```sql
select count(*) as total_customer from customers;

select count(*) as Total_delivery  from  deliveries;

select count(*)  as total_orders from orders;

select count(*) as total_restaurant from restaurants;

select Count(*) as total_riders from riders;




-- Customer Analysis
--1. Top 10 customers based on total spending

select top 10 c.customer_id,c.customer_name,round(sum(o.total_amount),2) as Total_spending,
count(o.order_id) as Total_orders
from customers as c
inner join orders as o on c.customer_id = o.customer_id
group by c.customer_id,c.customer_name
order by sum(o.total_amount) desc;


-- Top 10 Customers who placed Maximum orders in the last year

Select top 10 c.customer_id,c.customer_name,count(o.order_id) as order_count
from customers as c
inner join orders as o on c.customer_id = o.customer_id
where order_date >= dateadd(year,-1,getdate())
group by c.customer_id,c.customer_name
having count(o.order_id) > 250
```

```sql
order by count(o.order_id) desc;


-- Customer Age & Order Patterns

-- Which age group orders the most?

select case
when c.age between 18 and 25 then '18-25'
when c.age between 26 and 35 then '26-35'
when c.age between 36 and 45 then '36-45'
when c.age between 46 and 55 then '46-55'
else '56+' end as Age_group,
c.gender,count(o.order_id) as Total_orders
from customers as c
inner join orders as o on o.customer_id = c.customer_id
group by case
when c.age between 18 and 25 then '18-25'
when c.age between 26 and 35 then '26-35'
when c.age between 36 and 45 then '36-45'
when c.age between 46 and 55 then '46-55'
else '56+' end ,c.gender
order by count(o.order_id) desc,c.gender;


-- Restaurant Performance
-- Top 10 restaurants based on total sales

select top 10 r.restaurant_id,r.restaurant_name,r.city,count(o.order_id) as Total_orders,
round(sum(o.total_amount),2) as Total_sales
from restaurants as r
inner join orders as o on r.restaurant_id = o.restaurant_id
group by r.restaurant_id,r.restaurant_name,r.city
order by sum(o.total_amount) desc;
```

```sql
-- Order Trends by City
-- City with the highest number of orders

select r.city,count(o.order_id) as Total_orders,round(sum(o.total_amount),2) as Total_sales
from restaurants as r
inner join orders as o on r.restaurant_id = o.restaurant_id
group by r.city
order by count(o.order_id) desc;


-- Average order value per city

select r.city,round(avg(total_amount),2) as average_sales from restaurants as r
inner join orders as o on r.restaurant_id = o.restaurant_id
group by r.city
order by round(avg(total_amount),2) desc;

-- Order Patterns Over Time

-- Orders by day of the week

select datename(weekday,order_date) as Day, count(order_id) as Total_orders
from orders
group by datename(weekday,order_date)
order by count(order_id) desc;

-- Busiest time of the day for orders



select
case
    when cast(order_time as time) between '06:00:00' and '11:59:59' then 'Morning'
    when cast(order_time as time) between '12:00:00' and '17:59:59' then 'Afternoon'
    when cast(order_time as time) between '18:00:00' and '23:59:59' then 'Evening'
```

```sql
    else 'Night' end as Time_of_day,
count(order_id) as Total_orders from orders
group by
case
    when cast(order_time as time) between '06:00:00' and '11:59:59' then 'Morning'
    when cast(order_time as time) between '12:00:00' and '17:59:59' then 'Afternoon'
    when cast(order_time as time) between '18:00:00' and '23:59:59' then 'Evening'
    else 'Night' end
order by count(order_id) desc;


--Order Cancellations
-- Percentage of canceled orders

select count(case when order_status = 'cancelled' then 1 end)*100.0/count(order_id)
as Cancellation_percent from orders;

-- City with the highest cancellation rate

select r.city,count(o.order_id) as Total_orders,
count(case when o.order_status = 'cancelled' then 1 end) as cancel_orders,
count(case when o.order_status = 'cancelled' then 1 end)*100.0/count(o.order_id) as Cancellation_percent
from restaurants as r
inner join orders as o on r.restaurant_id = o.restaurant_id
group by r.city
order by count(case when o.order_status = 'cancelled' then 1 end)*100.0/count(o.order_id) desc;

--Food Preferences
-- Top 10 most ordered food items

select top 10 order_item, count(order_id) as total_order from orders
group by order_item
order by count(order_id) desc;

-- Food category with the highest average rating
```

```sql
select top 10 order_item, avg(rating) as Average_rating from orders
group by order_item
order by avg(rating) desc;

-- Delivery Performance
-- Percentage of successful vs. failed deliveries

select distinct delivery_status from deliveries;

select
count(case when delivery_status = 'Delivered' then 1 end)*100.0/count(delivery_id) as Success_percentage,
count(case when delivery_status = 'failed' then 1 end)*100.0/count(delivery_id) as failure_percentage
from deliveries;

-- City with the most failed deliveries percentage

select r.city,
count(case when d.delivery_status = 'failed' then 1 end)*100.0/count(d.delivery_id) as failure_percentage
from restaurants as r
inner join orders as o on r.restaurant_id = o.restaurant_id
inner join deliveries as d on d.order_id = o.order_id
group by  r.city
order by count(case when d.delivery_status = 'failed' then 1 end)*100.0/count(d.delivery_id) desc;


select distinct delivery_status from deliveries;

-- Rider Efficiency

-- Rider with the most Succesfull deliveries

select top 5 r.rider_id,r.Rider_name,count(d.delivery_id) Number_of_deliveries from riders as r
inner join deliveries as d on r.rider_id = d.rider_id
where d.delivery_status = 'Delivered'
```

```sql
group  by r.rider_id,r.Rider_name
order by count(d.delivery_id)  desc;



-- Average delivery time per city

select r.city, avg(datediff(minute,o.order_time,d.delivery_time)) as average_delivery_time
from restaurants as r
inner join orders as o on o.restaurant_id = r.restaurant_id
inner join deliveries as d on d.order_id = o.order_id
group by r.city
order by avg(datediff(minute,o.order_time,d.delivery_time))  ;

-- Revenue Forecasting & Trend Analysis
-- Monthly Revenue for the Last 6 Months


select year(order_date) as year,datename(month,order_date) as month,
round(sum(total_amount),2) as Total_revenue from orders
where order_date >= dateadd(month,-6,getdate())
group by year(order_date),datename(month,order_date),month(order_date)
order by year(order_date) desc,month(order_date) desc;

-- Order-to-Delivery Time Analysis


select o.order_id,o.customer_id,o.restaurant_id,d.rider_id,
datediff(minute,o.order_time,d.delivery_time) as Delivery_time_in_minutes
from orders as o
inner join deliveries as d on o.order_id = d.order_id
where d.delivery_status = 'Delivered'
order by datediff(minute,o.order_time,d.delivery_time) desc;
```