“Pizza Sales Analysis using SQL"

WELCOME TO PIZZA HUT

ORDER NOW

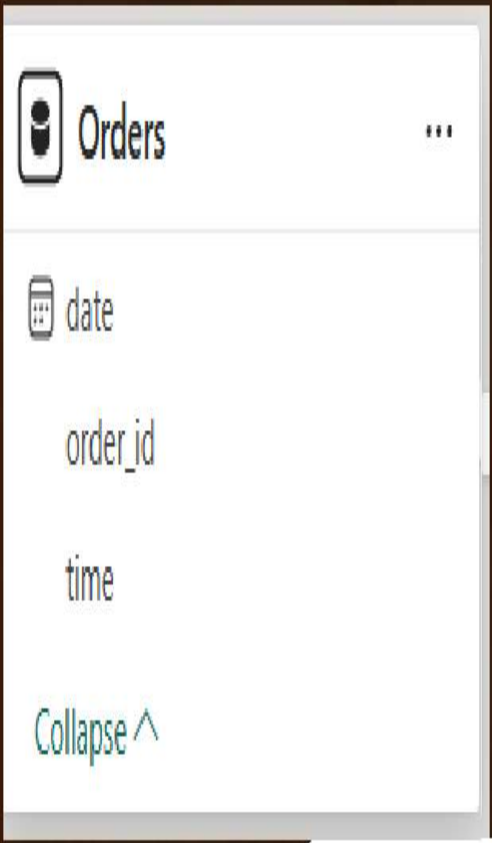
Start Your Slide

HELLO EVERYONE

My Name is Ramya Armugam

In this project I have utilized sql queries to solve a questions that were related to pizza sales.

“Powering Pizza Sales Analysis with SQL”



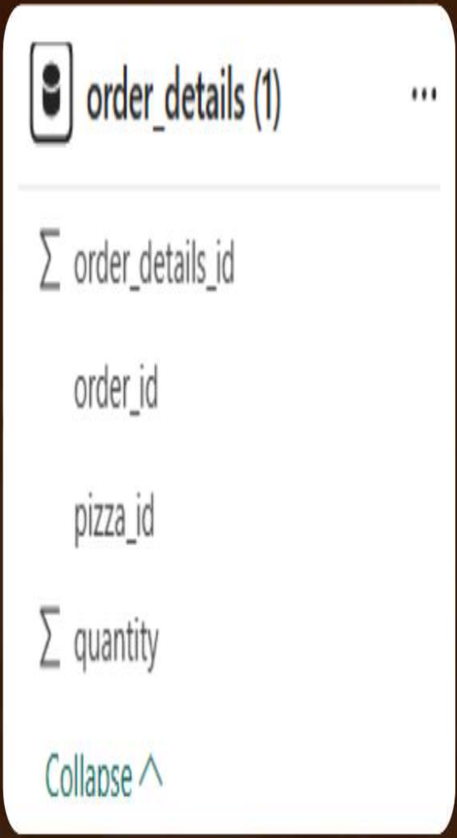
Orders ...

date

order_id

time

Collapse ^



order_details (1) ...

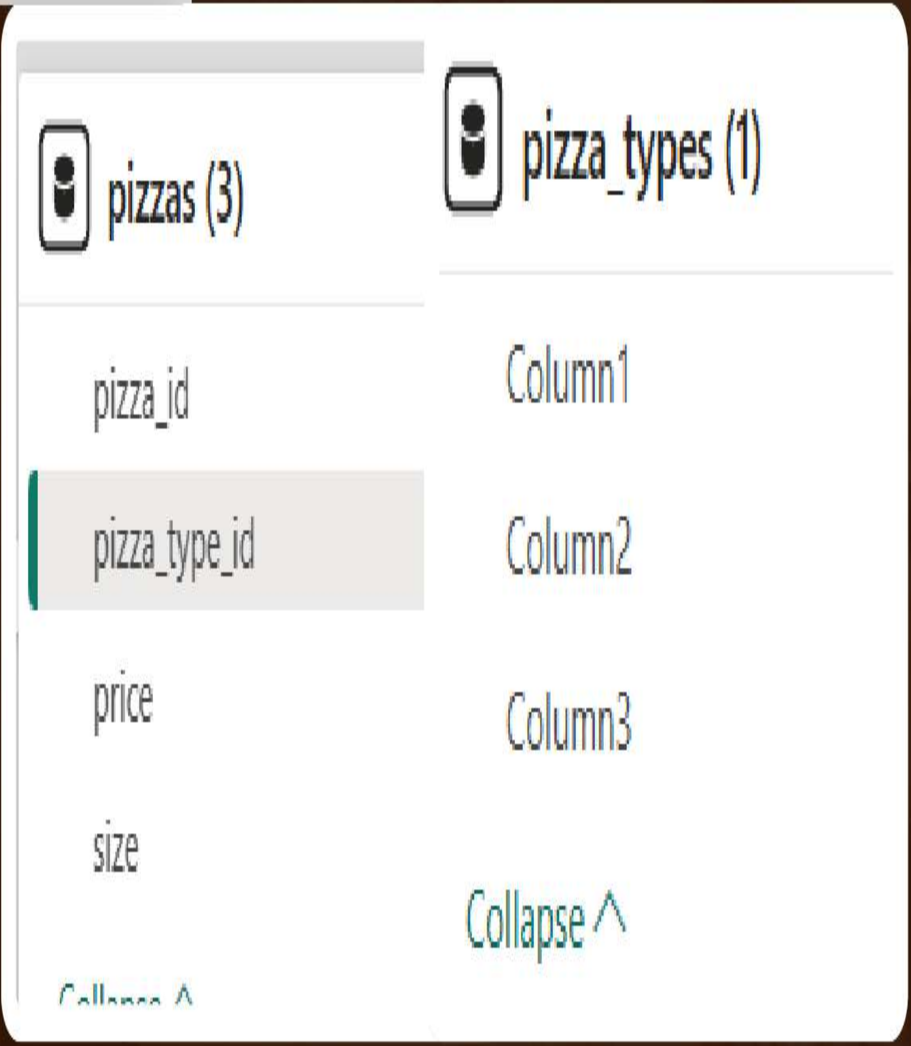
∑ order_details_id

order_id

pizza_id

∑ quantity

Collapse ^



pizzas (3)

pizza_id

pizza_type_id

price

size

Collapse ^

pizza_types (1)

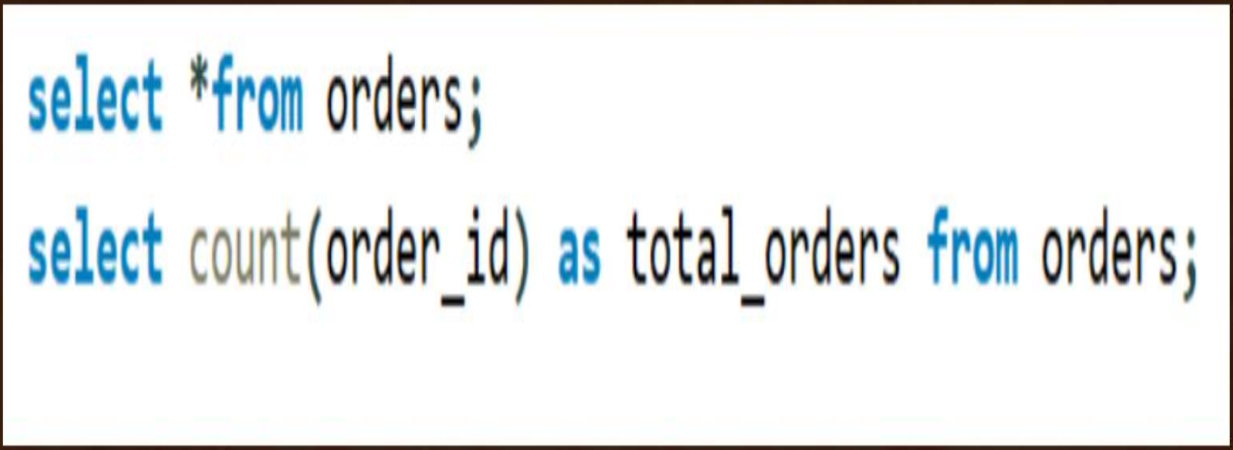
Column1

Column2

Column3

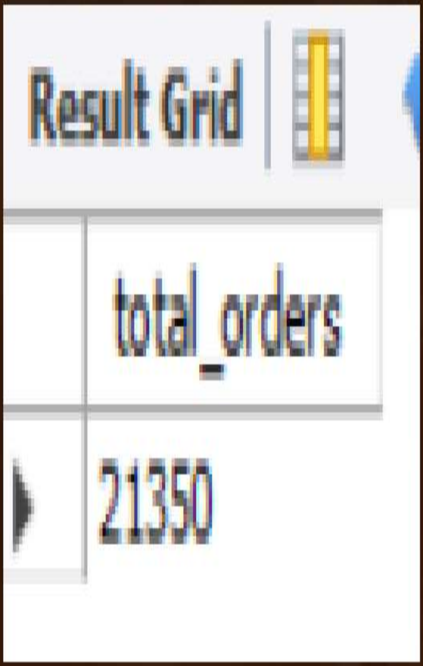
Collapse ^

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.



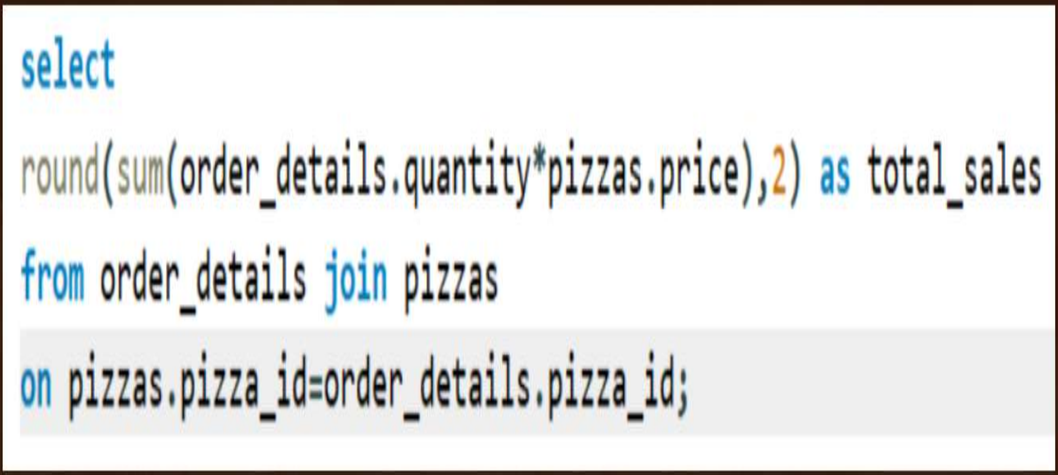
```sql
select *from orders;

select count(order_id) as total_orders from orders;
```

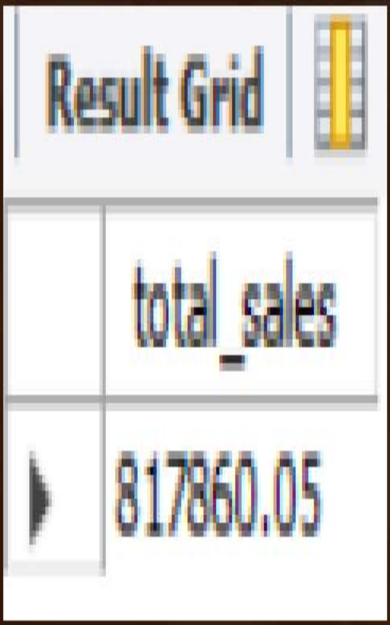


```
total_orders
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

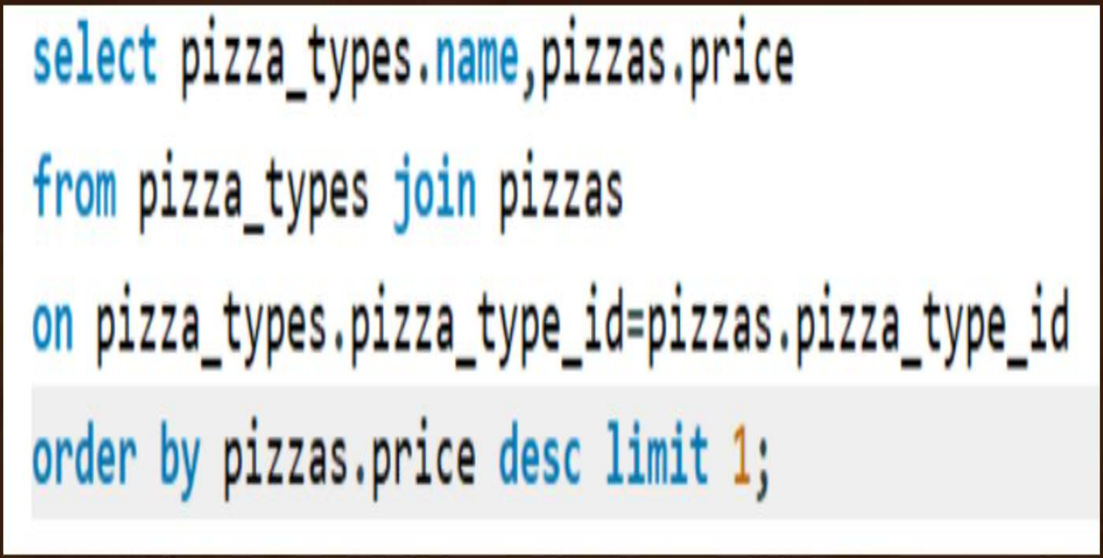


```
select
round(sum(order_details.quantity*pizzas.price),2) as total_sales
from order_details join pizzas
on pizzas.pizza_id=order_details.pizza_id;
```

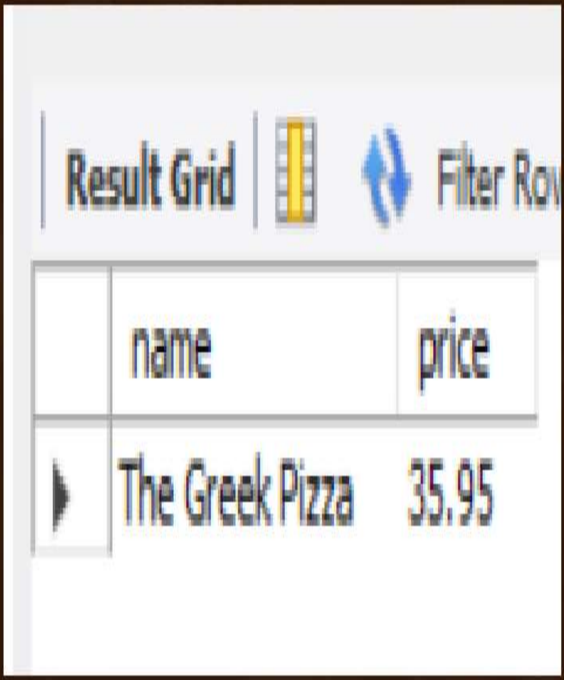


```
total_sales
817860.05
```

IDENTIFY THE HIGHEST-PRICED PIZZA.

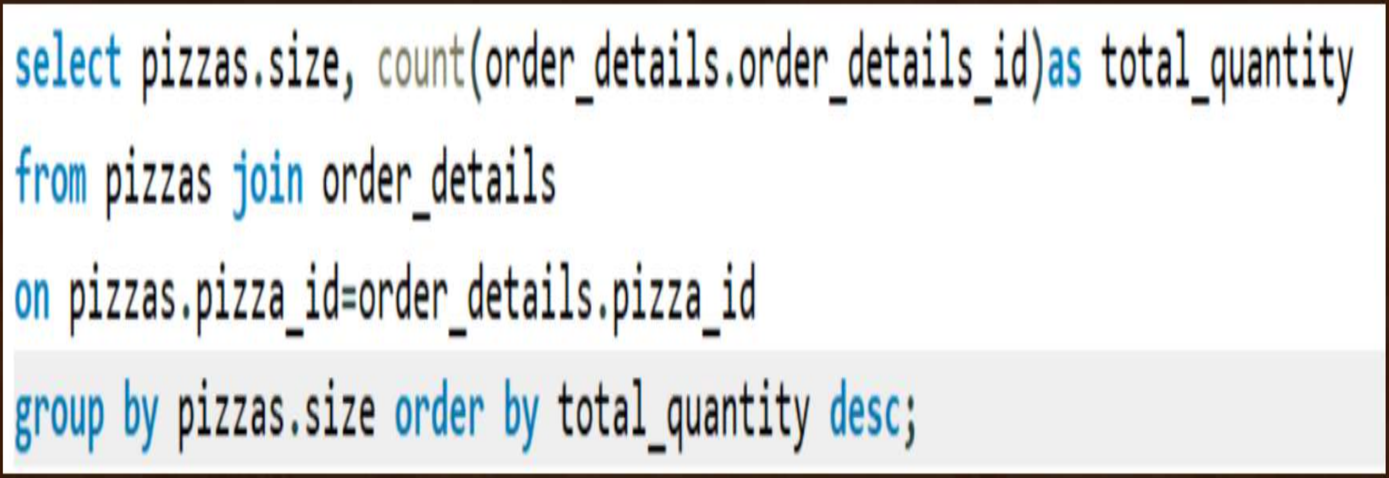


```
select pizza_types.name,pizzas.price
from pizza_types join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
order by pizzas.price desc limit 1;
```

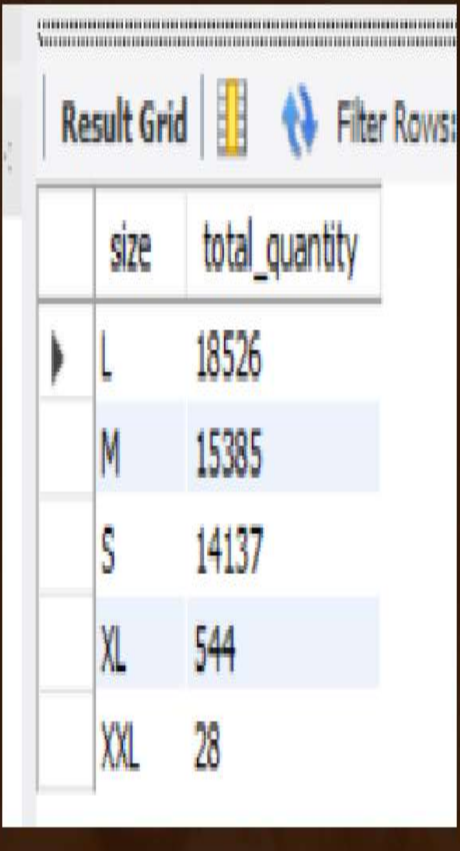


name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

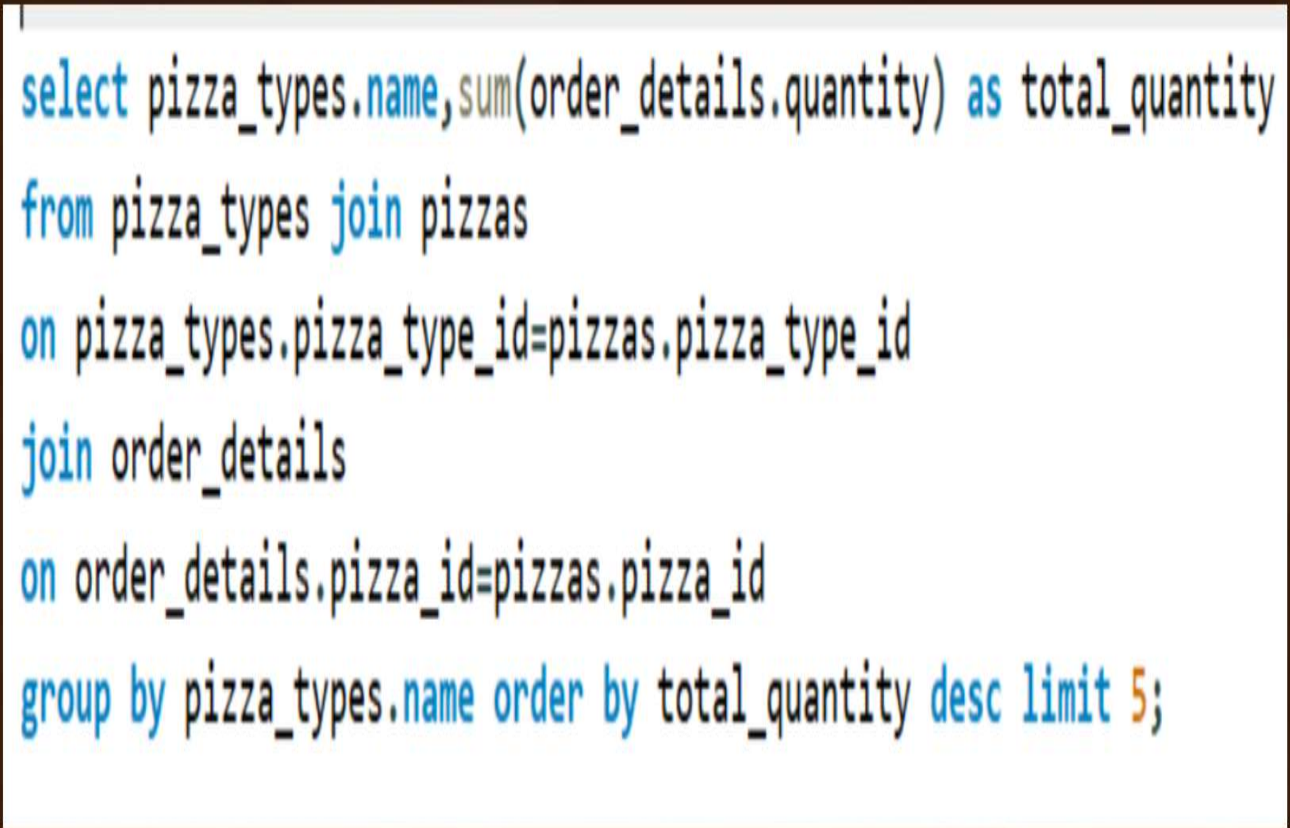


```sql
select pizzas.size, count(order_details.order_details_id)as total_quantity
from pizzas join order_details
on pizzas.pizza_id=order_details.pizza_id
group by pizzas.size order by total_quantity desc;
```

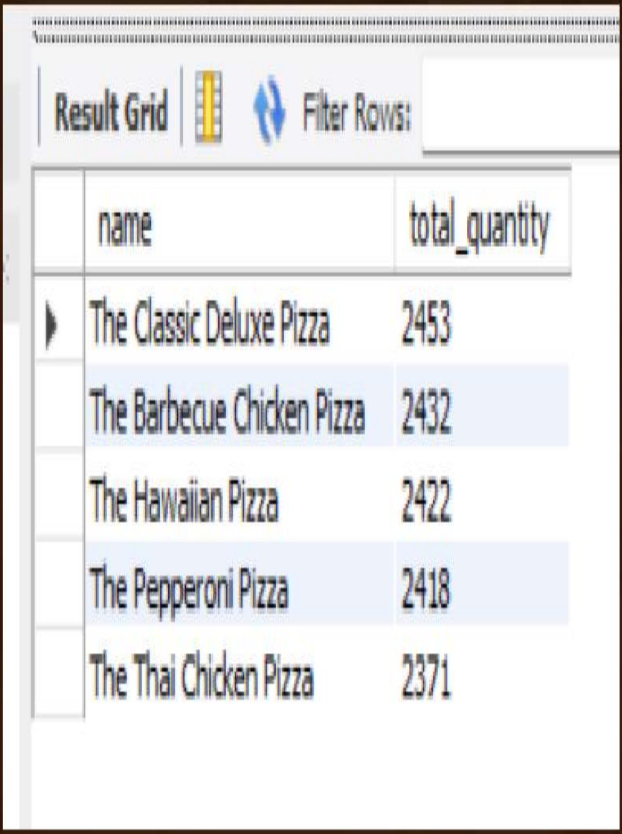


size	total_quantity
L	18526
M	15385
S	14137
XL	544
XXL	28

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

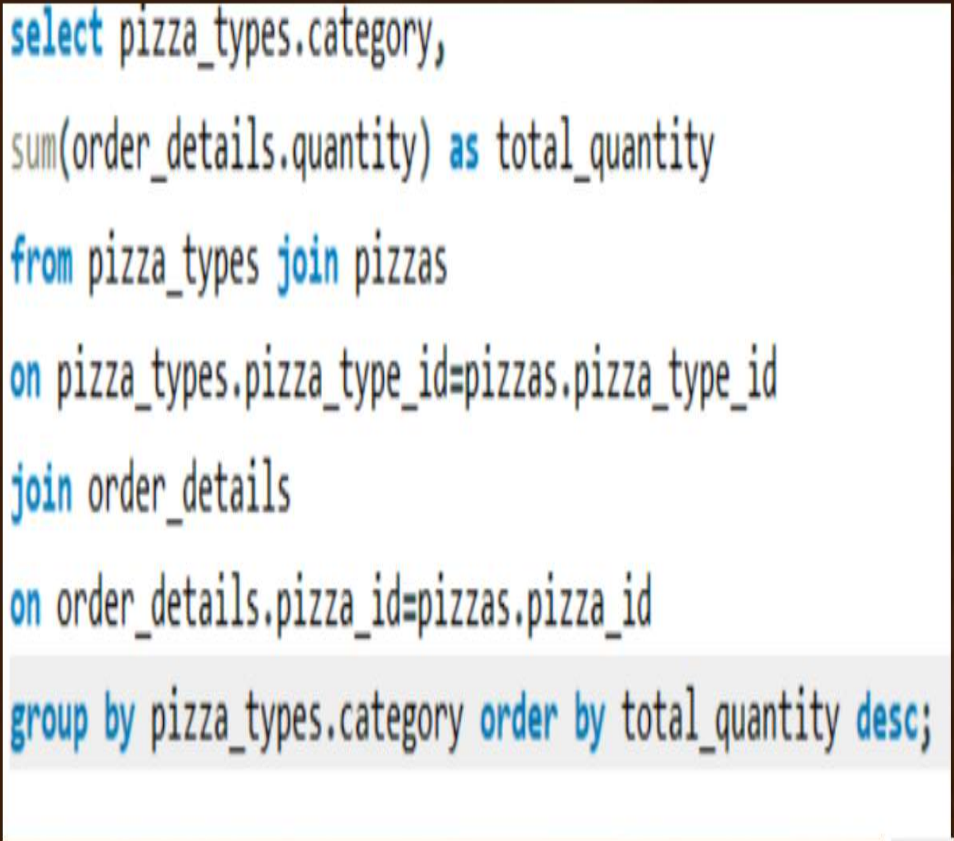


```
select pizza_types.name,sum(order_details.quantity) as total_quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.name order by total_quantity desc limit 5;
```

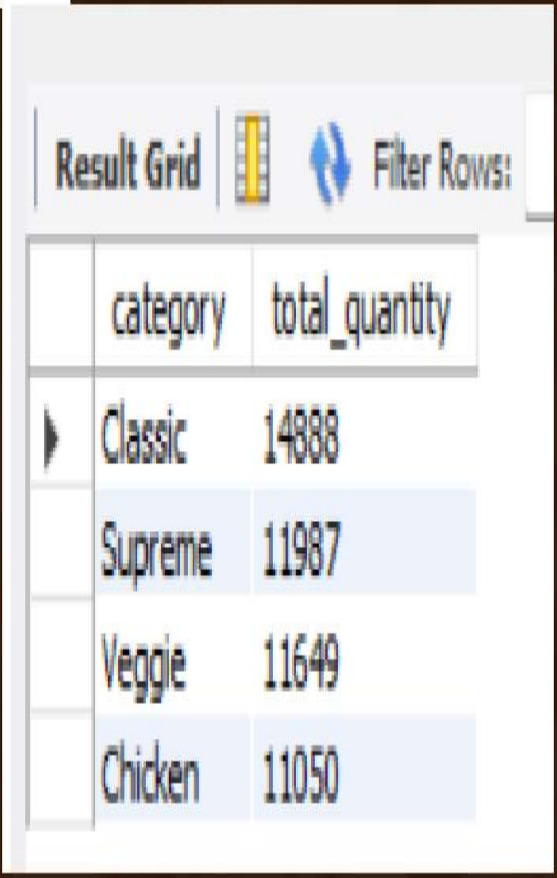


name	total_quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

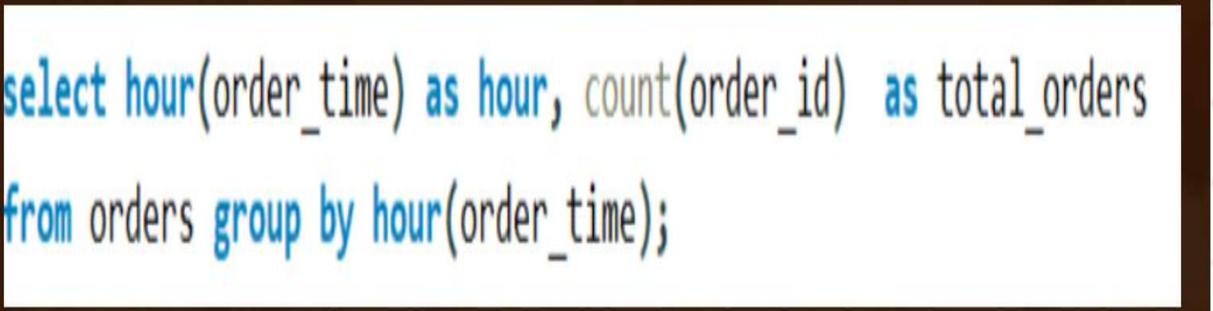


```sql
select pizza_types.category,
sum(order_details.quantity) as total_quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.category order by total_quantity desc;
```

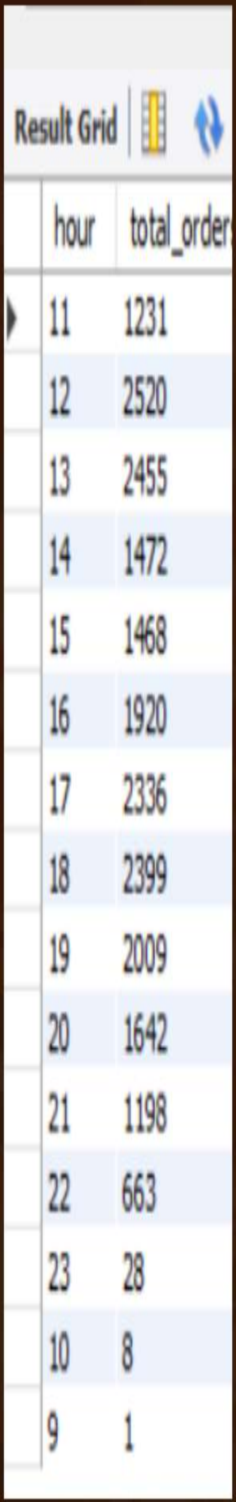


category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

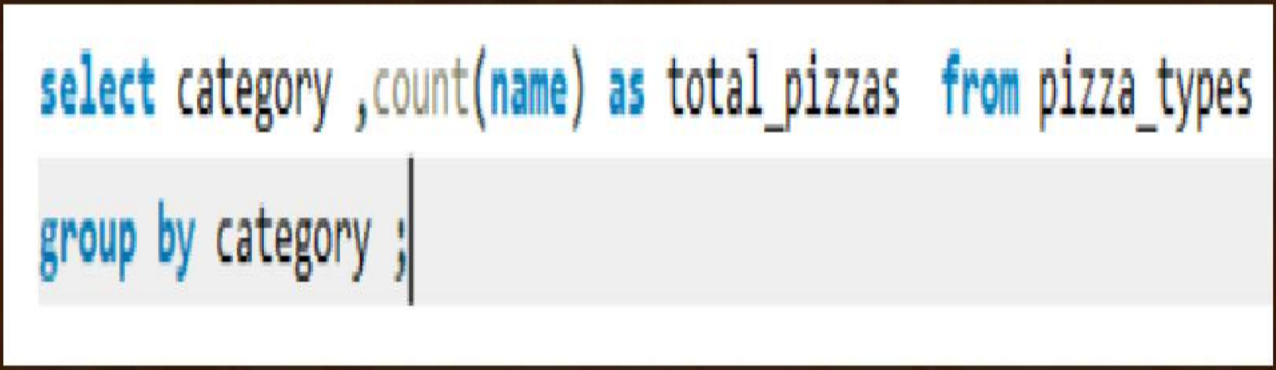


```sql
select hour(order_time) as hour, count(order_id)  as total_orders
from orders group by hour(order_time);
```

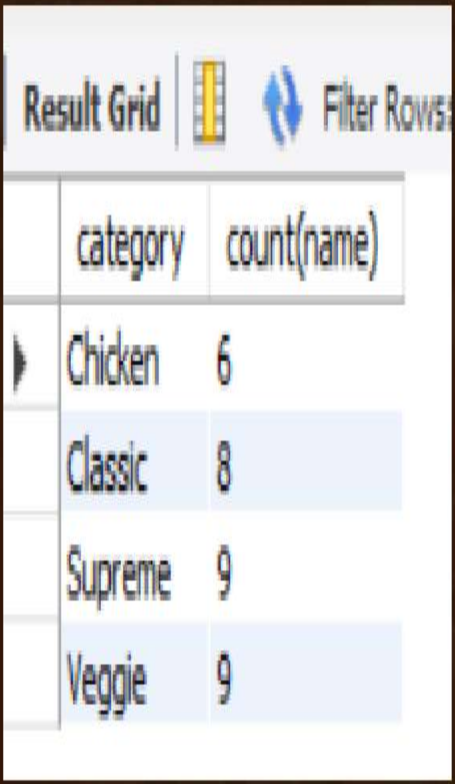


hour	total_order
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

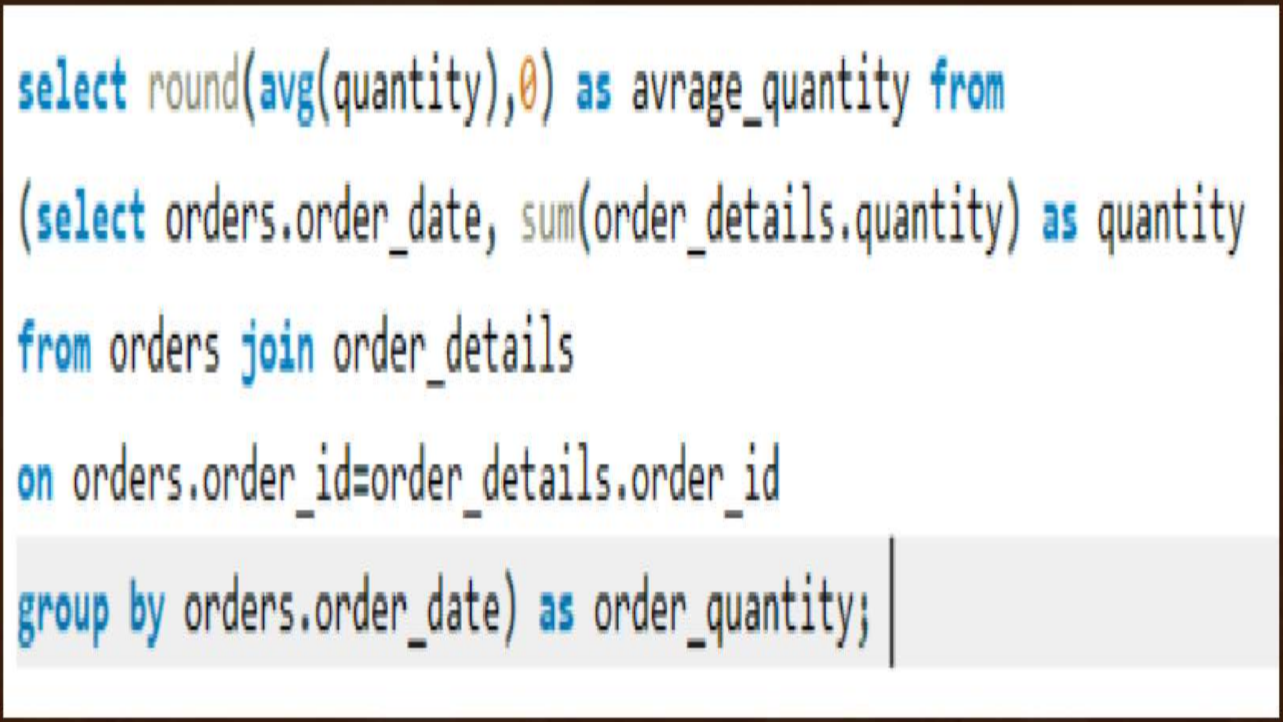


```
select category ,count(name) as total_pizzas  from pizza_types
group by category ;
```

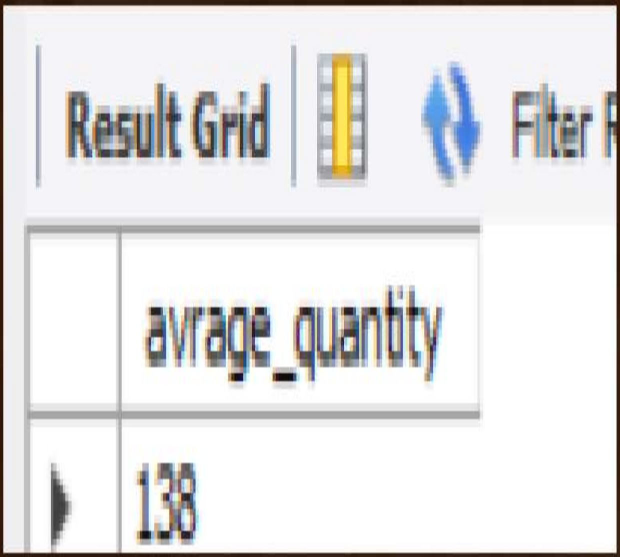


category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

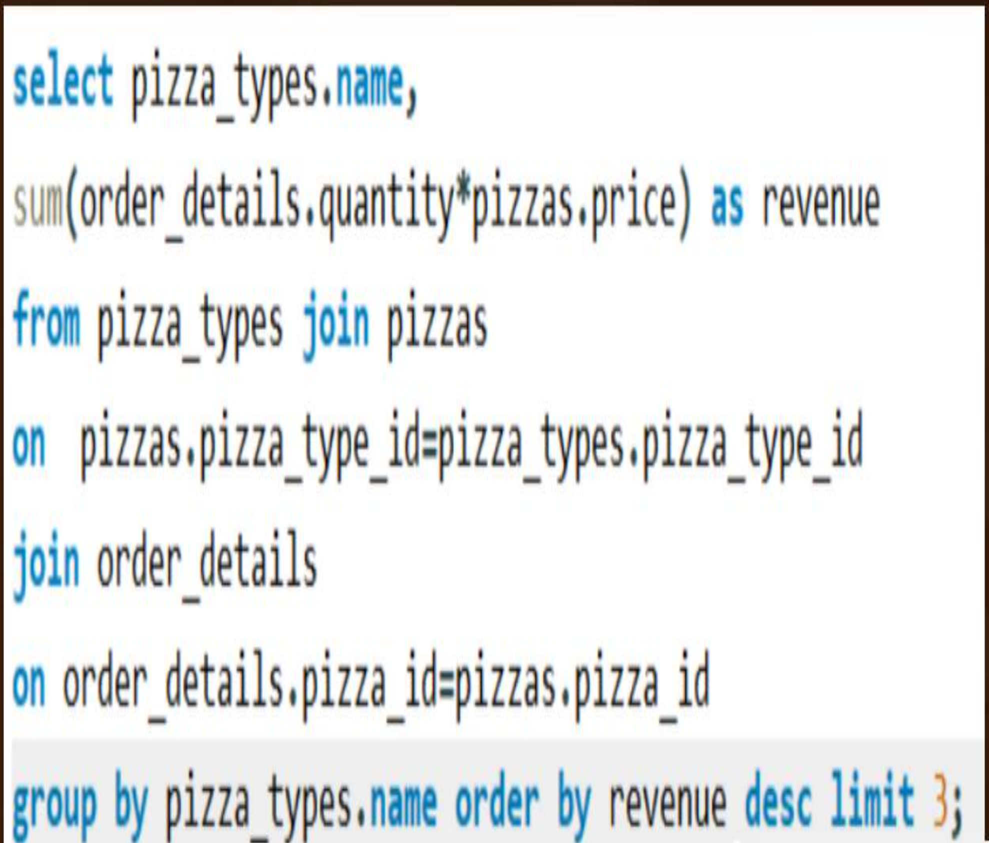


```sql
select round(avg(quantity),0) as avrage_quantity from
(select orders.order_date, sum(order_details.quantity) as quantity
from orders join order_details
on orders.order_id=order_details.order_id
group by orders.order_date) as order_quantity;
```

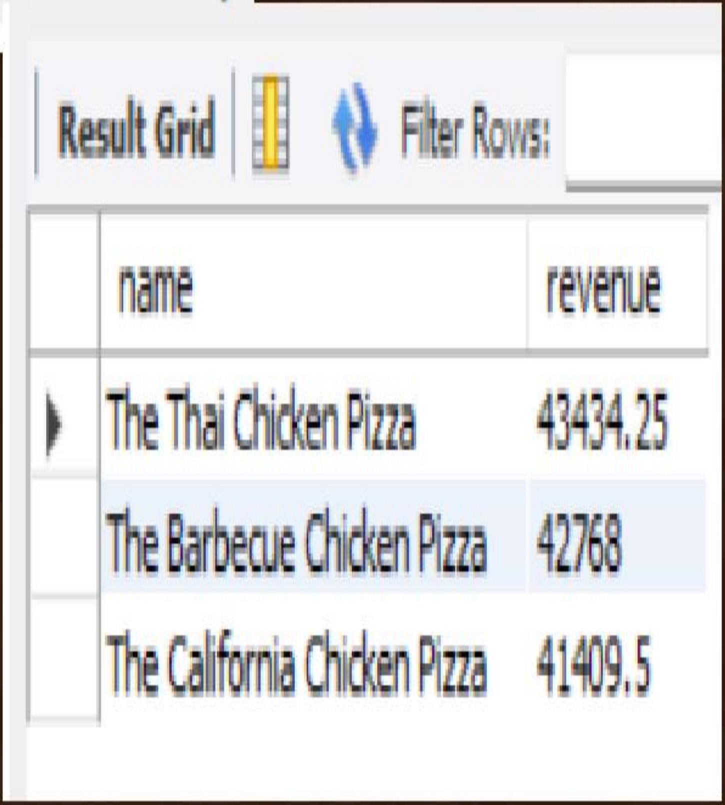


```
avrage_quantity
138
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

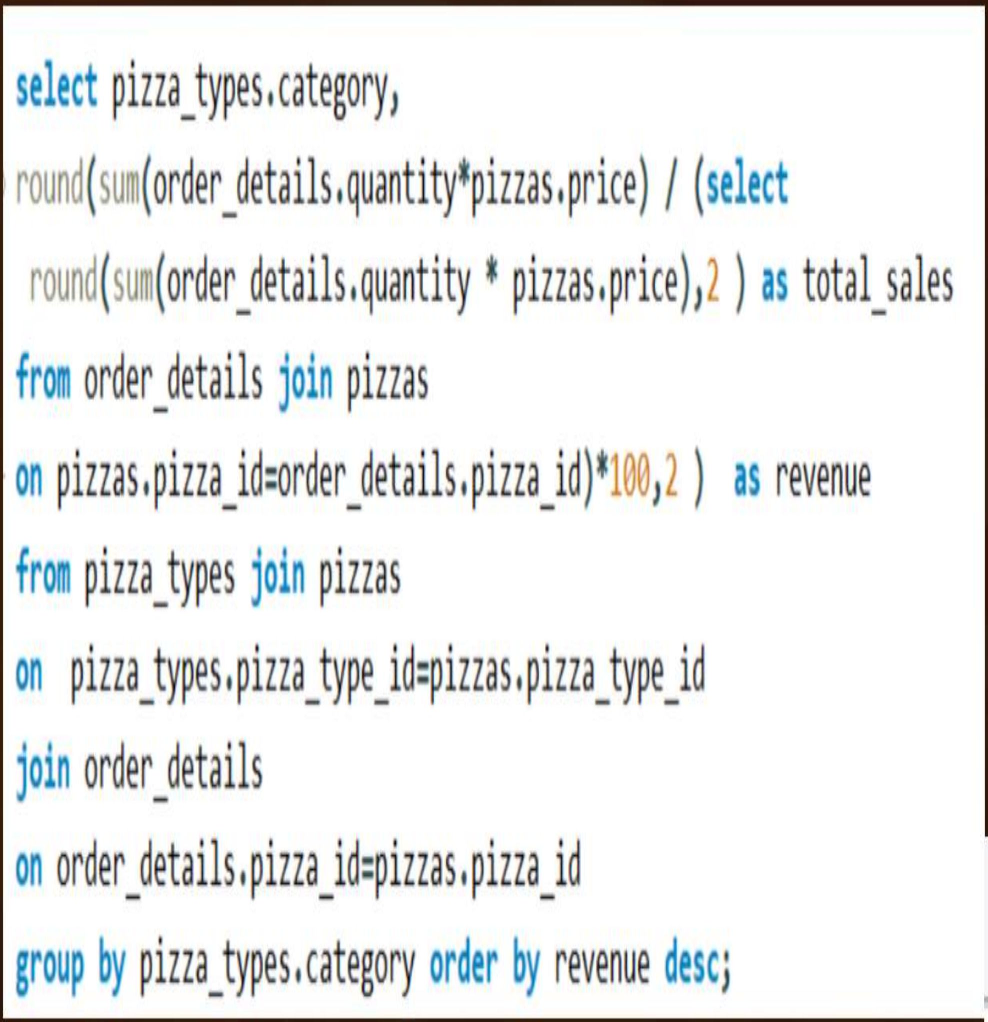


```sql
select pizza_types.name,
sum(order_details.quantity*pizzas.price) as revenue
from pizza_types join pizzas
on  pizzas.pizza_type_id=pizza_types.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.name order by revenue desc limit 3;
```

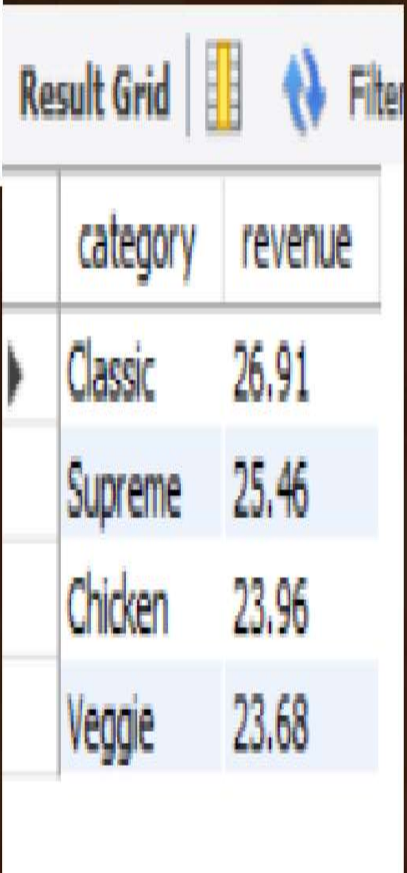


name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

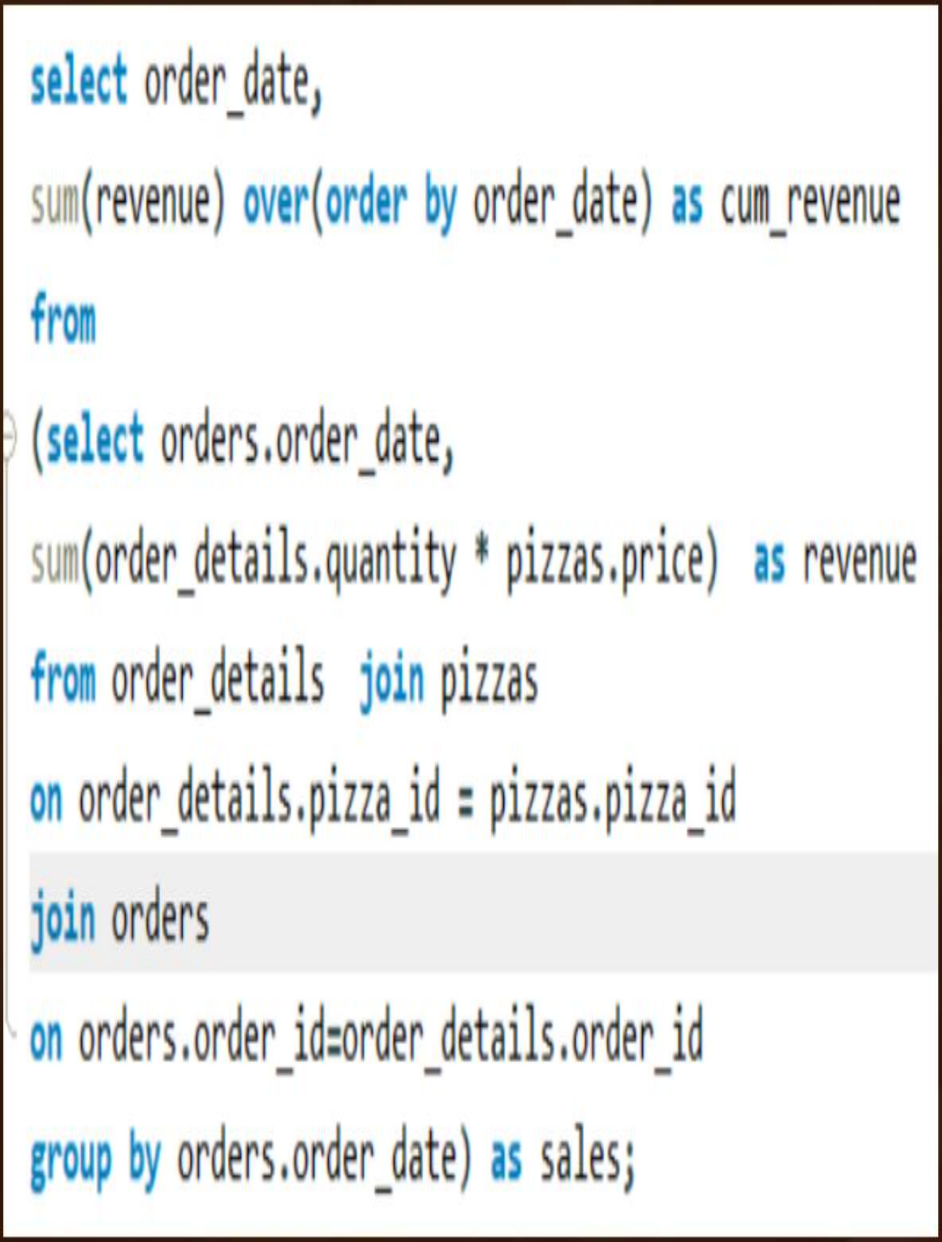


```
select pizza_types.category,
round(sum(order_details.quantity*pizzas.price) / (select
 round(sum(order_details.quantity * pizzas.price),2 ) as total_sales
from order_details join pizzas
on pizzas.pizza_id=order_details.pizza_id)*100,2 )  as revenue
from pizza_types join pizzas
on  pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_details
on order_details.pizza_id=pizzas.pizza_id
group by pizza_types.category order by revenue desc;
```

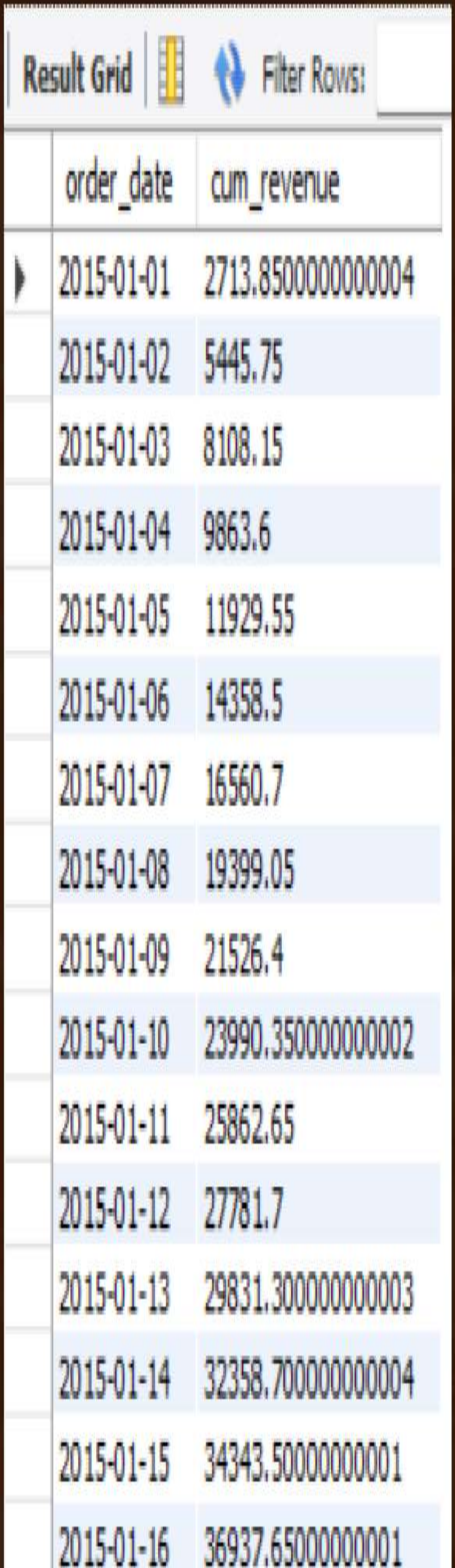


category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

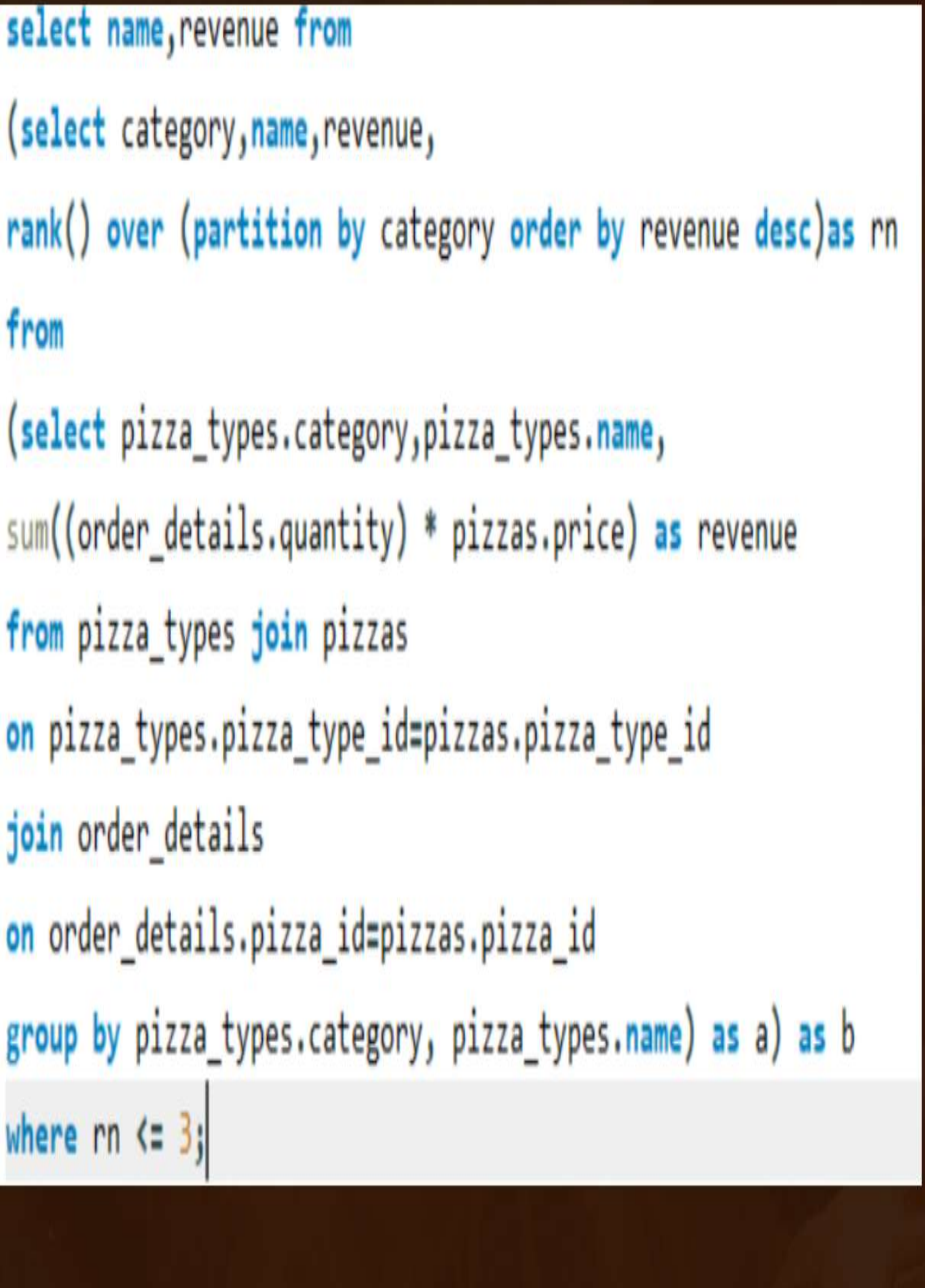


```
select order_date,
sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity * pizzas.price)  as revenue
from order_details  join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id=order_details.order_id
group by orders.order_date) as sales;
```

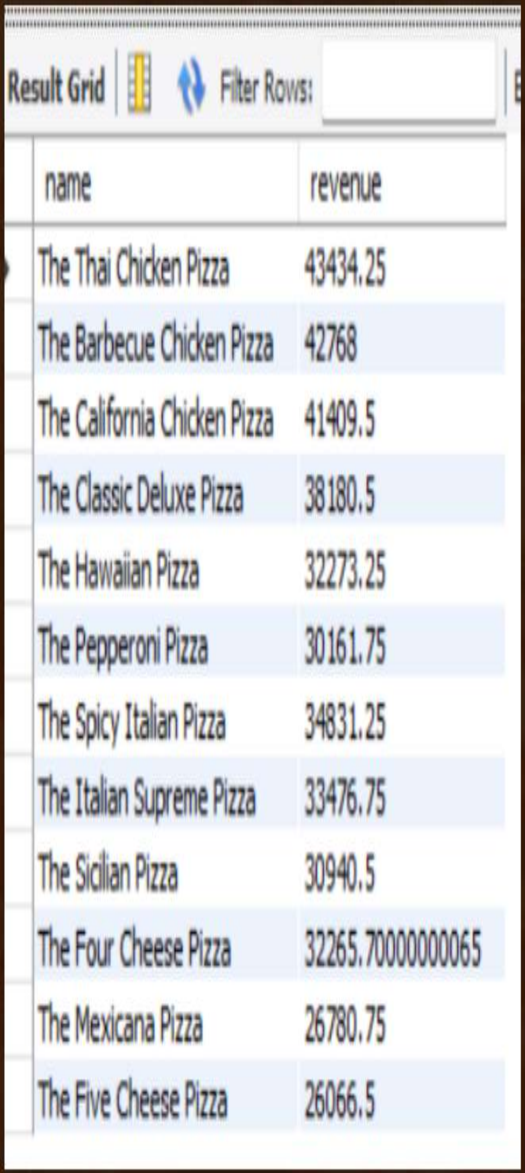


order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.



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



name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

Pizza Hut Presentation

“THANK YOU FOR YOUR ATTENTION”

See You Next