SWIGGY PROJECT PRESENTATION

Presented by AKASH SINGH

INTRODUCTION

This is MS SQL PROJECT

My name is Akash singh. I am from Mathura and i have completed polytechnic diploma adn b tech from civil branch. and I have utilized real based swiggy project and i covered basic problem, intermediate problem and advance problem
THANK YOU

DATA VALIDATION AND CLEANING

1. Null check

2. Blank or Empty Strings data.

3. Duplicate detection and delete duplication

NULL CHECK

```sql
select
sum(CASE when [State] is null then 1 else 0 end) as null_state,
sum(CASE when City is null then 1 else 0 end) as null_city,
sum(CASE when Order_Date is null then 1 else 0 end) as null_Order_Date,
sum(CASE when Restaurant_Name is null then 1 else 0 end) as null_Restaurant_Name,
sum(CASE when [Location] is null then 1 else 0 end) as null_Location,
sum(CASE when Category is null then 1 else 0 end) as null_Category,
sum(CASE when Dish_Name is null then 1 else 0 end) as null_Dish_Name,
sum(CASE when Price_INR is null then 1 else 0 end) as null_Price_INR,
sum(CASE when Rating is null then 1 else 0 end) as null_Rating,
sum(CASE when Rating_Count is null then 1 else 0 end) as null_Rating_Count
from Swiggy_Data;
```

CHECK BLANK DATA

```sql
select * from Swiggy_Data
where
State = '' OR City = ''
OR
Restaurant_Name = ''
OR
Location = '' OR Dish_Name = '';
```

DETECT DUPLICATE

```sql
select
[State], City, Order_Date, Restaurant_Name,[Location],Category, Dish_Name,
Price_INR,Rating, Rating_Count, count(*) as CNT
from Swiggy_Data
group by
[State],City, Order_Date, Restaurant_Name,[Location],Category, Dish_Name,
Price_INR,Rating, Rating_Count
HAVING COUNT(*) > 1;
```

DELETE DUPLICATION

```sql
with cte as(
select *, ROW_NUMBER() OVER(PARTITION BY state,
City, Order_Date, restaurant_name, Location, Category,
Dish_name, Price_INR, rating, Rating_Count
Order by (select Null)
)as rn
from Swiggy_Data
)
Delete from cte where rn > 1;
```

CREATING SCHEMA

1. Dimention table

2. Data table

1. Dimention tables are creating.

DIM_DATE TABLE

```
create table dim_date(
    date_id int identity(1,1) primary key,
    full_date date,
    year int,
    month int,
    month_name varchar(20),
    quarter int,
    day int,
    week int
);
```

DIM_LOCATION TABLE

```
create table dim_location(
location_id int identity(1,1) primary key,
state varchar(100),
city varchar(100),
location varchar(100)
);
```

DIM_RESTAURANT TABLE

```
create table dim_restaurant(
restaurant_id int primary key identity(1,1),
reataurant_name varchar(100)
);
```

DIM_CATEGORY TABLE

```
create table dim_category(
category_id int primary key identity(1,1),
category varchar(100)
);
```

DIM_DISH TABLE

```
create table dim_dish(
    dish_id int primary key identity(1,1),
    dish_name varchar(200)
);
```

2. Create data table

FACT TABLE

```
Create table fact_swiggy_orders(
order_id int identity(1,1) primary key,
price_inr decimal(10,2),
rating decimal(4,2),
rating_count int,

date_id int,
location_id int,
restaurant_id int,
category_id int,
dish_id int

foreign key (date_id) references dim_date(date_id),
foreign key (location_id) references dim_location(location_id),
foreign key (restaurant_id) references dim_restaurant(restaurant_id),
foreign key (category_id) references dim_category(category_id),
foreign key (dish_id) references dim_dish(dish_id)
);
```

Data Insert in all tables

INSERT DATA DIM_DATE

```sql
insert into dim_date(full_date, year, month, month_name, quarter, day, week)
    select distinct
    order_date,
    YEAR(order_date),
    MONTH(Order_Date),
    DATENAME(MONTH, Order_Date),
    DATEPART(QUARTER, Order_Date),
    DAY(Order_Date),
    DATEPART(WEEK, Order_Date)
from swiggy_data
where Order_Date is not null;
```

INSERT DATA DIM_LOCATION

```
insert into dim_location(state, city, location)
select distinct
    state, city, location
from Swiggy_Data;
```

INSERT DATA DIM_RESTAURANT

```
insert into dim_restaurant(restaurant_name)
select distinct
    restaurant_name
    from Swiggy_Data;
```

INSERT DATA DIM_CATEGORY

```sql
insert into dim_category(category)
    select distinct
    category
from Swiggy_Data;
```

INSERT DATA DIM_DISH

```sql
insert into dim_dish(Dish_name)
select distinct
    Dish_name
from Swiggy_Data;
```

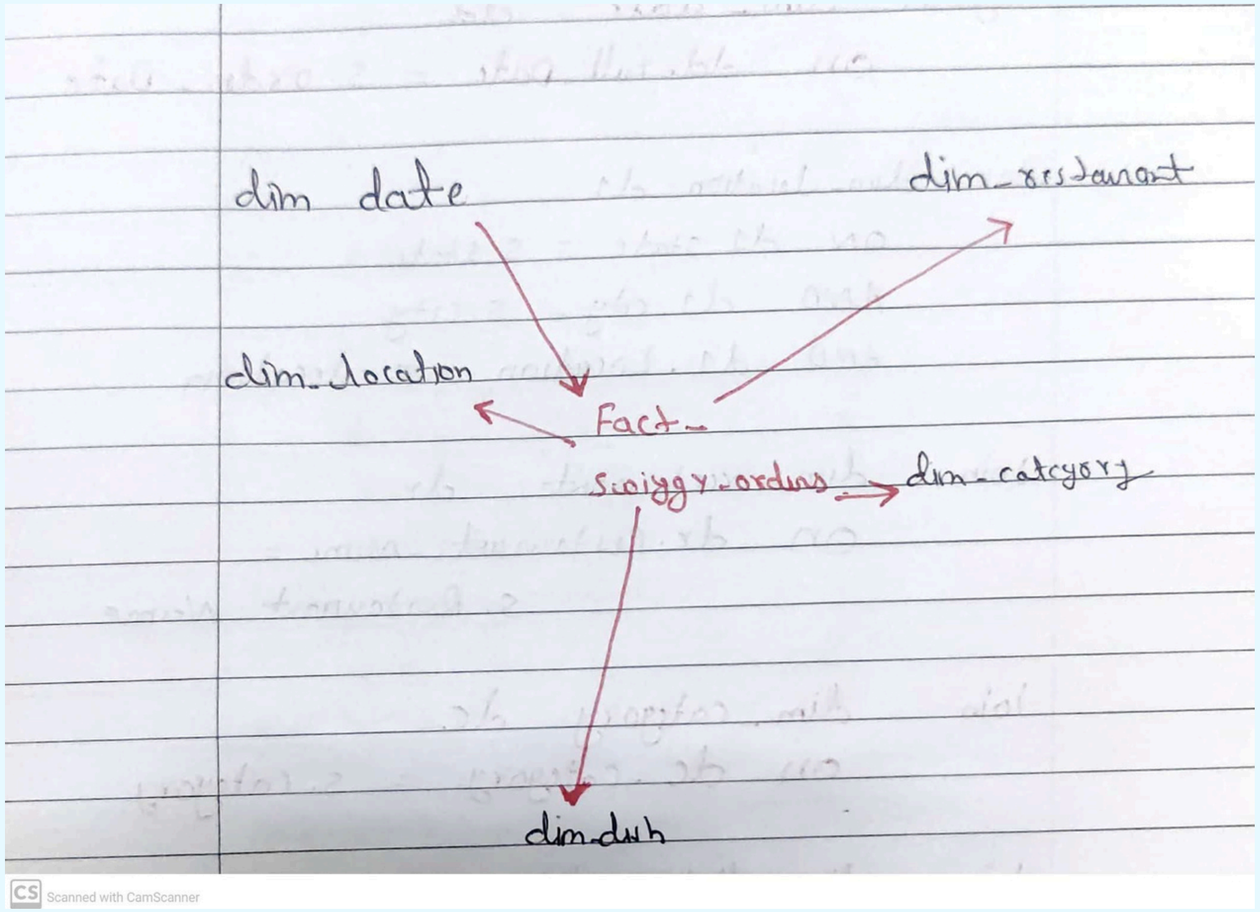
INSERT DATA FACT_SWIGGY_ORDERS

```sql
insert into fact_swiggy_orders
(date_id,price_inr,rating,rating_count,
location_id,restaurant_id,category_id,dish_id)
select
dd.date_id,s.Price_INR,s.Rating,s.Rating_Count,
d1.location_id,dr.restaurant_id,dc.category_id,
dsh.dish_id
from Swiggy_Data s

join dim_date dd
 on dd.full_date = s.Order_Date
join dim_location d1
    on d1.state  = s.State
    and d1.city = s.City
    and d1.location = s.Location
join dim_restaurant dr
on dr.Restaurant_name = s.Restaurant_Name
join dim_category dc
    on dc.category = s.Category
join dim_dish dsh
    on dsh.dish_name = s.Dish_Name;
```

JOIN ALL TABLES

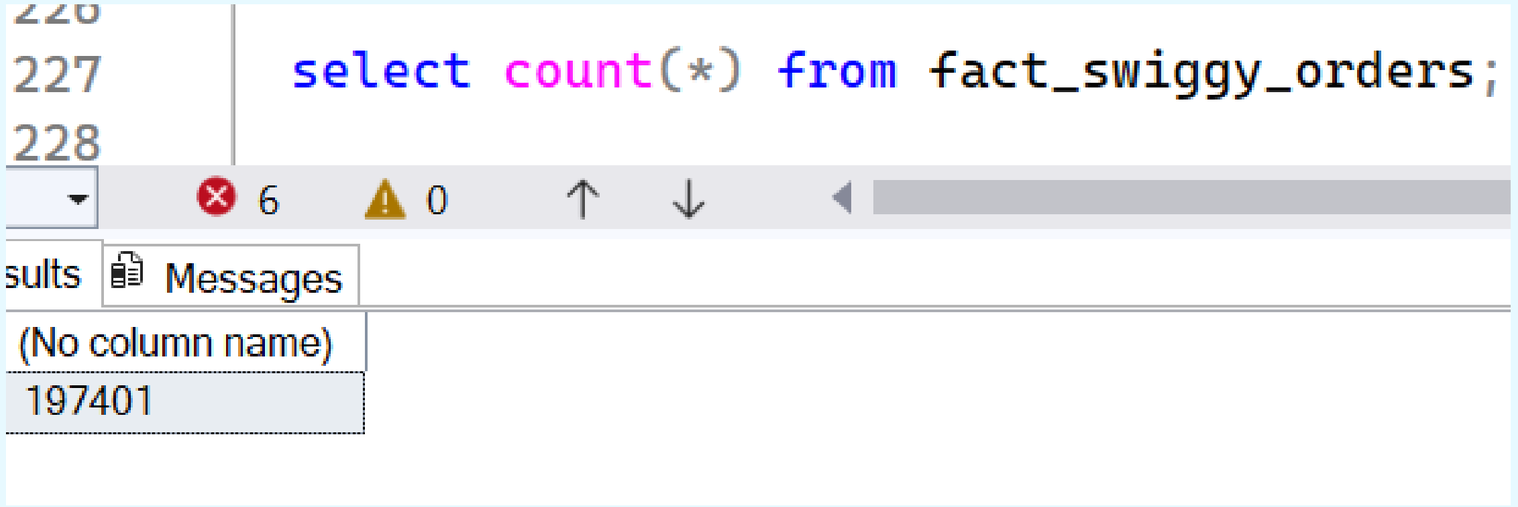
```sql
select * from fact_swiggy_orders f
join dim_date d
on f.date_id = d.date_id
join dim_location l
on l.location_id = f.location_id
join dim_restaurant r
on r.restaurant_id = f.restaurant_id
join dim_category c
on c.category_id = f.category_id
join dim_dish di
on di.dish_id = f.dish_id;
```

ALL JOIN TABLE DAIGRAM



→ BASIC KPIS

TOTAL ORDERS



TOTAL REVENUE(INR MILLION)

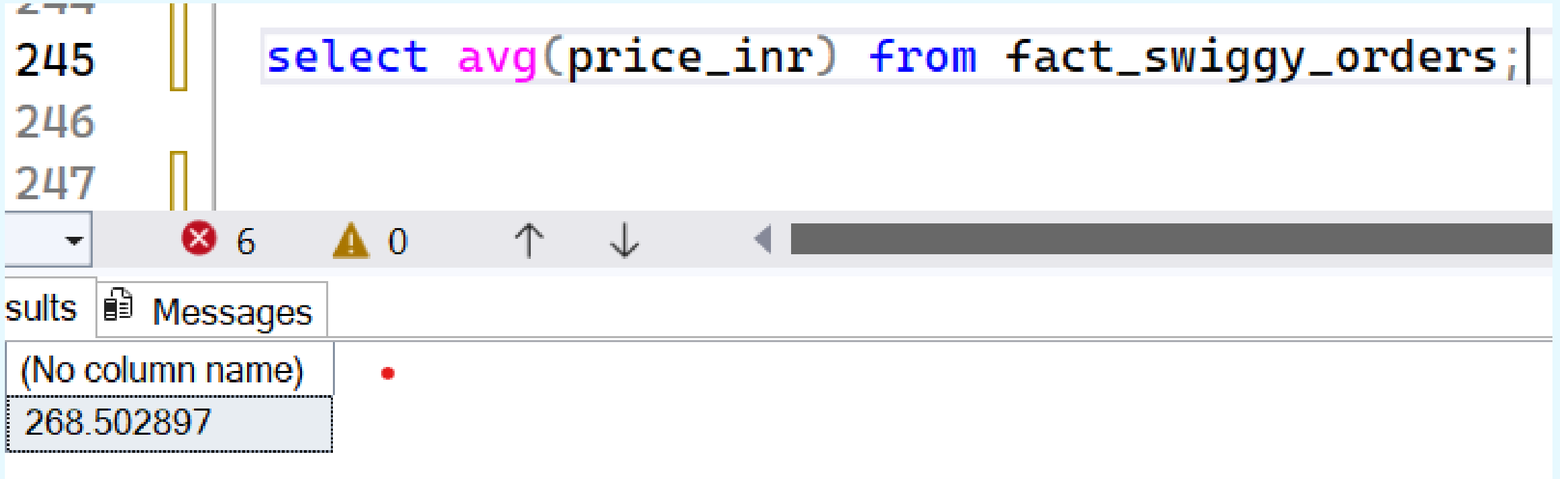
```
233
234  select format(sum(convert(float, price_inr))/1000000, 'n2') + ' milion'
235  as revenue
236  from fact_swiggy_orders;
237
```

6 0

esults Messages

revenue
53.00 milion

AVG DISH PRICE



AVG RATING

```
250    select avg(rating) AS AVG_RATING from fact_swiggy_orders;
```

% 6 0

Results Messages

(No column name)
4.341577

DEEP DIVE BUSINESS ANALYSIS
(DATE BASIS ANALYSIS)

MONTHLY ORDER TRENDS

```
259  select  top 3 dd.month, dd.month_name, count(*) as total_orders
260  from fact_swiggy_orders fso
261  join dim_date dd
262  on dd.date_id = fso.date_id
263  group by dd.month, dd.month_name
264  order by total_orders desc;
265
```

6 ⚠ 0

esults | Messages

month	month_name	total_orders
1	January	25393
8	August	25227
5	May	25188

TOP 3 MONTH REVENUE

```
268  select  top 3 dd.month, dd.month_name, sum(fso.price_inr)/1000000
269  as total_revenue_month_in_milion from fact_swiggy_orders fso
270  join dim_date dd
271  on dd.date_id = fso.date_id
272  group by dd.month, dd.month_name
273  order by total_revenue_month_in_milion desc;
274
```

6 0

sults Messages

month	month_name	total_revenue_month_in_milion
1	January	6.823981
5	May	6.792621
8	August	6.790899

QUARTER TREND

```
279  select dd.quarter, count(*) from fact_swiggy_orders fso
280  join dim_date dd
281  on dd.date_id = fso.date_id
282  group by dd.quarter
283
```

6 0

esults | Messages

quarter	(No column name)
1	73084
2	74154
3	50163

YEARLY TREND

```
289  select dd.year, count(*)as total_order_in_year
290  from fact_swiggy_orders  fso
291  join dim_date dd
292  on dd.date_id = fso.date_id
293  group by dd.year;
294
```

6 0

esults Messages

year	total_order_in_year
2025	197401

ORDER DAY OF WEEK

```
298  select
299  DATENAME(WEEKDAY, dd.full_date) as day_name,
300  count(*) as total_orders
301  from fact_swiggy_orders fso
302  join dim_date dd
303  on dd.date_id = fso.date_id
304  group by DATENAME(WEEKDAY, dd.full_date)
305  order by total_orders desc;
306
```

% 6 0

Results Messages

day_name	total_orders
Saturday	28933
Sunday	28469
Thursday	28450
Friday	28284
Wednesday	28284
Monday	27568
Tuesday	27413

LOCATION BASED PROBLEM

HIGHEST TOP 10 CITY ORDERS

```
310  select top 10 dl.city, COUNT(*) as tot_orders from fact_swiggy_orders fso
311  join dim_location dl
312  on dl.location_id = fso.location_id
313  group by  dl.city
314  order by COUNT(*) desc;
315
```

6 0

Results Messages

city	tot_orders
Bengaluru	20072
Mumbai	10507
Hyderabad	10308
Jaipur	10285
Lucknow	10192
New Delhi	10191
Ahmedabad	10175
Chandigarh	10060
Kolkata	10044
Chennai	10042

uery executed successfully. AKASH\SQLEXPRESS (16.0 RTM) AKASH

HIGHEST TOP 10 CITY REVENUE

```
317  select top 10 dl.city, sum(fso.price_inr) as tot_revenue from fact_swiggy_orders fso
318  join dim_location dl
319  on dl.location_id = fso.location_id
320  group by dl.city
321  order by tot_revenue desc;
322
```

09 % 6 0

Results Messages

	city	tot_revenue
1	Bengaluru	5455887.73
2	Lucknow	3117359.65
3	Hyderabad	3021656.62
4	Mumbai	3015573.35
5	New Delhi	2829180.60
6	Ahmedabad	2815536.27
7	Chandigarh	2804991.82
8	Kolkata	2662213.76
9	Chennai	2642594.63
10	Jaipur	2502833.61

Query executed successfully. AKASH\SQLEXPRESS (16.0 RTM) AKASH\alesh (77)

FOOD PERFORMENCE

TOP 10 RESTAURANT BY ORDERS

```
336
337  select top 10 dr.Restaurant_name, COUNT(*) as tot_orders
338  from fact_swiggy_orders fso
339  join dim_restaurant dr
340  on dr.restaurant_id = fso.restaurant_id
341  group by dr.Restaurant_name
342  order by tot_orders desc;
343
```

6 0

Results | Messages

Restaurant_name	tot_orders
McDonald's	13528
KFC	12957
Burger King	7115
Pizza Hut	6529
Domino's Pizza	5489
LunchBox - Meals and Thalis	4700
Baskin Robbins - Ice Cream Desserts	4197
Faasos - Wraps, Rolls & Shawarma	3256
Olio - The Wood Fired Pizzeria	3239
The Good Bowl	2665

Query executed successfully. AKASH\SQL

TOP CATEGORIES (INDIS, CHINESE ETC)

```
355
356  select dc.category, COUNT(*) as tot_category from fact_swiggy_orders fso
357  join dim_category dc
358  on dc.category_id = fso.category_id
359  group by dc.category
360  order by tot_category desc
361
```

6 0

sults Messages

category	tot_category
Recommended	24097
Desserts	3582
Main Course	2983
Beverages	2682
BURGERS	2538
Sweets	1954
McSaver Combos (2 Pc Meals)	1884
Exclusive Deals (Save upto 40%)	1717
Starters	1692
ROLLS	1652

MOST ORDERED DISHES

```
364
365  select top 10 dd.dish_name, count(*) as order_count
366  from fact_swiggy_orders fso
367  join dim_dish dd
368  on dd.dish_id = fso.dish_id
369  group by dd.dish_name
370  order by order_count desc;
```

09 % 6 0

Results Messages

	dish_name	order_count
1	Veg Fried Rice	321
2	Choco Lava Cake	303
3	Jeera Rice	265
4	Paneer Butter Masala	262
5	French Fries	248
6	Chicken Sausage	230
7	Chicken Fried Rice	228
8	Butter Naan	218
9	Margherita Pizza	203
10	Green Salad	197

ORDERS+RATINGS

```
376  select dd.dish_name, COUNT(*) as order_count,
377  AVG(fso.rating) as order_rating from fact_swiggy_orders fso
378  join dim_dish dd
379  on dd.dish_id = fso.dish_id
380  group by dd.dish_name
381  order by order_rating desc;
382
```

109 % 6 0

Results Messages

	dish_name	order_count	order_rating
1	Moongfali Kebab	2	5.000000
2	Jalli Weffers Pkt 200 Gm	1	5.000000
3	Chocolate Cake (500Gm.)	1	5.000000
4	Veg Kathi Roll Paneer Capsicum	1	5.000000
5	Cinnabon Stix - Pack Of 2 (8Pcs)	1	5.000000
6	Butter Cheese Onion Uttapam	1	5.000000
7	Jumbo Prawns Kebabs	1	5.000000
8	Khejur Dry Fruit	1	5.000000
9	Paneer Smoked Skewers [10 Pcs]	1	5.000000
10	Kulhad Rabri Lassi 5 Glass	1	5.000000

Query executed successfully. AKASH\SQLEXPR

CUSTOMER SPENDING TOTAL ORDER BY PRICE RANKING

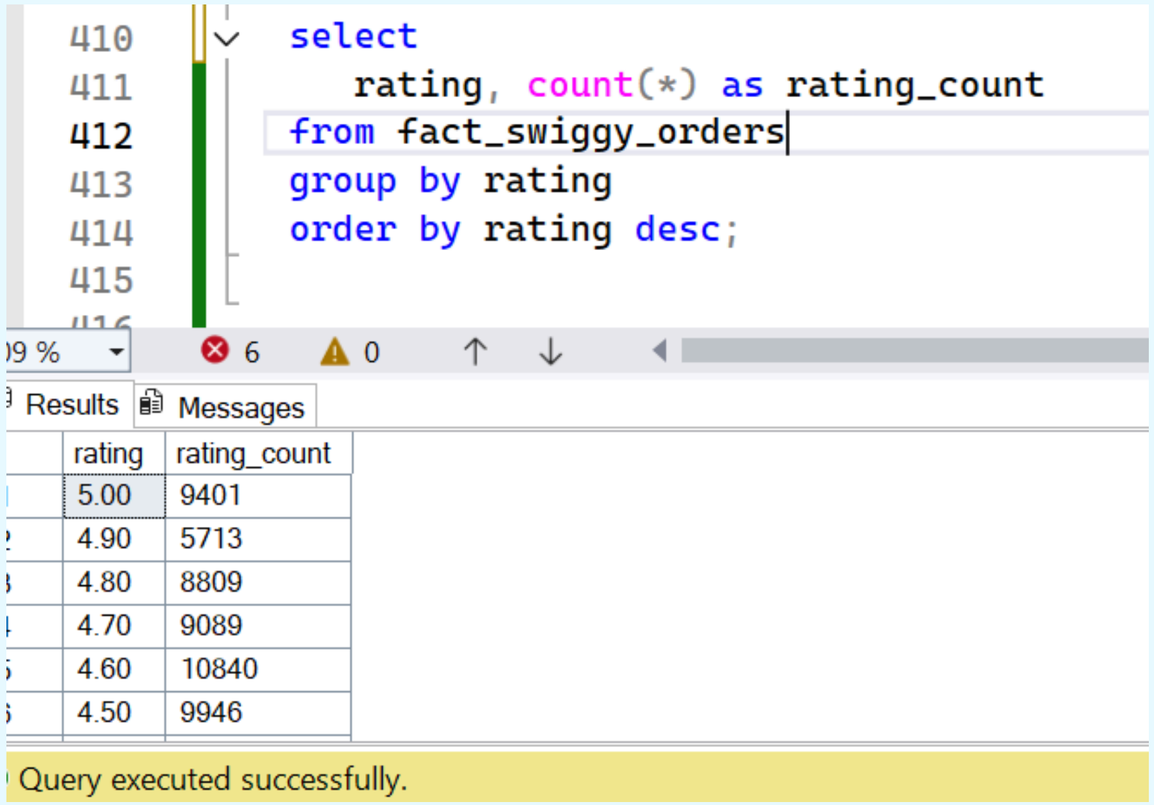
```sql
select
    case
        when CONVERT(float, price_inr)<100 then 'under 100'
        when CONVERT(float, price_inr) between 100 and 199 then '100-199'
        when CONVERT(float, price_inr) between 200 and 299 then '200 - 299'
        when CONVERT(float, price_inr) between 300 and 499 then '300-499'
        else '500+'
    end as price_range, COUNT(*) as total_orders
from fact_swiggy_orders
group by
    case
        when CONVERT(float, price_inr)<100 then 'under 100'
        when CONVERT(float, price_inr) between 100 and 199 then '100-199'
        when CONVERT(float, price_inr) between 200 and 299 then '200 - 299'
        when CONVERT(float, price_inr) between 300 and 499 then '300-499'
        else '500+'
    end
order by total_orders desc

--Rating count distribution
```

6 0

Results Messages

price_range	total_orders
100-199	56189
200 - 299	54567
300-499	43758
under 100	26795
500+	16092

RATING COUNT DISTRIBUTION(1 TO 5)



THANK YOU