SWIGGY

Advanced SQL Project

Presented By : Deepak Sharma

SWIGGY

About Swiggy

Swiggy is one of India's leading on-demand delivery platforms, founded in 2014 with a mission to bring unmatched convenience to customers. It allows users to order food from thousands of restaurants through a smooth and user-friendly app experience. Swiggy is known for its quick delivery, real-time tracking, and reliable service, making it one of the most preferred food delivery options in the country.

Over the years, Swiggy has expanded beyond food delivery by introducing services like Instamart for instant groceries and Swiggy Genie for pick-up and drop needs. With a strong network of delivery partners and a focus on technology-driven operations, Swiggy continues to transform how India orders food and essentials.

SWIGGY

PROJECT INTRODUCTION

This project analyses Swiggy's SQL dataset to uncover insights into customer behaviour, restaurant performance, and delivery partner efficiency. The analysis highlights key trends and operational gaps, enabling Swiggy to optimize services, improve decision-making, and enhance overall customer experience. It also provides data-driven recommendations that can support strategic planning and future business growth.

SWIGGY

Display all customers who live in 'Delhi'

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

SWIGGY

Find the average rating of all restaurants in 'Mumbai'

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

SWIGGY

List all customers who have placed at least one order

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.order_id)
FROM
    customers
        INNER JOIN
    orders USING (customer_id)
GROUP BY customers.customer_id , customers.name;
```

SWIGGY

Display the total number of orders placed by each customer.

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.customer_id) Total_order_Placed
FROM
    customers
        left JOIN
    orders USING (customer_id)
GROUP BY customer_id , customers.name;
```

SWIGGY

Find the total revenue generated by each restaurant

```sql
SELECT
    restaurants.restaurant_id,
    restaurants.name,
    coalesce(SUM(orders.total_amount),0) Total_revenue_generated
FROM
    restaurants
       left JOIN
    orders USING (restaurant_id)
GROUP BY restaurant_id , restaurants.name
order by restaurant_id;
```

SWIGGY

Find the top 5 restaurants with the highest average rating

```sql
SELECT
    restaurants.restaurant_id,restaurants.name,restaurants.rating
FROM
    restaurants
ORDER BY rating DESC
LIMIT 5;
```

SWIGGY

Display all customers who have never placed an order

```sql
SELECT
    customers.customer_id, customers.name
FROM
    customers
        LEFT JOIN
    orders USING (customer_id)
WHERE
    orders.order_id IS NULL
ORDER BY customer_id;
```

SWIGGY

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

```sql
SELECT
    customers.customer_id,
    customers.name,
    COUNT(orders.customer_id) Total_order_Placed
FROM
    customers
        LEFT JOIN
    orders USING (customer_id)
GROUP BY customer_id , customers.name
ORDER BY customer_id;
```

SWIGGY

Display all orders placed in the last 30 days

```
SELECT
    *
FROM
    orders
WHERE
    order_date >= DATE_ADD((SELECT
                MAX(order_date)
            FROM
                orders),
        INTERVAL - 30 DAY);
```

SWIGGY

List all delivery partners who have completed more than 1 delivery

```
SELECT
    deliverypartners.partner_id,
    deliverypartners.name,
    count( orderdelivery.order_id) Order_delivered,
    deliveryupdates.status
FROM
    deliverypartners
       JOIN
    orderdelivery USING (partner_id)
    Join
    deliveryupdates using( order_id)
    where status="delivered"
 group by       deliverypartners.partner_id, deliverypartners.name, deliveryupdates.status
    having Order_delivered >1
   order by partner_id ;
```

SWIGGY

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

```sql
SELECT
    customer_id,
    Customers.name,
    COUNT(DISTINCT order_date) Disctinct_order_days
FROM
    orders
        JOIN
    customers USING (customer_id)
GROUP BY customer_id,name
HAVING Disctinct_order_days = 3;
```

SWIGGY

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

```
With A as
    (SELECT customers.customer_id, customers.name, customers.city, orders.restaurant_id, orders.order_date
FROM
    customers JOIN orders using (customer_id))
SELECT A1.*, A2.*
FROM
    A AS A1
        JOIN
    A AS A2 ON A1.customer_id <> A2.customer_id
        AND A1.city = A2.city
        AND A1.restaurant_id = A2.restaurant_id
        AND A1.order_date <> A2.order_date
        Where A1.order_date< A2.order_date;
```

SWIGGY

SWIGGY

Thank You

Let’s Connect :

Deepak.official93@gmail.com

Deepak Sharma (deepakofficial93)