

## 1. Design Notes:

1. Three primary tables derived from original json objects: **brand\_table**, **receipt\_table**, and **user\_table** (as per the Relational Diagram).
2. An additional **item\_receipt** table is derived from **receipt\_table**, containing **receipt\_id**, **brand\_code**, and normalized fields from rewardsReceiptitemList column. This enables joins between receipt\_table and brand\_table for a relational data model.

Available Tables :

- receipt
- brand
- user
- Item\_receipt

## 3. SQL Dialect: MYSQL

## 2 SQL Queries

### 2.1 What are the top 5 brands by receipts scanned for the most recent month?

**\*\* Assuming most recent month = Month to Date**

```
SELECT
    brandcode,
    Count(*) AS receipt_count
FROM
    item_receipt src
JOIN
    receipt_table tgt ON src._id = tgt.receipt_id
WHERE
    tgt.dateScanned > LAST_DAY(CURDATE() - INTERVAL 1 MONTH) + INTERVAL 1 DAY
GROUP BY
    brandcode
ORDER BY
    receipt_count DESC
LIMIT 5;
```

2.2 How does the ranking of the top 5 brands by receipts scanned for the recent month compare to the ranking for the previous month?

```
WITH recentmonth
AS
(
    SELECT      brandcode,
                count(*)                               AS receipt_count,
                rank() over (ORDER BY count(*) DESC) AS ranking
    FROM        item_receipt src
    JOIN        receipt_table tgt
    ON          src.receipt_id = tgt._id
    WHERE       tgt.dateScanned > LAST_DAY(CURDATE() - INTERVAL 1 MONTH)
+ INTERVAL 1 DAY
    AND tgt.dateScanned < CURDATE()
    GROUP BY    brandcode
    ORDER BY    receipt_count DESC
    LIMIT       5 ),

previousmonth
AS
(
    SELECT      brandcode,
                count(*)                               AS receipt_count,
                rank() over (ORDER BY count(*) DESC) AS ranking
    FROM        item_receipt src
    JOIN        receipt_table tgt
    ON          src.receipt_id = tgt._id
    WHERE       tgt.dateScanned > LAST_DAY(CURDATE() - INTERVAL 2 MONTH)
    AND         tgt.dateScanned <= LAST_DAY(CURDATE() - INTERVAL 1 MONTH)
    GROUP BY    brandcode
    ORDER BY    receipt_count DESC) --No limit here to get correct join
with previous month

SELECT      recentmonth.brandcode
AS brand,
           recentmonth.ranking
AS recent_ranking,
           previousmonth.ranking
AS previous_ranking,

coalesce(recentmonth.ranking,0)-coalesce(previousmonth.ranking,0) AS
ranking_change
FROM        recentmonth
LEFT JOIN    previousmonth
ON          recentmonth.brandcode = previousmonth.brandcode
ORDER BY    recent_ranking;
```

2.3 When considering average spend from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?

```
WITH cte
  AS (SELECT rewardsreceiptstatus,
            Avg(totalspent) AS avg_spend
      FROM receipt_table
      WHERE rewardsreceiptstatus IN ( 'Accepted', 'Rejected'
    )
      GROUP BY rewardsreceiptstatus)

SELECT CASE
  WHEN MAX(CASE WHEN rewardsreceiptstatus = 'Accepted' THEN
avg_spend END) >
    MAX(CASE WHEN rewardsreceiptstatus = 'Rejected' THEN
avg_spend END)
  THEN 'Accepted'
  ELSE 'Rejected'
  END AS higher_spend
FROM cte
```

\*\* 'rewardsReceiptStatus' doesn't consist of "Accepted" as a unique status value. Hence, additional business rules might need to be setup to define the "Accepted" status based on existing "FINISHED"/"SUBMITTED" status.

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

```
WITH cte AS (
  SELECT
    rewardsReceiptStatus,
    SUM(item_receipt.purchasedItemCount) AS total_items
  FROM receipt_table
  JOIN item_receipt ON receipt_table._id =
item_receipt.receipt_id
  WHERE rewardsReceiptStatus IN ('Accepted', 'Rejected')
  GROUP BY rewardsReceiptStatus
)
```

```

SELECT
    CASE WHEN MAX(CASE WHEN rewardsreceiptstatus = 'Accepted' THEN
total_items END) >
                MAX(CASE WHEN rewardsreceiptstatus = 'Rejected' THEN
total_items END)

        THEN 'Accepted'
        ELSE 'Rejected'

END
    AS status_with_more_items
FROM cte

```

2.5 Which brand has the most spend among users who were created within the past 6 months?

```

SELECT brandcode,
       Sum(finalprice * quantitypurchased) AS totalSpent
FROM   item_receipt src
       JOIN receipt_table tgt
         ON src.receipt_id = tgt._id
-- joining to get user_id field from receipts table
WHERE  tgt.userid IN (SELECT _id
                      FROM   user_table
                      WHERE  From_unixtime(createdate / 1000) >=
                          Date_sub(Curdate(), INTERVAL 6 month
                          ))

GROUP BY brandcode
ORDER BY totalspent DESC
LIMIT 1

```

2.6 Which brand has the most transactions among users who were created within the past 6 months?

```

SELECT brandcode,
       Count(*) AS transaction_count
FROM   item_receipt src
       JOIN receipt_table tgt
         ON src.receipt_id = tgt._id
WHERE  tgt.userid IN (SELECT _id
                      FROM   user_table
                      WHERE  From_unixtime(createdate / 1000) >
                          Date_sub(Curdate(), INTERVAL 6
month)
                      )

```

```
GROUP BY brandcode  
ORDER BY totalspent DESC  
LIMIT 1;
```