SQL Queries: I used MySQL workbench to run these querries.

1. When considering total number of items purchased from receipts with 'rewardsReceiptStatus’ of ‘Accepted’ or ‘Rejected’, which is greater?

I could not find ‘Accepted’ , and I listed the unique options below:

A screen shot of a computer

Description automatically generated

So I assumed that ‘Accepted’ refers to ‘Finished’ because of the provided info below:

**rewardsReceiptStatus:** status of the receipt through receipt validation and processing

solution:

**SELECT rewardsReceiptStatus, SUM(purchasedItemCount) AS total\_items\_purchased**

**FROM receipts**

**WHERE rewardsReceiptStatus IN ('Finished', 'Rejected')**

**GROUP BY rewardsReceiptStatus;**

A screenshot of a computer

Description automatically generated

1. When considering average spend from receipts with 'rewardsReceiptStatus’ of ‘Accepted’ or ‘Rejected’, which is greater?

**WITH receipt\_status\_averages AS (**

**SELECT**

**CASE WHEN rewardsReceiptStatus = 'FINISHED' THEN avg(totalSpent) END AS finished\_avg,**

**CASE WHEN rewardsReceiptStatus = 'Rejected' THEN avg(totalSpent) END AS rejected\_avg**

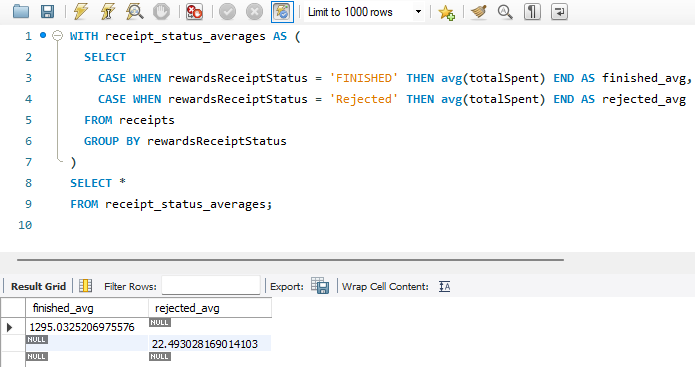
**FROM receipts**

**GROUP BY rewardsReceiptStatus**

**)**

**SELECT \***

**FROM receipt\_status\_averages;**

****

1. Which brand has the most spend among users who were created within the past 6 months?

SELECT b.name AS brand\_name, SUM(r.totalSpent) AS total\_spend

FROM receipts r

JOIN receipt\_items ri ON r.receiptId = ri.receiptId

JOIN brands b ON ri.brandCode = b.brandCode

JOIN users u ON r.userId = u.UserId

WHERE u.createdDate >= CURRENT\_DATE - INTERVAL '6' MONTH

GROUP BY b.name

ORDER BY SUM(r.totalSpent) DESC

LIMIT 1;

A screenshot of a computer

Description automatically generated

Hello Adam,

I hope you are doing well. My name is Mohammed and I am part of the analytics team at Fetch. While analyzing dataframes containing receipt, user, and brand information, I encountered a few questions and issues about the data quality and overall performance, which include:

* **CPG Collection**: the info about the data mentions that CPG references the CPG collection, without any additional information, leaving it ambiguous, and any additional guidance would be much appreciated.
* **Missing and non-unique primary keys**: After preprocessing, the data was divided into five dataframes. Each dataframe should ideally have a unique identifier (primary key) for each row. However, many candidate keys have null values and appear repeatedly, violating the requirements for a primary key (unique, singular, and not allowing null values).
* **Inconsistent data**: the number of unique brands in the brands dataframe is significantly higher than the one in the receipt dataframe, with a difference in the range of 600, indicating that some of the databases are missing much information, making advanced analytics a more challenging task.
* **High percentage of null values**: Several columns contain up to 90% null values. This poses a challenge for collaboration with data science and machine learning teams, as such high missing values make advanced statistical analysis difficult.
* **Missing item description**: The receipt\_items table has many null values in the "description" field. This is problematic for analyzing shopping patterns and trends, as missing descriptions can prevent extracting insights, especially during events like the Super Bowl (e.g., difficulty in observing sales trends for pizza or beer). Additionally, lacking detailed item information hinders data-driven supply planning for partners, potentially leading to missed opportunities to maximize profits by matching demand.

We need to make sure that when a user is making a purchase, that all of the data is captured correctly, and the database does not have a high percentage of empty values.

A high daily influx of receipts poses several performance challenges, such as:

* **Ingesting this data** can create backlogs and slow loading times.
* **Storing the growing dataset** can strain storage capacity, impacting query performance.
* **Extracting and cleaning valuable information** from these receipts can be computationally expensive, leading to processing delays.
* **Generating reports** and analyzing massive amounts of data can further strain resources, resulting in longer response times.

To mitigate these challenges, we can utilize :

* **Batch processing**: where we group receipts into batches, instead of processing each receipt individually. This improves efficiency and reduces the overhead associated with starting and stopping the processing for each receipt individually.
* **Data partitioning**: we can also organize the data by date, store ID, or other relevant criteria (like labeling and organizing boxes in the warehouse). This allows for faster retrieval of specific subsets of data (like quickly finding boxes from a specific date).
* **Horizontal scaling**: adding more resources could be an option, where we can add more machines to distribute the workload across them.

I would love to discuss these topics with you in detail and collaborate to solve those challenges. Could you please provide me with your availability so we can set up a meeting? Please let me know if you have any questions regarding this email.

Thank you for your time,

Mohammed Younis