```
use swiggy

select * from Swiggy

--Data Validation and Cleaning
--Null values

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_state,
SUM(case when Restaurant_Name is null then 1 else 0 end) as null_restarurant_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

--BLANK OR EMPTY STRING

select * from Swiggy
```

```
where

State='' or City='' or Restaurant_Name='' or Location='' or Category='' or

Dish_Name=''


--Dupliactes record


select
state,City,Order_Date,Restaurant_Name,Location,Category,Dish_Name,Price_INR,Rating,Ratin
g_Count,COUNT(*) as CNT

from Swiggy

group by
state,City,Order_Date,Restaurant_Name,Location,Category,Dish_Name,Price_INR,Rating,Ratin
g_Count

having COUNT(*)>1


--Delete Duplications


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)

delete from CTE where rn>1


--Creating Schema
```

```
--Dimensnal tabel

--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)


select * from dim_date


create table dim_location(

location_id int identity(1,1) primary key,

state varchar(100),

city varchar(100),

location varchar(200)

)


select * from dim_location
```

```
create table dim_restaurant(
restaurant_id int identity(1,1) primary key,
restaurant_name varchar(200))

select * from dim_restaurant

create table dim_category(
category_id int identity(1,1) primary key,
category varchar(200))

select * from dim_category

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

select * from dim_dish

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),
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)
)

select * from fact_swiggy_orders

--insert data in tables
--dim date
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

where Order_Date is not null


select * from dim_date


--dim location

insert into dim_location(state,city,location)

select distinct

state,

city,

location

from Swiggy


--dim_restaurant

insert into dim_restaurant(restaurant_name)

select distinct

restaurant_name
```

```
from Swiggy

insert into dim_category(category)
select distinct
category
from Swiggy

insert into dim_dish(dish_name)
select distinct
dish_name
from Swiggy

select * from dim_dish

--fact_table

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,

dl.location_id,
dr.restaurant_id,
dc.category_id,
dsh.dish_id
from Swiggy 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.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


select * from fact_swiggy_orders


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 di on f.dish_id=di.dish_id


--KPI's

--Total orders

select COUNT(*) as total_orders

from fact_swiggy_orders


--Total revenue(inr million)

select format(SUM(convert(float,price_inr))/1000000,'N2')+'INR Million' as total_revenue

from fact_swiggy_orders
```

```
--Average dish price
select format(avg(convert(float,price_inr)),'N2')+'INR' as total_revenue
from fact_swiggy_orders

--Average rating
select AVG(rating) average_rating
from fact_swiggy_orders

--Deep Dive BusinessAnalysis
--Monthly order trends

select d.year,d.month,d.month_name,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.month_name
order by COUNT(*) desc

select d.year,d.month,d.month_name,SUM(price_inr) as total_revenue
from fact_swiggy_orders f
join dim_date d on f.date_id=d.date_id
group by d.year,d.month,d.month_name
order by SUM(price_inr) desc
```

```
--Quaterly trends
select d.year,d.quarter,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.quarter
order by COUNT(*) desc

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

--Order by Day of week (Mon-Sun)
select
DATENAME(WEEKDAY,d.full_date) as day_name,
COUNT(*) as total_orders
from fact_swiggy_orders f
join dim_date d on f.date_id=d.date_id
group by DATENAME(weekday,d.full_date),DATEPART(weekday,d.full_date)
order by DATEPART(weekday,d.full_date)
```

```
--10 cities by order volume
select top 10 l.city,
count(*) as total_orders from fact_swiggy_orders f
join dim_location l
on l.location_id=f.location_id
group by l.city
order by COUNT(*) desc

--Revenue contribtion by states
select l.state,
SUM(f.price_inr) as total_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

--top 10 restaurant by orders
select r.restaurant_name,
SUM(f.price_inr) as total_revenue from fact_swiggy_orders f
join dim_restaurant r
on r.restaurant_id=f.restaurant_id
group by r.restaurant_name
```

```
order by SUM(f.price_inr) desc

--top cateogary by order volumn
select c.category,
COUNT(*) as total_orders from fact_swiggy_orders f
join dim_category c
on f.category_id=c.category_id
group by c.category
order by COUNT(*) desc

--most ordered dishes
select d.dish_name,COUNT(*) as order_count
from fact_swiggy_orders f
join dim_dish d on f.dish_id=d.dish_id
group by d.dish_name
order by order_count desc

--Cusine perfomance(orders+avg rating)
select
c.category,
COUNT(*) as total_orders,
AVG(CONVERT(float,f.rating)) as avg_rating
from fact_swiggy_orders f
```

```
join dim_category c on f.category_id=c.category_id

group by c.category

order by total_orders desc


--Total orders by price range

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 '200-499'

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 '200-499'

else '500+'

end
```

```
order by total_orders desc


--Rating count distribution(1-5)
select rating,
COUNT(*) as rating_count
from fact_swiggy_orders
group by rating
order by COUNT(*) desc
```