

**Experiment No.:** 01

**Aim:** To perform a multidimensional data model using SQL queries. e.g., snowflake, star and fact constellation schema.

**Name:** Kiran K Patil

**Enrolment No.:** 211070904

**Branch:** Computer Engineering

**Batch:** D

**Theory :**

Multidimensional Data Modeling Multidimensional Schema is especially designed to model data warehouse systems. The schemas are designed to address the unique needs of very large databases designed for the analytical purpose (OLAP).

**Schema**

* Schema gives the logical description of the entire database.
* It gives the details about the constraints placed on the table , how the key values are linked between the different tables.

**Types of Data Warehouse Schema**

1. Star schema
2. Snowflake schema
3. Fact Constellation Schema

**1. Star schema**

* Star schema is a mature modelling approach widely adopted by relational data warehouses. It requires modelers to classify their model tables as either dimension or fact.
* Dimension tables describe business entities—the things you model. Entities can include products, people, places, and concepts including time itself. The most consistent table you'll find in a star schema is a date dimension table. A dimension table contains a key column (or columns) that acts as a unique identifier, and descriptive columns.
* Fact tables store observations or events, and can be sales orders, stock balances, exchange rates, temperatures, etc. A fact table contains dimension key columns that relate to dimension tables, and numeric measure columns. The dimension key columns determine the dimensionality of a fact table, while the dimension key values determine the granularity of a fact table.
* For example, consider a fact table designed to store sale targets that has two- dimension key columns Date and Product Key. It's easy to understand that the table has two dimensions. The granularity, however, can't be determined without considering the dimension key values. In this example, consider that the values stored in the Date column are the first day of each month. In this case, the granularity is at month-product level.
* Generally, dimension tables contain a relatively small number of rows. Fact tables, on the other hand, can contain a very large number of rows and continue to grow over time.
* The star schema is intensely suitable for data warehouse database design because of the following features:
* It creates a DE-normalized database that can quickly provide query responses.
* It provides a flexible design that can be changed easily or added to throughout the development cycle, and as the database grows.
* It provides a parallel in design to how end-users typically think of and use the data.
* It reduces the complexity of metadata for both developers and end-users.

**Advantages of Star Schema**

1. **Query Performance**: A star schema database has a limited number of table and clear join paths, the query run faster than they do against OLTP systems. Small single-table queries, frequently of a dimension table, are almost instantaneous. Large join queries that contain multiple tables takes only seconds or minutes to run. In a star schema database design, the dimension is connected only through the central fact table. When the two-dimension table is used in a query, only one join path, intersecting the fact tables, exist between those two tables. This design feature enforces authentic and consistent query results.
2. **Load performance and administration:** Structural simplicity also decreases the time required to load large batches of record into a star schema database. By describing facts and dimensions and separating them into the various table, the impact of a load structure is reduced. Dimension table can be populated once and occasionally refreshed. We can add new facts regularly and selectively by appending records to a fact table.
3. **Built-in referential integrity:** A star schema has referential integrity built- in when information is loaded. Referential integrity is enforced because each data in dimensional tables has a unique primary key, and all keys in the fact table are legitimate foreign keys drawn from the dimension table. A record in the fact table which is not related correctly to a dimension cannot be given the correct key value to be retrieved.
4. **Easily Understood**: A star schema is simple to understand and navigate, with dimensions joined only through the fact table. These joins are more significant to the end-user because they represent the fundamental relationship between parts of the underlying business. Customer can also browse dimension table attributes before constructing a query.

**Disadvantage of Star Schema**

1. There is some condition which cannot be meet by star schemas like the relationship between the user, and bank account cannot describe as star schema as the relationship between them is many to many.

**Implementing star schema:**

Doctor Assigned

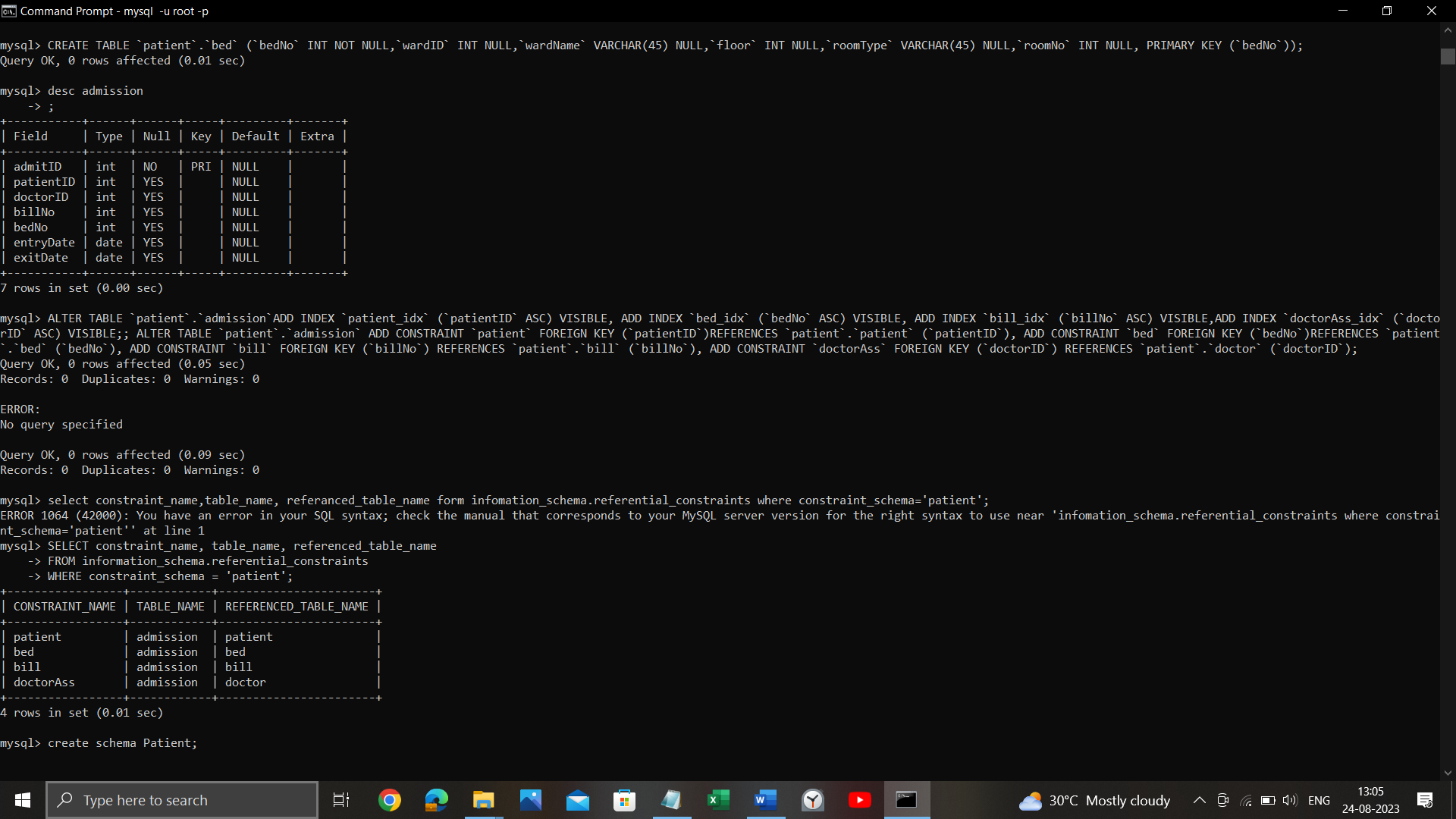
Patient

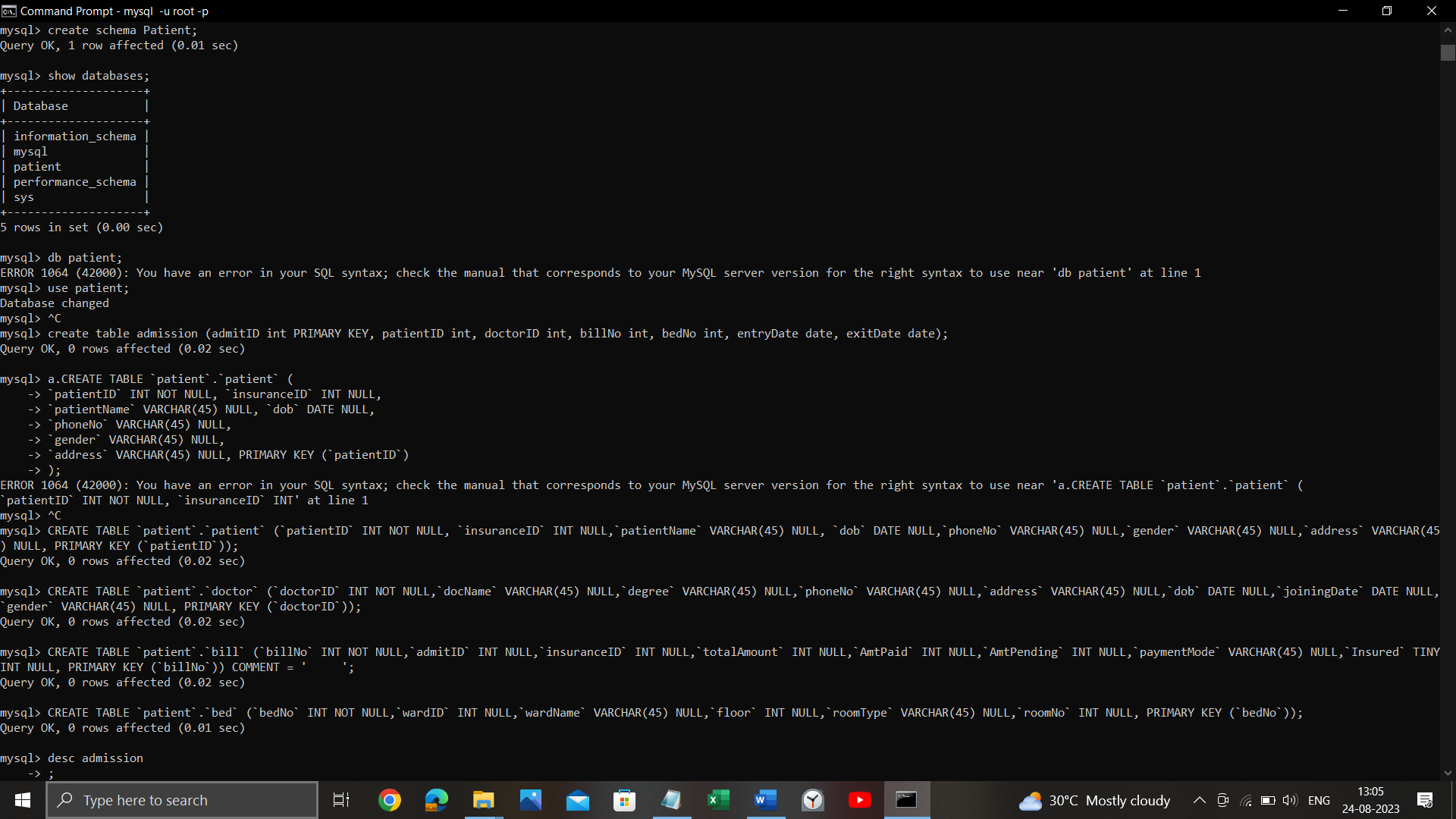
Admission

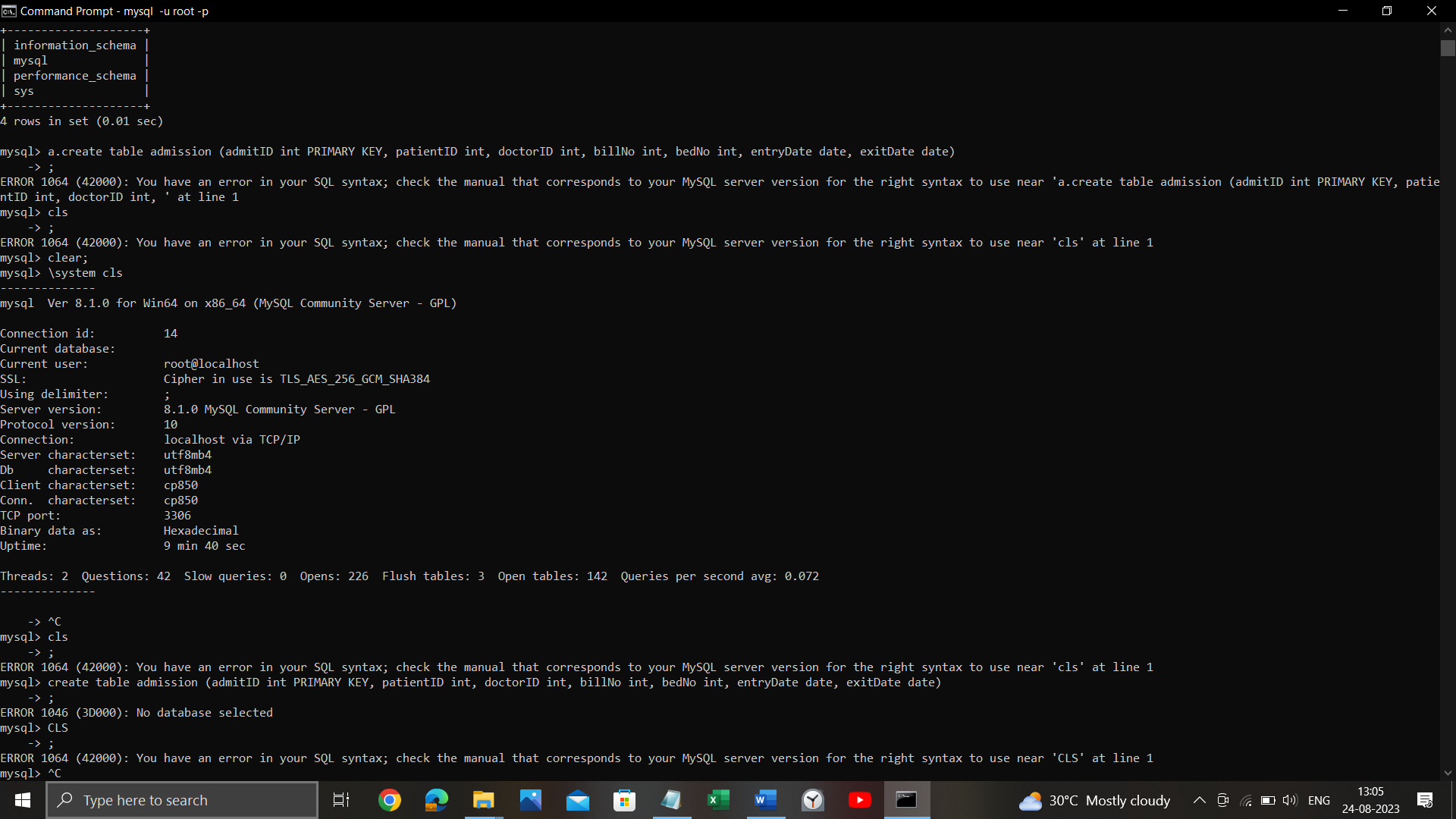
Bed

Bill

**Implementation :**

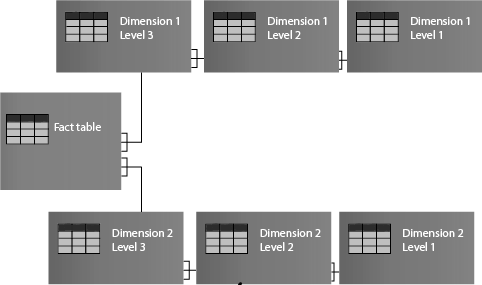




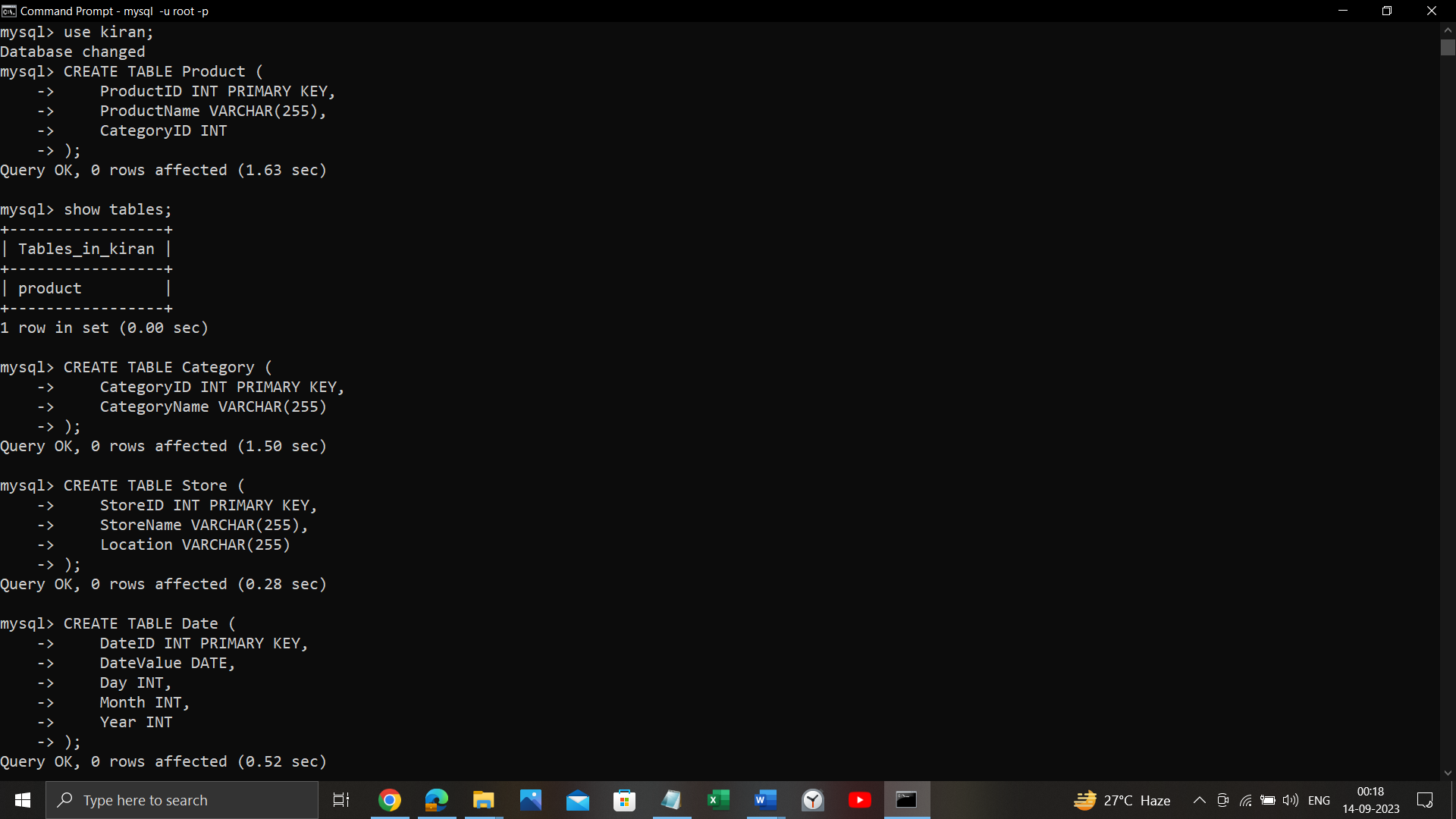
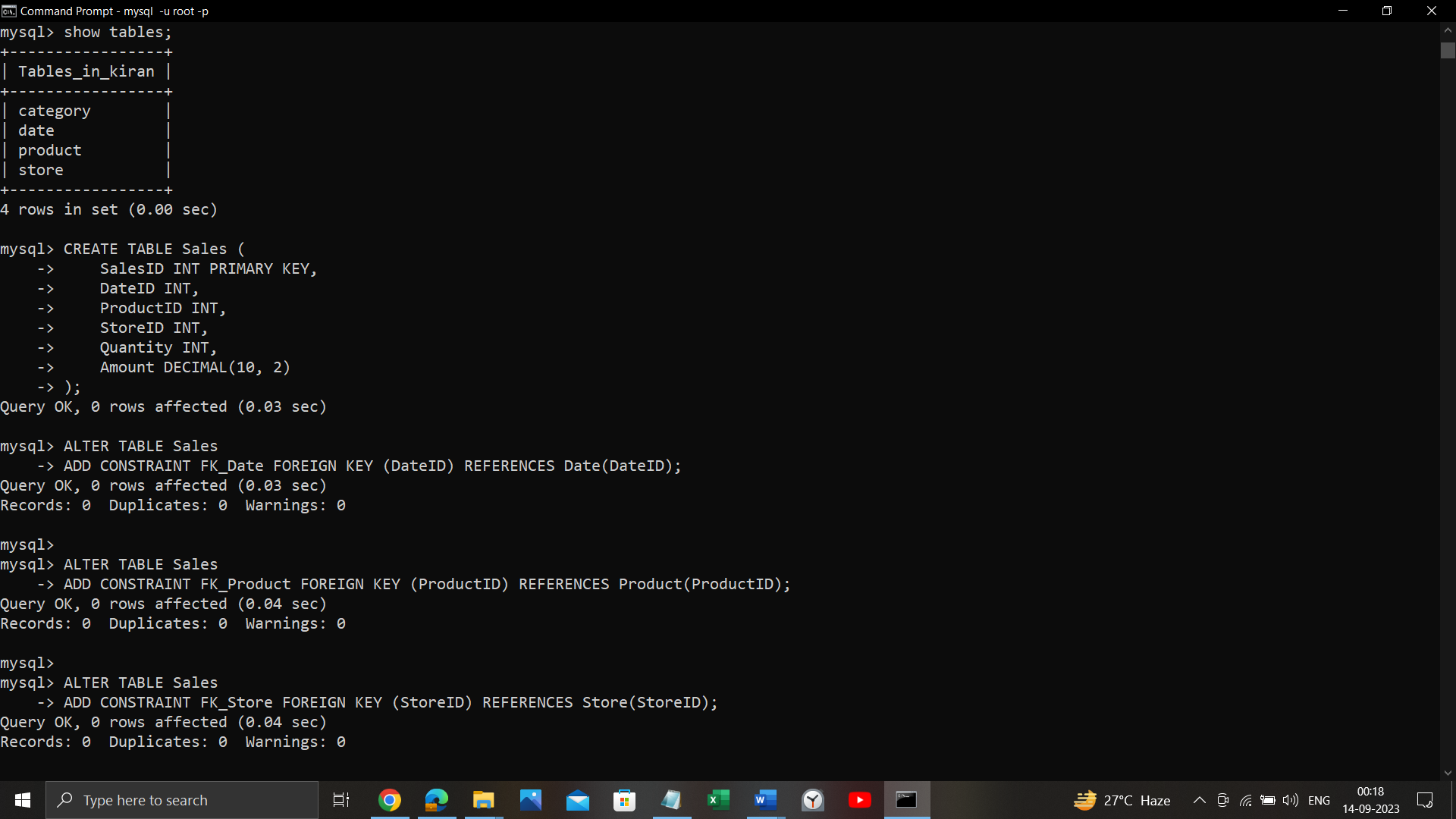


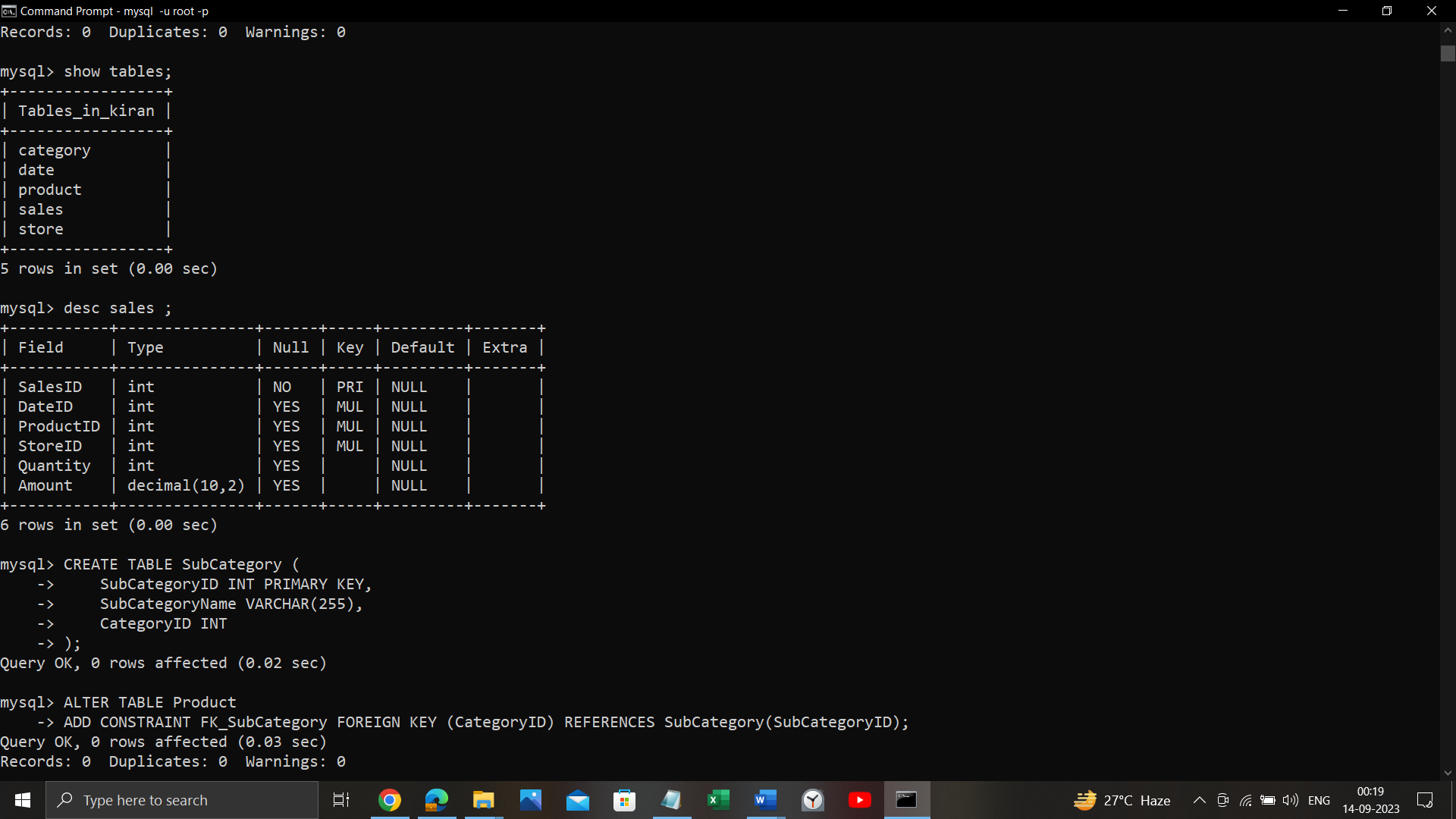
**Snowflake schema**

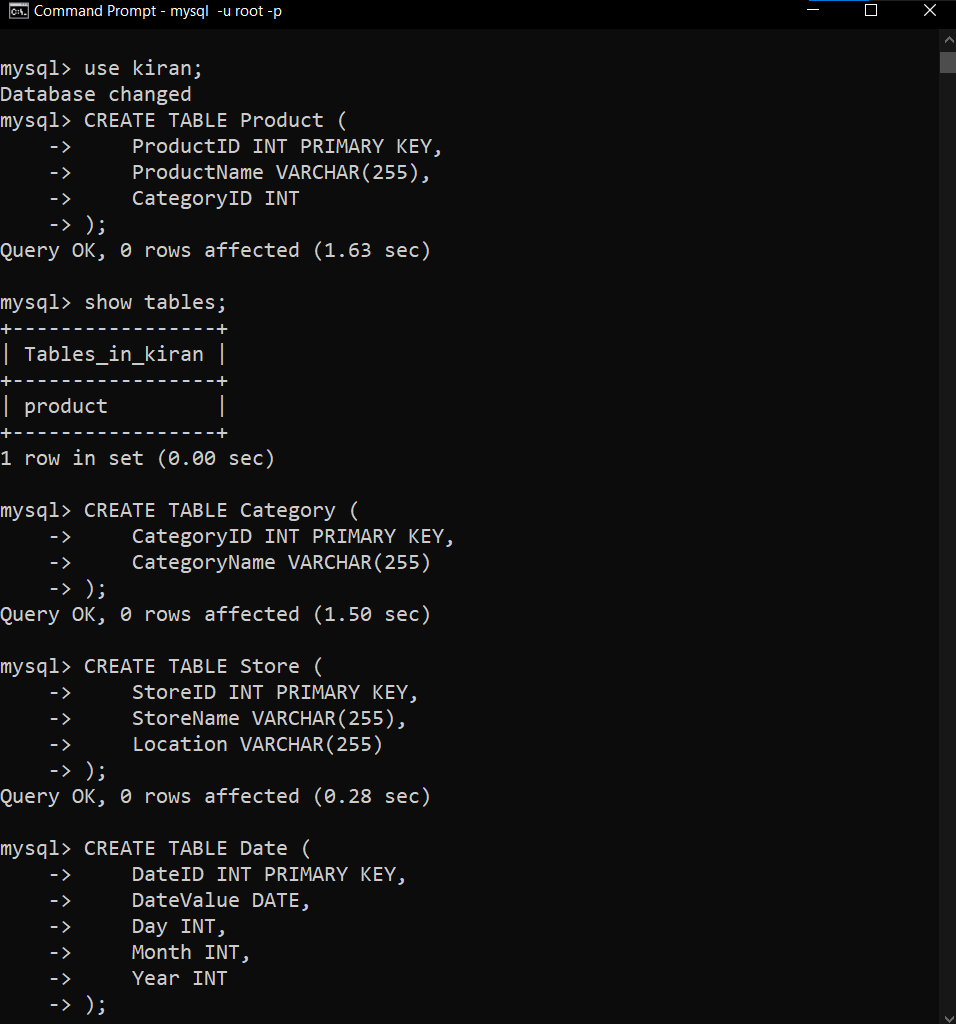
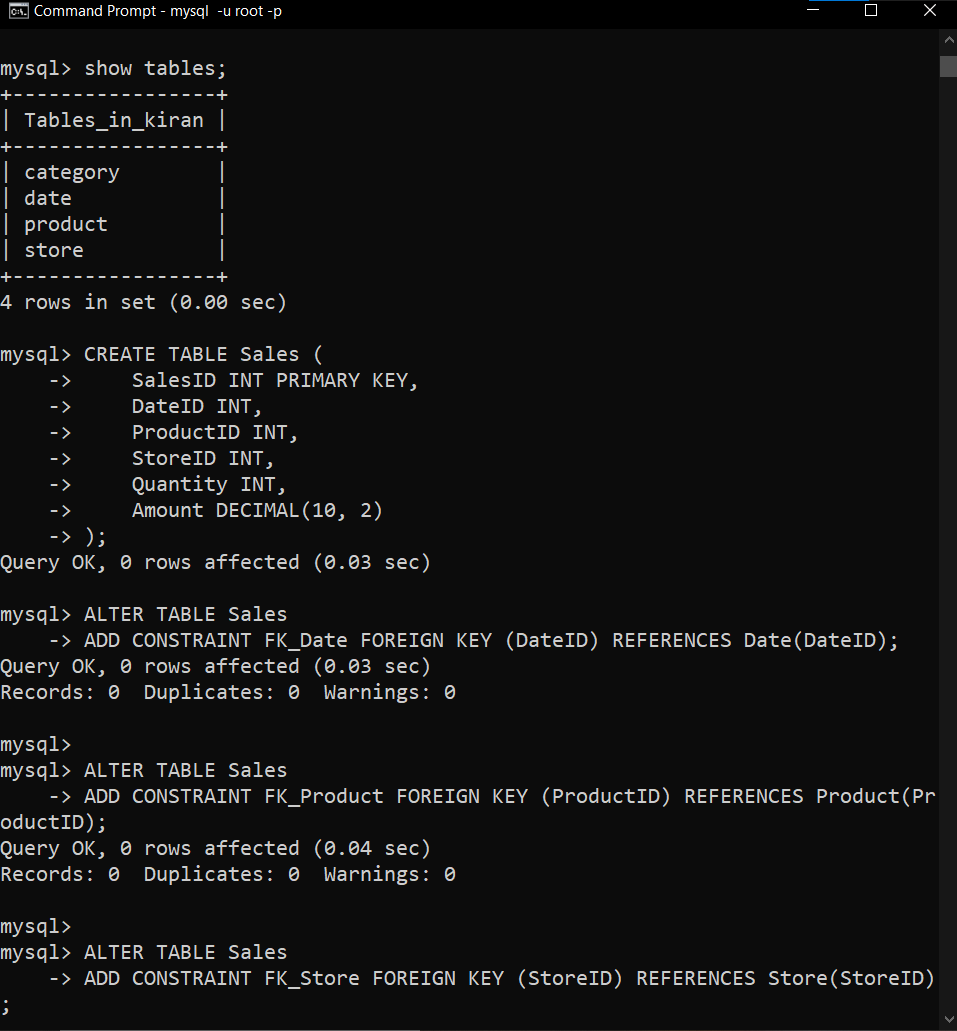
* Snowflake schema is equivalent to the star schema. "A schema is known as a snowflake if one or more-dimension tables do not connect directly to the fact table but must join through other dimension tables."
* The snowflake schema is an expansion of the star schema where each point of the star explodes into more points. It is called snowflake schema because the diagram of snowflake schema resembles a snowflake. Snowflaking is a method of normalizing the dimension tables in a STAR schema. When we normalize all the dimension tables entirely, the resultant structure resembles a snowflake with the fact table in the middle.
* Snowflaking is used to develop the performance of specific queries. The schema is diagrammed with each fact surrounded by its associated dimensions, and those dimensions are related to other dimensions, branching out into a snowflake pattern.
* The snowflake schema consists of one fact table which is linked to many dimension tables, which can be linked to other dimension tables through a many-to-one relationship. Tables in a snowflake schema are generally normalized to the third normal form. Each dimension table performs exactly one level in a hierarchy.
* The following diagram shows a snowflake schema with two dimensions, each having three levels. A snowflake schema can have any number of dimensions, And each dimension can have any number of levels.

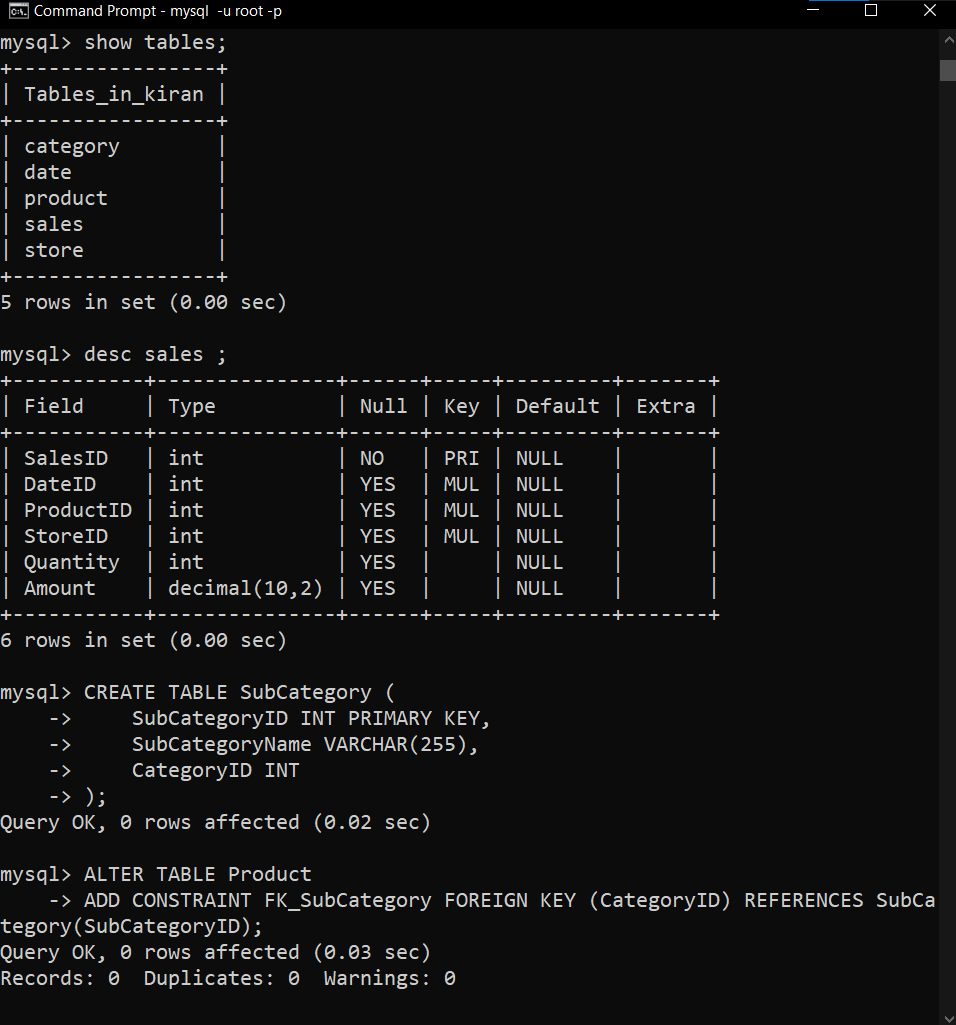
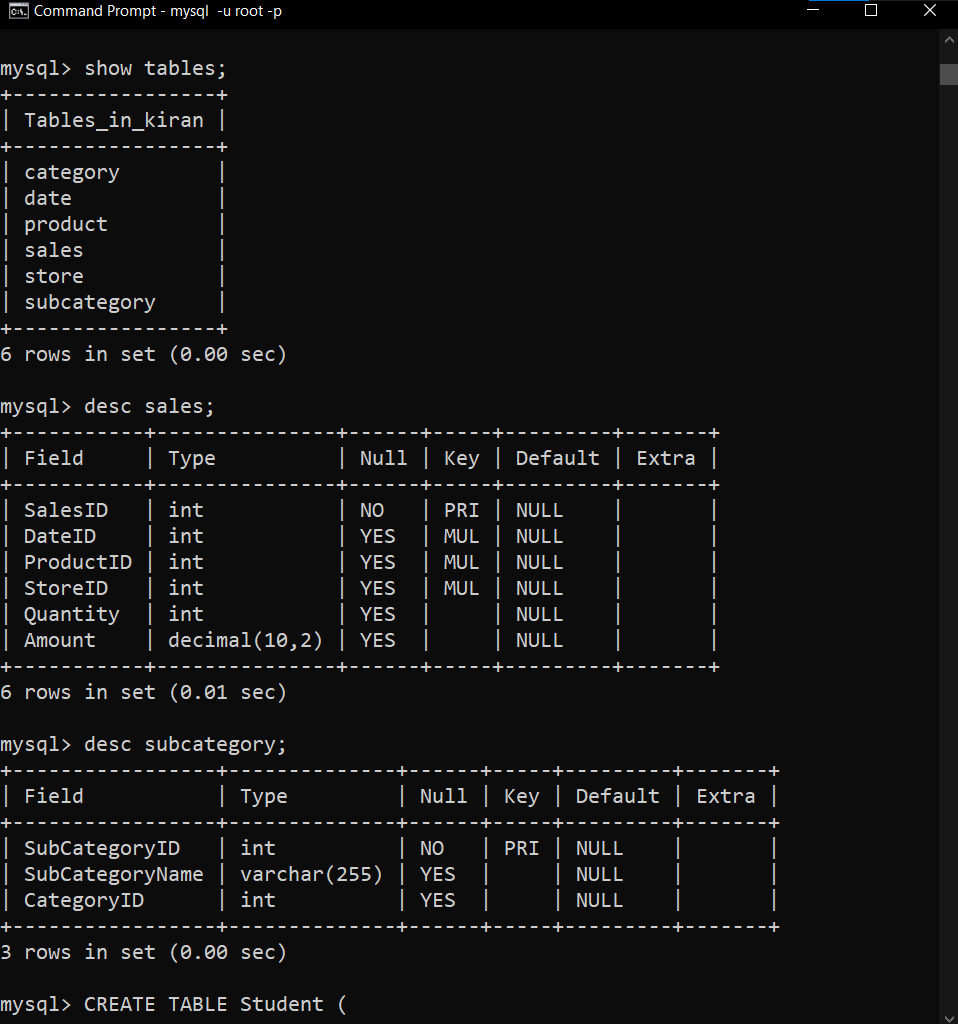


**Implementation And Output:**

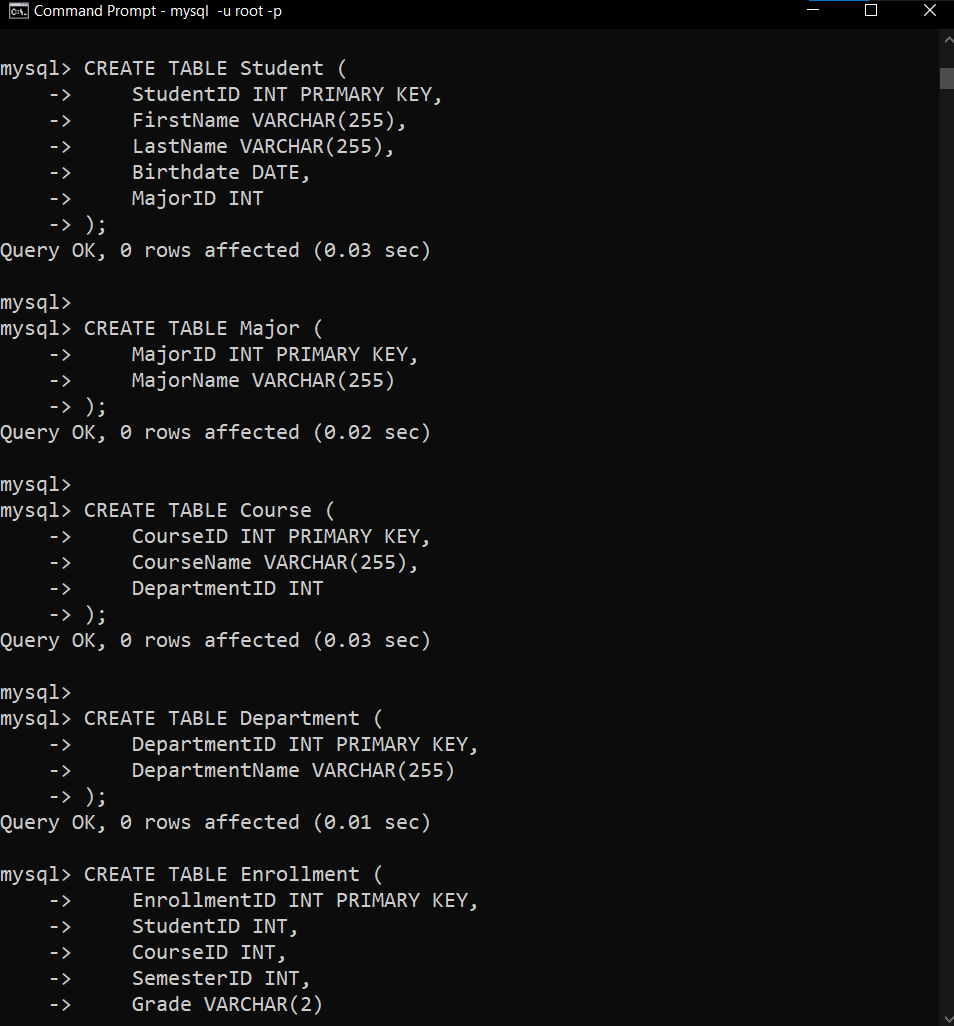
 

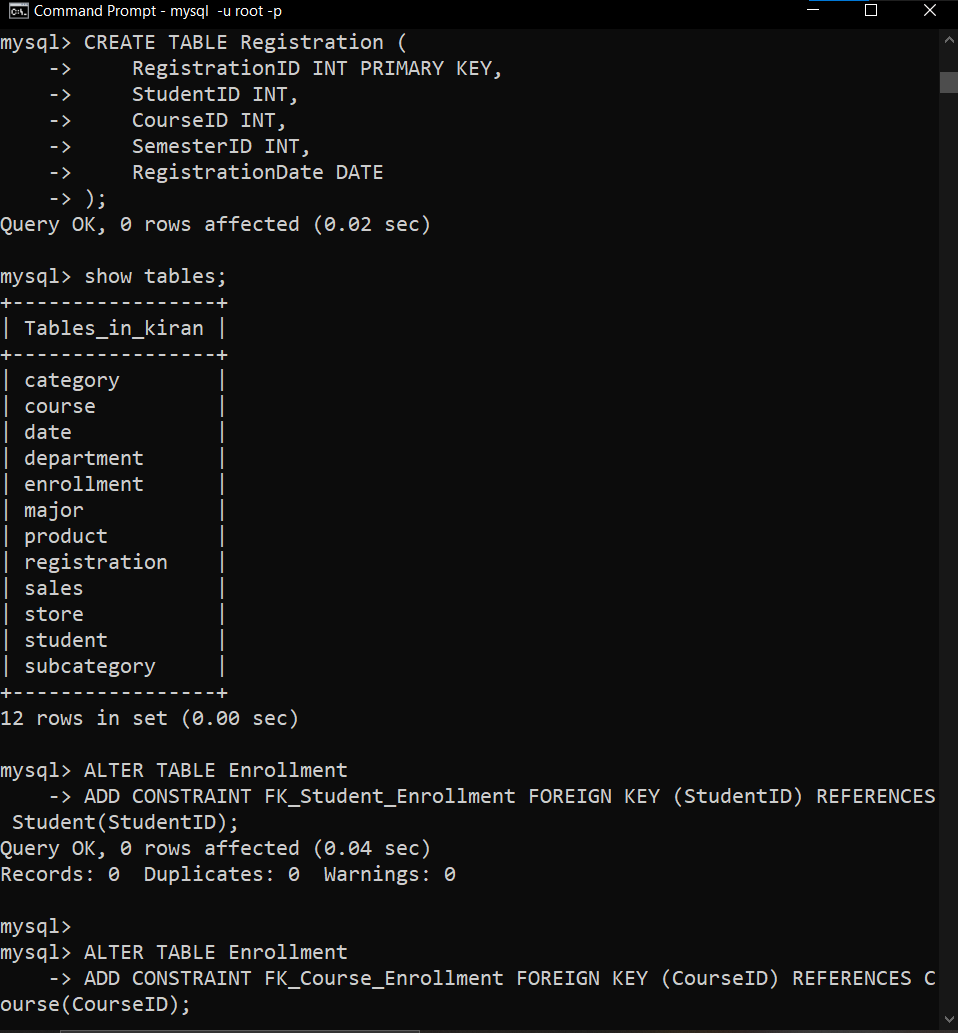


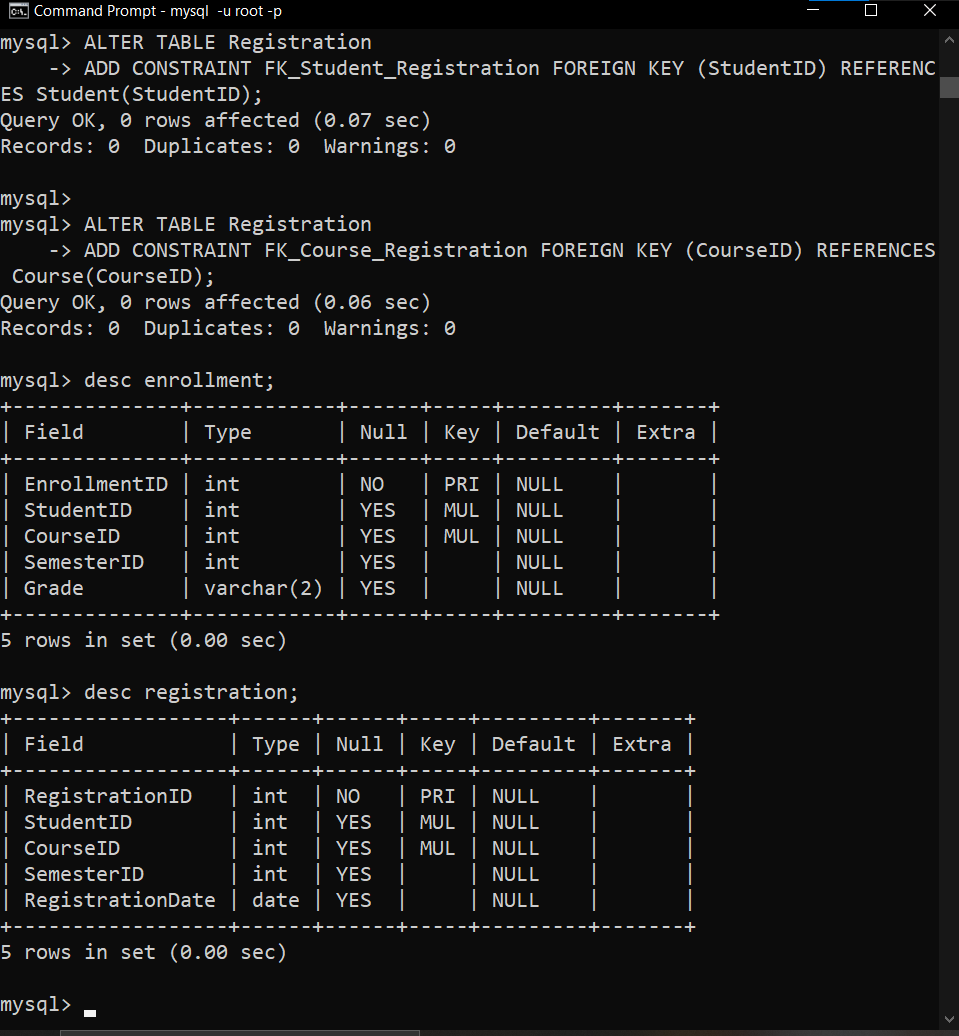
** **

**** ****

**Fact Constellation Schema**







**Conclusion**: In conclusion, database schema design plays a crucial role in organizing and optimizing data warehousing solutions. The Star Schema offers simplicity and efficient query performance but may involve data redundancy. Snowflake Schema normalizes data to reduce redundancy while maintaining data integrity. Fact Constellation Schema, with its multiple fact tables, provides flexibility for complex scenarios but requires more intricate maintenance. The choice among these schema types should align with specific business needs, balancing performance, data integrity, and complexity for an effective data warehousing solution.