**Fetch Rewards Coding Exercise**

**First: Review Existing Unstructured Data and Diagram a New Structured Relational Data Model**

After reviewing 3 data sample files and identified they were all nested JSON files. To access the data files, I used an iterative approach to flatten these nested JSON files using python. With the help of MySQL Workbench, I generated the ERD for the 3 data files.

A picture containing timeline

Description automatically generated

**Second: Write a query that directly answers a predetermined question from a business stakeholder.**

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

SELECT rewardsReceiptStatus, SUM(purchasedItemCount) as purchased\_total

FROM receipts

GROUP BY rewardsReceiptStatus

ORDER BY purchased\_total DESC*;*

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

SELECT rewardsReceiptStatus, AVG(totalSpent) as avg\_total

FROM receipts

GROUP BY rewardsReceiptStatus

ORDER BY avg\_total DESC;

**Third: Evaluate Data Quality Issues in the Data Provided**

The data at hand are nested JSON files with multi records that are not delimitated. After flattening the nested JSON objects, I presumed the data is extracted correctly and performed EDA accordingly. The distribution of data is quite unclear as well as the datatypes are not appropriately appended. The data is poorly formatted, due the nested nature of the JSON files, upon flattening the json file, each nested key creates a new column. Furthermore, the data relation between the 3 sample tables aren’t clearly explained, especially for the nested columns.

Furthermore, over half of the columns had 85-90% of missing data. The missing data can be addressed as follows:

* Address business question with limited data : For example, considering rewardsReceiptItemList\_userFlaggedBarcode column, it has 935 missing values but since it is a usserflaggedbarcode values I’m concluding that this column points to the data by the user that directs to some reason why this was flagged.
* Drop Columns: Taking rewardsReceiptItemList\_originalMetaBriteItemPrice column for an example, it has only 3 values out of 1119 rows. After carefully reviewing the relevant columns, I presume only one product is included in the MetaBrite rewards and since it does not answer any business question we can drop all the relevant columns.
* Use relevant columns to fill the null values: From brands table, we can use category column to fill at least 495 null columns present in categoryCode.

**Fourth: Communicate with Stakeholders**

Dear Sir/Madam,

I have reviewed the 3 sample data files and was hoping if you could help me find answers for a few of many following questions:

1. Questions regarding the data

* What is the source of the data?
* What is the Importance of each column?
* What are the Business KPI’s that reflect your business questions or priorities?

1. Data Quality Issues

Upon careful review of data, I came across following data issues:

* Poorly formatted data files because of nested JSON files (Nested JSON provides higher clarity in that it decouples objects into different layers, making it easier to maintain)
* Unclear data distribution
* Majority data missing

1. To resolve the data quality issues

* Inconsistency – There are many columns present in the data that share common information, this increases redundancy within the tables.
* Incomplete Information – A few columns like category and CategoryCode have incomplete information, for example where a category has Bakery mentioned the relevant categoryCode identifies some whereas others are just Null values.

1. Information needed to optimize the data assets

* Business background – Learn more about domain knowledge, to understand the target variables, establish business questions and define KPI’s for the same.
* Business use cases – What will the data be used for?
* Source of the data and data extraction method used.
* Data relation description between the 3 sample data files, it will help us explore the data from different perspectives and clean the data accordingly.

1. Performance and Scaling Concerns, with suggestions to solve any issues

A production ready service is scalable and efficient. Performance and scaling concerns arise when,

* Traffic is too high – Increase hardware or processing power, may result into cases of bottlenecks. Can be addressed using horizontal scaling to increase throughput by adding additional nodes, which is more appropriate than vertical scaling as it may result into resource deficit.
* Scalable data storage – Choosing the wrong database or schema, it is crucial to select, build and maintain database, preferably assign a team to completely focus on this