Final Project Assignment – Pharmacy Claims



ALY 6030: Data Warehousing and SQL Prof. Kayal Chandrasekaran

March 31, 2024

By- Ayushi Walia

[walia.ay@northeastern.edu](mailto:walia.ay@northeastern.edu)

# Introduction:

# The data in the "ALY 6030 Final Project Data Set.xlsx" Excel file offers a glimpse into the world of pharmacy claims processed by a fictitious insurance company. These records include information about prescriptions and how they're paid for. It also comes with a data dictionary that explains what each piece of information means. We'll use this data to learn about how people use prescription drugs and how much they cost, helping us make better healthcare decisions.

# Part 1) Normalization

A screenshot of a computer

Description automatically generated

After looking at the data, we noticed it doesn't follow the first normal form (1NF). This is because there are repetitive attributes like "date filled 1," "date filled 2," etc., along with "copay1," "copay2," etc. We need to rearrange the data to remove these repetitions and any empty values. This way, the new data will follow the 1NF rules.

Once the data meets the requirements of 1NF, we find that it doesn't meet 2NF due to partial dependency in the table. To resolve this, I divided the data into three separate tables: two dimension tables (member\_dim and drug\_ndc\_dim) and one fact table (fact\_table).

Furthermore, to ensure 3NF compliance, I thought of splitting the drug\_ndc\_dim table into two tables (drug\_form\_dim and drug\_dim) to eliminate any transitive dependency.

Hence we got three dimension tables and one fact table:

Member\_dim

Drug\_dim

Drug\_form\_dim

Fact\_table

# Q1.

# Member\_id: Additive - Serves as a unique identifier for each member and is not subject to aggregation.

# Drug\_ndc: Additive - Similar to Member ID, Drug NDC serves as a unique identifier for each drug and is not subject to aggregation.

# Fill\_date: Non-additive - Represents specific dates on which prescriptions were filled and cannot be summed.

# Copay: Additive - Represents the amount paid by the member for each prescription and is directly summable.

# Insurance\_paid: Additive - Represents the amount paid by insurance for each prescription and is directly summable.

1. In the fact table, the grain is at the level of each fact row, where each row represents a single prescription filled by a member for a specific drug on a particular date, along with associated copay and insurance payment details.

# Part 2) Primary and Foreign Key Setup in MySQL

Now we have all the tables we need to perform queries, so I manually created the table and inserted the values from the main XLSX file (there was some glitch while trying to import the data) but it was worth it.

|  |  |  |
| --- | --- | --- |
| **Table** | **Primary Key** | **Natural / Surrogate key** |
| member\_dim | member\_id | Natural key |
| drug\_dim | drug\_ndc | Natural key |
| drug\_form\_dim | drug\_form\_code, | Natural key |
| fact\_table | fact\_id | Surrogate key |

1. In the fact\_table, there are two foreign keys:

member\_id references the member\_dim table, where member\_id is the primary key.

drug\_ndc references the drug\_dim table, where drug\_ndc is the primary key.

1. For the foreign keys defined in the fact\_table:

* **member\_id:**

On Delete: SET NULL

On Update: SET NULL

If a member record is deleted or updated, setting the foreign key to NULL ensures that any related fact records still exist but are not associated with a member. This option is suitable when a member may be removed from the system or updated without affecting the historical data of fact records.

* **drug\_ndc:**

On Delete: SET NULL

On Update: SET NULL

If a drug record is deleted or updated, setting the foreign key to NULL ensures that any related fact records still exist but are not associated with a drug. Like the member foreign key, this option is suitable when a drug may be removed from the system or updated without affecting the historical data of fact records.

**Part 3) Entity Relationship Diagram (ERD)**

A screenshot of a computer

Description automatically generated

The above ERD diagram depicts a STAR schema comprising a fact table connected to dimension tables. All dimension tables establish one-to-many relationships with the fact table.

**Part 4) Analytics and Reporting**

**4a.**

A screenshot of a computer

Description automatically generated

* **Ambien precriptions:** Total 3 Ambien prescriptions

A screenshot of a computer

Description automatically generated

# 4b. Age 65 and 65+ query:

# A computer screen shot of a computer Description automatically generated

* 1. **unique members are over 65 years with total prescriptions query:**

A screenshot of a computer

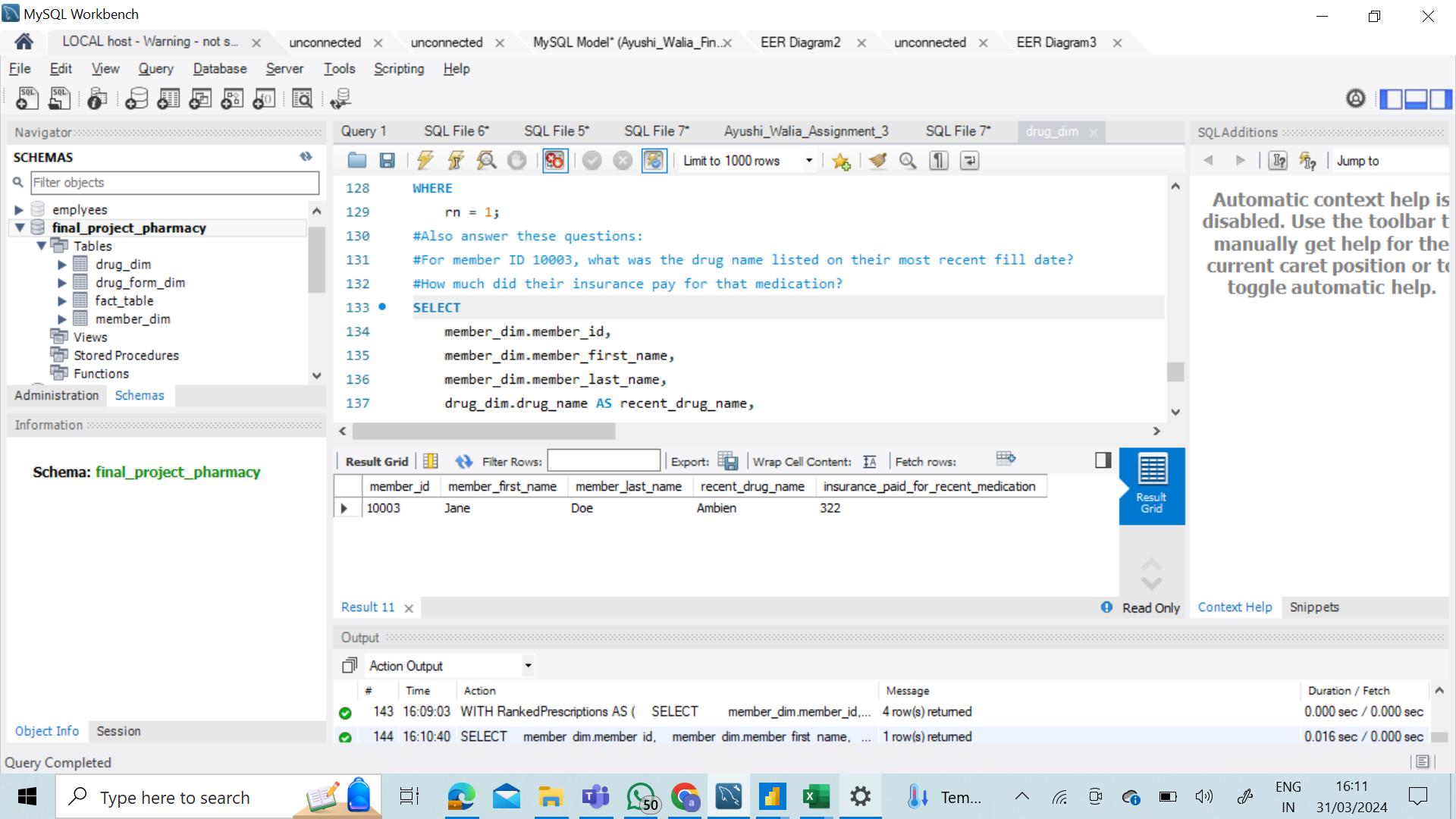
Description automatically generated

# The member aged above 65 has filled four prescriptions.

# 4c.

# 

* For member ID 10003, the most recent medication filled on 16th May 2018 is Ambien.



A screenshot of a computer

Description automatically generated

* The insurance paid for member ID 10003 for the drug Ambien is $322.

**References:**

1. *Setting up table relations what do “Cascade”, “Set Null” and “Restrict” do?* (2011, March 21). Stack Overflow. https://stackoverflow.com/questions/5383612/setting-up-table- relations-what-do-cascade-set-null-and-restrict-do
2. *How to add a primary key and foreign key to an existing table in Microsoft SQL Server database*. (2017, March 3). Stack

Overflow. https://stackoverflow.com/questions/42577979/how-to-add-a-primary-key-and- foreign-key-to-an-existing-table-in-microsoft-sql-s