**Day 7 Assignment - 11/12/2023 - Vamsi Viswanadham**

Worked on the below concepts.

* **Data Cleansing and Manipulation**

Data cleaning (also called data cleansing) involves identifying any inaccuracies in a dataset and then fixing them.

Here is a screenshot of the work.

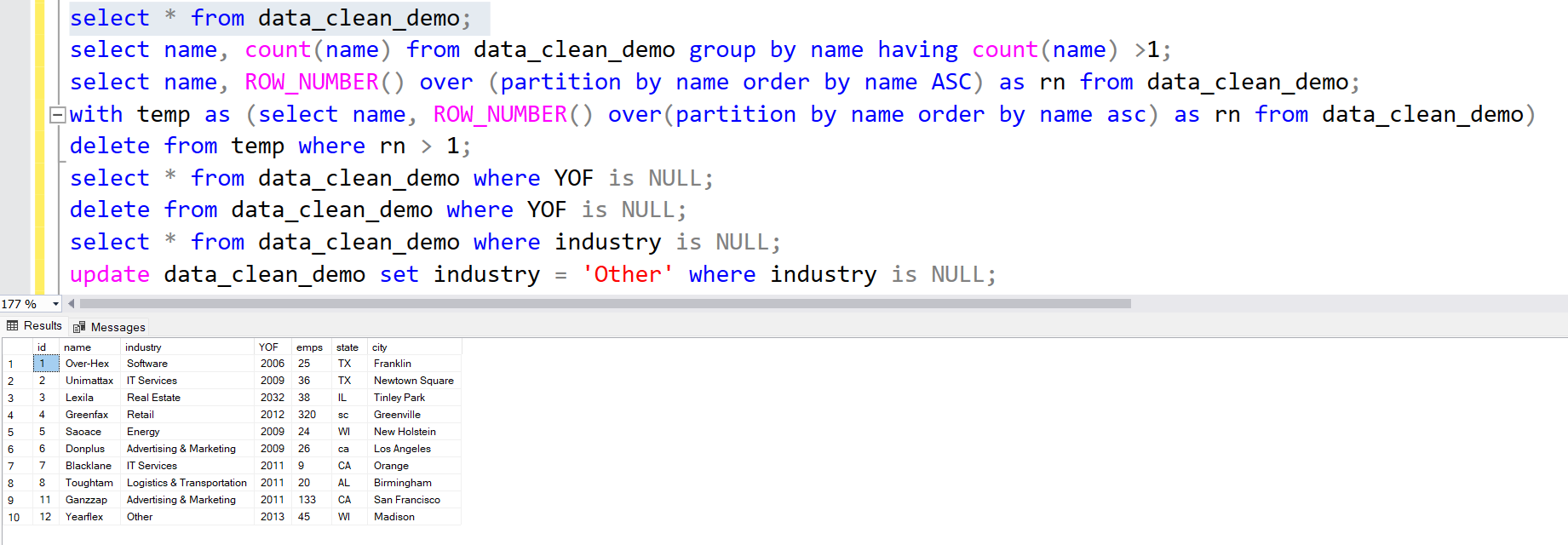
For eg: lets say you want to check if any null values present in the data, so you will do as below:

**select \* from data\_clean\_demo where YOF is NULL;**

This query will show the data where YOF feature is null. In order to delete that data where null values is present in your data especially at the YOF feature, you can do as shown below.

**delete from data\_clean\_demo where YOF is NULL;**

Here is a screenshot of the work.

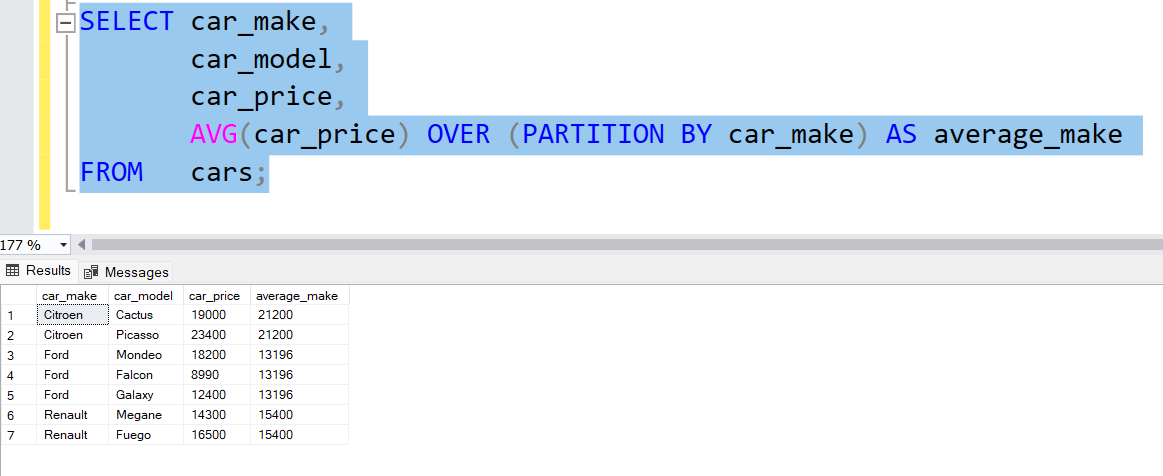


* **Partition By Clause**

PARTITION BY expression is a subclause of the OVER clause, which is used in almost all invocations of window functions like AVG(), MAX(), and RANK().

Here is a screenshot of the work.

Below, we are partitioning the results based on the car\_make feature to the average of the car price values of each record.



* Fact and Dimension Table Types
* Star Schema
* Snow Flake Schema

Snowflaking is a method of dimensional modeling in data warehousing where dimensions are normalized into multiple related tables. This approach creates a snowflake schema, which is a more complex variation of the star schema.

Star Schema vs. Snowflake Schema:

In a star schema, a central fact table is linked to multiple denormalized dimension tables. This setup facilitates quick and efficient queries but can lead to higher data redundancy and potential data integrity issues.

The snowflake schema evolves from the star schema by normalizing the dimension tables, reducing redundancy and improving data integrity. This schema uses a more complex structure with additional join operations.

Normalization in Snowflake Schema:

Normalization in snowflaking involves breaking down the dimensions into more specific tables, thereby reducing data redundancy.

It involves separating attributes with low cardinality into distinct tables and forming hierarchical relationships among these tables.

For example, in a product dimension, attributes like ProductType and ProductSupplier could be moved to separate tables, reducing redundant data.

Advantages of Snowflake Schema:

Reduced data redundancy enhances data integrity.

Simplified data maintenance due to less redundant data across multiple tables.

Potentially reduced disk space requirements.

Better suited for specific query needs or data types, particularly in ROLAP (Relational OLAP) architectures.

Trade-offs and Considerations:

Snowflake schemas are more complex to set up and maintain than star schemas.

Queries in snowflake schemas can be slower due to more join operations.

It's often chosen based on specific needs, such as when dealing with sparsely populated attributes, supporting many-to-many relationships, managing large dimensions with redundant data, or needing to query low cardinality attributes independently.

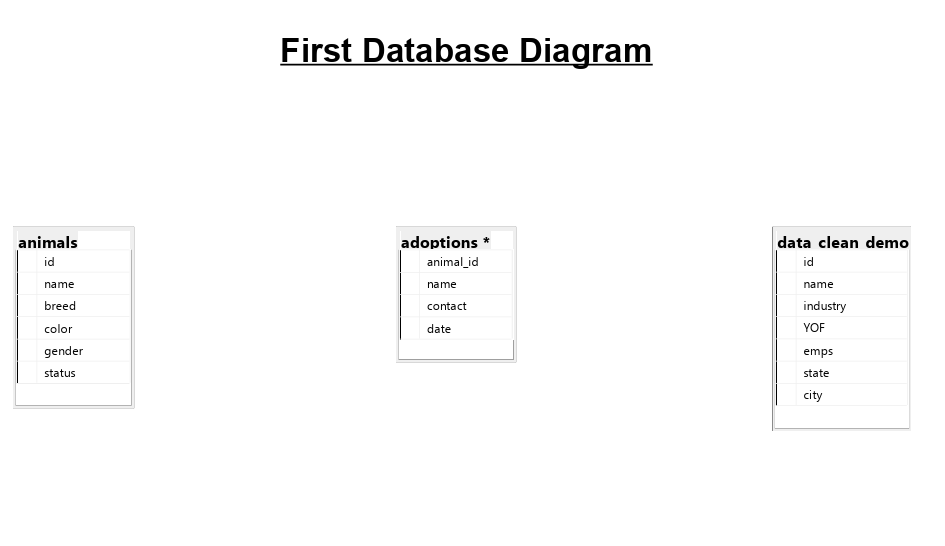
The decision to use a snowflake schema often depends on factors like query performance, data maintenance, storage, and development resources.

Use Cases for Snowflaking:

In situations where dimensions have many null values, support many-to-many relationships with limitations, contain a large amount of redundant data, or have attributes that are frequently queried independently.

Useful in hierarchies where different levels are queried independently.

Here is a screenshot of the work.



Here is a [SQL File](https://drive.google.com/file/d/1iQn8yZW0Q6LupiDIFMAMCseZ-POLS0QQ/view?usp=sharing) of the work.