**Exercise:**

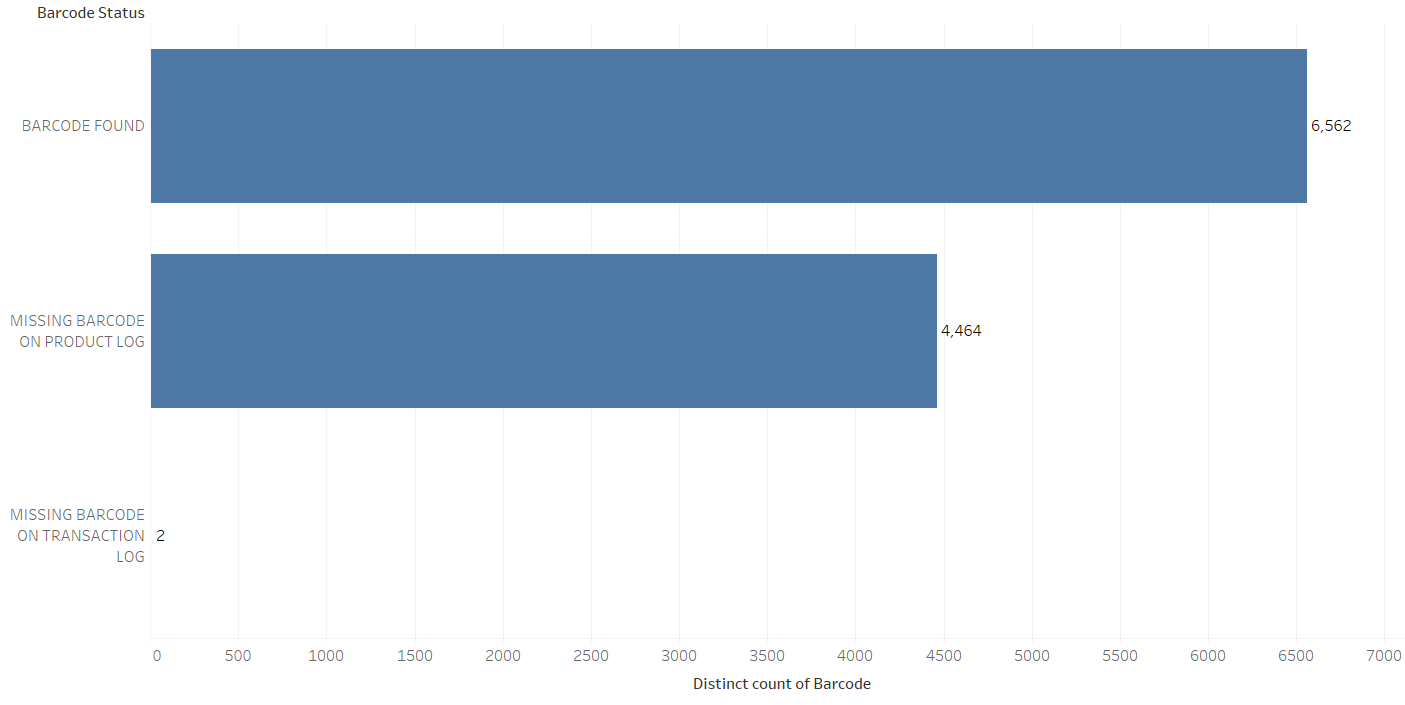
**First: explore the data**

**Review the unstructured csv files and answer the following questions with code that supports your conclusions:**

* Are there any data quality issues present?
* ----------------------------- USERS TABLE ------------------------------------
  + The users table is missing a lot of records that are present in the transactions table.
    - SELECT COUNT(DISTINCT USER\_ID) FROM TRANSACTIONS\_EV A WHERE NOT EXISTS(SELECT 1 FROM USERS\_D WHERE ID = A.USER\_ID)
    - 
  + People that use Fetch are apparently long-lived. The users table contains around 60 folks that are over 100 years old. Several of them were around at the turn of the last century which is impressive. I’m guessing some of these records should have a ‘20’ for the year rather than a ‘19’, but not entirely sure where the cutoff would be other than arbitrarily picking a maximum age.
* SELECT \*

FROM users\_d

WHERE birth\_date < DATEADD(YEAR, -100, GETDATE());

* + The gender data has 12 variations, many of which are essentially the same thing. Need to standardize how it’s being stored.
    - SELECT DISTINCT GENDER FROM USERS\_D
* ----------------------------- PRODUCTS TABLE ------------------------------------
  + Similar to the users table this table is also missing a lot of records that are present in the transactions table.
    - 
    - SELECT \* FROM TRANSACTIONS\_EV A WHERE NOT EXISTS(SELECT 1 FROM PRODUCTS\_D WHERE BARCODE = A.BARCODE)
  + There’s duplication within the table and a number of nulls, which is bad as barcode is our main linkage between the transaction table and the product table.
    - SELECT BARCODE, COUNT(1) FROM PRODUCTS\_D GROUP BY BARCODE HAVING COUNT(1) > 1 --184 NUMERIC BARCODES ARE IN THERE/DUPLICATED, ALSO 4025 EMPTY BARCODES
* ----------------------------- TRANSACTIONS TABLE ------------------------------------
  + Some issues centered around the final\_quantity and final\_sale columns. Specifically, there were many instances where either one or the other would be null. I was able to merge some of the records using combination of barcode/receipt, and user/store/date, but there are still null records left. One assumption I did make based on looking at the differences is that these records are all legitimate, but they need to be merged. There were the exact same number of records with missing quantities and those with missing sales, which leads me to believe a better merge is necessary. Adding additional fields to the users table and maybe some additional product details could help with this endeavor.

--JOIN THE SPLIT RECORDS BASED ON RECEIPT ID AND BARCODE-------------------------

* select a.receipt\_id, a.purchase\_date, a.scan\_date, a.store\_name, a.user\_id, a.barcode, a.final\_quantity, b.final\_sale

from transactions\_ev a inner join transactions\_ev b ON a.receipt\_id = b.receipt\_id and a.barcode = b.barcode

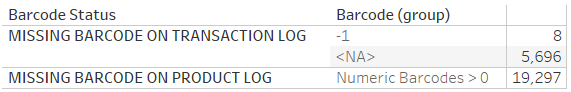
where a.final\_sale is null and b.final\_quantity is null

--HERES ANOTHER ONE I FOUND THAT WORKED THAT I'M STILL RELATIVELY CONFIDENT IN (SAME STORE, SAME USER, SAME DATE)

* select a.receipt\_id, a.purchase\_date, a.scan\_date, a.store\_name, a.user\_id, a.barcode, a.final\_quantity, b.final\_sale

from transactions\_ev a inner join transactions\_ev b ON a.user\_id = b.user\_id and a.store\_name = b.store\_name and a.purchase\_date = b.purchase\_date

where a.final\_sale is null and b.final\_quantity is null and a.receipt\_id != b.receipt\_id order by a.user\_id, a.store\_name

* + - Select count(1) from transactions\_ev where final\_quantity is null --12500
    - select count(1) from transactions\_ev where final\_sale is null –12500
  + There was also an issue with the transaction log specifically where it had a number of missing barcodes
    - 
    - SELECT BARCODE, COUNT(1) FROM TRANSACTIONS\_EV WHERE BARCODE < 0 GROUP BY BARCODE
* Are there any fields that are challenging to understand?
  + For me the fields weren’t as challenging to understand as the logging method especially as it relates to the transaction table. For a while I thought that a large number of those records were either bad or duplicated. After spending some time looking through it, I’m more inclined to think that they are in need of a merge. I think the datasets could use some standardization in terms of field types and think that the tables would be more useful with additional elements.

**We recommend using SQL or python and data visualization to examine the data.**

**Second: provide SQL queries**

**Answer three of the following questions with at least one question coming from the closed-ended and one from the open-ended question set. Each question should be answered using one query.**

**Closed-ended questions:**

* What are the top 5 brands by receipts scanned among users 21 and over?
* What are the top 5 brands by sales among users that have had their account for at least six months?
* What is the percentage of sales in the Health & Wellness category by generation?
  + My first step here was to define what years the generations are classified into. For this I googled it to get a range of years for each generation.
* SELECT CASE WHEN YEAR(b.birth\_date) BETWEEN 1901 AND 1924 THEN 'The Greatest Generation'

WHEN YEAR(b.birth\_date) BETWEEN 1925 AND 1945 THEN 'The Silent Generation'

WHEN YEAR(b.birth\_date) BETWEEN 1946 AND 1964 THEN 'Baby Boomers'

WHEN YEAR(b.birth\_date) BETWEEN 1965 AND 1980 THEN 'Generation X'

WHEN YEAR(b.birth\_date) BETWEEN 1981 AND 1996 THEN 'Millennials'

WHEN YEAR(b.birth\_date) BETWEEN 1997 AND 2012 THEN 'Generation Z'

WHEN YEAR(b.birth\_date) BETWEEN 2013 AND 2022 THEN 'Generation Alpha'

WHEN YEAR(b.birth\_date) >= 2025 THEN 'Generation Beta'

ELSE 'Unknown Generation'

END AS Generation,

(SUM(a.final\_sale) / SUM(SUM(a.final\_sale)) OVER ()) \* 100 AS Perc\_of\_Sales

FROM transactions\_ev a JOIN users\_d b ON a.user\_id = b.id JOIN products\_d c ON a.barcode = c.barcode

WHERE c.category\_1 = 'Health & Wellness' AND a.final\_sale IS NOT NULL

GROUP BY CASE WHEN YEAR(b.birth\_date) BETWEEN 1901 AND 1924 THEN 'The Greatest Generation'

WHEN YEAR(b.birth\_date) BETWEEN 1925 AND 1945 THEN 'The Silent Generation'

WHEN YEAR(b.birth\_date) BETWEEN 1946 AND 1964 THEN 'Baby Boomers'

WHEN YEAR(b.birth\_date) BETWEEN 1965 AND 1980 THEN 'Generation X'

WHEN YEAR(b.birth\_date) BETWEEN 1981 AND 1996 THEN 'Millennials'

WHEN YEAR(b.birth\_date) BETWEEN 1997 AND 2012 THEN 'Generation Z'

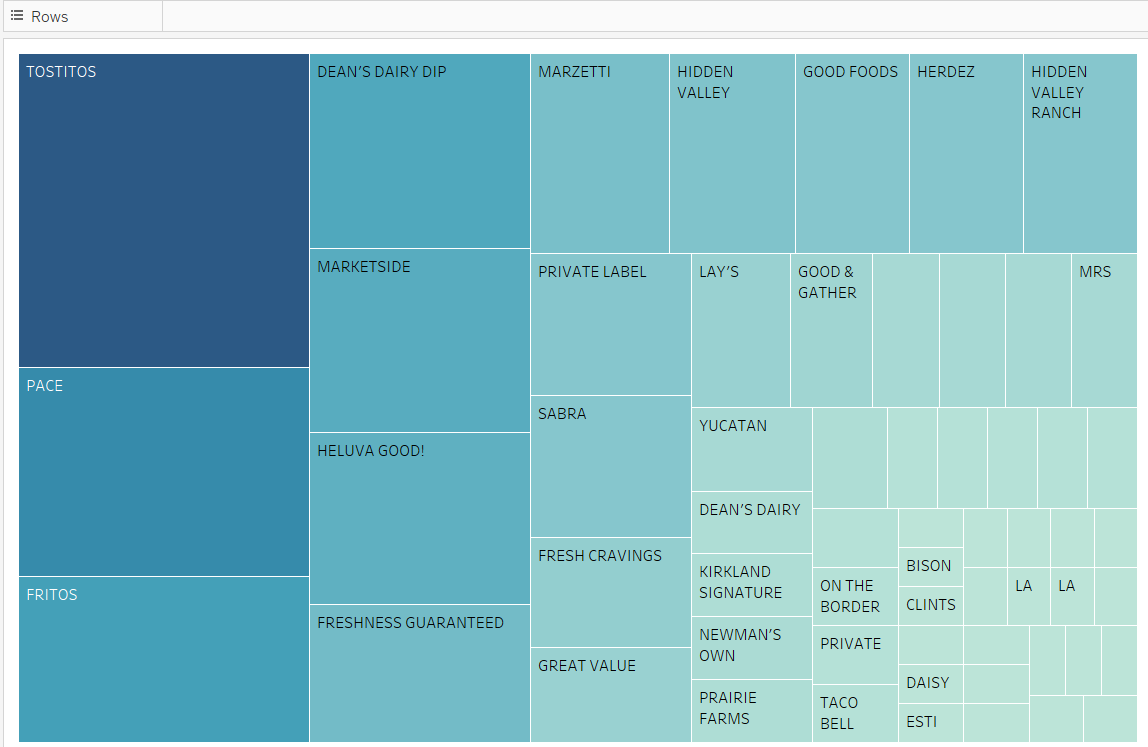
WHEN YEAR(b.birth\_date) BETWEEN 2013 AND 2022 THEN 'Generation Alpha'

WHEN YEAR(b.birth\_date) >= 2025 THEN 'Generation Beta'

ELSE 'Unknown Generation' END ORDER BY Perc\_of\_Sales DESC;

**Open-ended questions: for these, make assumptions and clearly state them when answering the question.**

* Who are Fetch’s power users?
* Which is the leading brand in the Dips & Salsa category?
  + Tostitos is the leader in the dataset. It is worth noting that since the product log is incomplete (see image 2 below), and brand name comes from Product Log, this result could be skewed



* SELECT c.category\_2, Brand, COUNT(a.receipt\_id) AS transaction\_count

FROM transactions\_ev a

INNER JOIN products\_d c ON a.barcode = c.barcode

WHERE c.category\_2 = 'Dips & Salsa' AND a.final\_sale IS NOT NULL and brand is not null

GROUP BY c.category\_2, brand ORDER BY transaction\_count DESC;

* At what percent has Fetch grown year over year?
  + For this specific question I have to assume you are referring to user growth as opposed to transaction growth, as the transaction table only contains roughly 3 months of data from 2024. I also assume you are looking for growth in a full 12 month period, not just since the beginning of 2024 which would only be a 9 month growth rate. Based on those assumptions, user growth has increased by 18.15% in the last 12 months.
  + SELECT total\_users, last\_year\_total\_users,

CAST((total\_users - last\_year\_total\_users) as decimal(18,2))/ last\_year\_total\_users \* 100 as growth\_last\_12\_months

FROM(

SELECT

COUNT(DISTINCT CASE

WHEN CAST(t.created\_date AS DATETIMEOFFSET) < d.dt\_id

THEN t.id

ELSE NULL

END) AS last\_year\_total\_users,

COUNT(DISTINCT t.id) AS total\_users

FROM t\_users t

CROSS JOIN (

SELECT DATEADD(DAY, -365, MAX(CAST(created\_date AS DATETIMEOFFSET))) AS dt\_id

FROM t\_users

) d) e;

**Third: communicate with stakeholders**

**Construct an email or slack message that is understandable to a product or business leader who is not familiar with your day-to-day work. Summarize the results of your investigation. Include:**

* Key data quality issues and outstanding questions about the data
* One interesting trend in the data
  + Use a finding from part 2 or come up with a new insight
* Request for action: explain what additional help, info, etc. you need to make sense of the data and resolve any outstanding issues

--------------------------- EMAIL -------------------------------------------------------------------------------------

Hey Mark/Katie,

I’ve had a chance to dig through the data sets you sent me on Wednesday and wanted to share what I’ve found so far. Here’s a summary of the data quality issues I encountered, some interesting trends I noticed, and a couple of questions I still have.

* **Missing Data**: When I compare the 3 tables to one another I notice there are a lot of records missing from both the Users and the Product Tables. This is probably my biggest concern at the moment. For products, a little less than half of the barcodes in the transaction log have corresponding products. For users, the story is actually a lot worse. Currently we show less than 1% of users in the user log. I’m thinking/hoping that I have just a subset of data, and would like to request access to the full data set. Can you help with that?
* **Null Values in Transaction Data**: The transactions log has numerous records where either the final\_quantity or final\_sale columns are null. While I’ve been able to merge some of these records by matching barcode, receipt, and user/store/date, there are still many that need cleaning. Based on what I’m seeing, I think that the problem is potentially that the records are being split for some reason, but I need to talk to someone in Engineering to make sure they aren’t pulling in any bad data from source and to make sure that the issue isn’t just related to the exports I’ve received.
* **Other issues**: I’m seeing some other smaller problems within the data like duplicate records within the products log and some odd birthdays in the users log. These aren’t quite as big of deal, but still need to understand them more and hopefully get this data fixed.
* **Some good news**: So, it’s not all bad news. Based on some initial findings with the current user table it seems like we’ve had some good user growth in the 12 months leading up to September. We have over 18% growth during that time period, and I’m thinking that it might look even better once we have the full user logs to analyze.

**In terms of next steps, I’d like to propose we schedule a meeting with someone from the DBA team and the Engineering team as soon as possible.** It would be helpful to confirm record counts on each of the datasets, and I’d like to better understand if there’s any additional data being logged that could support our analysis. I’ll go ahead and put something on the calendar in the next couple of days and will send out invites shortly.

**In the meantime, could you let me know if there are any Confluence pages, documentation, or internal resources related to the data pipelines or point me to the right contact person?** This will help me get a better picture of the data flow and help expedite resolving some of the issues we’ve encountered.

Please let me know if you have any questions or if there's anything specific you’d like me to focus on during these meetings.

Thanks for your help!

*--Signature would go here if I weren’t posting to GIT*