SWIGGY

ADVANCED SQL PROJECT

Presented by :- ABHISHEK PANDEY

Contents

- About SWIGGY
- What are 'JOINS'
- Problem Statement
- SQL Query Analysis
- Connect with me

About SWIGGY

SWIGGY founded in India in 2014, is a leading food delivery platform that has transformed dining experiences with its easy-to-use app, broad restaurant selection, and fast delivery service has also expanded into grocery deliveries, continually evolving in the competitive market.

Swiggy is headquartered in Bangalore and operates in more than 580 Indian cities, as of July 2023.

Besides food delivery, the platform also provides quick commerce services under the name **Swiggy** Instamart, and same-day package deliveries with **Swiggy** Genie.

What are 'JOINS'

A JOIN is used to combine data from two or more tables, based on a related column between them. In SQL, a JOIN is used to combine rows from two or more tables based on a related column between them. Joins are essential for querying relational databases where data is often distributed across multiple tables.

Here's a rundown of the main types of SQL joins:

- **INNER JOIN** : Returns only matching rows.
- **LEFT JOIN** : Returns all left table rows, matching or not.
- **RIGHT JOIN** : Returns all right table rows, matching or not.
- **FULL JOIN** : Returns rows when there’s matching or not.
- **CROSS JOIN** : Combines all rows from both tables.
- **SELF JOIN** : Joins a table with itself.

Problem Statement:

- Display all customers who live in 'Delhi'.
- Find the average rating of all restaurants in 'Mumbai'.
- List all customers who have placed at least one order.
- Display the total number of orders placed by each customer.
- Find the total revenue generated by each restaurant.
- Find the top 5 restaurants with the highest average rating.
- Display all customers who have never placed an order.

- Find the number of orders placed by each customer in 'Mumbai'.
- Display all orders placed in the last 30 days.
- List all delivery partners who have completed more than 1 delivery.
- Find the customers who have placed orders on exactly three different days.
- Find the delivery partner who has worked with the most different customers.
- Identify customers who have the same city and have placed orders at the same restaurants, but on different dates.

SWIGGY

SQL QUERY ANALYSIS

Display all customers who live in 'Delhi'.

```sql
SELECT
    customer_id, name as Customer_Name
FROM
    customers
WHERE
    city = 'Delhi';
```

Result Grid | Filter Rows:

customer_id	Customer_Name
2	Rohini Verma
5	Manish Kumar
18	Sonali Mishra
NULL	NULL

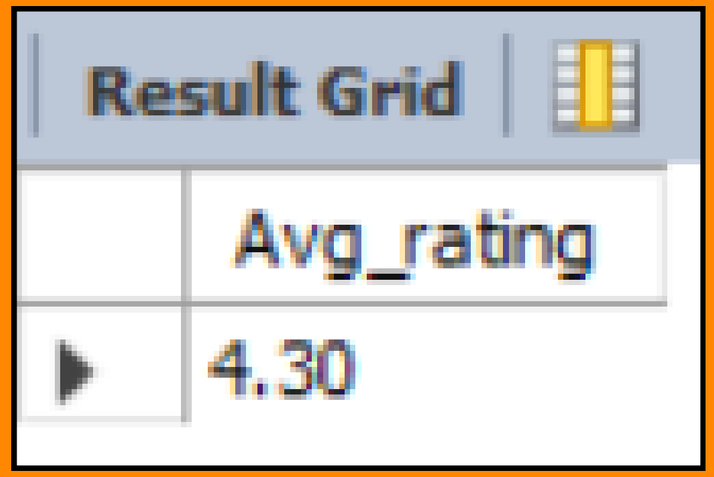
PURPOSE

The purpose of this query is to retrieve the information of the customers of a specific area to make useful decision in accordance to that.

SWIGGY

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

```sql
SELECT
    ROUND(AVG(rating), 2) AS Avg_rating
FROM
    restaurants
WHERE
    city = 'Mumbai';
```



PURPOSE

The purpose of this query is to retrieve the average rating of all the restaurants of "Mumbai" and to have an overall rating of the restaurants.

SWIGGY

List all customers who have placed at least one order.

```sql
SELECT DISTINCT
    customers.customer_id,
    customers.name,
    COUNT(orders.order_id) AS no_of_orders_placed
FROM
    customers
        INNER JOIN
    orders ON customers.customer_id = orders.customer_id
GROUP BY customers.customer_id , customers.name;
```

Result Grid | Filter Rows: | Export:

customer_id	name	no_of_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

PURPOSE

The purpose of this query is to retrieve the information of the customers who have placed at least one order and to make rightful decision in accordance like forming sales boosting strategies.

SWIGGY

Display the total number of orders placed by each customer.

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.order_id) AS no_of_order_placed
FROM
    customers
        LEFT JOIN
    orders ON customers.customer_id = orders.customer_id
GROUP BY customers.customer_id , customers.name;
```

Result Grid | Filter Rows: | Expor

customer_id	name	no_of_order_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

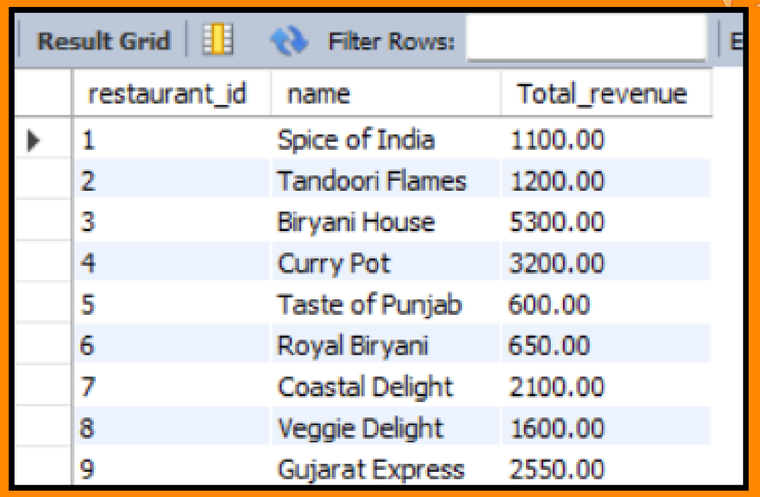
SWIGGY

PURPOSE

The purpose of this query is to retrieve the information regarding total number of order placed by each customer and to make rightful decision in accordance to that, like giving discount on next order or giving coupons.

Find the total revenue generated by each restaurant.

```sql
SELECT
    restaurants.restaurant_id,
    restaurants.name,
    COALESCE(SUM(orders.total_amount), 00) AS Total_revenue
FROM
    restaurants
        LEFT JOIN
    orders ON restaurants.restaurant_id = orders.restaurant_id
GROUP BY restaurants.restaurant_id , restaurants.name;
```



Result Grid | Filter Rows:

restaurant_id	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

SWIGGY

PURPOSE

The purpose of this query is to retrieve the revenue generated by each restaurants in order to compare the sales and popularity between multiple restaurants .

Find the top 5 restaurants with the highest average rating.

```sql
SELECT
    restaurants.restaurant_id,
    restaurants.name,
    ROUND(AVG(restaurants.rating),2) AS Avg_rating
FROM
    restaurants
GROUP BY restaurants.restaurant_id , restaurants.name
ORDER BY AVG(restaurants.rating) DESC
LIMIT 5;
```

Result Grid | Filter Rows:

restaurant_id	name	Avg_rating
3	Biryani House	4.80
22	Paradise Biryani	4.80
30	Lucknowi Nawabi	4.70
6	Royal Biryani	4.70
12	Flavours of Bengal	4.60

SWIGGY

PURPOSE

The purpose of this query is to retrieve the information of top restaurants with highest average ratings to get better search results for good experience of restaurants.

Display all customers who have never placed an order.

```sql
SELECT
    customers.customer_id, customers.name
FROM
    customers
        LEFT JOIN
    orders ON customers.customer_id = orders.customer_id
WHERE
    orders.customer_id IS NULL;
```

Result Grid | Filter Rows:

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

PURPOSE

The purpose of this query is to retrieve the information of the customers who have never placed an order and make some better strategies for more interaction.

SWIGGY

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

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.order_id) AS Orders_placed
FROM
    customers
        LEFT JOIN
    orders ON customers.customer_id = orders.customer_id
WHERE
    city = 'Mumbai'
GROUP BY customers.customer_id , customers.name;
```

Result Grid | Filter Rows:

customer_id	name	Orders_placed
1	Amit Sharma	2
3	Rajesh Gupta	3
19	Arjun Desai	2
23	Ravi Singh	2

PURPOSE

The purpose of this query is to retrieve the information of number of orders placed by each customers of "MUMBAI" to make a region specific algorithm and make decision in accordance to that.

SWIGGY

Display all orders placed in the last 30 days.

```sql
SELECT
    *
FROM
    orders
WHERE
    order_date >= DATE_SUB(CURDATE(), INTERVAL 30 DAY);
```

PURPOSE

The purpose of this query is to retrieve the information of all the orders placed in a specific period to review and analyze it for improvement.

SWIGGY

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

```sql
SELECT
    deliverypartners.partner_id,
    deliverypartners.name,
    COUNT(deliveryupdates.order_id) AS Count_of_orders
FROM
    deliverypartners
        JOIN
    orderdelivery ON deliverypartners.partner_id = orderdelivery.partner_id
        JOIN
    deliveryupdates ON orderdelivery.order_delivery_id = deliveryupdates.delivery_id
WHERE
    deliveryupdates.status = 'Delivered'
GROUP BY deliverypartners.partner_id , deliverypartners.name;
```

Result Grid | Filter Rows:

partner_id	name	Count_of_orders
4	Suresh Reddy	3
3	Priya Patel	1
5	Anita Desai	2
7	Sonia Agarwal	1
2	Ravi Kumar	2
12	Reena Rao	2
18	Meera Gupta	1
10	Kiran Mehta	1

PURPOSE

The purpose of this query is to retrieve the information of the delivery partners who have completed more than 1 delivery to make better incentive plans and provide bonuses.

SWIGGY

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

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.order_date) AS Order_date
FROM
    customers
        JOIN
    orders ON customers.customer_id = orders.customer_id
GROUP BY customers.customer_id , customers.name
HAVING COUNT(DISTINCT orders.order_date) = 3;
```

Result Grid | Filter Rows:

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

PURPOSE

The purpose of this query is to retrieve the information of the customers who have placed order on three different dates to understand the customer's prefered time and choices of order for improvement.

SWIGGY

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

```sql
SELECT
    deliverypartners.partner_id,
    deliverypartners.name,
    COUNT(DISTINCT orders.customer_id) AS Count_of_customers
FROM
    deliverypartners
        JOIN
    orderdelivery ON deliverypartners.partner_id = orderdelivery.partner_id
        JOIN
    orders ON orderdelivery.order_id = orders.order_id
GROUP BY deliverypartners.partner_id , deliverypartners.name
ORDER BY COUNT(DISTINCT orders.customer_id) DESC
LIMIT 1;
```

Result Grid | Filter Rows:

partner_id	name	Count_of_customers
4	Suresh Reddy	6

PURPOSE

The purpose of this query is to retrieve the information of the Delivery partner who have worked with most different customer to motivate them with monetary or non-monetary rewards for their dynamism.

SWIGGY

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

```sql
SELECT
    c1.name AS Customer1,
    c2.name AS Customer2,
    c1.city AS City1,
    c2.city AS City2,
    restaurants.name
FROM
    customers AS c1
        JOIN
    orders AS o1 ON c1.customer_id = o1.customer_id
        JOIN
    orders AS o2 ON o2.restaurant_id = o1.restaurant_id
        JOIN
    customers AS c2 ON c1.city = c2.city AND c1.name <> c2.name
        AND o2.customer_id = c2.customer_id
        JOIN
    restaurants ON o1.restaurant_id = restaurants.restaurant_id
WHERE
    o1.order_date <> o2.order_date;
```

Result Grid | Filter Rows: | Export:

Customer1	Customer2	City1	City2	name
Manish Kumar	Sonali Mishra	Delhi	Delhi	Biryani House
Sonali Mishra	Manish Kumar	Delhi	Delhi	Biryani House
Sonali Mishra	Manish Kumar	Delhi	Delhi	Biryani House
Arjun Desai	Ravi Singh	Mumbai	Mumbai	Veggie Delight
Manish Kumar	Sonali Mishra	Delhi	Delhi	Biryani House
Ravi Singh	Arjun Desai	Mumbai	Mumbai	Veggie Delight

PURPOSE

The purpose of this query is to retrieve the information of the customers who are from same city and have placed order at same restaurants on different dates to understand the complex algorithms for improvement of the model.

SWIGGY

SWIGGY

Thank you

Connect with me

- **GMAIL**
  abhishekp1322@gmail.com

- **LinkedIn**
  Abhishek Pandey