**A red and black logo

Description automatically generated**

**Public Housing Inspections Star Schema**

**Data Warehousing and SQL**

**Project**

**Prepared by:Sachit Gopal**

**Introduction**

I was given a realistic problem to work on for my final homework. Actually, data on pharmacy costs for fictitious insurance company members is stored in an Excel file. This organization uses a third party known as a Pharmacy Benefit Manager (PBM) to pay for pharmacy expenses. A guide (data dictionary) outlining the contents of the file and its organization is also included. This data dictionary is a great tool for comprehending the problem's context.

I have these fictitious entries from the PBM so that I can build a practice database in my capacity as a developer. Setting up and creating some standard SQL query reports is my primary goal. Our company's analysts and business users will require these reports after the actual pharmacy spending information become available in a few months. When the real data warehouse is deployed for production, this practice database work is crucial to a smooth and successful launch.

Part 1 - Normalization

My goal is to put this data into a practice database, but in order to establish a start schema, I need to convert it into a simpler format, such as 3NF. However, since the data is not already in 3NF, getting it into 3NF will be my top focus. I therefore separated the entire set of data into 5 parts based on what I knew. The fact table will be the remaining one, and the other four will be dimension tables. I focused on all of the conditions listed above in order to join 3NF by the table.



Figure 1: Fact Table.

Question 1:

For each fact variable in your fact table, what type of fact is it? Additive, semi-additive, or non-additive?

Copay and insurance\_paid are the two facts in the fact table. Given that we may apply both truths to any aggregate operation, I think they are both additive types of facts. Put another way, we can use average() and sum() on these data.

Question 2:

In your fact table, describe the grain in one sentence. What does each fact row represent?

As is well known, the word "grain" refers to the data table's granular level. In my instance, the information regarding copay and insurance paid is contained in each row of the fact table. Moreover, insurance paid denotes the payment made by the insurance company, whereas copay denotes the payment made by the shareholder.

Part 2 -Primary key and Foreign Key setup in MySQL

As we all know, the primary key and foreign key are the most essential components of any data model. Therefore, in the second section of my project, I have to use SQL to set up these temporary keys in my tables.

A screen shot of a computer code

Description automatically generatedA computer screen shot of a black screen

Description automatically generatedA computer code on a black background

Description automatically generated

I finally had to change the data type of this column. For example, I changed the value of "drug\_form\_code" to VARCHAR (100) in the "dim\_drug\_form\_code" database.

Question 1:

Primary keys that I defined and its type:

|  |  |  |
| --- | --- | --- |
| Data Table | Primary Key | Natural or Surrogate Key |
| fact\_drug | id | Surrogate Key |
| dim\_brands\_generic | drug\_brand\_generic\_code | Natural Key |
| dim\_drug\_from\_code | drug\_form\_code | Natural Key |
| dim\_drug\_ndc | drug\_ndc | Natural Key |
| dim\_member | member\_id | Natural Key |

Question 2:

Foreign key and corresponding reference table with its primary key:

In the fact\_drug table, I have defined foreign keys here. The foreign keys from the fact\_drug, together with the primary keys of the reference table they belong to, are listed below.

|  |  |  |
| --- | --- | --- |
| Foreign Key from fact\_drug | Reference table | Primary Key of Reference table |
| member\_id | dim\_member | member\_id |
| drug\_ndc | dim\_drug\_ndc | drug\_ndc |
| drug\_brand\_generic\_code | dim\_brands\_generic | drug\_brand\_generic\_code |
| drug\_form\_code | dim\_drug\_form\_code | drug\_form\_code |

Question 3:

Explanation of why I choose SET NULL over CASCADE and RESTRICT.

Since I believed that "SET NULL" was the best option for this situation, I chose to go with it. First, we can remove or delete any entry from the fact table (Main table) and set NULL for its corresponding entry in the child table (Dimension table) by setting SET NULL for every foreign key. In this manner, we can make changes to the main table without encountering any difficulties.

Why not CASCADE now?

If the CASCADE constraint is enabled, each row deletion in the parent table will automatically remove the corresponding row from the child table.

As opposed to RESTRICT, why not?

I would not be able to remove or delete any rows from the parent or main table until I deleted the equivalent record from the dimension table or child table if I had set the RESSTRICT on foreign key. In real world situations, this will not be desired.

Part 3: Entity Relationship Diagram (ERD).

The best approach to see our data model's structure is with an ERD diagram. We may obtain a summary of the primary and foreign keys connected to every data table from the ERD. It also shows the kind of link that exists between two data tables. Thus, I utilized MySQL Workbench's "Reverse Engineering" option to create the ERD. Thanks to my prior upload of the database and my definition of the foreign and main keys, using reverse engineering made this operation quite simple.

A diagram of a drug

Description automatically generated

Figure: ERD for my Star Schema.

Part 4: Analytics and Reporting.

Question: (number of prescriptions grouped by drug name)

A screenshot of a computer

Description automatically generated

How many prescriptions were filled for the drug Ambien?

Question: Total prescriptions, counts unique (i.e. distinct) members, sums copay, sums insurance paid, for members grouped as either ‘age 65+’ or ’< 65’.

A screenshot of a computer

Description automatically generated

Question: Number of unique members over 65 years of age.

Based on the analysis's findings, there is only one individual who is over 65.

Question: Number of prescriptions filled.

Five insurances in total are filled out by people under 65. On the other hand, six insurances were filled out by those over 65.

Question: Amount paid by the insurance for the most recent prescription fill date using windows function.

A screenshot of a computer

Description automatically generated

Question: Most recent drug taken by member id of 10003.

The most recent substance used by member ID 10003 was ambien.

Additionally, on May 8, 2018, David Dennison (10001) took Risperidone at an insurance cost of $55. On June 14, 2018, John Smith (10002) received Amoxicillin at a cost of $130 covered by insurance. When Jane Doe (10003) took Ambien on May 16, 2018, it was 322. Finally, Elaine Rogers (10004) covered the cost of the medication Diprosone that he took on December 8, 2017, with insurance, paying 712.

References

Darnley, N. (2023, March 30). *How to use CASE WHEN in GROUP BY*. LearnSQL.com. https://learnsql.com/blog/group-by-case-when/

*SQL GROUP BY CASE statement with aggregate function*. (n.d.). Stack Overflow. https://stackoverflow.com/questions/1208854/sql-group-by-case-statement-with-aggregate-function