**Stakeholder Questions**

1. What questions do you have about the data?
   1. Is there any difference between pointsPayer and rewardsProductPartner?
   2. How do we differentiate between points and bonus points? What is the criteria?
   3. Importance of User Flagged fields?
   4. The Relation between brand, category and CPG?
   5. What does modify date signify wrt the points and status?
   6. What does target price mean?
2. How did you discover the data quality issues?
   1. I applied basic data modeling rules and data assertions to check data quality,
   2. While checking for unique barcodes, I found that there are brand items sharing one barcode.
   3. While checking for the users who have receipts associated with them, I found that some users are not in the user database but have a receipt with reward points.
   4. While calculating total spend on each item, I found that final price and item price have same values irrespective of the quantity purchased.
   5. While looking for top 5 brands, I found that the brand data only had barcodes of 5111 series, while there are receipts with different barcodes
3. What do you need to know to resolve the data quality issues?
   1. For the brands with same barcodes, which ones have the different barcode.
   2. Why are the barcodes other than the 5111 series missing?
   3. For the users not in user database, Were the users delete after the transaction or the data was never captured for those users?
   4. Which is the right amount, item price or final price?
   5. discountedItemPrice is same as itemPrice, why?
4. What other information would you need to help you optimize the data assets you're trying to create?
   1. Importance of some fields such as metaBriteCampaignID?
   2. BrandCode and categoryCode required?
5. What performance and scaling concerns do you anticipate in production and how do you plan to address them?
   1. Increase Data Consistency workload - by Normalizing data probably upto 3N form so as to not affect performance much.
   2. transient data such as last login - Developing a deletion plan, to systematically delete login data.
   3. Latency Increase as more and more users join - Query optimization that reduces join queries, breaks a big query into multiple smaller queries, and adds their results up in the application layer.
   4. Speed issue due to scaling - Database indexing and partitioning