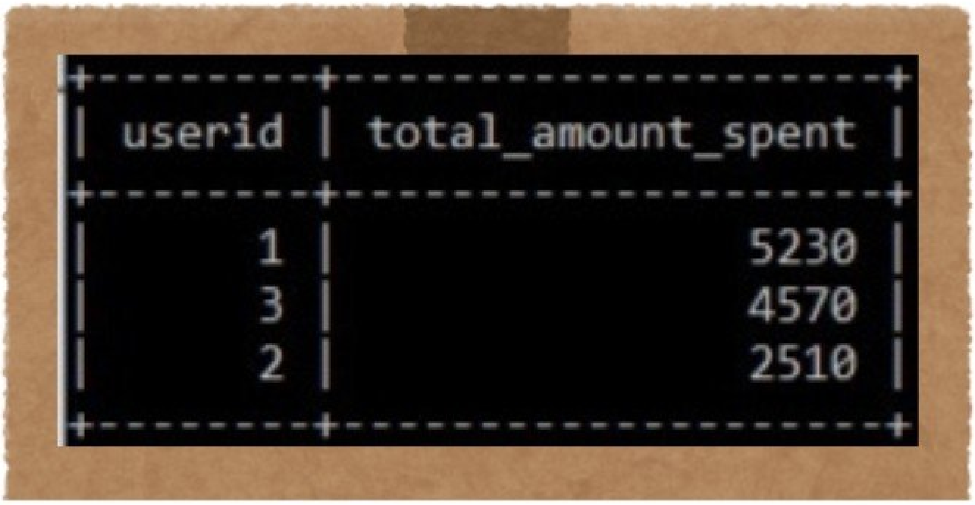
ZOMATO
Data Analysis Portfolio

OBJECTIVE

The objectives of analyzing these queries are to understand customer spending patterns, track purchase frequency, identify initial product preferences, and evaluate product popularity and engagement before and after users became gold members. These insights help in optimizing marketing strategies, improving customer retention, and enhancing overall sales performance.

1. What is the total amount spent by each user on all purchased products?

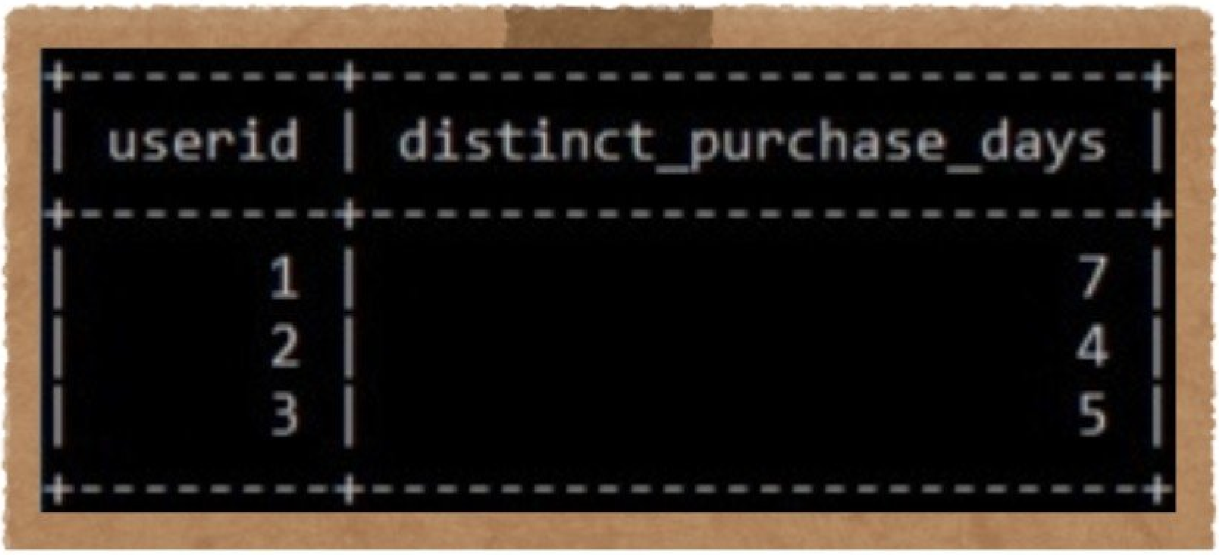
```
SELECT s.userid, SUM(p.price) AS total_amount_spent
FROM sales s
INNER JOIN product p
ON s.product_id = p.product_id
GROUP BY s.userid;
```



```
+--------+--------------------+
| userid | total_amount_spent |
+--------+--------------------+
|      1 |               5230 |
|      3 |               4570 |
|      2 |               2510 |
+--------+--------------------+
```

2. How many distinct days did each user make purchases?

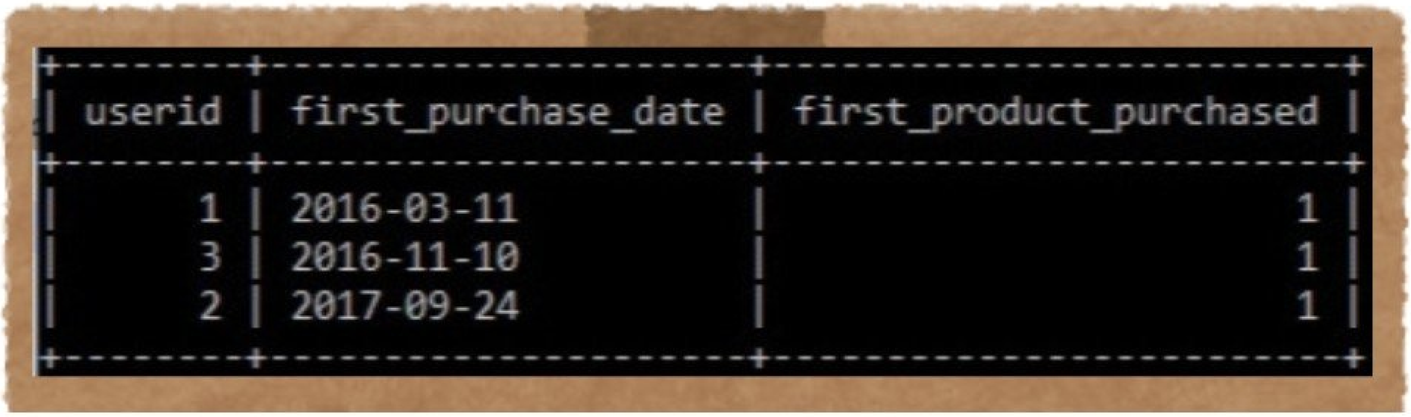
```
SELECT userid, COUNT(DISTINCT created_date) AS
distinct_purchase_days
FROM sales
GROUP BY userid;
```



```
+--------+------------------------+
| userid | distinct_purchase_days |
+--------+------------------------+
|      1 |                      7 |
|      2 |                      4 |
|      3 |                      5 |
+--------+------------------------+
```

3. What was the first product purchased by each user?

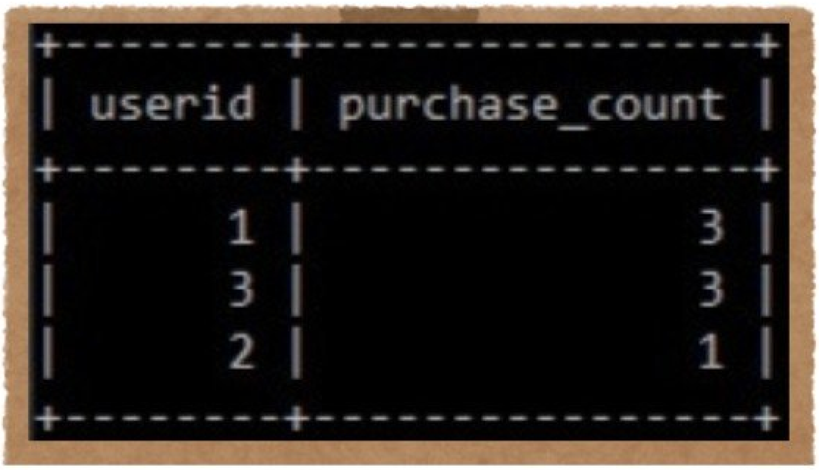
```
SELECT userid, MIN(created_date) AS
first_purchase_date, MIN(product_id) AS
first_product_purchased
FROM sales
GROUP BY userid;
```



```
+--------+---------------------+-------------------------+
| userid | first_purchase_date | first_product_purchased |
+--------+---------------------+-------------------------+
|      1 | 2016-03-11          |                       1 |
|      3 | 2016-11-10          |                       1 |
|      2 | 2017-09-24          |                       1 |
+--------+---------------------+-------------------------+
```

4. For each user, which product did they purchase the most frequently, and how many times did they purchase it?

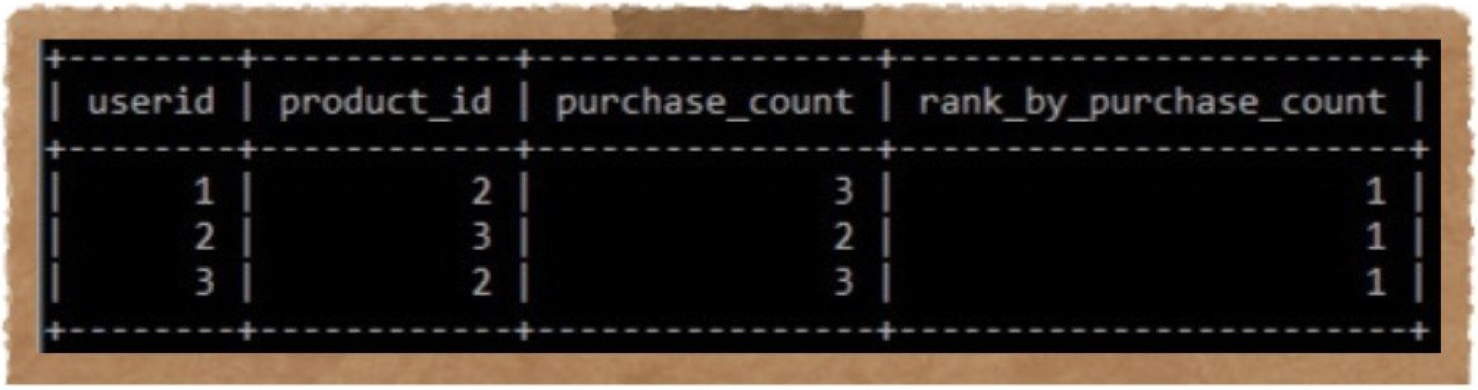
```
SELECT userid, COUNT(product_id) AS
purchase_count
FROM sales
WHERE product_id = (SELECT product_id FROM
sales GROUP BY product_id ORDER BY
COUNT(product_id) DESC LIMIT 1)
GROUP BY userid;
```



```
+--------+----------------+
| userid | purchase_count |
+--------+----------------+
|      1 |              3 |
|      3 |              3 |
|      2 |              1 |
+--------+----------------+
```

5. For each user, which product has the highest purchase frequency, and how many times was it purchased?

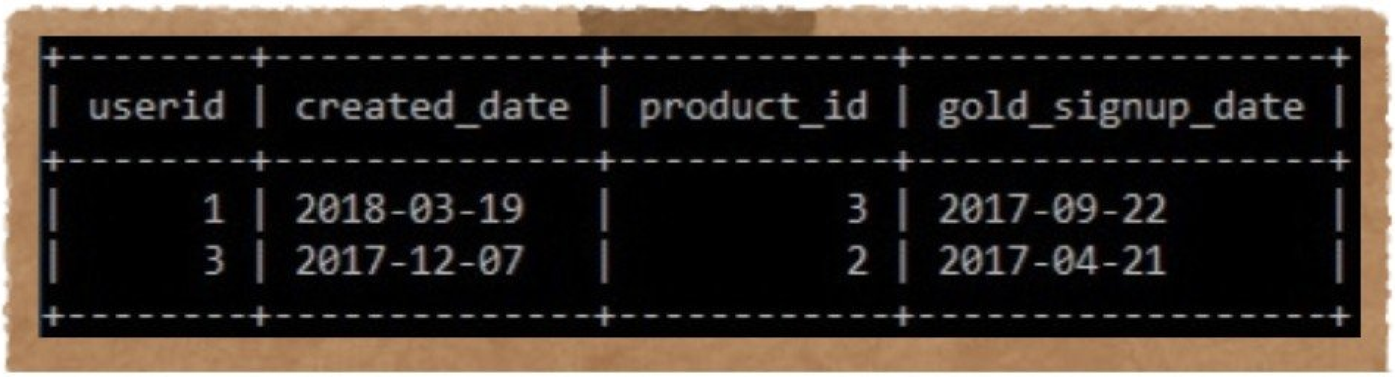
```
SELECT *
FROM (SELECT *, RANK() OVER(PARTITION BY userid
ORDER BY purchase_count DESC) AS
rank_by_purchase_count
FROM (SELECT userid, product_id, COUNT(*) AS
purchase_count
FROM sales GROUP BY userid, product_id) AS
product_purchase_counts) AS ranked_products
WHERE rank_by_purchase_count = 1;
```



```
+--------+------------+----------------+------------------------+
| userid | product_id | purchase_count | rank_by_purchase_count |
+--------+------------+----------------+------------------------+
|      1 |          2 |              3 |                      1 |
|      2 |          3 |              2 |                      1 |
|      3 |          2 |              3 |                      1 |
+--------+------------+----------------+------------------------+
```

6. Find the first product purchased by each user after they became a gold member.

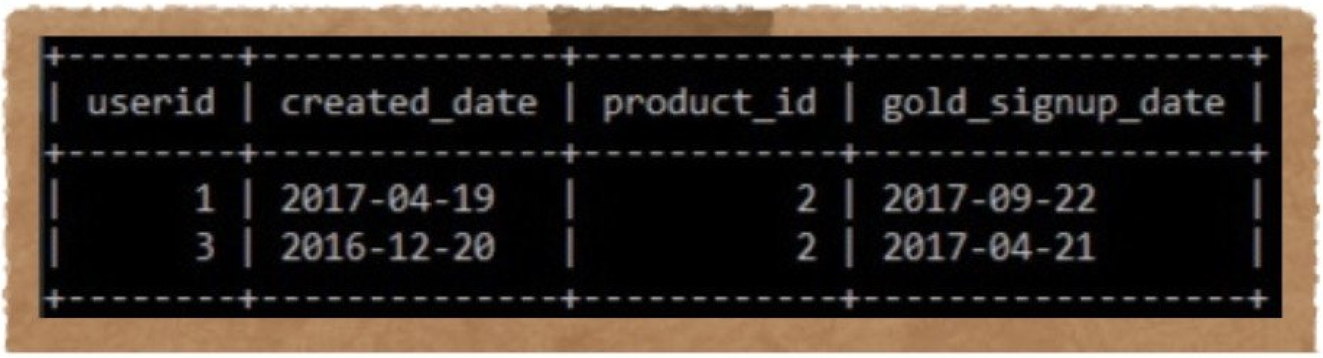
```
SELECT d.userid, d.created_date, d.product_id,
d.gold_signup_date
FROM (SELECT c.*, RANK() OVER(PARTITION BY userid
ORDER BY created_date) AS rnk
FROM (SELECT a.userid, a.created_date, a.product_id,
b.gold_signup_date FROM sales a
INNER JOIN goldusers_signup b ON a.userid = b.userid
WHERE a.created_date >= b.gold_signup_date) c) d
WHERE d.rnk = 1;
```



```
+--------+--------------+------------+------------------+
| userid | created_date | product_id | gold_signup_date |
+--------+--------------+------------+------------------+
|      1 | 2018-03-19   |          3 | 2017-09-22       |
|      3 | 2017-12-07   |          2 | 2017-04-21       |
+--------+--------------+------------+------------------+
```

7. Which were the most recent products purchased by users just before they became gold members, including their respective gold membership signup dates?

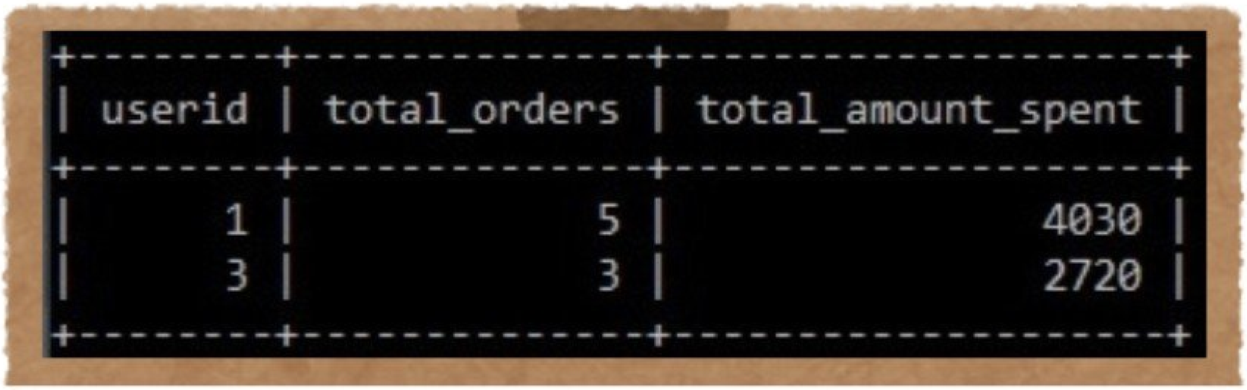
```
SELECT d.userid, d.created_date, d.product_id,
d.gold_signup_date
FROM (SELECT c.*, RANK() OVER(PARTITION BY userid
ORDER BY created_date DESC) AS rnk
FROM (SELECT a.userid, a.created_date, a.product_id,
b.gold_signup_date
FROM sales a INNER JOIN goldusers_signup b ON a.userid
= b.userid
WHERE a.created_date <= b.gold_signup_date) c) d
WHERE d.rnk = 1;
```



```
+--------+--------------+------------+------------------+
| userid | created_date | product_id | gold_signup_date |
+--------+--------------+------------+------------------+
|      1 | 2017-04-19   |          2 | 2017-09-22       |
|      3 | 2016-12-20   |          2 | 2017-04-21       |
+--------+--------------+------------+------------------+
```

8. What is the total number of orders and the total amount spent for each customer before they became a member?

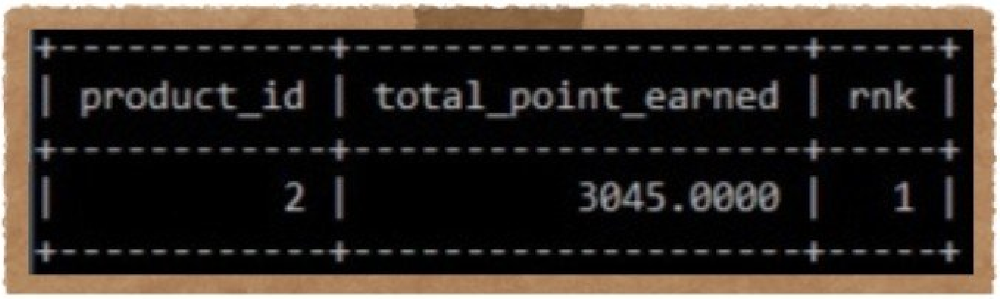
```
SELECT a.userid, COUNT(s.product_id) AS
total_orders, SUM(p.price) AS total_amount_spent
FROM users a LEFT JOIN sales s ON a.userid =
s.userid LEFT JOIN product p ON s.product_id =
p.product_id
WHERE s.created_date < (SELECT gold_signup_date
FROM goldusers_signup b WHERE b.userid =
a.userid)
GROUP BY a.userid;
```



```
+--------+--------------+--------------------+
| userid | total_orders | total_amount_spent |
+--------+--------------+--------------------+
|      1 |            5 |               4030 |
|      3 |            3 |               2720 |
+--------+--------------+--------------------+
```

9. Identify the Top Product by Points Earned, using the following points system: product 1 earns 5 points, product 2 earns 2 points, and product 3 earns 5 points, and rank the products to find the top performer?

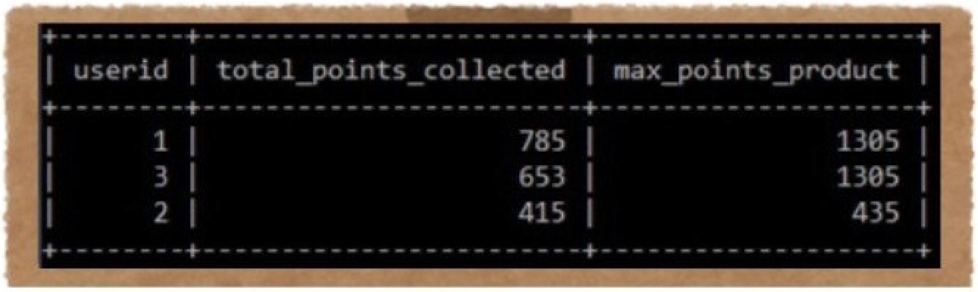
```
SELECT * FROM (SELECT *, RANK() OVER(ORDER BY total_point_earned
DESC) AS rnk
FROM (SELECT product_id, SUM(total_points) AS total_point_earned
FROM (SELECT e.*, amt / points AS total_points
FROM (SELECT d.*, CASE
WHEN product_id = 1 THEN 5
WHEN product_id = 2 THEN 2
WHEN product_id = 3 THEN 5 ELSE 0 END AS points
FROM (SELECT c.userid, c.product_id, SUM(price) AS amt
FROM (SELECT a.*, b.price
FROM sales a INNER JOIN product b ON a.product_id = b.product_id) c
GROUP BY userid, product_id) d) e) f
GROUP BY product_id) g) h
WHERE rnk = 1;
```



```
+------------+--------------------+-----+
| product_id | total_point_earned | rnk |
+------------+--------------------+-----+
|          2 |          3045.0000 |   1 |
+------------+--------------------+-----+
```

10. Calculate Total and Maximum Points for Each User, using the following points system: product 1 earns 5 points, product 2 earns 10 points, and product 3 earns 5 points?

```
SELECT userid, SUM(total_points) AS total_points_collected, MAX(points)
AS max_points_product
FROM (SELECT f.userid, f.product_id, FLOOR(f.total_amt /
f.points_per_zomato) AS total_points, CASE
WHEN f.product_id = 1 THEN FLOOR(f.total_amt / 5)
WHEN f.product_id = 2 THEN FLOOR(f.total_amt / 10) * 5
WHEN f.product_id = 3 THEN FLOOR((f.total_amt - 2) / 5) ELSE 0 END AS
points
FROM (SELECT e.userid, e.product_id, SUM(e.price) AS total_amt, CASE
WHEN e.product_id = 1 THEN 5 WHEN e.product_id = 2 THEN 10 WHEN
e.product_id = 3 THEN 5 ELSE 0 END AS points_per_zomato
FROM (SELECT d.userid, d.product_id, b.price
FROM sales d INNER JOIN product b ON d.product_id = b.product_id) e
GROUP BY e.userid, e.product_id) f) g
GROUP BY userid;
```



```
+--------+------------------------+--------------------+
| userid | total_points_collected | max_points_product |
+--------+------------------------+--------------------+
|      1 |                    785 |               1305 |
|      3 |                    653 |               1305 |
|      2 |                    415 |                435 |
+--------+------------------------+--------------------+
```