

SWIGGY CASE STUDY

Contents

- INTRODUCTION

- TABLES

- QUESTIONS

- USING SQL



INTRODUCTION

1

This project delves into the world of Swiggy, a leading Indian food delivery platform, using the power of SQL. We'll be exploring a simulated Swiggy database containing information on users, restaurants, orders, and more. By crafting SQL queries, we'll unlock valuable insights about Swiggy's operations. We'll uncover trends, identify top performers, and gain a deeper understanding of customer behavior. This case study aims to demonstrate the effectiveness of SQL in analyzing real-world data and extracting knowledge to inform strategic decision-making for businesses like Swiggy.

2 TABLES

USERS

	user_id	name	email	password
▶	1	Nitish	nitish@gmail.com	p252h
	2	Khushboo	khushboo@gmail.com	hxN9b
	3	Vartika	vartika@gmail.com	9hu7j
	4	Ankit	ankit@gmail.com	Ikko3
	5	Neha	neha@gmail.com	3i7qm
	6	Anupama	anupama@gmail.com	46rdw2
	7	Rishabh	rishabh@gmail.com	4sw123

ORDERS

	order_id	user_id	r_id	amount	date	partner_id	delivery_time	delivery_rating	restaurant_rating
▶	1001	1	1	550	2022-05-10	1	25	5	3
	1002	1	2	415	2022-05-26	1	19	5	2
	1003	1	3	240	2022-06-15	5	29	4	
	1004	1	3	240	2022-06-29	4	42	3	5
	1005	1	3	220	2022-07-10	1	58	1	4
	1006	2	1	950	2022-06-10	2	16	5	
	1007	2	2	530	2022-06-23	3	60	1	5
	1008	2	3	240	2022-07-07	5	33	4	5
	1009	2	4	300	2022-07-17	4	41	1	
	1010	2	5	650	2022-07-31	1	67	1	4
	1011	3	1	450	2022-05-10	2	25	3	1
	1012	3	4	180	2022-05-20	5	33	4	1

2 TABLES

MENU

	menu_id	r_id	f_id	price
▶	1	1	1	450
	2	1	2	400
	3	1	3	100
	4	2	3	115
	5	2	4	230
	6	2	5	300
	7	3	3	80
	8	3	6	160
	9	3	7	140
	10	4	6	230
	11	4	8	180
	12	4	9	120
	13	5	6	250
	14	5	10	220
	15	5	11	180

FOOD

	f_id	f_name	type
▶	1	Non-veg Pizza	Non-veg
	2	Veg Pizza	Veg
	3	Choco Lava cake	Veg
	4	Chicken Wings	Non-veg
	5	Chicken Popcorn	Non-veg
	6	Rice Meal	Veg
	7	Roti meal	Veg
	8	Masala Dosa	Veg
	9	Rava Idli	Veg
	10	Schezwan Noodles	Veg
	11	Veg Manchurian	Veg

2 TABLES

ORDER DETAILS

	<u>id</u>	<u>order_id</u>	<u>f_id</u>
▶	1	1001	1
	2	1001	3
	3	1002	4
	4	1002	3
	5	1003	6
	6	1003	3
	7	1004	6
	8	1004	3
	9	1005	7
	10	1005	3
	11	1006	1
	12	1006	2
	13	1006	3

RESTAURANTS

	<u>r_id</u>	<u>r_name</u>	<u>cuisine</u>
	1	dominos	Italian
	2	kfc	American
	3	box8	North Indian
	4	Dosa Plaza	South Indian
	5	China Town	Chinese

3 QUESTIONS

Q1. FIND CUSTOMERS WHO HAVE NEVER ORDERED?

QUERY

```
select name from users  
where user_id not in  
(select user_id from orders);
```

RESULT

	name
▶	Anupama
	Rishabh

3 QUESTIONS

Q2. FIND AVERAGE PRICE PER DISH IN DECREASING ORDER?

QUERY

```
select f.f_name, avg(m.price) as 'Avg Price'  
from food f  
join menu m  
on f.f_id = m.f_id  
group by f.f_name  
order by avg(m.price) desc;
```

RESULT

	f_name	Avg Price
▶	Non-veg Pizza	450.0000
	Veg Pizza	400.0000
	Chicken Popcorn	300.0000
	Chicken Wings	230.0000
	Schezwan Noodles	220.0000
	Rice Meal	213.3333
	Masala Dosa	180.0000
	Veg Manchurian	180.0000
	Roti meal	140.0000
	Rava Idli	120.0000
	Choco Lava cake	98.3333

3 QUESTIONS

Q3. TOP RESTAURANT IN TERMS OF NO. OF ORDERS FOR MONTH OF JUNE

QUERY

```
select r.r_name as 'Restaurant Name',
       count(order_id) 'No. of orders'
  from orders o
 join restaurants r on o.r_id=r.r_id
 where monthname(date) like 'june'
 group by r.r_name
 order by count(order_id) desc limit 1;
```

RESULT

Restaurant Name	No. of orders
kfc	3

3 QUESTIONS

Q4.FIND RESTAURANTS WITH MONTLY SALES(JULY) > RS 500 ?

QUERY

```
select r.r_name as 'Restaurant Name',
sum(amount) as 'Revenue'
from orders o
join restaurants r
on o.r_id=r.r_id
where monthname(date) like 'july'
group by r.r_name
having sum(amount) > 500;
```

RESULT

Restaurant Name	Revenue
China Town	1050
dominos	1100
kfc	1935

3 QUESTIONS

Q5.SHOW ALL ORDERS WITH ORDER DETAILS FOR A PARTICULAR CUSTOMER IN A PARTICULAR DATE RANGE.

QUERY

```
select o.order_id,r.r_name,f.f_name from orders o
join restaurants r
on r.r_id=o.r_id
join order_details od
on od.order_id=o.order_id
join food f
on f.f_id=od.f_id
where user_id =
(select user_id from users where name like 'Ankit')
and (date between '2022-05-01' and '2022-06-01');
```

RESULT

order_id	r_name	f_name
1017	Dosa Plaza	Masala Dosa
1016	Dosa Plaza	Masala Dosa
1017	Dosa Plaza	Rava Idli
1016	Dosa Plaza	Rava Idli

3 QUESTIONS

Q6. RESTAURANT WITH MAX REPEATED CUSTOMERS.

QUERY

```
select r.r_name as 'Restaurant Name',
count(user_id) as 'Loyal Customers'
from
  (select r_id,user_id,count(*) as 'visits'
   from orders
   group by r_id,user_id
   having visits>1) t
join restaurants r on r.r_id=t.r_id
group by r.r_name
order by count(user_id) desc limit 1;
```

RESULT

Restaurant Name	Loyal Customers
kfc	2

3 QUESTIONS

Q7.MONTH OVER MONTH REVENUE GROWTH OF SWIGGY.

QUERY

```
select Month,((revenue - prev)/prev)*100 as 'Growth(%)'  
from(  
With sales as (  
select monthname(date) as 'Month',sum(amount) as 'Revenue'  
from orders o  
join restaurants r on o.r_id=r.r_id  
group by monthname(date)  
)  
select month ,revenue,lag(revenue,1)  
over(order by revenue) as'prev'  
from sales)t
```

RESULT

Month	Growth(%)
May	NULL
June	32.7835
July	50.4658

3 QUESTIONS

Q8. FIND CUSTOMER'S FAVOURITE FOOD.

QUERY

```
with temp as (
    select o.user_id,od.f_id,count(od.f_id) as 'frequency'
    from orders o
    join order_details od
    on o.order_id=od.order_id
    group by o.user_id,od.f_id
)
select u.Name,f.f_name as 'Food Item',t1.frequency from
temp t1
join users u
on u.user_id=t1.user_id
join food f
on f.f_id=t1.f_id
where t1.frequency=
(
    select max(frequency)
    from temp t2
    where t2.user_id=t1.user_id
)
```

RESULT

Name	Food Item	frequency
Neha	Choco Lava cake	5
Khushboo	Choco Lava cake	3
Nitish	Choco Lava cake	5
Vartika	Chicken Wings	3
Ankit	Schezwan Noodles	3
Ankit	Veg Manchurian	3

3 QUESTIONS

Q8. FIND CUSTOMER'S FAVOURITE FOOD.

QUERY

```
with temp as (
    select o.user_id,od.f_id,count(od.f_id) as 'frequency'
    from orders o
    join order_details od
    on o.order_id=od.order_id
    group by o.user_id,od.f_id
)
select u.Name,f.f_name as 'Food Item',t1.frequency from
temp t1
join users u
on u.user_id=t1.user_id
join food f
on f.f_id=t1.f_id
where t1.frequency=
(
    select max(frequency)
    from temp t2
    where t2.user_id=t1.user_id
)
```

RESULT

Name	Food Item	frequency
Neha	Choco Lava cake	5
Khushboo	Choco Lava cake	3
Nitish	Choco Lava cake	5
Vartika	Chicken Wings	3
Ankit	Schezwan Noodles	3
Ankit	Veg Manchurian	3

THANK YOU