SWIGGY
FOOD DELIVERY APP

WsCubetech - Cohort 5

Week 5 - Case Study

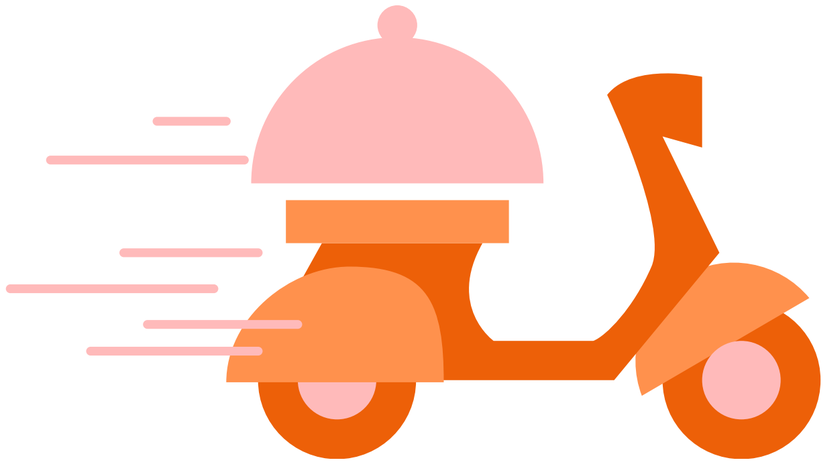
SWIGGY INSIGHTS

Binay Kumar Naik

5 May, 2025

PROJECT OVERVEIW

Swiggy seeks insights from its SQL dataset. Implement sophisticated SQL queries with intricate joins for in-depth analysis and strategic decision-making.



TASKS

1. Display all customers who live in 'Delhi'.
2. Find the average rating of all restaurants in 'Mumbai'.
3. List all customers who have placed at least one order.
4. Display the total number of orders placed by each customer.
5. Find the total revenue generated by each restaurant.
6. Find the top 5 restaurants with the highest average rating.
7. Display all customers who have never placed an order.
8. Find the number of orders placed by each customer in 'Mumbai'.
9. Display all orders placed in the last 30 days.
10. List all delivery partners who have completed more than 1 delivery
11. Find the customers who have placed orders on exactly three different days.
12. Find the delivery partner who has worked with the most different customers.
13. Identify customers who have the same city and have placed orders at the same restaurants, but on different dates.

TASK - 1

1. Display all customers who live in 'Delhi'.

QUERY

```
1 • SELECT
2       name, city
3   FROM
4       customers
5   WHERE
6       city = 'Delhi';
```

OUTPUT

	name	city
▶	Rohini Verma	Delhi
	Manish Kumar	Delhi
	Sonali Mishra	Delhi

- There are only 3 customers who live in Delhi.

TASK - 2

2. Find the average rating of all restaurants in 'Mumbai'.

QUERY

```
3 •  SELECT
4        ROUND(AVG(COALESCE(rating,0)), 2) AS avg_rating
5    FROM
6        restaurants
7    WHERE
8        city = 'Mumbai';
```

OUTPUT

	avg_rating
▶	3.23

- The average rating of all restaurants in 'Mumbai' is **3.23**

TASK - 3

3. List all customers who have placed at least one order.

QUERY

```sql
SELECT
    c.customer_id,
    c.name,
    COUNT(o.order_id) AS total_orders_placed
FROM
    customers c
        JOIN
    orders o ON c.customer_id = o.customer_id
GROUP BY 1 , 2
ORDER BY 1;
```

OUTPUT

customer_id	name	total_orders_placed
1	Amit Sharma	2
2	Rohini Verma	3
3	Rajesh Gupta	3
4	Sneha Mehta	2
5	Manish Kumar	4
6	Priya Singh	3
7	Vikas Reddy	3
8	Anjali Patel	3
9	Suresh Nair	1
10	Kavita Deshmukh	2

- There are **23** customers who have placed at least one order

TASK - 4

4. Display the total number of orders placed by each customer.

QUERY

```
3   SELECT
4       c.customer_id,
5       c.name,
6       COUNT(o.order_id) AS total_orders_placed
7   FROM
8       customers c
9           JOIN
10      orders o ON c.customer_id = o.customer_id
11  GROUP BY 1 , 2
12  ORDER BY 3 desc;
```

OUTPUT

customer_id	name	total_orders_placed
5	Manish Kumar	4
2	Rohini Verma	3
3	Rajesh Gupta	3
6	Priya Singh	3
7	Vikas Reddy	3
8	Anjali Patel	3
14	Nidhi Saxena	3
15	Ashok Kumar	3
18	Sonali Mishra	3
1	Amit Sharma	2

TASK - 5

5. Find the total revenue generated by each restaurant.

QUERY

```
3 •  SELECT
4        r.restaurant_id,
5        r.name AS restaurant_name,
6        COALESCE(SUM(o.total_amount), 0) AS total_revenue
7    FROM
8        restaurants r
9            LEFT JOIN
10       orders o ON r.restaurant_id = o.restaurant_id
11   GROUP BY 1 , 2;
```

OUTPUT

restaurant_id	restaurant_name	total_revenue
1	Spice of India	1100.00
2	Tandoori Flames	1200.00
3	Biryani House	5300.00
4	Curry Pot	3200.00
5	Taste of Punjab	600.00
6	Royal Biryani	650.00
7	Coastal Delight	2100.00
8	Veggie Delight	1600.00
9	Gujarat Express	2550.00
10	Andhra Spice	4050.00
11	Punjabi Tadka	900.00
12	Flavours of Bengal	4050.00
13	South Treat	2950.00

TASK - 6

6. Find the top 5 restaurants with the highest average rating.

QUERY

```sql
SELECT
    name AS restaurant_name,
    ROUND(AVG(COALESCE(rating, 0)), 2) AS ratings
FROM
    restaurants
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

OUTPUT

restaurant_name	ratings
Biryani House	4.80
Paradise Biryani	4.80
Lucknowi Nawabi	4.70
Royal Biryani	4.70
Flavours of Bengal	4.60

TASK - 7

7. Display all customers who have never placed an order.

QUERY

```
3 •  SELECT
4        c.customer_id,
5        c.name AS customer_name,
6        o.order_id AS orders_placed
7    FROM
8        customers c
9            LEFT JOIN
10       orders o ON c.customer_id = o.customer_id
11   WHERE
12       o.order_id IS NULL;
```

OUTPUT

customer_id	customer_name	orders_placed
24	Sonal Kaur	NULL
25	Vivek Malhotra	NULL
26	Divya Iyer	NULL
27	Rakesh Yadav	NULL
28	Mona Sharma	NULL
29	Sudha Pillai	NULL
30	Gaurav Khanna	NULL

TASK - 8

8. Find the number of orders placed by each customer in 'Mumbai'.

QUERY

```sql
SELECT
    c.customer_id,
    c.name AS customer_name,
    c.city AS city,
    COUNT(o.order_id) AS orders_placed
FROM
    customers c
        LEFT JOIN
    orders o ON c.customer_id = o.customer_id
WHERE
    c.city = 'Mumbai'
GROUP BY 1 , 2;
```

OUTPUT

customer_id	customer_name	city	orders_placed
1	Amit Sharma	Mumbai	2
3	Rajesh Gupta	Mumbai	3
19	Arjun Desai	Mumbai	2
23	Ravi Singh	Mumbai	2

TASK - 9

9. Display all orders placed in the last 30 days.

QUERY

```sql
SELECT
    c.customer_id, c.name, o.order_id, o.order_date
FROM
    customers c
        LEFT JOIN
    orders o ON c.customer_id = o.customer_id
WHERE
    o.order_date >= '2024-08-15' - INTERVAL 30 DAY
ORDER BY o.order_date;
```

OUTPUT

customer_id	name	order_id	order_date
22	Neha Kaushik	31	2024-08-11 00:00:00
11	Vivek Bhatt	42	2024-08-11 00:00:00
15	Ashok Kumar	44	2024-08-12 00:00:00
5	Manish Kumar	28	2024-08-12 00:00:00
10	Kavita Deshmukh	32	2024-08-13 00:00:00
18	Sonali Mishra	51	2024-08-13 00:00:00
6	Priya Singh	33	2024-08-14 00:00:00
16	Deepa Rao	52	2024-08-14 00:00:00
19	Arjun Desai	46	2024-08-15 00:00:00

TASK - 10

10. List all delivery partners who have completed more than 1 delivery

QUERY

```sql
SELECT
    dp.partner_id,
    dp.name,
    COUNT(od.partner_id) AS delivery_count
FROM
    deliverypartners dp
        JOIN
    orderdelivery od ON dp.partner_id = od.partner_id
GROUP BY 1 , 2
HAVING delivery_count > 1;
```

OUTPUT

partner_id	name	delivery_count
1	Amit Sharma	2
2	Ravi Kumar	5
3	Priya Patel	3
4	Suresh Reddy	6
5	Anita Desai	4
6	Rajesh Gupta	4
7	Sonia Agarwal	3
8	Vikram Singh	2
9	Sneha Iyer	2

TASK - 11

11. Find the customers who have placed orders on exactly three different days.

QUERY

```sql
SELECT
    c.customer_id,
    c.name,
    COUNT(DISTINCT o.order_date) AS order_days
FROM
    customers c
        LEFT JOIN
    orders o ON c.customer_id = o.customer_id
GROUP BY 1 , 2
HAVING order_days = 3;
```

OUTPUT

customer_id	name	order_days
2	Rohini Verma	3
6	Priya Singh	3
8	Anjali Patel	3
14	Nidhi Saxena	3
15	Ashok Kumar	3
18	Sonali Mishra	3

TASK - 12

12. Find the delivery partner who has worked with the most different customers.

QUERY

```sql
SELECT
    dp.partner_id, dp.name, COUNT(o.customer_id) AS customers
FROM
    deliverypartners dp
        JOIN
    orderdelivery od ON dp.partner_id = od.partner_id
        JOIN
    orders o ON o.order_id = od.order_id
GROUP BY 1 , 2
ORDER BY 3 DESC
LIMIT 1;
```

OUTPUT

partner_id	name	customers
4	Suresh Reddy	6

"Suresh Reddy" has worked with the most different customers

TASK - 13

13. Identify customers who have the same city and have placed orders at the same restaurants, but on different dates.

QUERY

```sql
WITH cust_orders AS (
    SELECT
        c.customer_id,
        c.name          AS customer_name,
        c.city,
        o.order_id,
        o.restaurant_id,
        o.order_date
    FROM customers   c
    JOIN orders o ON o.customer_id
)

SELECT DISTINCT
       co1.customer_id  AS customer1_id,
       co1.customer_name AS customer1_name,
       co2.customer_id  AS customer2_id,
       co2.customer_name AS customer2_name,
       co1.restaurant_id AS restaurant_id
FROM   cust_orders co1
JOIN   cust_orders co2
       ON  co1.city          = co2.city
       AND co1.restaurant_id = co2.restaurant_id
       AND co1.customer_id   <> co2.customer_id
       AND co1.order_date    <> co2.order_date;
```

OUTPUT

customer1_id	customer1_name	customer2_id	customer2_name	restaurant_id
5	Manish Kumar	18	Sonali Mishra	3
18	Sonali Mishra	5	Manish Kumar	3
19	Arjun Desai	23	Ravi Singh	8
23	Ravi Singh	19	Arjun Desai	8

SWIGGY
FOOD DELIVERY APP

THANK YOU

You can check my github link for the detailed projrct:
https://github.com/Binay005X/Swiggy-Sales