***GROUP 36***

**Cleaning Our Dataset**

* We had already cleaned our dataset by inserting value 0 whenever we encountered null values in the attribute for example in the case of days\_since\_prior\_order attribute of the orders table. We also converted the values for our attributes to integer and text types as required.

Text

Description automatically generated

**Integration of Dataset**

* For data integration we created different kinds of views, one of which, dept\_aisle( view2), sorts the aisles according to departments and displays an array of aisles for each department. For this we integrated the product department and aisle table and created a view out of it to find which different aisles belong to what departments.

**Text

Description automatically generated**

* **Similar task was done for Aisle\_Product, wherein we sorted different unique products into their aisle by joining aisle and product table. This way any user can simply find the location of their desired product.(This one may look complex as there are a lot of products for an aisle)**

**Text

Description automatically generated**

**Itemset Mining**

* First, we formed a table named temp2. This table contains all the important information that we need. The table is formed by joining order and order\_product table on order\_id, call this newly formed table temp1, and then joining product table on the temp1 table on product\_id thus resulting in our temp2 table.
* We then applied itemset mining on this table for product\_id, as our goal here was to find the products that were bought together frequently in multiple different orders.
* The basic process is a python program that keeps on writing sql queries to form k-items lattice, till the rows in the lattices are zero or no more k number of elements are being brought together at least n number of times where n is our threshold set by us.
* We have set the threshold to 10,000. This means only those elements are included who were being brought in minimum 10000 number of orders.
* We then take each row of the final lattice and take each unique element from them and print at the last the most frequently brought items.
* In Total we got three lattice and we have printed the third lattice for you. The fourth lattice contains 0 rows.

**Shape

Description automatically generated with medium confidence**

**Shape

Description automatically generated with medium confidence**

**Which model is a best fit for our dataset?**

* In the case of our project, the Instacart Market Basket Analysis dataset, a relational model is a better fit for the task of itemset mining to discover interesting association rules.
* The dataset consists of structured data with well-defined entities such as orders, products, aisles, and departments. Each entity has its own set of attributes that can be easily mapped to columns in a relational database.
* This makes it easy to create a schema that represents the entities and their relationships, as we described in our proposed schema.
* Relational databases are designed for structured data and excel at handling large volumes of structured data with complex relationships. They also provide powerful query capabilities for joining and aggregating data from multiple tables, which is important for tasks such as itemset mining.
* On the other hand, document-oriented databases are better suited for unstructured or semi-structured data that can vary in schema and format. They provide more flexibility in terms of schema design and can handle large volumes of data with high write and read throughput.
* However, in the case of the Instacart dataset, the structured nature of the data and the well-defined relationships between entities make a relational model a more appropriate choice for itemset mining.