SQL Code

```sql
--select * from swiggy_Data
--SWIGGY DATA ANALYSIS

--DATA CLEANING ABD VALIDATION
--NULL CHECK
SELECT *
FROM swiggy_data
where state is null
or city is null
or order_date is null
or restaurant_name is null
or location is null
or category is null
or dish_name  is null
or price_inr is null
or rating is null
or rating_count is null ;

--BLANK OR EMPTY STRINGS

SELECT *
FROM swiggy_data
where state = ''
or city = ''
or restaurant_name = ''
or location = ''
or category= ''
or dish_name= ''

--DUPLICATE DETECTION

select
  state,city,order_date,restaurant_name,location,category,dish_name,price_inr,rating,rating_count,count(*) as duplicate_value
  from swiggy_data
  group by state,city,order_date,restaurant_name,location,category,dish_name,price_inr,rating,rating_count having count(*)>1;

  -- DELETE DUPLICATES

  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 t
  from swiggy_data )
  delete from cte where t>1

    --CREATING SCHEMA

  --DIMENSION TABLES
  --DATE TABLE
  drop table dim_date
```

```sql
  create table dim_date
  (date_id int identity(1,1) primary key,
  full_date date,
  year INT,
  month INT,
  monthname varchar(20),
  quater INT,
  day INT,
  week INT
  )

    --DIMENSION LOCATION
    drop table dim_location
  create table dim_location
  (location_id int identity(1,1) primary key,
  state VARCHAR(50),
  city VARCHAR(50),
  location VARCHAR(200)
  )

  --DIMENSION CATEGORY
  drop table dim_category
create table dim_category
( category_id int identity(1,1) primary key ,
category varchar(200)
)
--DIMENSION RESTAURANT
drop table dim_restaurant
create table dim_restaurant
( restaurant_id int identity(1,1) primary key,
restaurant_name varchar(200)
)

--DIMENSION DISH
drop table dim_dish
create table dim_dish
(dish_id int identity(1,1) primary key,
dish_name varchar(200)
)
--FACT TABLE
drop table fact_swiggy_orders
create table fact_swiggy_orders
(
order_id int identity(1,1) primary key,
date_id int ,
price_inr decimal(10,2),
rating decimal (4,2),
rating_count 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),
```

```
105  foreign key(restaurant_id) references dim_restaurant(restaurant_id),
106  foreign key(category_id) references dim_category(category_id)
107  );
108
109  -- INSERT DATA INTO TABLES
110
111  --dim_date
112
113  insert into dim_date
114  select distinct
115  order_date,
116  year(order_date),
117  month(order_date),
118  datename(month,order_date),
119  datepart(quarter, order_date),
120  day(order_date),
121  datepart(week,order_date)
122  from swiggy_data
123  where order_date is not null;
124  select * from dim_date
125
126  -- dim_location
127
128  insert into dim_location (state,city,location)
129  select distinct state,city, location
130  from swiggy_data;
131
132  select * from dim_location
133
134  -- dim_category
135
136  insert into dim_category
137  select distinct
138  category
139  from swiggy_data;
140  select * from dim_category
141
142  --dim restaurant
143
144  insert into dim_restaurant
145  select distinct
146  restaurant_name
147  from swiggy_data
148  select * from dim_restaurant
149
150  --dim_dish
151
152  insert into dim_dish
153  select distinct
154  dish_name
155  from swiggy_data;
156  select * from dim_dish
157  -- fact_table
158  insert into fact_swiggy_orders
159  (date_id ,
```

```
    price_inr ,
    rating,
    rating_count,
    location_id ,
    restaurant_id,
    category_id,
    dish_id)
    select distinct
    dd.date_id,
    s.price_inr,
    s.Rating,
    s.rating_count,
    dl.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 dl on dl.state= s.state and dl.city=s.city and dl.location=s.loca
tion
    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;

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

    -- KPI DEVELOPMENT
    select * from fact_swiggy_orders
    --TOTAL ORDERS

    select
    count(order_id) as total_orders
    from fact_swiggy_orders

    --Total Revenue (INR Million)

    select
    cast(cast(sum(price_inr) /1000000 as decimal(20,2))as varchar(20)) + ' '+ 'INR MILL
ION' as total_revenue
    from fact_swiggy_orders


    --Average Dish Price

    select
    cast(cast(avg(price_inr)  as decimal(20,2))as varchar(20)) + ' INR' as avg_dish_pri
ce
    from fact_swiggy_orders

    --Average Rating
```

```sql
select
cast(avg(rating) as decimal(10,2)) as avg_rating
from fact_swiggy_orders

--DEEP-DIVE BUSINESS ANALYSIS

--DATE BASED ANALYSIS

--Monthly order trends
select
d.year,
d.month,
d.monthname,
count(*) as total_orders
from fact_swiggy_orders f
join dim_date d on f.date_id= d.date_id
group by
d.year ,
d.month ,
d.monthname
order by count(*) desc



--Quarterly orders

select
d.year ,
d.quater,
count(*) as qua_total_orders
from fact_swiggy_orders f
join dim_date d on d.date_id = f.date_id
group by d.year,d.quater
order by count(*) desc


--Yearly Orders

select
d.year ,
count(*) as year_total_orders
from fact_swiggy_orders f
join dim_date d on d.date_id = f.date_id
group by d.year,d.quater
order by count(*) desc


--Order by day of week ( MON-SUN)

select
datename(weekday,d.full_date) as day_name ,
count(*) as week_total_orders
from fact_swiggy_orders f
join dim_date d on d.date_id = f.date_id
group by datename(weekday,d.full_date)
```

```sql
order by count(*) desc


--LOCATION BASED ANALYSIS

--Top 10 cities by order volume

select top 10
c.city,
count(*)
from fact_swiggy_orders f
join dim_location c on c.location_id = f.location_id
group by c. city
order by count(*) desc

--Revenue contribution by states

select
l.state,
sum(f.price_inr) as state_revenue from fact_swiggy_orders f
join dim_location l on l.location_id = f.location_id
group by l.state
order by sum(f.price_inr) desc

--FOOD PERFORMANCE

--Top 10 restaurants by orders

select top 10
r.restaurant_name  ,
sum(price_inr) as top_res_sales from fact_swiggy_orders f
join dim_restaurant r on r.restaurant_id = f.restaurant_id
group by r.restaurant_name
order by sum(price_inr) desc

--Top categories (Indian, Chinese, etc.)

select top 10
c.category  ,
count(*) as top_cat_sales from fact_swiggy_orders f
join dim_category c on c.category_id  = f.category_id
group by c.category
order by count(*)  desc


--Most ordered dishes
select top 10
d.dish_name ,
count(*) as top_ord_dish from fact_swiggy_orders f
join dim_dish d on d.dish_id  = f.dish_id
group by d.dish_name
order by count(*)  desc

--Cuisine performance ? Orders + Avg Rating

```

```sql
select
c.category ,
count(*) as total_orders,
avg(f.rating) as avg_rating
from fact_swiggy_orders f
join dim_category c on c.category_id = f.category_id
group by c.category
order by total_orders desc

--CUSTOMERS SPENDING INSIGHTS

/*Buckets of customer spend:
Under 100
100?199
200?299
300?499
500+
With total order distribution across these ranges. */

select

    case
        when price_inr <100 then 'under 100'
        when price_inr between 100 and 199 then '100-199'
        when price_inr between 200 and 299 then '299-400'
        when 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 price_inr<100 then 'under 100'
            when price_inr between 100 and 199 then '100-199'
            when price_inr between 200 and 299 then '299-400'
            when price_inr between 300 and 499 then '300-499'
            else '500+'
        end
order by price_range


--RATING ANALYSIS

--Distribution of dish ratings from 1-?5.
select
rating,
count(*) as rating_count
from fact_swiggy_orders
group by rating
order by count(*) desc

```