of water for irrigation.
  - Example Values: ["Well", "River", "Reservoir"]
- **TOTAL_IRRIGATION_DURATION_MIN**
  - Data Type: NUMBER(17,0)
  - Unit of Measurement: Minutes
  - Description: Total duration of irrigation in minutes.
  - Example Values: [30, 45, 60]

Table: FACT_SENSOR

- Description: Records sensor data, including various environmental factors.
- Columns:
  - **TIMESTAMP**
    - Data Type: TIMESTAMP_NTZ(9)
    - Unit of Measurement: None
    - Description: Date and time of sensor data observation.
    - Example Values: ["2023-09-18 12:00:00", "2023-09-19 09:30:00", "2023-09-20 15:45:00"]
  - **WEATHER_ID**
    - Data Type: NUMBER(38,0)
    - Unit of Measurement: None
    - Description: Unique identifier for the weather condition during the sensor reading.
    - Example Values: [1, 2, 3]
  - **AVG_TEMPERATURE**
    - Data Type: NUMBER(10,2)
    - Unit of Measurement: Celsius
    - Description: Average temperature reading.
    - Example Values: [25.5, 28.3, 23.8]
  - **AVG_HUMIDITY**
    - Data Type: NUMBER(10,2)
    - Unit of Measurement: Percentage (%)
    - Description: Average humidity reading.
    - Example Values: [60.2, 55.8, 62.4]
  - **AVG_SOIL_MOISTURE**
    - Data Type: NUMBER(10,2)
    - Unit of Measurement: Percentage (%)
    - Description: Average soil moisture reading.
    - Example Values: [32.1, 28.7, 35.2]
  - **AVG_LIGHT_INTENSITY**
    - Data Type: NUMBER(10,2)

        - Unit of Measurement: Lux
        - Description: Average light intensity reading.
        - Example Values: [1200, 1100, 1300]
    - **AVG_BATTERY_LEVEL**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: Percentage (%)
        - Description: Average battery level of the sensor.
        - Example Values: [87.5, 92.1, 85.0]
    - **AVG_SOIL_COMP**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: Percentage (%)
        - Description: Average soil composition.
        - Example Values: [15.3, 17.8, 14.5]
    - **AVG_SOIL_PH**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: pH
        - Description: Average soil pH level.
        - Example Values: [6.5, 6.8, 6.2]
    - **AVG_NITROGEN_LEVEL**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: Percentage (%)
        - Description: Average nitrogen level in the soil.
        - Example Values: [18.7, 17.2, 19.5]
    - **AVG_PHOSPHORUS_LEVEL**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: Percentage (%)
        - Description: Average phosphorus level in the soil.
        - Example Values: [7.2, 7.5, 6.8]
    - **AVG_ORGANIC_MATTER**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: Percentage (%)
        - Description: Average organic matter content in the soil.
        - Example Values: [3.8, 4.1, 3.5]
    - **AVG_WIND_SPEED**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: meters/second
        - Description: Average wind speed reading.
        - Example Values: [8.2, 7.5, 9.0]
    - **AVG_PRECIPITATION**
        - Data Type: NUMBER(10,2)
        - Unit of Measurement: millimeters
        - Description: Average precipitation reading.
        - Example Values: [10.1, 23.0, 0.5]

Table: FACT_YIELD

- Description: Records crop yield, including crop type, pest infestation, and total yield.
- Columns:
    - **TIMESTAMP**
        - Data Type: TIMESTAMP_NTZ(9)
        - Unit of Measurement: None
        - Description: Date and time of the yield measurement.
        - Example Values: ["2023-09-18 18:00:00", "2023-09-19 17:30:00", "2023-09-20 19:45:00"]
    - **CROP_ID**
        - Data Type: NUMBER(38,0)
        - Unit of Measurement: None
        - Description: Unique identifier for a crop.
        - Example Values: [201, 202, 203]
    - **PEST_ID**
        - Data Type: NUMBER(38,0)
        - Unit of Measurement: None
        - Description: Unique identifier for a pest infestation (if applicable).
        - Example Values: [1,2,3]
    - **TOTAL_CROP_YIELD**
        - Data Type: NUMBER(22,2)
        - Unit of Measurement: None
        - Description: Total yield of the crop at the given timestamp.
        - Example Values: [1250.75, 1300.50, 1100.25]

Table: DIM_LOCATION

- Description: Contains information about sensor locations, including their coordinates and region.
- Columns:
- **SENSOR_ID**
    - Data Type: VARCHAR(8)
    - Unit of Measurement: None
    - Description: Unique identifier for a sensor.
    - Example Values: ["S001", "S002", "S003"]
- **LOCATION_ID**
    - Data Type: VARCHAR(8)
    - Unit of Measurement: None
    - Description: Unique identifier for a location.
    - Example Values: ["LXDFE", "HYTED", "MDFIE"]
- **LOCATION_NAME**
    - Data Type: VARCHAR(16)
    - Unit of Measurement: None
    - Description: Name of the location.
    - Example Values: ["Field A", "Field B", "Field C"]

- **LATITUDE**
  - Data Type: NUMBER(15,9)
  - Unit of Measurement: Degrees
  - Description: Latitude coordinate of the location.
  - Example Values: [34.123456789, 35.234567890, 33.987654321]
- **LONGITUDE**
  - Data Type: NUMBER(15,9)
  - Unit of Measurement: Degrees
  - Description: Longitude coordinate of the location.
  - Example Values: [-118.987654321, -119.876543210, -120.123456789]
- **ELEVATION**
  - Data Type: NUMBER(8,2)
  - Unit of Measurement: Meters
  - Description: Elevation of the location.
  - Example Values: [245.67, 312.45, 178.90]
- **REGION**
  - Data Type: VARCHAR(16)
  - Unit of Measurement: None
  - Description: Region where the location is situated.
  - Example Values: ["Central", "Eastern", "Western"]