Kevin Zhao (zhaokevin79@gmail.com)

Section 2: Completing SQL queries + generating initial insights about the database

Initially, I chose to complete question 3 as it seemed to be the simplest and cheapest query with the fewest number of joins. However, this query gives little insight into the nature of the data, and I believed the result from this query to be not as important to the stakeholders as some of the other queries. After re-framing my mindset to “What queries would give the most knowledge and insight into this data and help drive stakeholder decisions?” I decided to complete questions 1 and 5.

Below are my notes and thoughts on each of the 3 queries I completed.

Question 1: *What are the top 5 brands by receipts scanned for the most recent month?*
After exploring the data initially, I found that the most recent receipt scanned from this dataset was March 1, 2021. Today is March 1, 2025, meaning this dataset is exactly 4 years old. Thus, to answer this question, I will assume this data is “current” and move “today” back to the date if the most recent receipt scanned, and find receipts scanned within 1 month of the new “today”.

This question is also ambiguous – “What are the top 5 brands” has no concrete metric (“best” could mean highest selling, most sold items, etc.). This is also something I will bring up with the stakeholder in the future, but for now I will calculate just the highest selling items.

After initially running the query, I found that most of the Items did not have an associated “barcode” or “brandCode” field – meaning they could not be traced back to the Brand that they belong to, and thus this query would not produce actionable, insightful data for the stakeholder. This discovery will be added to my list of insights for the stakeholder in section 3/4.

This insight will be important to bring up, as this query can generate valuable data for which brands users are buying now, and can help predict which brands are rising or falling in popularity along with query 2, and not having this data can negatively impact stakeholder decisions which could hurt Fetch's revenue.

QUERY:

```
one_month_in_unix = 2629743 # Approximately 1 month in seconds of UNIX time
most_recent_receipt_scanned = cur.execute("""
    SELECT strftime('%s', dateScanned)
    FROM Receipts
    ORDER BY dateScanned DESC
    LIMIT 1
""").fetchone()[0]

highest_selling = cur.execute("""
WITH temp AS (
    SELECT r.id, i.brandCode, COUNT(*) AS itemCount
    FROM Receipts r JOIN Items i
    ON r.id = i.receiptId
    WHERE unixepoch(r.dateScanned) >= ? - ?
    GROUP BY r.id, i.brandCode
)
SELECT b.name AS Brand, SUM(temp.itemCount > 0) AS NumReceipts
FROM Brands b JOIN temp
ON b.brandCode = temp.brandCode
GROUP BY b.name
ORDER BY NumReceipts DESC
LIMIT 5
""", (
    most_recent_receipt_scanned, one_month_in_unix
))
```

RESULT:

```
[('Viva', 1)]
```

Question 3: *When considering average spend from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?*

This question is a proxy reflection of how Fetch's receipt verification system performs – the higher the ratio between avg. spend of 'Accepted' and 'Rejected' receipts, the more Fetch's verification system allows high-spending receipts and denies low-spending receipts.

However, without supporting analysis on different parameters that could affect if a receipt is accepted or rejected (e.g. Number of items bought, brands bought, account age, store the receipt is from, etc.), this insight does not mean much on its own.

While looking through the Receipts table, I could not find rows with ‘rewardsReceiptStatus’ = ‘Accepted’. I did find statuses = ‘Finished’, which was the closest synonym to ‘Accepted’ of all statuses I could find in the Table, so I will be using ‘Finished’ instead.

QUERY:

```
SELECT rewardsReceiptStatus, ROUND(AVG(totalSpent), 2)
FROM Receipts
WHERE rewardsReceiptStatus = 'REJECTED'
OR rewardsReceiptStatus = 'FINISHED'
AND totalSpent NOT NULL
GROUP BY rewardsReceiptStatus
```

RESULT:

```
[('FINISHED', 80.85), ('REJECTED', 23.33)] # Accepted/Finished is greater
```

Question 5: *Which brand has the most spend among users who were created within the past 6 months?*

Similarly to Question 1, the most recent account opened was Feb 12, 2021, which is over 4 years from today’s date of March 1, 2025. Thus, to make this query more insightful, I am moving “today” to be the date of the most recent account creation.

I decided that this question would provide actionable insight to stakeholders because it shows which brands new customers are attracted to. Thus, giving bonus fetch points and rewards to customers who buy from these brands will promote sales from new customers, which increases revenue for Fetch from those brands who rely on Fetch to promote their products.

QUERY:

```
six_months_in_unix = 2629743 * 6 # Approximately 6 months in seconds of UNIX time
```

```
most_recent_created_account_date = cur.execute("""
    SELECT strftime('%s', dateScanned)
    FROM Receipts
    ORDER BY dateScanned DESC
    LIMIT 1
""").fetchone()[0]

res = cur.execute(f"""
WITH temp AS (
    WITH recent_users AS (
        SELECT id FROM USERS
        WHERE unixepoch(createdDate) > ? - ?
    )
    SELECT i.brandCode, SUM(i.finalPrice) AS brandCodeSum
    FROM Receipts r JOIN recent_users u
    ON r.userId = u.id
    JOIN Items i
    ON r.id = i.receiptId
    GROUP BY i.brandCode
)
SELECT b.name AS BrandName, SUM(temp.brandCodeSum) AS TotalSales
FROM Brands b JOIN temp ON
b.brandCode = temp.brandCode
GROUP BY b.name
ORDER BY TotalSales DESC
LIMIT 1
```

RESULT:

```
[('Cracker Barrel Cheese', 703.5)]
```