Second: Write a query that directly answers a predetermined question from a business stakeholder

1) **What are the top 5 brands by receipts scanned for most recent month?**

```
SELECT b.brand_id, b.name, COUNT(DISTINCT r.receipt_id) AS receipt_count
FROM Brand b JOIN Items i ON b.brand_id = i.brand_id
JOIN Receipt_Barcode_User rbu ON i.barcode = rbu.barcode
JOIN Receipts r ON rbu.receipt_id = r.receipt_id
WHERE MONTH(r.dateScanned) = MONTH(CURRENT_DATE) AND YEAR(r.dateScanned) =
YEAR(CURRENT_DATE)
GROUP BY b.brand_id
ORDER BY receipt_count DESC
LIMIT 5;
```

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 CurrentMonthData AS (
   SELECT b.brand_id, b.name, COUNT(DISTINCT r.receipt_id) AS receipt_count
   FROM Brand b
   JOIN Items i ON b.brand_id = i.brand_id
   JOIN Receipt_Barcode_User rbu ON i.barcode = rbu.barcode
   JOIN Receipts r ON rbu.receipt_id = r.receipt_id
   WHERE MONTH(r.dateScanned) = MONTH(CURRENT_DATE) AND YEAR(r.dateScanned)
= YEAR(CURRENT_DATE)
   GROUP BY b.brand_id
   ORDER BY receipt_count DESC
   LIMIT 5
),
PreviousMonthData AS (
   SELECT b.brand_id, b.name, COUNT(DISTINCT r.receipt_id) AS receipt_count
   FROM Brand b
   JOIN Items i ON b.brand_id = i.brand_id
   JOIN Receipt_Barcode_User rbu ON i.barcode = rbu.barcode
   JOIN Receipts r ON rbu.receipt_id = r.receipt_id
   WHERE MONTH(r.dateScanned) = IF(MONTH(CURRENT_DATE) = 1, 12,
MONTH(CURRENT_DATE) - 1)
      AND YEAR(r.dateScanned) = IF(MONTH(CURRENT_DATE) = 1,
YEAR(CURRENT_DATE) - 1, YEAR(CURRENT_DATE))
   GROUP BY b.brand_id
   ORDER BY receipt_count DESC
   LIMIT 5
)
```

```
SELECT
    c.brand_id AS current_month_brand_id,
    c.name AS current_month_brand,
    'Current Month Top 5' AS status
FROM CurrentMonthData c
UNION ALL
SELECT
    p.brand_id AS previous_month_brand_id,
    p.name AS previous_month_brand,
    'Previous Month Top 5' AS status
FROM PreviousMonthData p;
```

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

```
SELECT rewardsReceiptStatus, AVG(totalSpent) AS average_spent
FROM Receipts
WHERE rewardsReceiptStatus IN ('Accepted', 'Rejected')
GROUP BY rewardsReceiptStatus;
```

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

```
SELECT rewardsReceiptStatus, SUM(purchasedItemCount) AS total_items
FROM Receipts
WHERE rewardsReceiptStatus IN ('Accepted', 'Rejected')
GROUP BY rewardsReceiptStatus;
```

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

```
SELECT b.brand_id, b.name, SUM(r.totalSpent) AS total_spent
FROM Users u
JOIN Receipts r ON u.user_id = r.userId
JOIN Receipt_Barcode_User rbu ON r.receipt_id = rbu.receipt_id
JOIN Items i ON rbu.barcode = i.barcode
JOIN Brand b ON i.brand_id = b.brand_id
WHERE u.createdDate >= DATE_SUB(CURRENT_DATE, INTERVAL 6 MONTH)
GROUP BY b.brand_id
ORDER BY total_spent DESC
LIMIT 1;
```

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

```
SELECT b.brand_id, b.name, COUNT(DISTINCT r.receipt_id) AS transaction_count
FROM Users u
JOIN Receipts r ON u.user_id = r.userId
JOIN Receipt_Barcode_User rbu ON r.receipt_id = rbu.receipt_id
JOIN Items i ON rbu.barcode = i.barcode
JOIN Brand b ON i.brand_id = b.brand_id
WHERE u.createdDate >= DATE_SUB(CURRENT_DATE, INTERVAL 6 MONTH)
GROUP BY b.brand_id
ORDER BY transaction_count DESC
LIMIT 1;
```