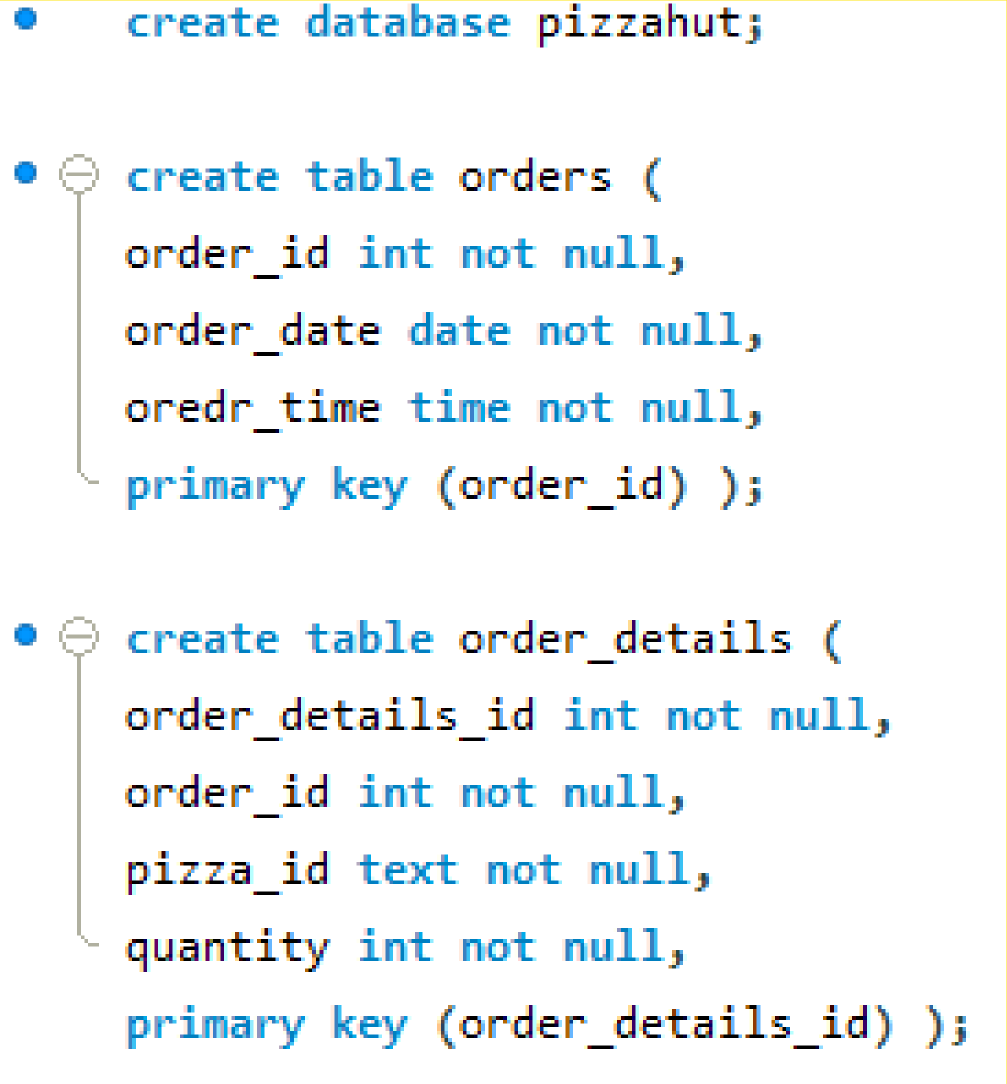
SQL PIZZAHUT_SALES PROJECT

EXPLANATION

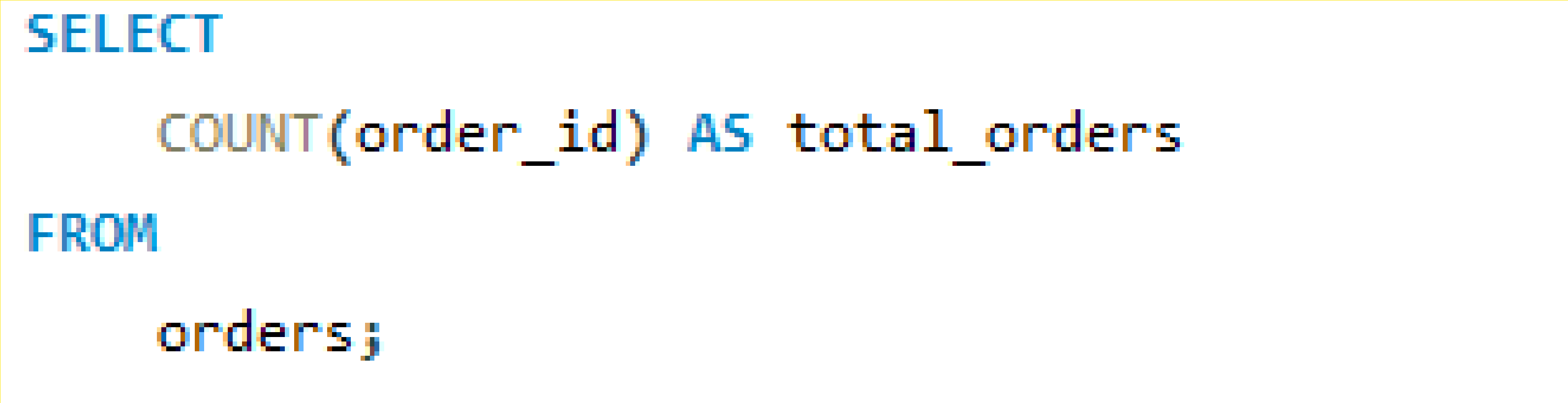
My name is Muhammad Karim Abdullah Khan, In this Pizzahut Sales Project I have utilize the sql quires to solve the questions that was related to pizzahut sales

CREATE DATABASE & TABLES

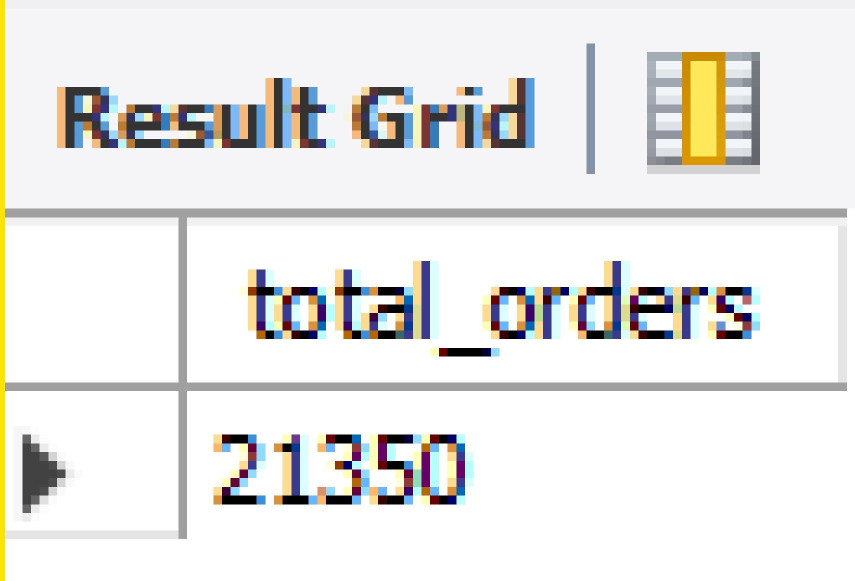


```sql
create database pizzahut;

create table orders (
order_id int not null,
order_date date not null,
oredr_time time not null,
primary key (order_id) );

create table order_details (
order_details_id int not null,
order_id int not null,
pizza_id text not null,
quantity int not null,
primary key (order_details_id) );
```

RETRIEVE THE TOTAL NUMBERS OF ORDERS PLACED

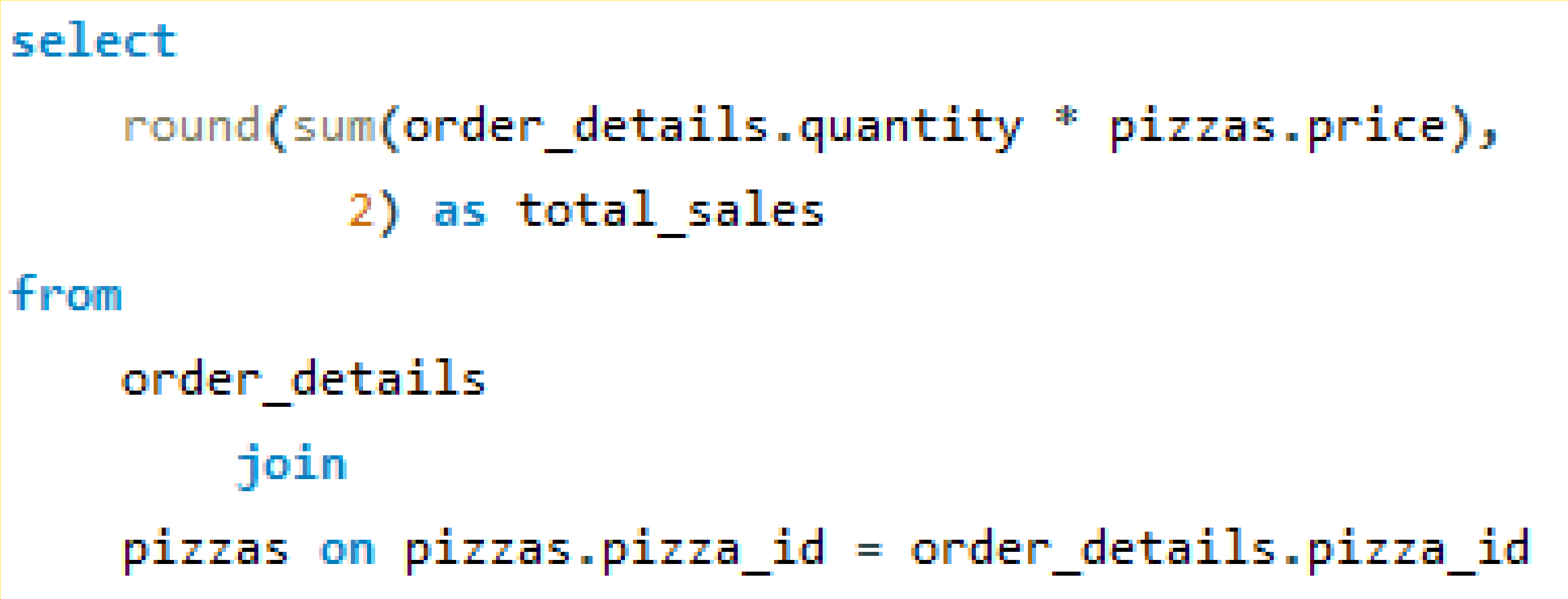


```
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

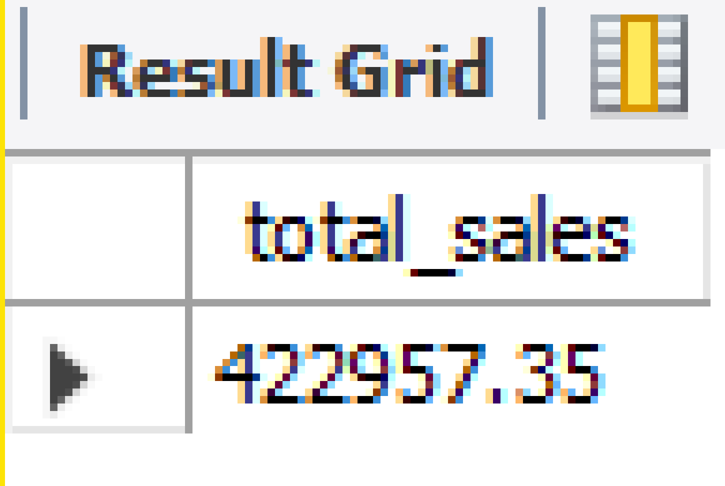


total_orders
21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

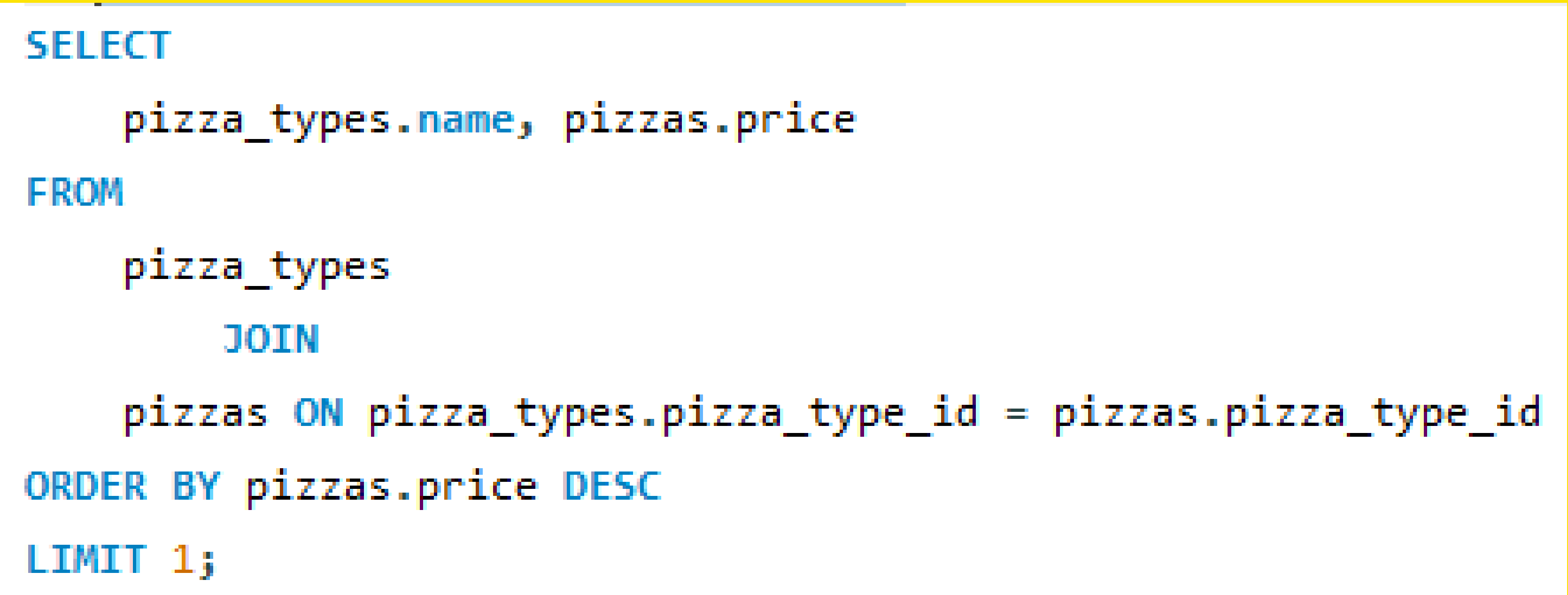


```sql
select
    round(sum(order_details.quantity * pizzas.price),
            2) as total_sales
from
    order_details
        join
    pizzas on pizzas.pizza_id = order_details.pizza_id
```

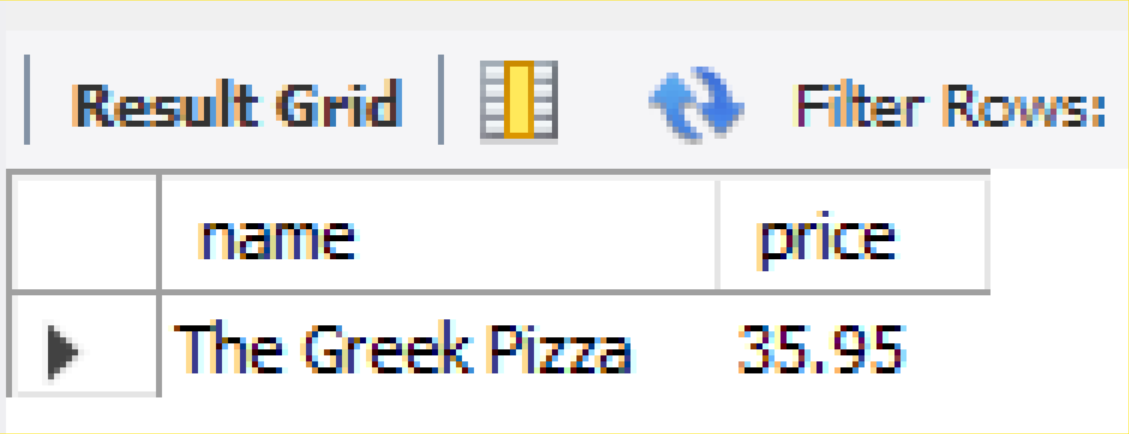


	total_sales
▶	422957.35

IDENTIFY THE HIGHEST PRICED PIZZA

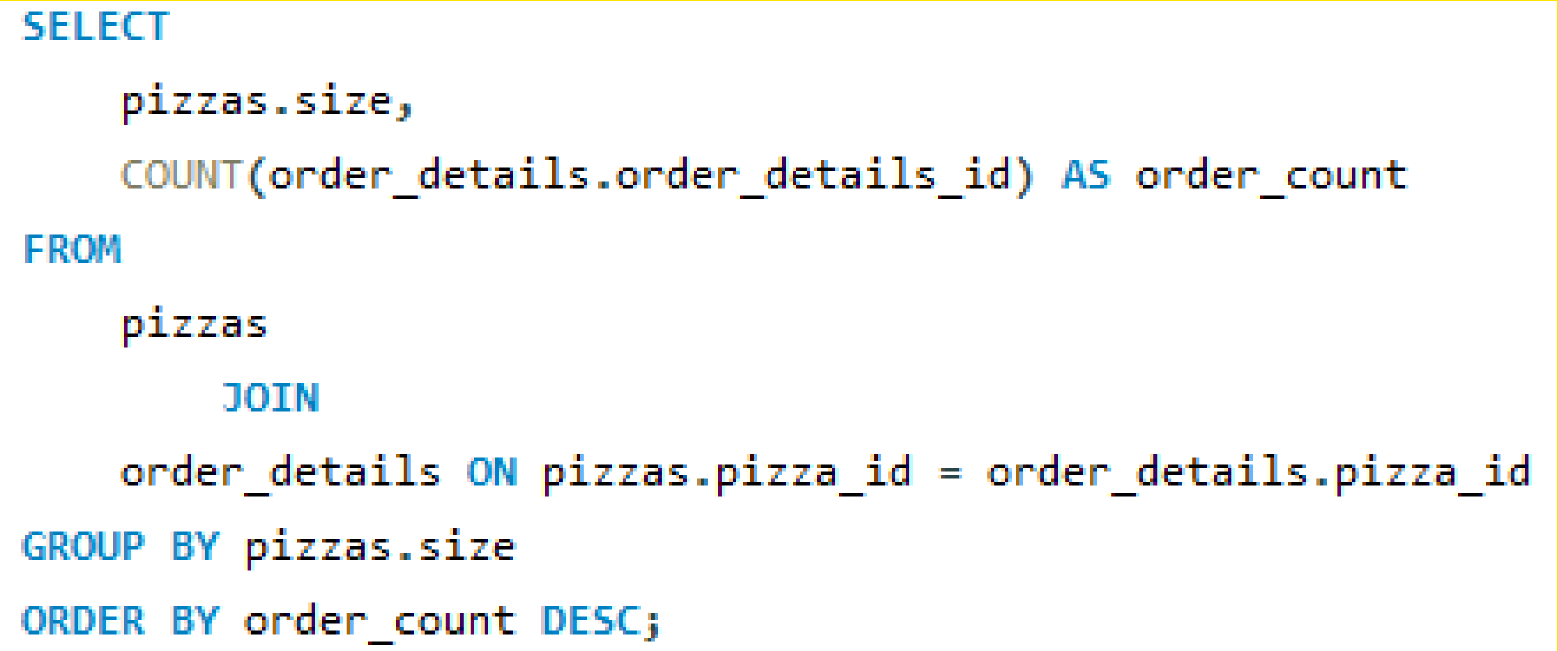


```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

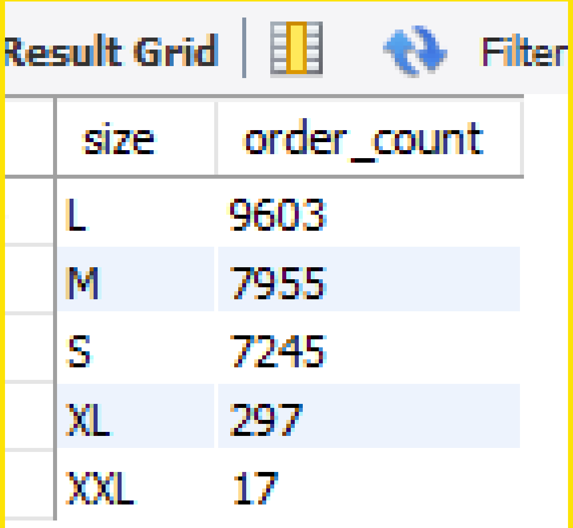


name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

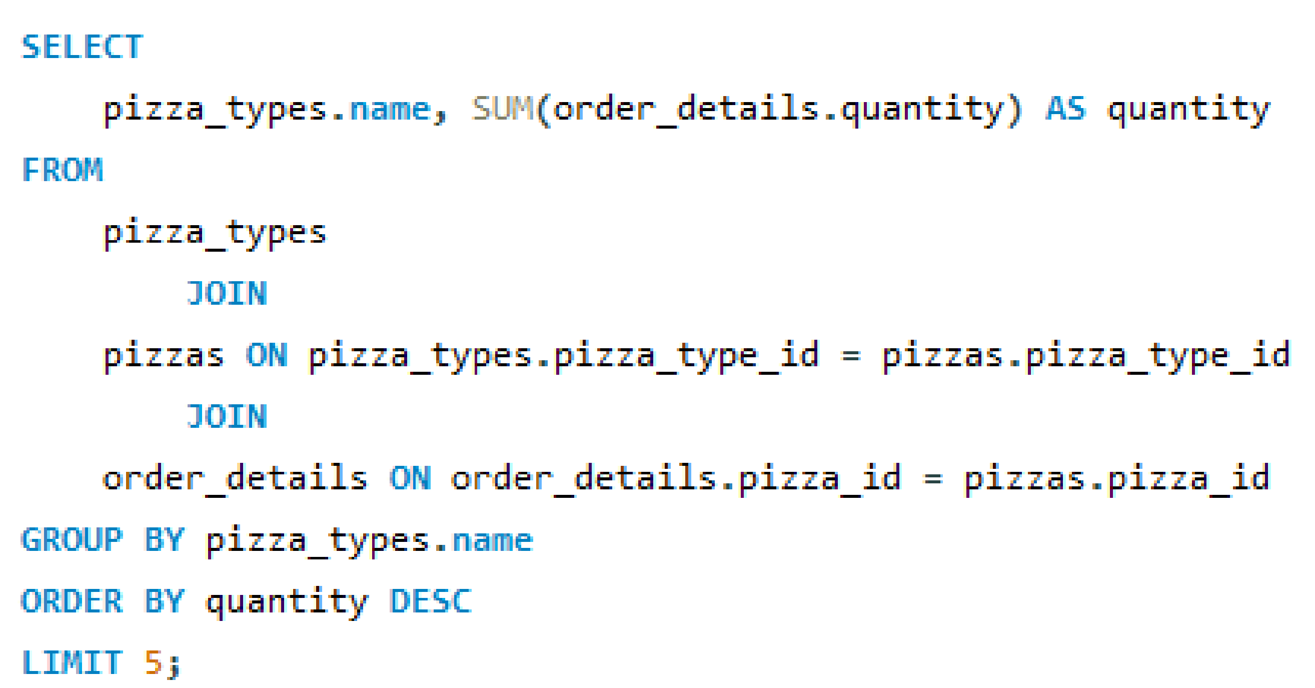


```sql
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

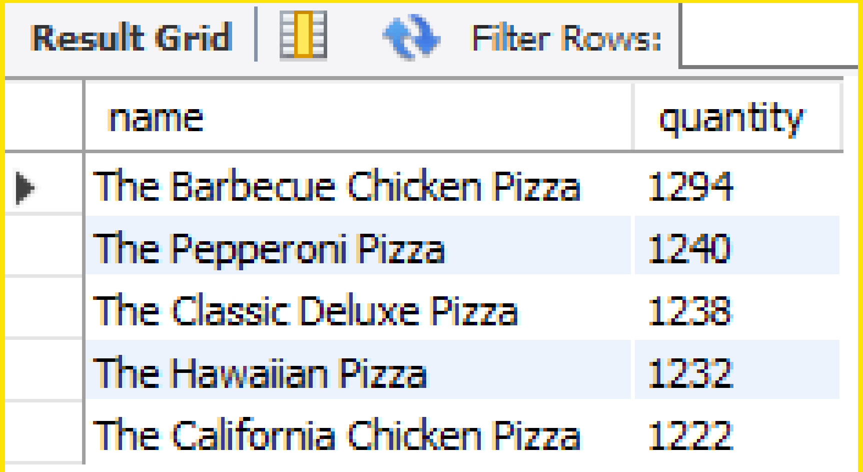


size	order_count
L	9603
M	7955
S	7245
XL	297
XXL	17

LIST THE MOST TOP 5 ORDERED PIZZAS TYPES ALONG WITH THEIE QUANTITY

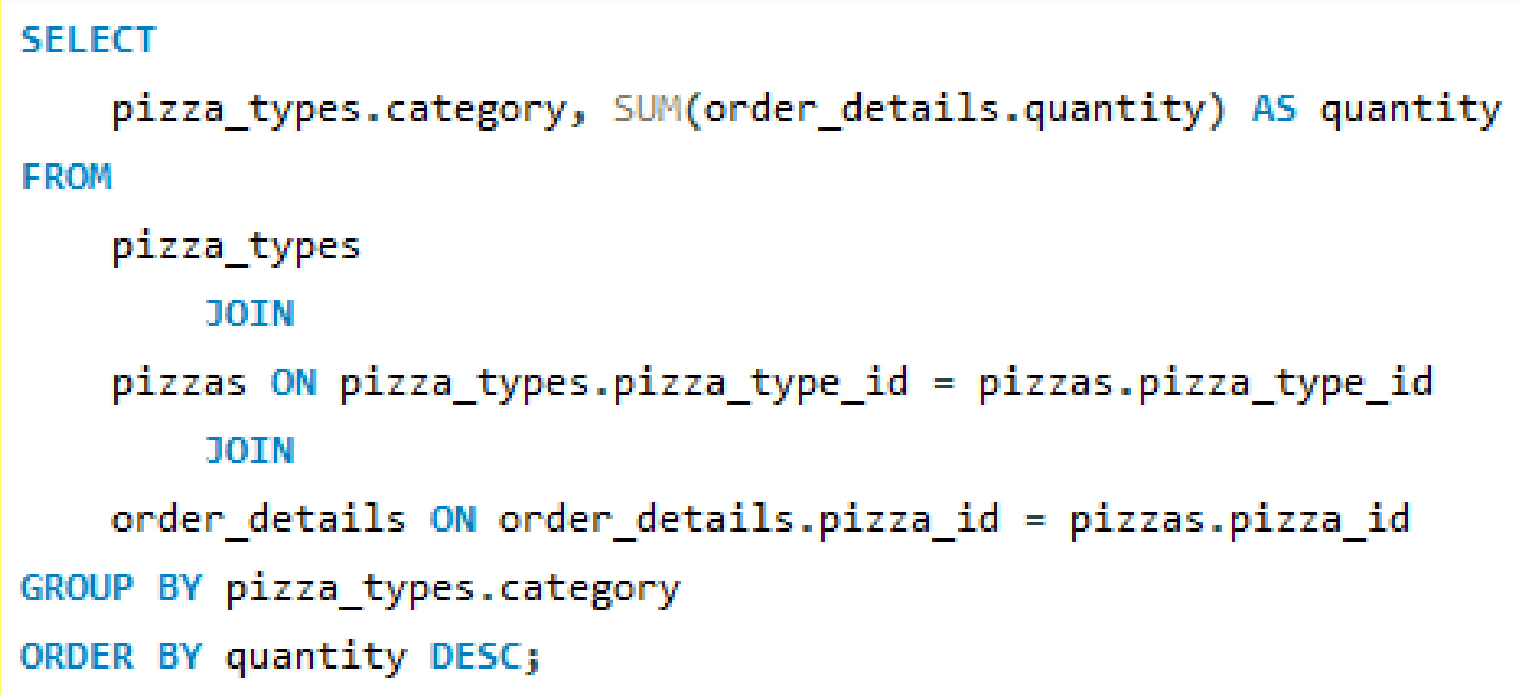


```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

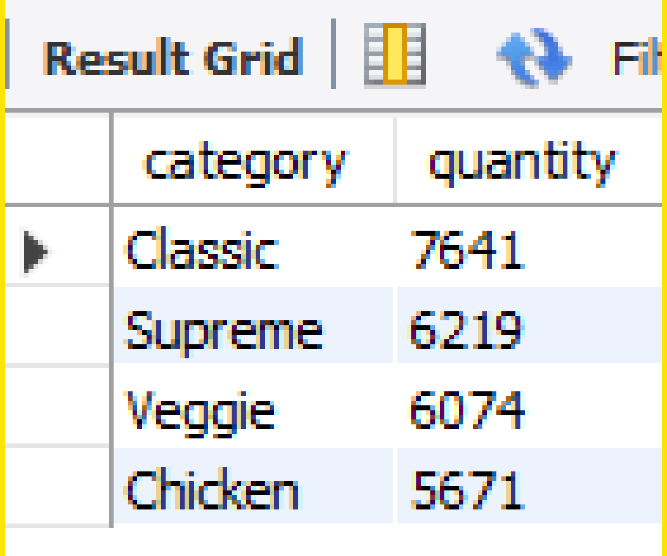


name	quantity
The Barbecue Chicken Pizza	1294
The Pepperoni Pizza	1240
The Classic Deluxe Pizza	1238
The Hawaiian Pizza	1232
The California Chicken Pizza	1222

JOIN THE NECESSARY TABLE TO FIND THE TOTAL QUANTITY OF EACH PIZZAS CATEGORY ORDERED

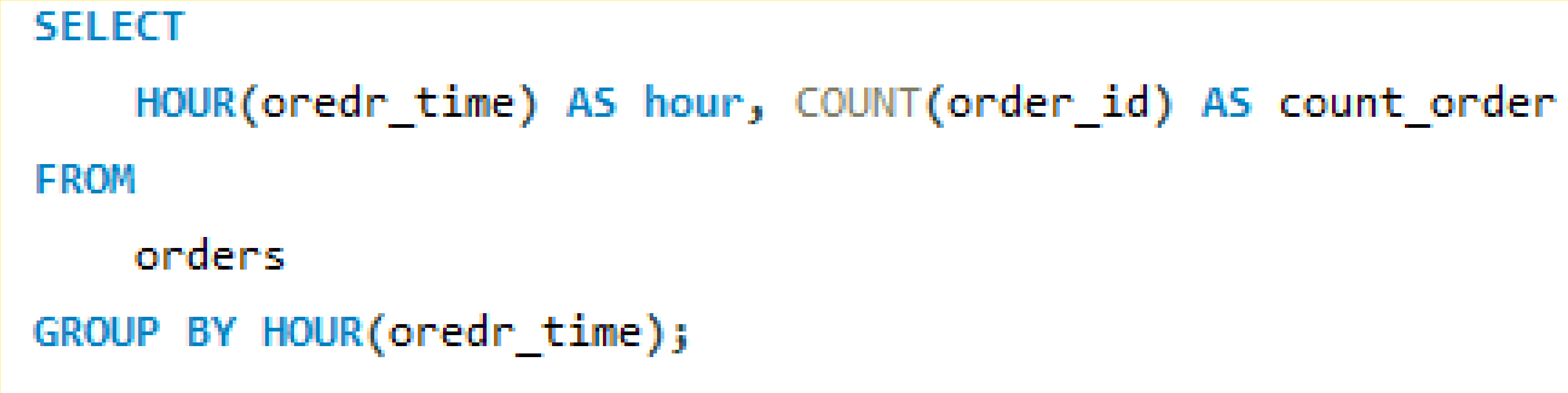


```
SELECT
    pizza_types.category, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

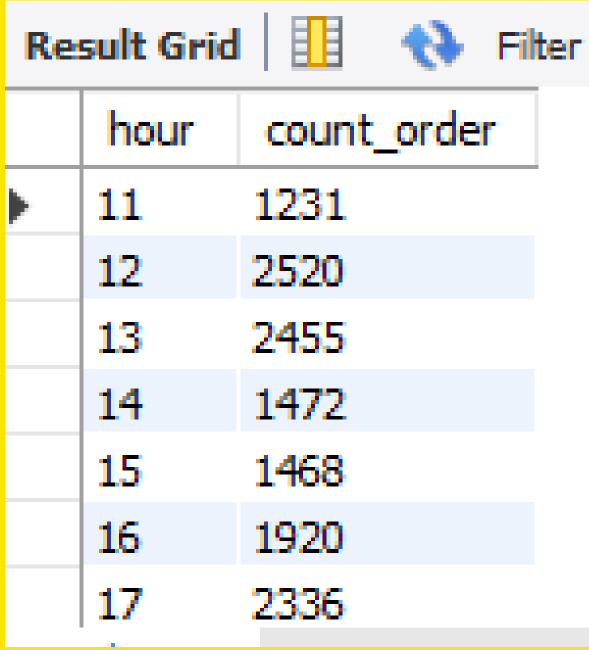


category	quantity
Classic	7641
Supreme	6219
Veggie	6074
Chicken	5671

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

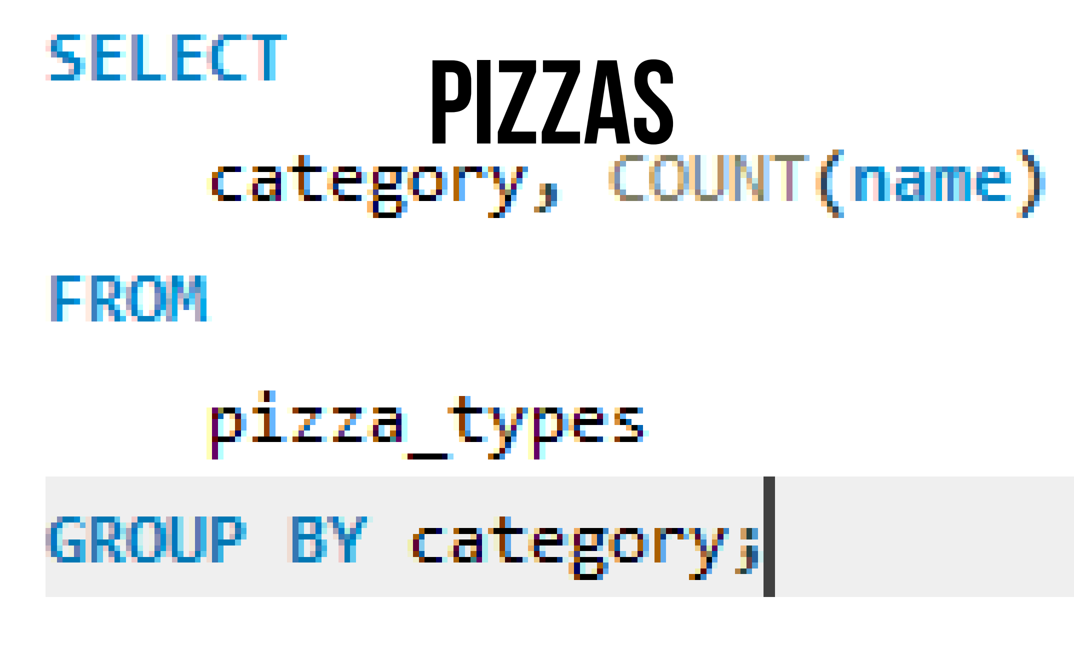


```
SELECT
    HOUR(oredr_time) AS hour, COUNT(order_id) AS count_order
FROM
    orders
GROUP BY HOUR(oredr_time);
```

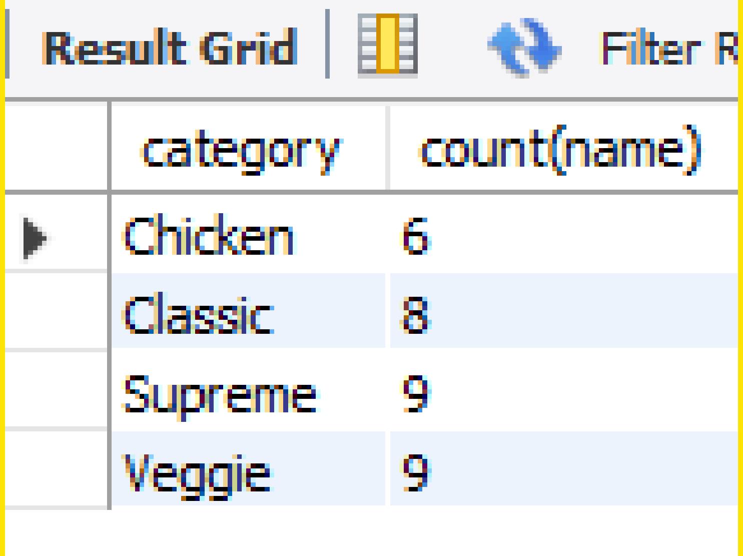


hour	count_order
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336

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

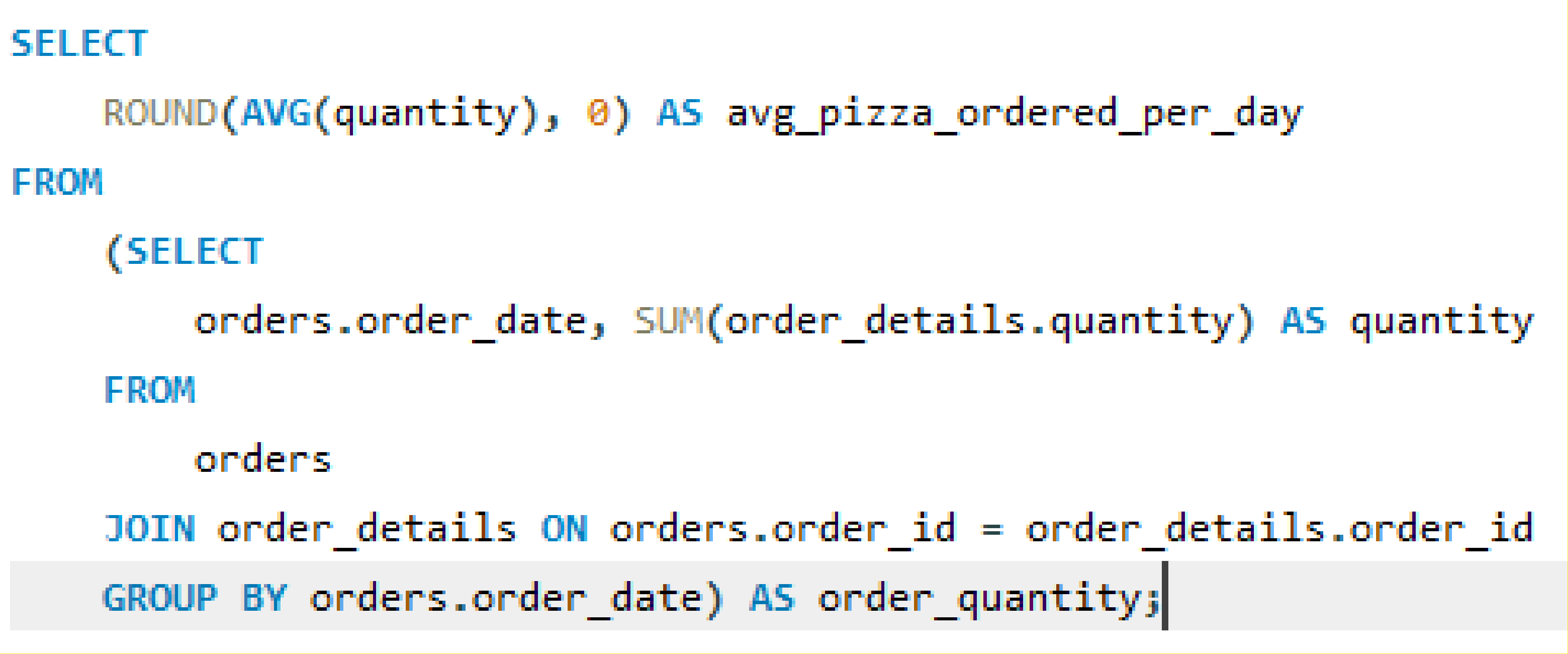


```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

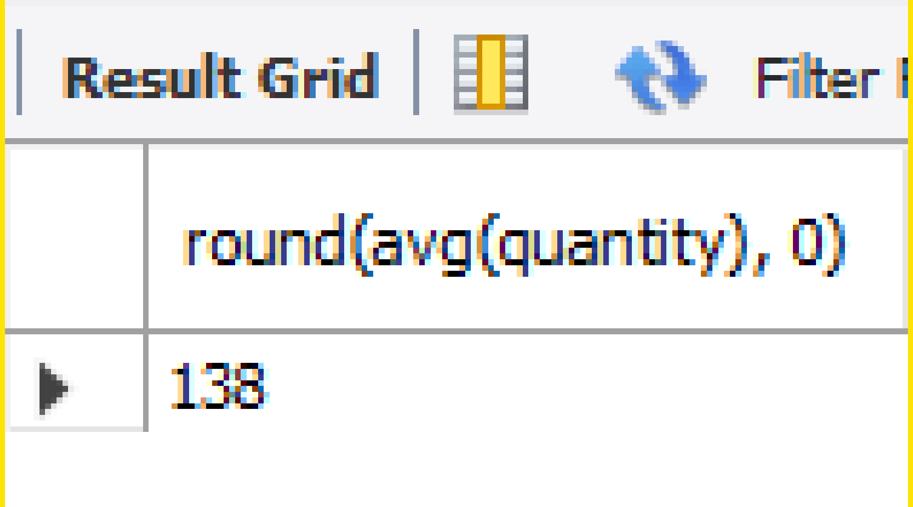


category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

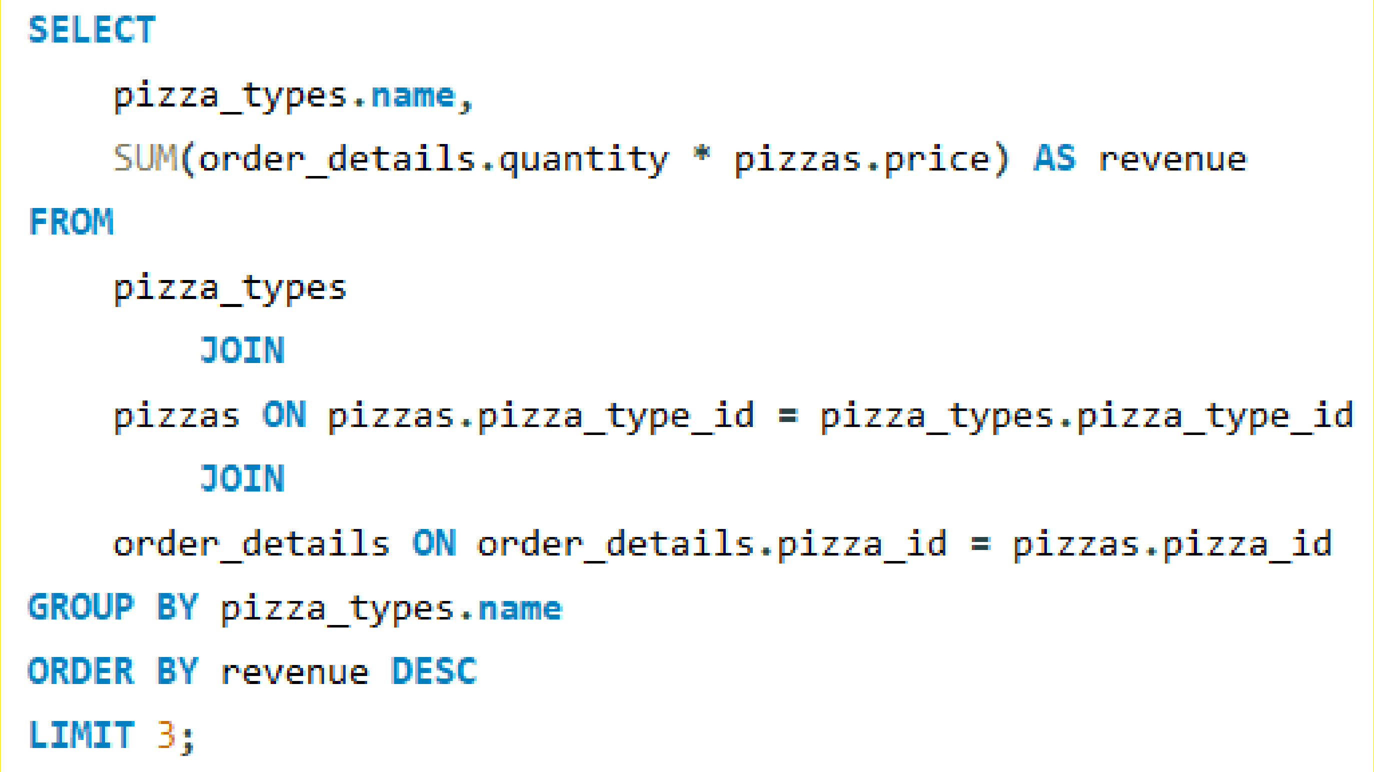


```sql
SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

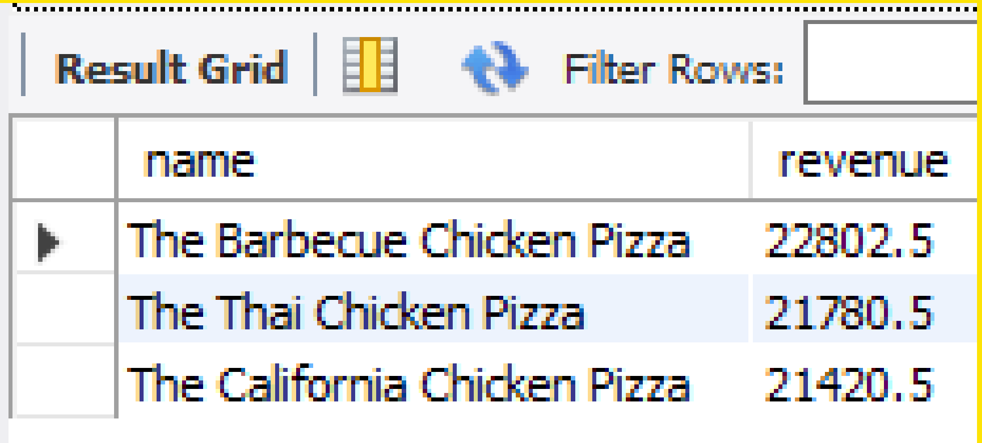


```
round(avg(quantity), 0)
138
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

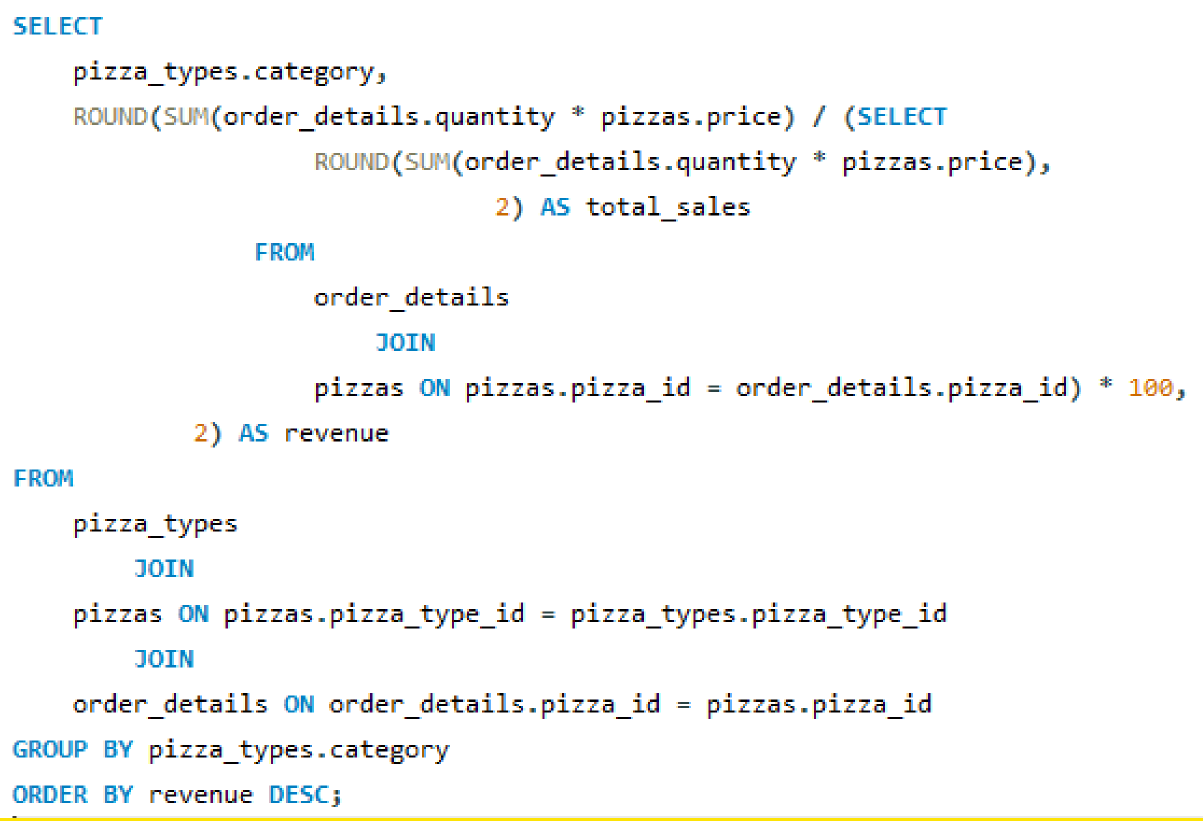


```sql
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

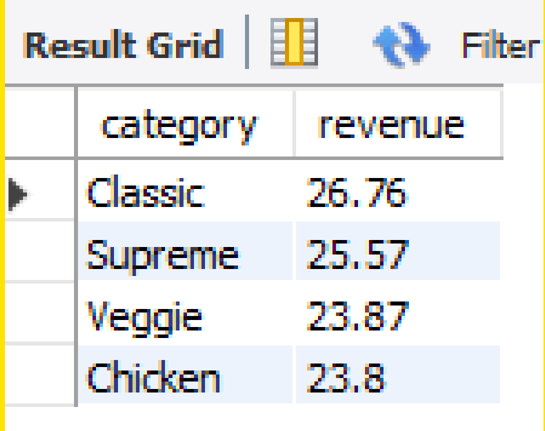


name	revenue
The Barbecue Chicken Pizza	22802.5
The Thai Chicken Pizza	21780.5
The California Chicken Pizza	21420.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPES OF TOTAL REVENUE

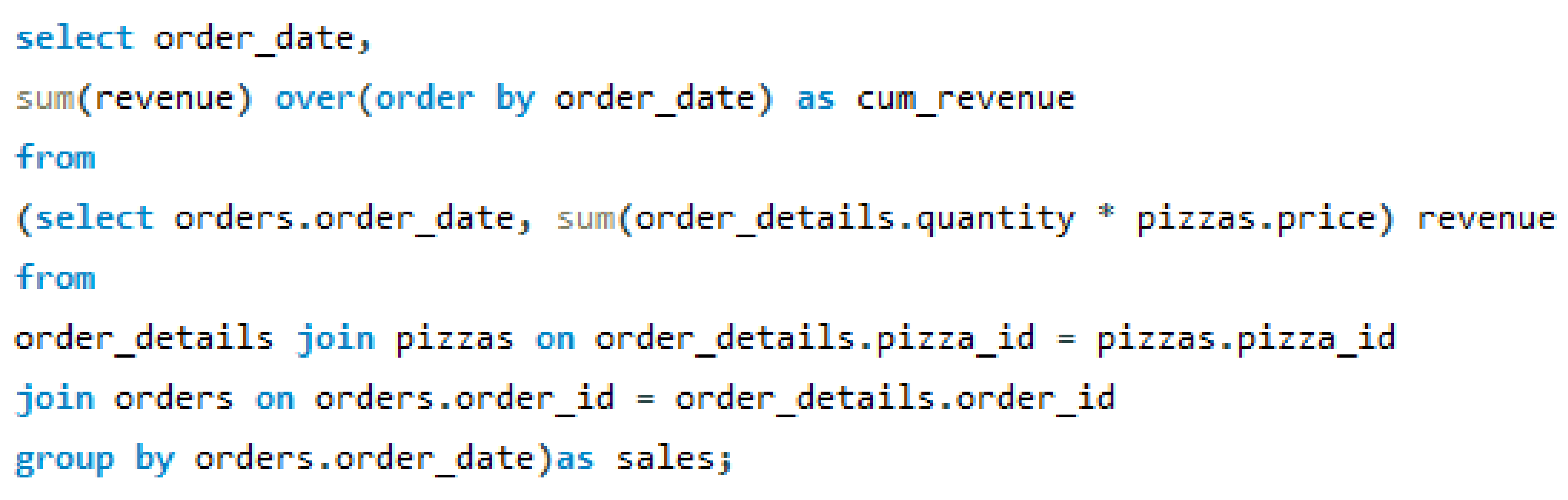


```
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),
                                2) AS total_sales
                FROM
                    order_details
                        JOIN
                    pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
            2) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

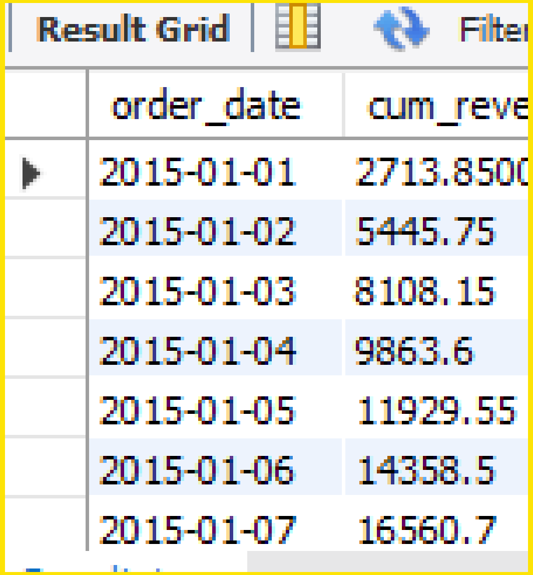


category	revenue
Classic	26.76
Supreme	25.57
Veggie	23.87
Chicken	23.8

ANALYZE THE COMMULATIVE 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) 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_reve
2015-01-01	2713.850
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