Subject: Data Quality Findings and Next Steps for Fetch Rewards Data Warehouse

Hi [Stakeholder's Name],

I hope you're doing well. As part of our efforts to enhance the accuracy and reliability of the Fetch Rewards data warehouse, we've conducted an initial data quality assessment across the `Users`, `Receipts`, and `Brands` datasets. Below are our findings, outstanding questions, and next steps to ensure data integrity and performance.

Key Findings:

1. **Users Data:**
    - Missing values in `state` and `signUpSource` fields.
    - Duplicate `user_id` entries identified.
    - Inconsistent date formats in `createdDate` and `lastLogin` fields.
2. **Receipts Data:**
    - Some receipts are missing critical fields such as `totalSpent` and `purchaseDate`.
    - Data type inconsistencies (e.g., numeric values stored as strings).
    - No detected duplicate receipt IDs.
3. **Brands Data:**
    - Missing `brandCode` values impacting reporting.
    - Some `topBrand` values stored as strings instead of booleans.
    - No duplicate `brand_id` entries found.

Outstanding Questions:

- Are the missing values in `state` and `brandCode` expected, or should we apply default values?
- How should duplicate user records be handled? Should we prioritize the most recent record or apply deduplication logic?
- For inconsistent date formats, is there a preferred standard format that stakeholders would like to enforce?

Resolution Needs:

To address the identified data quality issues, we require the following:

- Business validation on the treatment of missing data fields.
- Clarification on deduplication strategy for the `Users` dataset.
- Guidelines on expected data formats and validation rules moving forward.

Additional Information Needed:

To optimize our data assets, it would be helpful to gather insights on:

- Expected data update frequency and volume to optimize indexing and storage.
- Reporting requirements to prioritize critical data fields.
- Any upcoming changes in data sources that could impact ingestion workflows.

Performance and Scaling Considerations:

Looking ahead to production deployment, we anticipate the following concerns and mitigation strategies:

1. **Data Growth:** Implementing partitioning strategies and indexing key columns to ensure query performance.
2. **Concurrency:** Optimizing queries and leveraging caching mechanisms to handle high concurrent access.
3. **Monitoring:** Establishing alerts and automated checks to detect data inconsistencies early.

Please let us know your thoughts on the above findings and the required next steps. We appreciate your input and look forward to collaborating on improving data quality.

Best Regards,
Justin Warren