

MYSQL Queries to Answer Business Stakeholder Questions

Question 1: What are the top 5 brands by receipts scanned for the most recent month?

```
SELECT B.name AS 'Brand Name', COUNT (DISTINCT R.receipt_id) AS 'Receipt Count'
FROM Receipts R
JOIN Receipt_Items RI ON R.receipt_id = RI.receipt_id
JOIN Brands B ON RI.brand_id = B.brand_id
WHERE R.date_scanned >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH)
GROUP BY B.name
ORDER BY `Receipt Count` DESC
LIMIT 5;
```

Steps:

1. **Join tables:** Connect the Receipts, Receipt_Items, and Brands tables using receipt and brand IDs.
2. **Filter receipts:** Select only the receipts scanned in the last month.
3. **Count unique receipts:** For each brand, count distinct receipts.
4. **Group by brand:** Group results by brand name.
5. **Order and limit:** Sort by receipt count in descending order and limit the result to the top 5 brands

Example Output:

Brand Name	Receipt Count
Starbucks	25
Doritos	18
Campbell's Home	15
Kraft	12
Pepsi	10

This output represents the top 5 brands based on the number of distinct receipts scanned in the most recent month.

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

```
SELECT R.rewards_receipt_status as 'Rewards Receipt Status', AVG(R.total_spent) AS  
'Average Spend'
```

```
FROM Receipts R
```

```
WHERE R.rewards_receipt_status IN ('Accepted', 'Rejected')
```

```
GROUP BY R.rewards_receipt_status;
```

Steps:

1. **Filter by status:** Select receipts that have a status of either 'Accepted' or 'Rejected'.
2. **Calculate average spend:** Compute the average of the total spend (total_spent) for each status.
3. **Group by status:** Group the results by the receipt status ('Accepted' or 'Rejected').

Example Output:

Rewards Receipt Status	Average Spend
Accepted	\$23.45
Rejected	\$15.32

This output shows that receipts with an 'Accepted' status had a higher average spend compared to those with a 'Rejected' status.

Question 4: When considering total number of items purchased from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?

```
SELECT R.rewards_receipt_status as 'Rewards Receipt Status',  
SUM(R.purchased_item_count) AS 'Total Items Purchased'  
  
FROM Receipts R  
  
WHERE R.rewards_receipt_status IN ('Accepted', 'Rejected')  
  
GROUP BY R.rewards_receipt_status;
```

Steps:

1. **Filter by status:** Select receipts that have a status of either 'Accepted' or 'Rejected'.
2. **Sum item counts:** Calculate the total number of items purchased by summing the purchased_item_count for each status.
3. **Group by status:** Group the results by the receipt status ('Accepted' or 'Rejected').

Example Output:

Rewards Receipt Status	Total Items Purchased
Accepted	150
Rejected	100

This output indicates that more items were purchased from receipts with an 'Accepted' status compared to those with a 'Rejected' status.

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

```
SELECT B.name as 'Brand Name', SUM(RI.final_price) as 'Total Spend'
FROM Users U
JOIN Receipts R ON U.user_id = R.user_id
JOIN Receipt_Items RI ON R.receipt_id = RI.receipt_id
JOIN Brands B ON RI.brand_id = B.brand_id
WHERE U.created_date >= DATE_SUB(CURDATE(), INTERVAL 6 MONTH)
GROUP BY B.name
ORDER BY total_spend DESC
LIMIT 1;
```

Steps:

1. **Join tables:** Join Users, Receipts, Receipt_Items, and Brands tables using the relevant keys (user_id, receipt_id, and brand_id).
2. **Filter users:** Select users who created accounts within the last 6 months.
3. **Calculate total spend:** Sum the final_price of items for each brand.
4. **Group by brand:** Group the results by brand name.
5. **Order and limit:** Sort by total spend in descending order and return the top brand.

Example Output:

Brand Name	Total Spend
Starbucks	\$1500.00

This output shows that Starbucks had the highest total spend among users who created accounts in the past six months.

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

```
SELECT B.name as 'Brand Name', COUNT(DISTINCT R.receipt_id) as 'Transaction Count'
FROM Users U
JOIN Receipts R ON U.user_id = R.user_id
JOIN Receipt_Items RI ON R.receipt_id = RI.receipt_id
JOIN Brands B ON RI.brand_id = B.brand_id
WHERE U.created_date >= DATE_SUB(CURDATE(), INTERVAL 6 MONTH)
GROUP BY B.name
ORDER BY transaction_count DESC
LIMIT 1;
```

Steps:

1. **Join tables:** Join Users, Receipts, Receipt_Items, and Brands tables using the relevant keys (user_id, receipt_id, and brand_id).
2. **Filter users:** Select users who created accounts within the last 6 months.
3. **Count distinct transactions:** Count distinct receipt_id for each brand to get the number of unique transactions.
4. **Group by brand:** Group the results by brand name.
5. **Order and limit:** Sort by transaction count in descending order and return the brand with the most transactions.

Example Output:

Brand Name	Transaction Count
Pepsi	200

This output shows that Pepsi had the highest number of transactions among users created in the past six months.