Name : Sonika Shrinivas Kulkarni
Subject : Data Warehousing and Mining
Roll no : 34 **Batch** : B **Class** : TY 3

Experiment No : 1

Title :

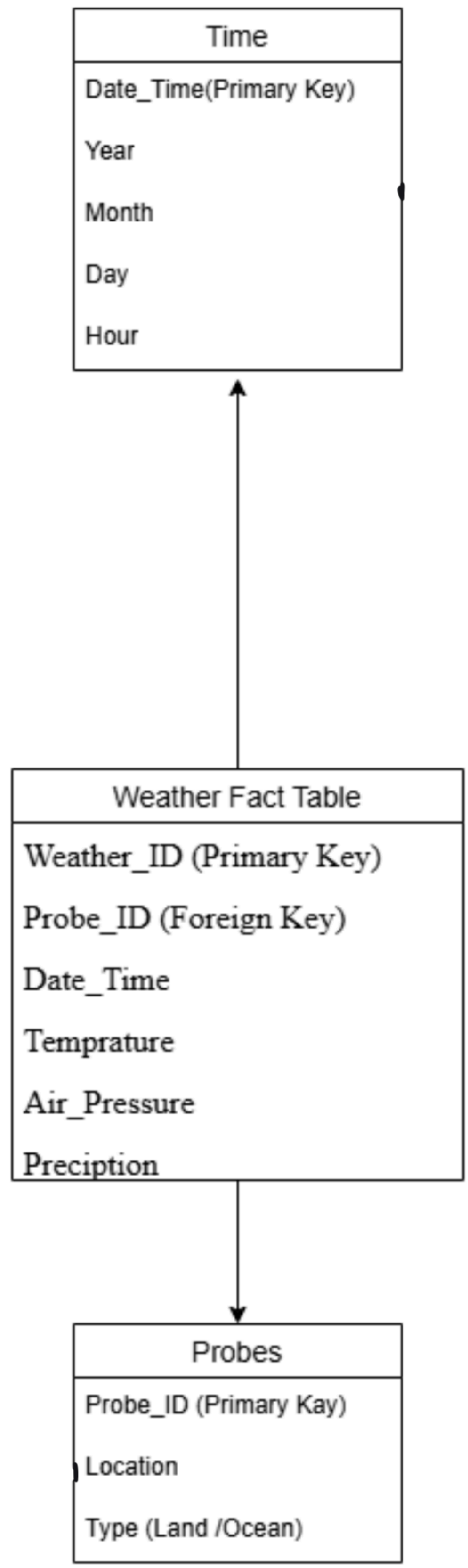
Implementation of all dimension tables and fact tables.

Problem Statement :

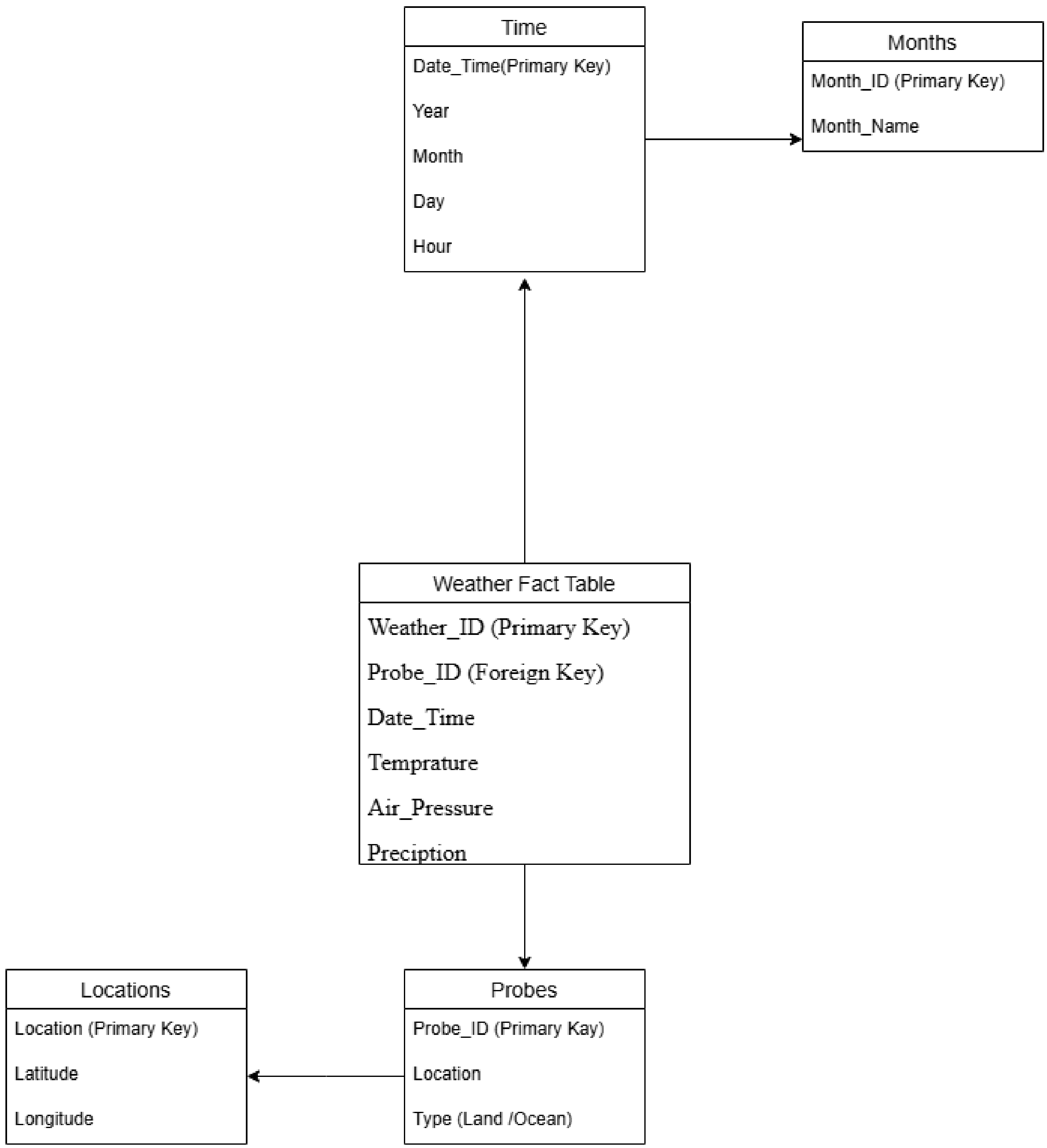
Design a data warehouse for a regional weather bureau. The weather bureau has about 100 probes, which are scattered throughout various land and ocean locations in the region to collect basic weather data, including air pressure, temperature and precipitation at each hour. All data is sent to the central station, which has collected such data for more than 10 years. Design Star schema and Snowflake schema such that it should facilitate efficient querying and online analytical processing and derive general weather patterns in multidimensional space.
Explain all aspects of the diagram. Design Star and Snowflake schema for above case.

Diagram:

Star Schema:


Time
Date_Time(Primary Key)
Year
Month
Day
Hour
Weather Fact Table
Weather_ID (Primary Key)
Probe_ID (Foreign Key)
Date_Time
Temprature
Air_Pressure
Preciption
Probes
Probe_ID (Primary Kay)
Location
Type (Land /Ocean)

SnowFlake Schema:



Preview Questions :

1.In a star schema, how is the fact table typically related to the dimension tables?

In a star schema, the fact table is connected to multiple dimension tables through foreign key relationships. These relationships establish a many-to-one connection between the fact table and each dimension table.

How the Relationship Works:

1. Fact Table as the Core:
   - The fact table stores quantitative data (metrics or measures) related to business operations, such as sales, revenue, or transactions.
   - It contains foreign keys that reference the primary keys of the dimension tables.
2. Dimension Tables Provide Context:
   - Each dimension table contains descriptive attributes that provide context for the numerical data in the fact table.
   - These attributes include details like time periods, product names, customer information, or geographical locations.
3. One-to-Many Relationship:
   - Each row in a dimension table can be associated with multiple rows in the fact table.
   - However, each row in the fact table typically refers to only one row in each related dimension table.
4. Denormalized Structure:
   - Unlike a snowflake schema, the star schema keeps dimension tables in a denormalized format, meaning data redundancy is allowed to improve query performance.
   - This structure simplifies queries and enhances retrieval speed in analytical processing.

Key Advantages of This Relationship:

- Faster Query Performance: Since joins are simple and involve fewer tables.
- Easy to Understand: The structure is straightforward, making it easier for analysts to interpret.
- Efficient for OLAP (Online Analytical Processing): Optimized for aggregation and reporting.

2. What is the main difference between a star schema and a snowflake schema?

The primary difference between a star schema and a snowflake schema lies in the level of normalization applied to the dimension tables.

In a star schema, dimension tables are denormalized, meaning they contain redundant data and are not split into smaller related tables. Each dimension table connects directly to the fact table, resulting in a simple structure that is optimized for query performance. This schema is preferred for its simplicity and faster retrieval times, making it well-suited for analytical processing. However, it requires more storage space due to data redundancy and may be less efficient when updates or modifications are needed.

In contrast, a snowflake schema follows a normalized structure, where dimension tables are further broken down into sub-dimensions to eliminate redundancy. This results in a more complex schema with multiple joins required for queries. The snowflake schema is beneficial for reducing storage requirements and improving data integrity, as updates and modifications are easier to manage. However, due to the increased number of joins, query performance may be slightly slower compared to a star schema.

In summary, a star schema prioritizes simplicity and speed, while a snowflake schema focuses on storage efficiency and data integrity. The choice between the two depends on specific use cases, with star schemas being more common in business intelligence and data warehousing applications where performance is critical.

Conclusion :

A data warehouse for a regional weather bureau is designed to efficiently store and analyze weather data collected from 100 probes across various locations over a period of more than 10 years. The star schema consists of a central fact table storing weather measurements such as air pressure, temperature, and precipitation, linked to denormalized dimension tables for time, location, and probe details, ensuring faster query performance. In contrast, the snowflake schema normalizes dimension tables by breaking them into sub-dimensions, reducing data redundancy while increasing query complexity. The star schema is preferred for OLAP and analytical queries due to its simplicity and speed, while the snowflake schema is beneficial for data integrity and storage efficiency. The choice between them depends on the need for either performance (star schema) or normalized structure (snowflake schema) to facilitate weather pattern analysis.