```
create database pizza_project;
use  pizza_project;

create table pizza_type
(pizza_type_id varchar(50) primary key,
name varchar(50),
category varchar(50),
ingredients varchar(200));



select *from pizza_type;

create table pizzas (
pizza_id varchar(50) primary key,
pizza_type_id varchar(50),
foreign key (pizza_type_id) references pizza_type(pizza_type_id),
size varchar(2),
price double);

select * from  pizzas;



create table orders (
order_id int(10) primary key,
date date,
time time);



select * from orders;
```

```
select * from order_details
left join orders on order_details.order_id = orders.order_id
left join pizzas on order_details.pizza_id = pizzas.pizza_id ;

SHOW COLUMNS FROM order_details;

-- retrieve the total nuber of orders placed
select count(order_id) as total
from
orders;

 -- calculate the total revenue genrated from pizza sales.

select round(sum(a.quantity * b.price),2) as total_sales
from
order_details as a
join pizzas as b
on a.pizza_id = b.pizza_id;

-- Identify the Highest - priced pizza.

select pizza_type.name,pizzas.price
from pizza_type
join
pizzas
on pizza_type.pizza_type_id = pizzas.pizza_type_id
order by pizzas.price desc
limit 1;

-- Idetify the most common pizza size ordered.
```

```
select count(order_details.order_id)as total_order,pizzas.size
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id
group by pizzas.size
order by total_order desc
limit 1;

-- List the top 5 most ordered pizza
-- types along with thier quntities.

select sum(order_details.quantity) as total_quantity,pizza_type.name as pi_type
from order_details
join
pizzas
on order_details.pizza_id =pizzas.pizza_id
join pizza_type
on pizza_type.pizza_type_id =pizzas.pizza_type_id
group by pi_type
order by total_quantity desc
limit 5 ;

-- join the necessary table to find the
-- total quantity of each category ordered

select sum(order_details.quantity)as Quantity,pizza_type.category as pizza_category
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id
join pizza_type
```

```
on pizza_type.pizza_type_id = pizzas.pizza_type_id
group by pizza_category
order by Quantity desc;

-- determine the distribution of orders by hour of the day

select hour(time) as hour_day,count(order_id) as count_order
from orders
group by hour_day
order by count_order desc;

-- join relevant tables to find the
-- category-wise distribution of pizzas

select category,count(name) from pizza_type
group by category;


-- group the orders by date and calculate the average
-- number of pizzas ordered per day


select round(avg(quantity),2) from
(select month(orders.date),sum(order_details.quantity) quantity
from order_details
join orders
on order_details.order_id = orders.order_id
group by orders.date) as order_quantity;

-- determine the top 3 most ordered pizza types based on revenue
```

```
select round(sum(order_details.quantity * pizzas.price),2) as revenue ,pizza_type.name  as n
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id
join pizza_type
on pizza_type.pizza_type_id = pizzas.pizza_type_id
group by n
order by revenue desc
limit 3;
```

```
select sum(order_details.quantity * pizzas.price) /
(select (sum(order_details.quantity * pizzas.price)) as total_rev
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id)*100 as persontage ,pizza_type.category AS n
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id
join pizza_type
on pizza_type.pizza_type_id = pizzas.pizza_type_id
group by n
order by persontage desc ;
```

```
-- analyze the cumulative revenue generated over time

select date,round(sum(rev) over (order by date),2)as cum_revenue
from
```

```
(select sum(order_details.quantity*pizzas.price) as rev ,orders.date
from order_details
join pizzas
on order_details.pizza_id =pizzas.pizza_id
join orders
on orders.order_id =order_details.order_id
group by date) as sales;

-- determine the 3 most ordered pizzas types
-- based on revenue of each pizza category

select category, name,rev from
(select category,name,rev,
rank() over(partition by category order by rev desc) as renks
from
(select pizza_type.category,pizza_type.name,
(sum(order_details.quantity*pizzas.price)) as rev
from order_details
join pizzas
on order_details.pizza_id =pizzas.pizza_id
join pizza_type
on pizzas.pizza_type_id = pizza_type.pizza_type_id
group by pizza_type.category, pizza_type.name) as a) as b
where renks <=3 ;
```