- *What are the top 5 brands by sales among users that have had their account for at least six months?*

```
WITH Merged_table AS (
    SELECT *
    FROM Users u
    JOIN Transactions t ON u.id = t.user_id
    JOIN Products p ON p.barcode = t.barcode
    WHERE p.barcode IS NOT NULL
        AND t.barcode IS NOT NULL
)

SELECT
    brand,
    SUM(final_sale) AS total_sales
FROM Merged_table
WHERE created_date <= DATE('now', '-6 months')
    AND brand IS NOT NULL
GROUP BY brand
ORDER BY total_sales DESC
LIMIT 5;
```

- *What is the percentage of sales in the Health & Wellness category by generation?*

```
WITH Merged_table AS (
    SELECT *
    FROM Users u
    JOIN Transactions t ON u.id = t.user_id
    JOIN Products p ON p.barcode = t.barcode
    WHERE p.barcode IS NOT NULL
        AND t.barcode IS NOT NULL
),
Category_sale AS (
    SELECT
        CASE
            WHEN CAST(strftime('%Y', birth_date) AS INTEGER) BETWEEN 1997 AND 2012 THEN 'Generation Z'
            WHEN CAST(strftime('%Y', birth_date) AS INTEGER) BETWEEN 1981 AND 1996 THEN 'Millennials'
            WHEN CAST(strftime('%Y', birth_date) AS INTEGER) BETWEEN 1965 AND 1980 THEN 'Generation X'
            WHEN CAST(strftime('%Y', birth_date) AS INTEGER) BETWEEN 1946 AND 1964 THEN 'Baby Boomers'
            ELSE 'Others'
        END AS generation,
        SUM(final_sale) AS total_sale
    FROM Merged_table
    WHERE category_1 = 'Health & Wellness'
    GROUP BY generation
),
All_sale AS (
    SELECT SUM(final_sale) as all_sale
    FROM Merged_table
    WHERE category_1 = 'Health & Wellness'
)

SELECT generation, ROUND(total_sale * 100/(SELECT all_sale FROM All_sale), 2) AS percentage
FROM Category_sale;
```

- *Which is the leading brand in the Dips & Salsa category?*

  *My assumptions: The **leading brand** is identified as the one with the highest total sales. In cases where multiple brands have similar sales figures, the total quantity sold, transaction count, and average sale per transaction are used to further refine the ranking.*

```
SELECT
    p.brand,
    SUM(t.final_sale) AS total_sales,
    SUM(t.final_quantity) AS total_quantity_sold,
    COUNT(DISTINCT t.receipt_id) AS transaction_count,
    SUM(t.final_sale)/COUNT(DISTINCT t.receipt_id) AS average_sale_per_transaction
FROM Transactions t
JOIN Products p ON t.barcode = p.barcode
WHERE p.barcode IS NOT NULL
        AND t.barcode IS NOT NULL
        AND p.category_2 = 'Dips & Salsa'
GROUP BY p.brand
ORDER BY
            total_sales DESC,
            total_quantity_sold DESC,
            transaction_count DESC,
            average_sale_per_transaction DESC
LIMIT 1;
```