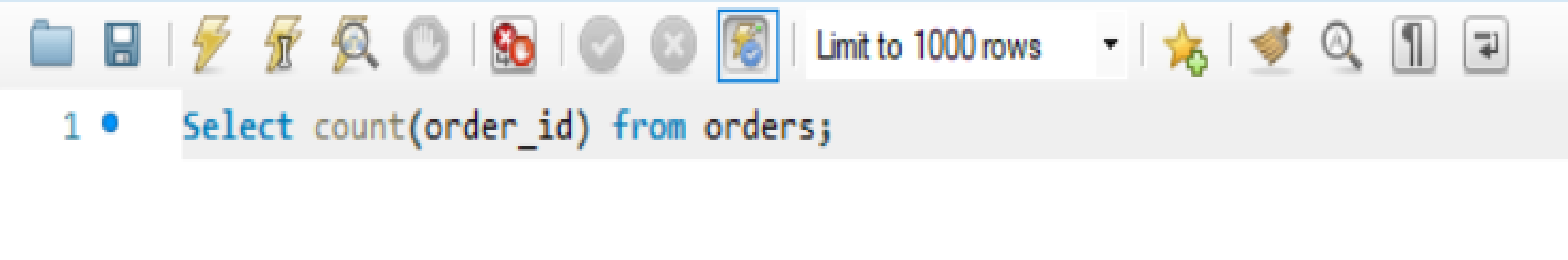
PIZZA SALES ANALYSIS – SQL PROJECT

OVERVIEW

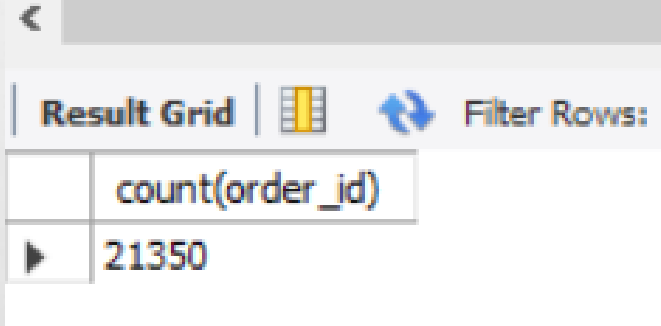
THE PIZZA COMPANY HAS A DATABASE THAT CONTAINS INFORMATION ABOUT THEIR ORDERS, CUSTOMERS, AND PRODUCTS. THEY WANT ME TO CALCULATE SOME METRICS AND TRENDS THAT CAN HELP THEM UNDERSTAND HOW THEIR BUSINESS IS DOING AND WHAT THEY CAN DO TO IMPROVE IT.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED



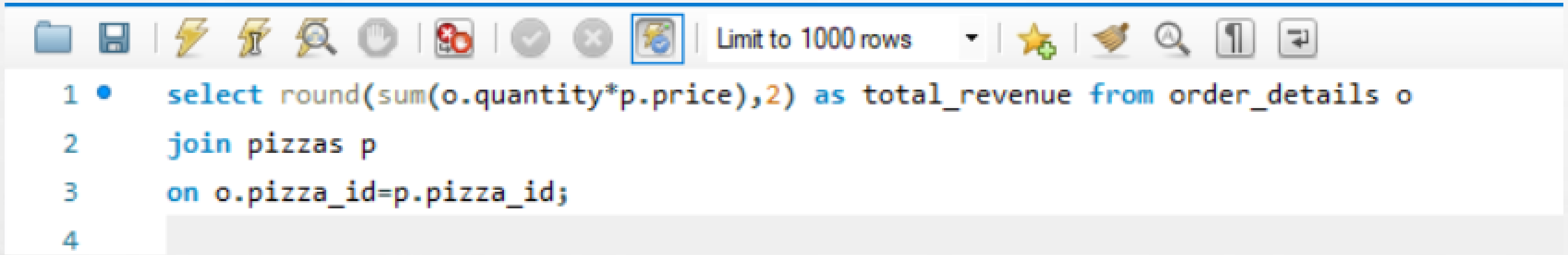
Limit to 1000 rows

```sql
Select count(order_id) from orders;
```

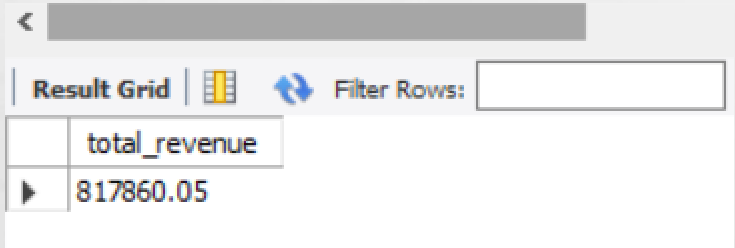


```
count(order_id)
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

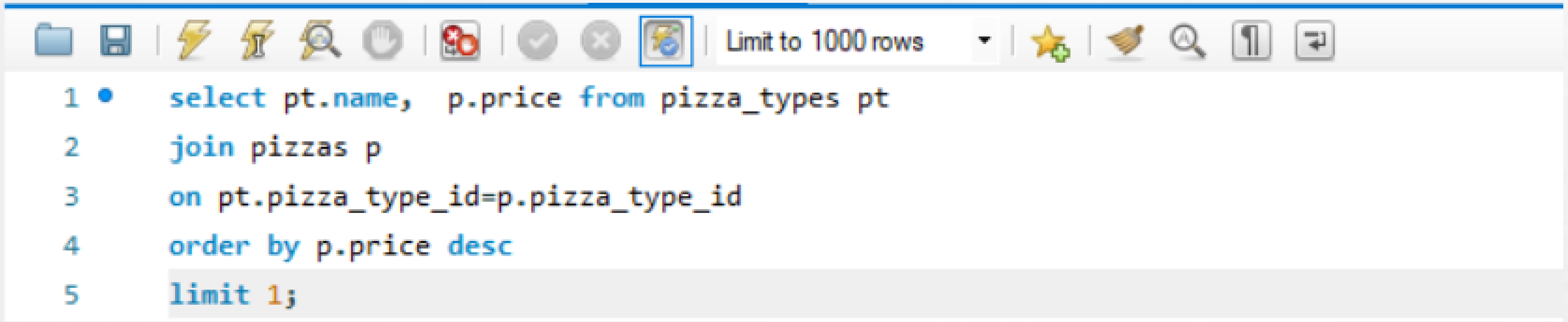


```sql
select round(sum(o.quantity*p.price),2) as total_revenue from order_details o
join pizzas p
on o.pizza_id=p.pizza_id;
```

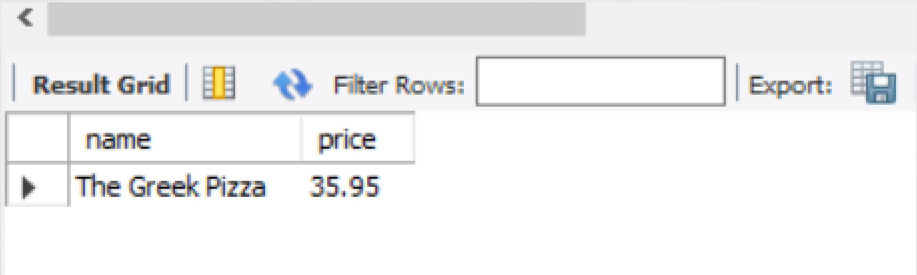


```
total_revenue
817860.05
```

IDENTIFY THE HIGHEST-PRICED PIZZA

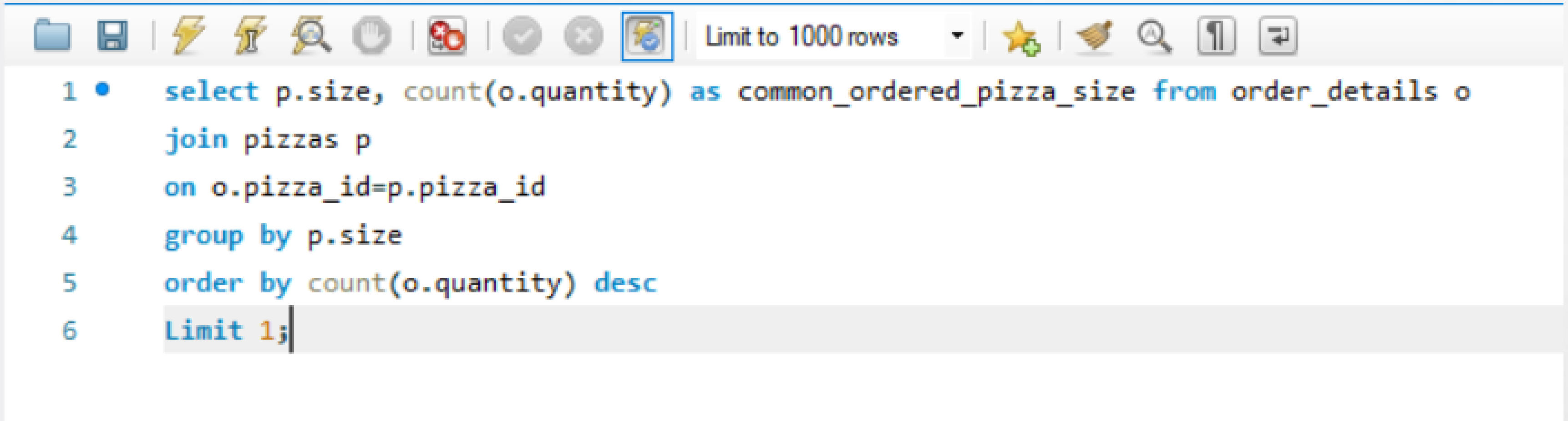


```
select pt.name,  p.price from pizza_types pt
join pizzas p
on pt.pizza_type_id=p.pizza_type_id
order by p.price desc
limit 1;
```

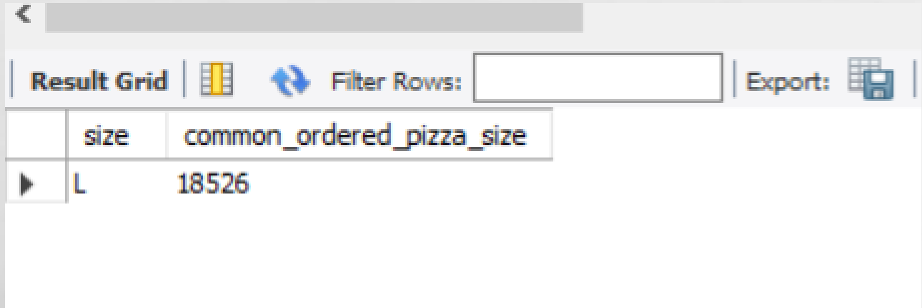


name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

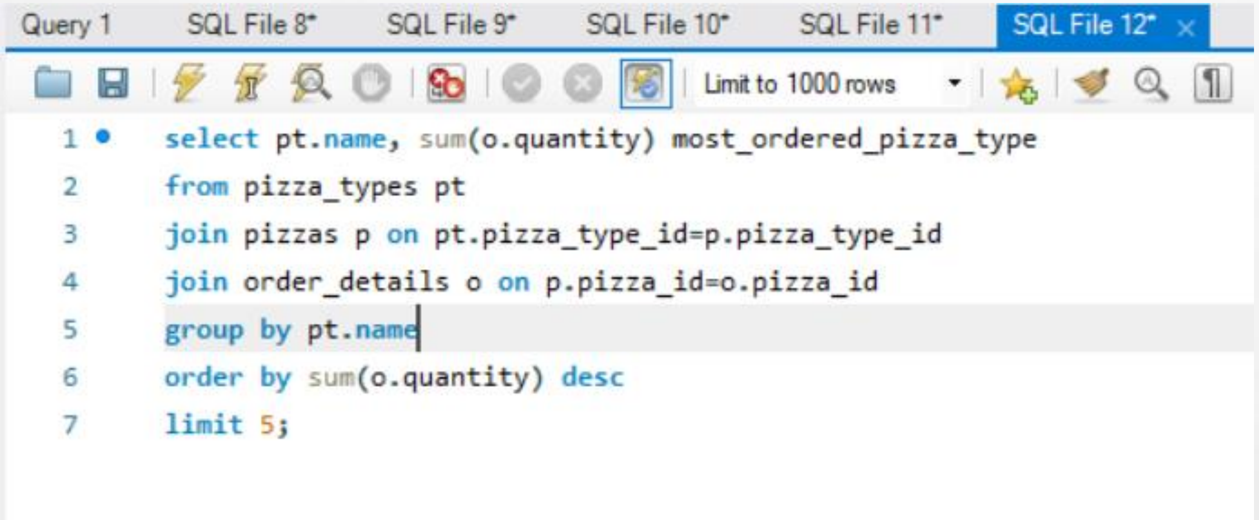


```sql
select p.size, count(o.quantity) as common_ordered_pizza_size from order_details o
join pizzas p
on o.pizza_id=p.pizza_id
group by p.size
order by count(o.quantity) desc
Limit 1;
```

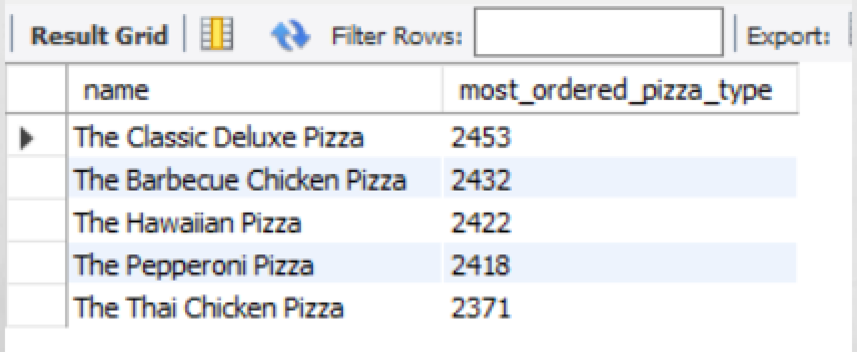


size	common_ordered_pizza_size
L	18526

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

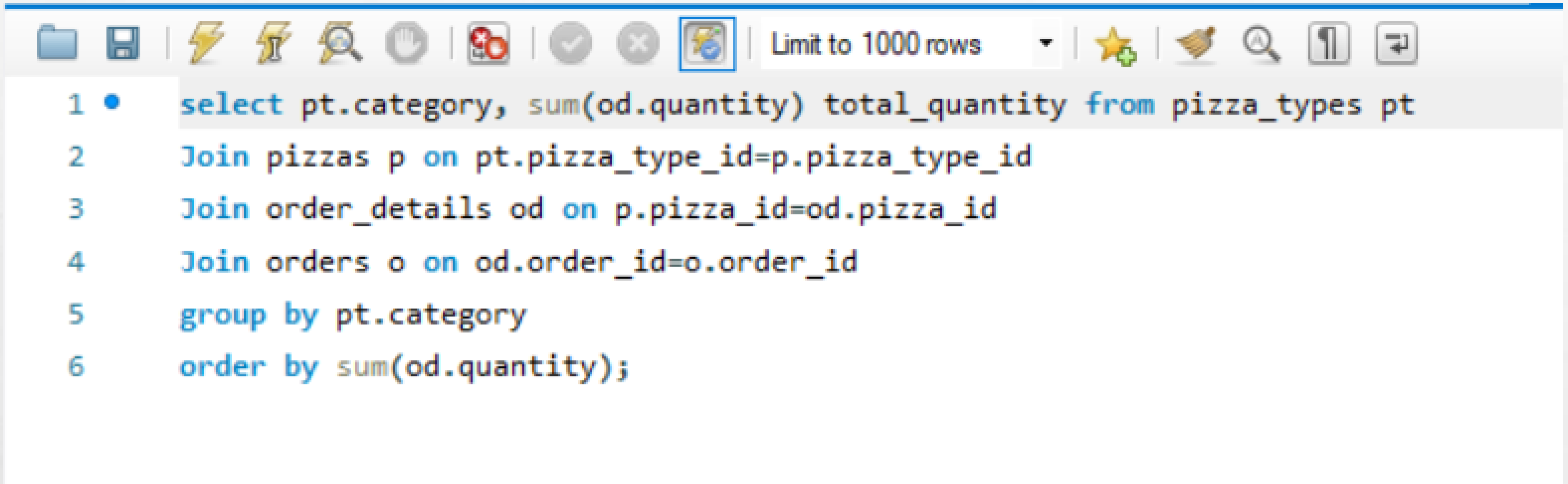


```
select pt.name, sum(o.quantity) most_ordered_pizza_type
from pizza_types pt
join pizzas p on pt.pizza_type_id=p.pizza_type_id
join order_details o on p.pizza_id=o.pizza_id
group by pt.name
order by sum(o.quantity) desc
limit 5;
```

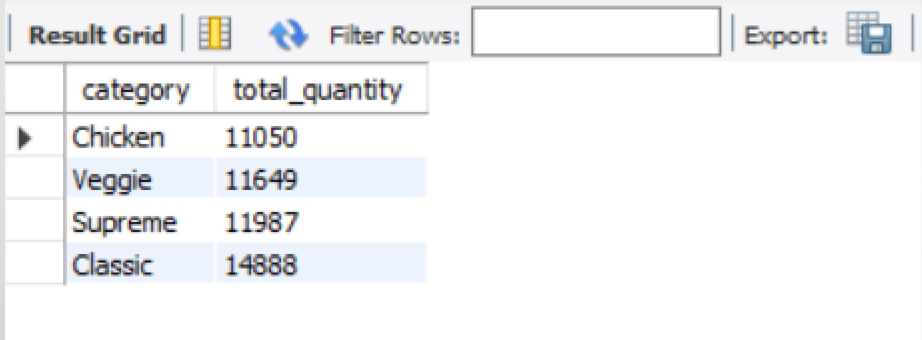


name	most_ordered_pizza_type
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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

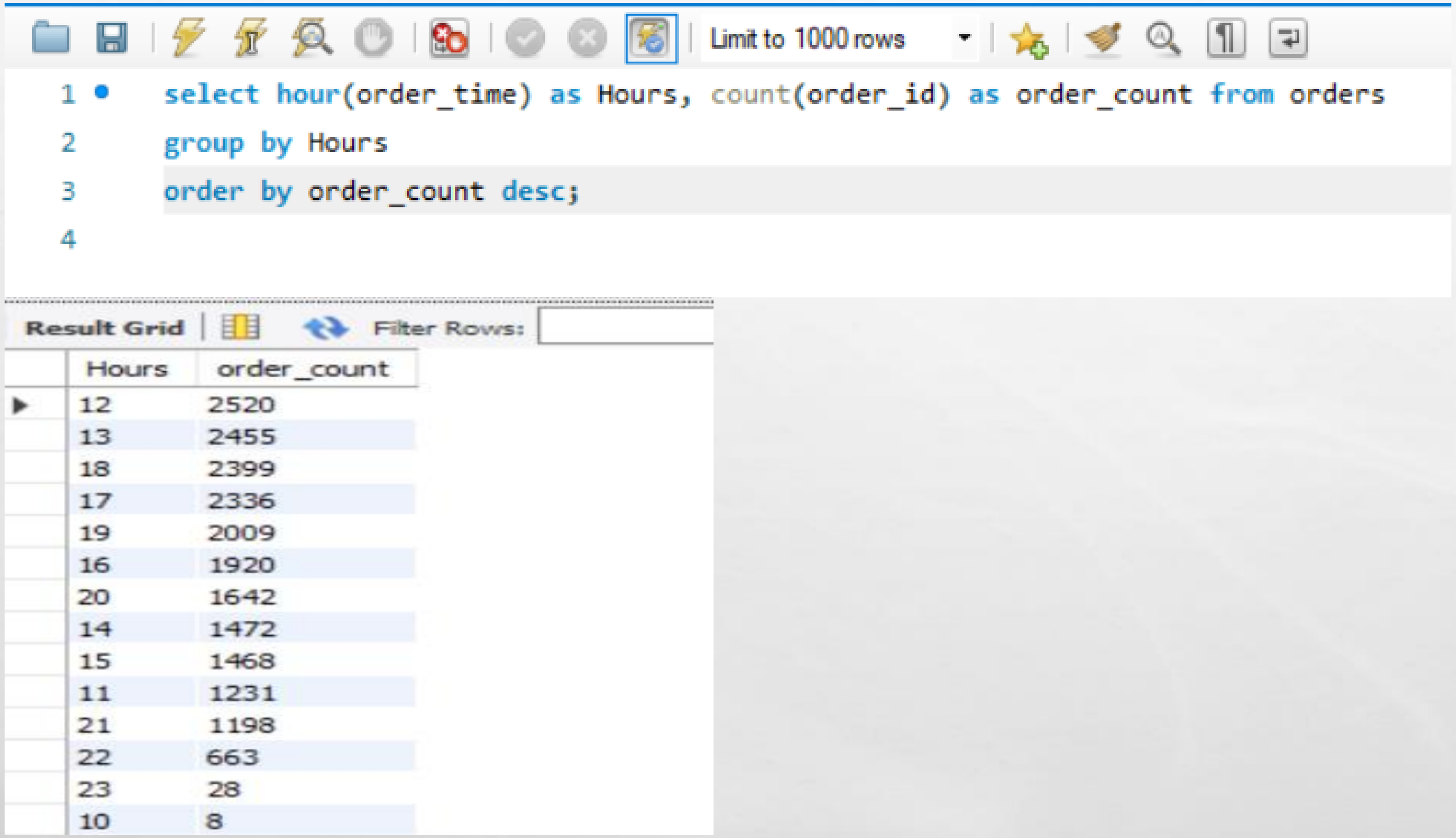


```
select pt.category, sum(od.quantity) total_quantity from pizza_types pt
Join pizzas p on pt.pizza_type_id=p.pizza_type_id
Join order_details od on p.pizza_id=od.pizza_id
Join orders o on od.order_id=o.order_id
group by pt.category
order by sum(od.quantity);
```



category	total_quantity
Chicken	11050
Veggie	11649
Supreme	11987
Classic	14888

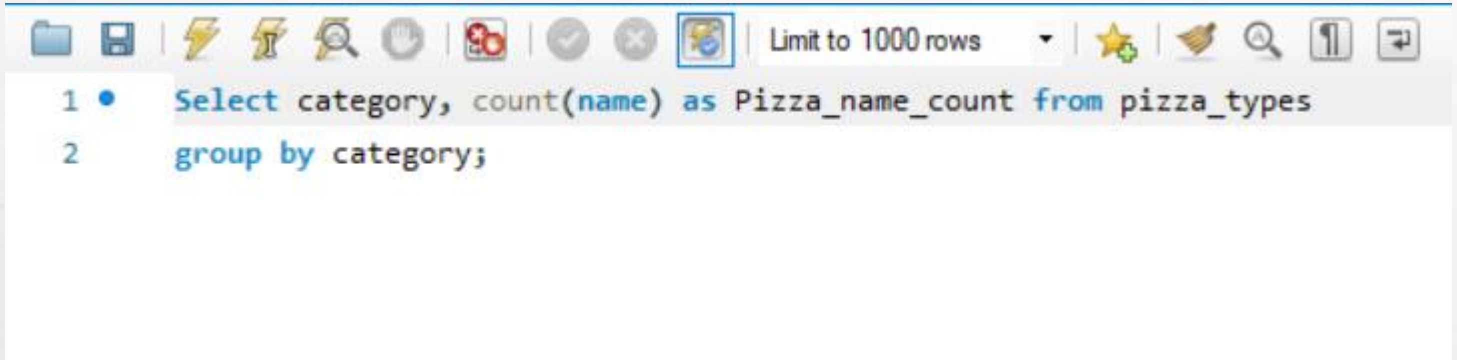
DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.



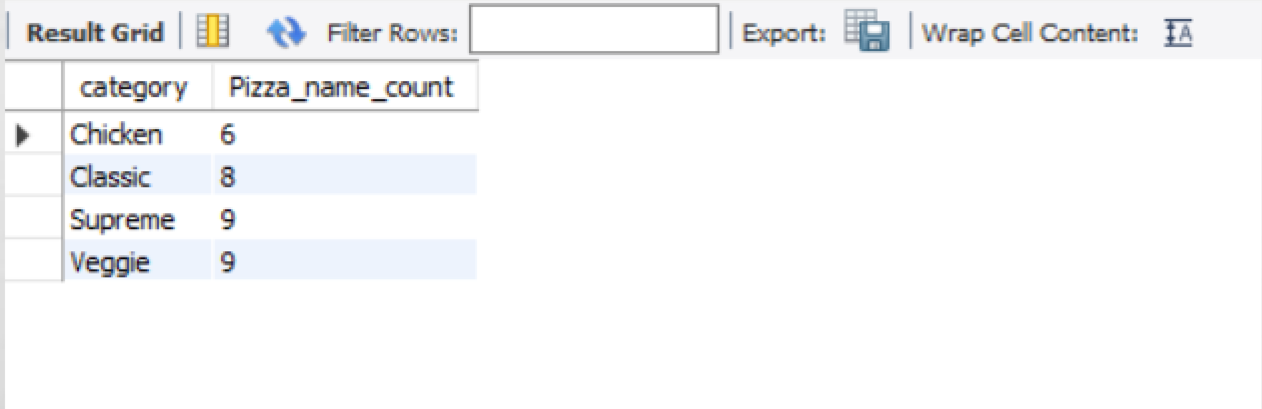
```sql
select hour(order_time) as Hours, count(order_id) as order_count from orders
group by Hours
order by order_count desc;
```

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

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

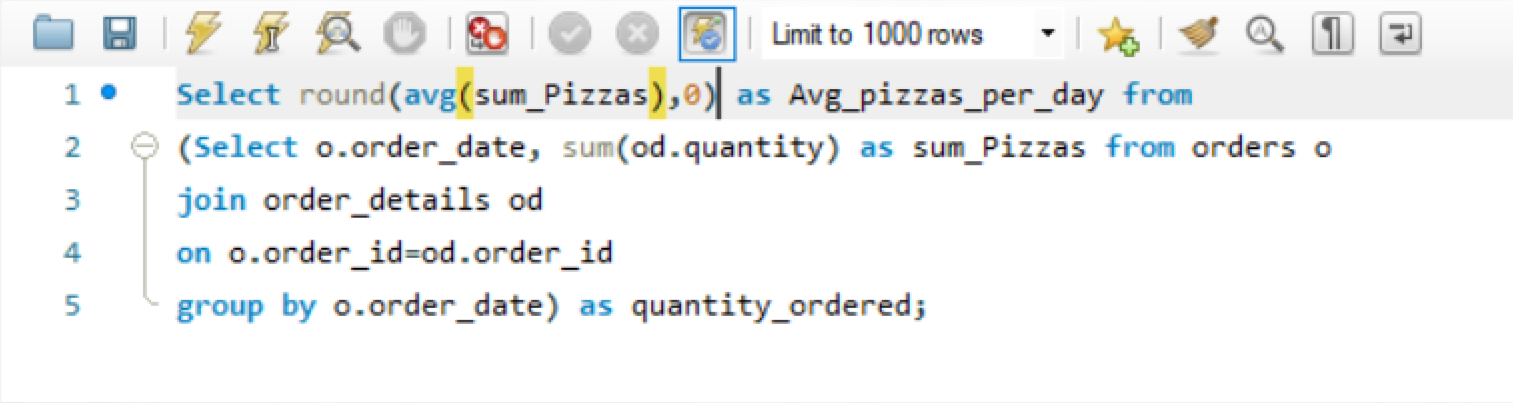


```sql
Select category, count(name) as Pizza_name_count from pizza_types
group by category;
```

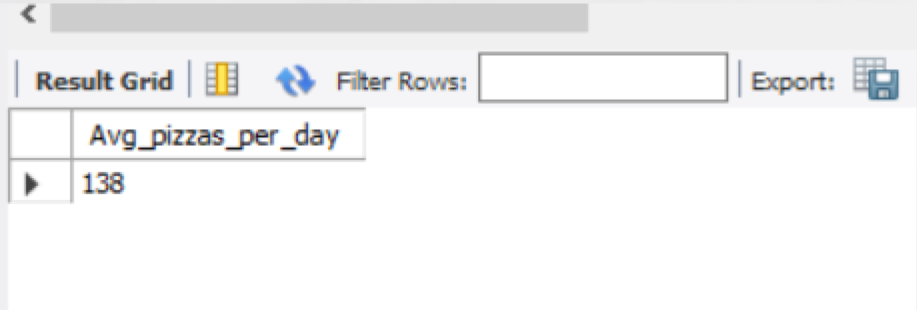


```
category   Pizza_name_count
Chicken    6
Classic    8
Supreme    9
Veggie     9
```

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

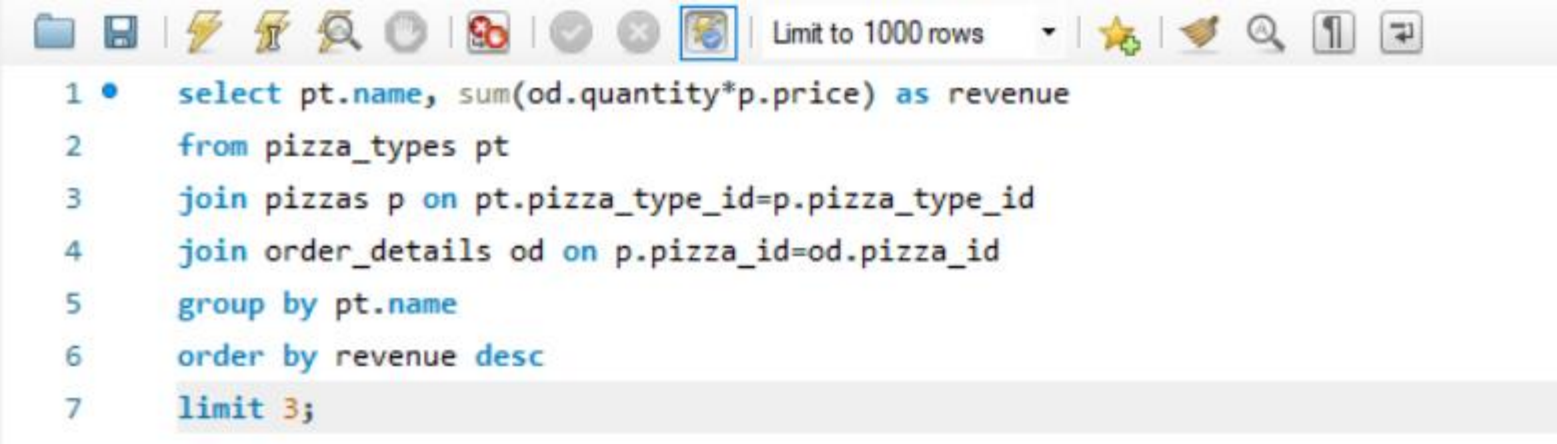


```sql
Select round(avg(sum_Pizzas),0) as Avg_pizzas_per_day from
(Select o.order_date, sum(od.quantity) as sum_Pizzas from orders o
join order_details od
on o.order_id=od.order_id
group by o.order_date) as quantity_ordered;
```

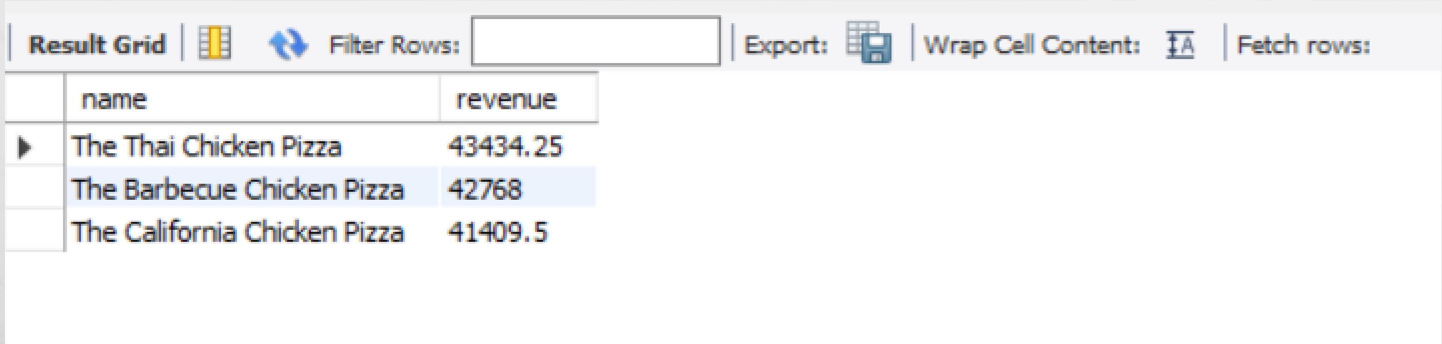


```
Avg_pizzas_per_day
138
```

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

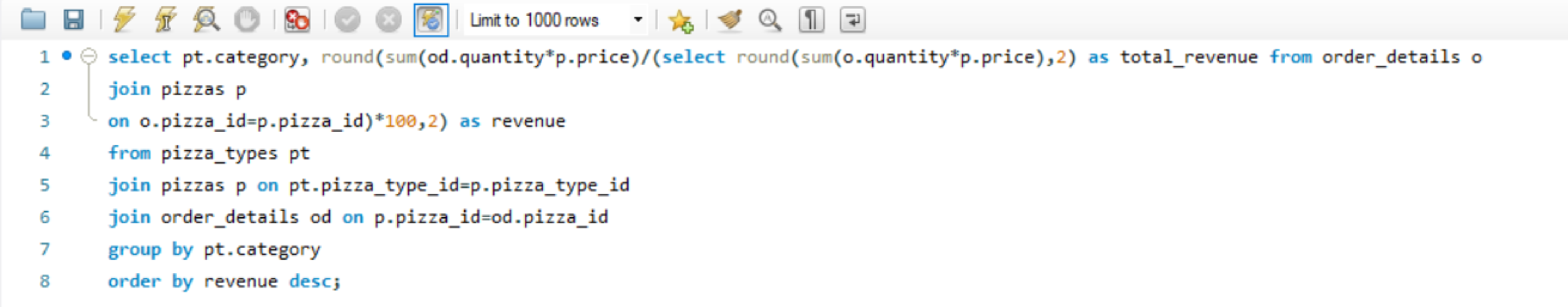


```sql
select pt.name, sum(od.quantity*p.price) as revenue
from pizza_types pt
join pizzas p on pt.pizza_type_id=p.pizza_type_id
join order_details od on p.pizza_id=od.pizza_id
group by pt.name
order by revenue desc
limit 3;
```



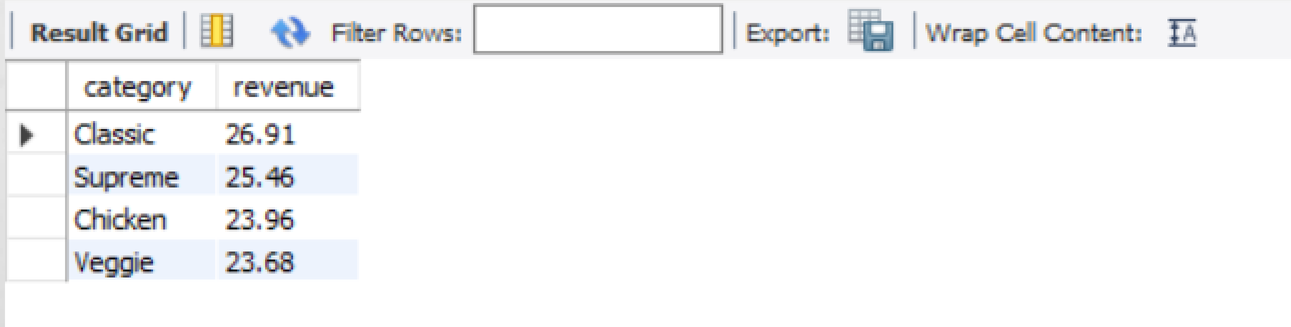
name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

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



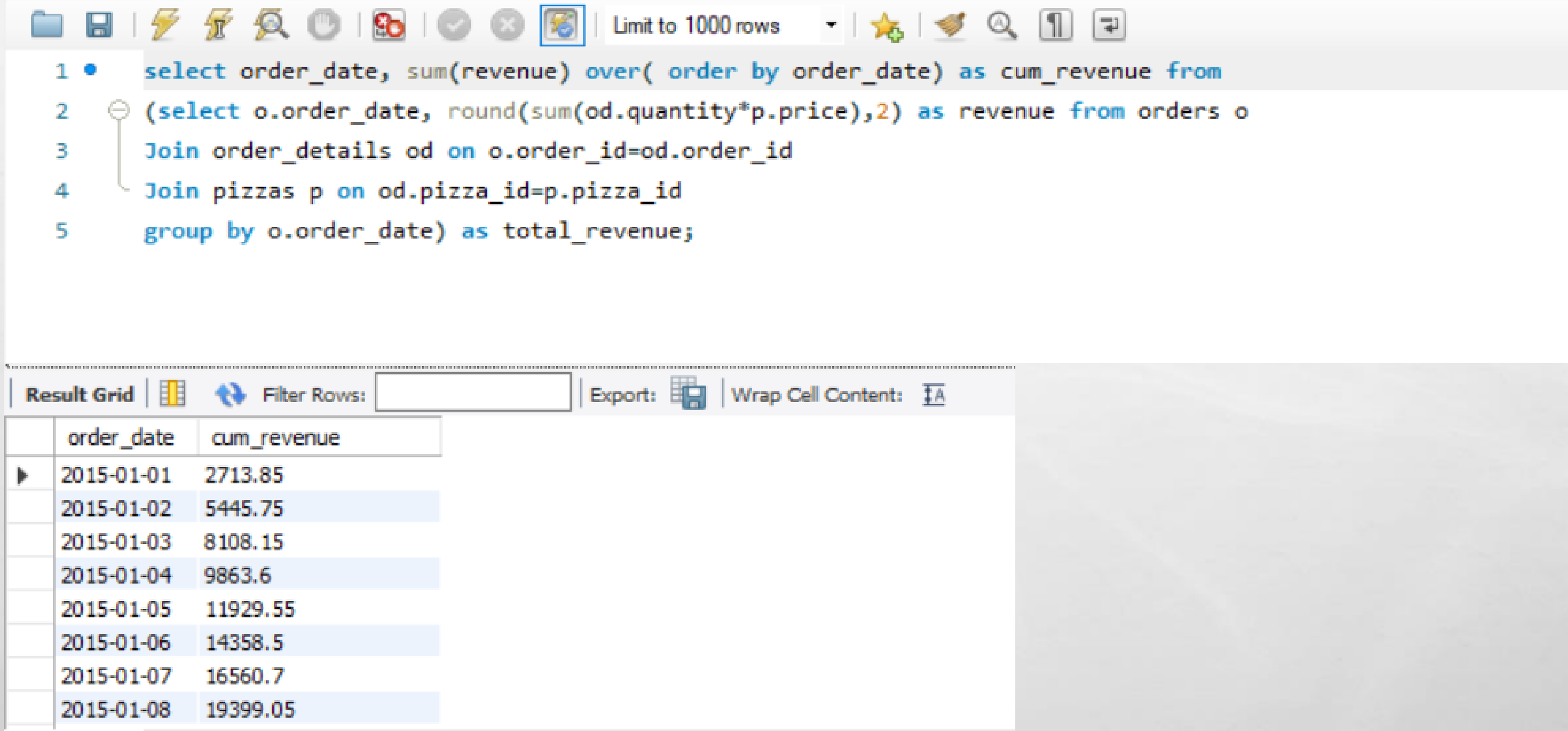
Limit to 1000 rows

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



category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

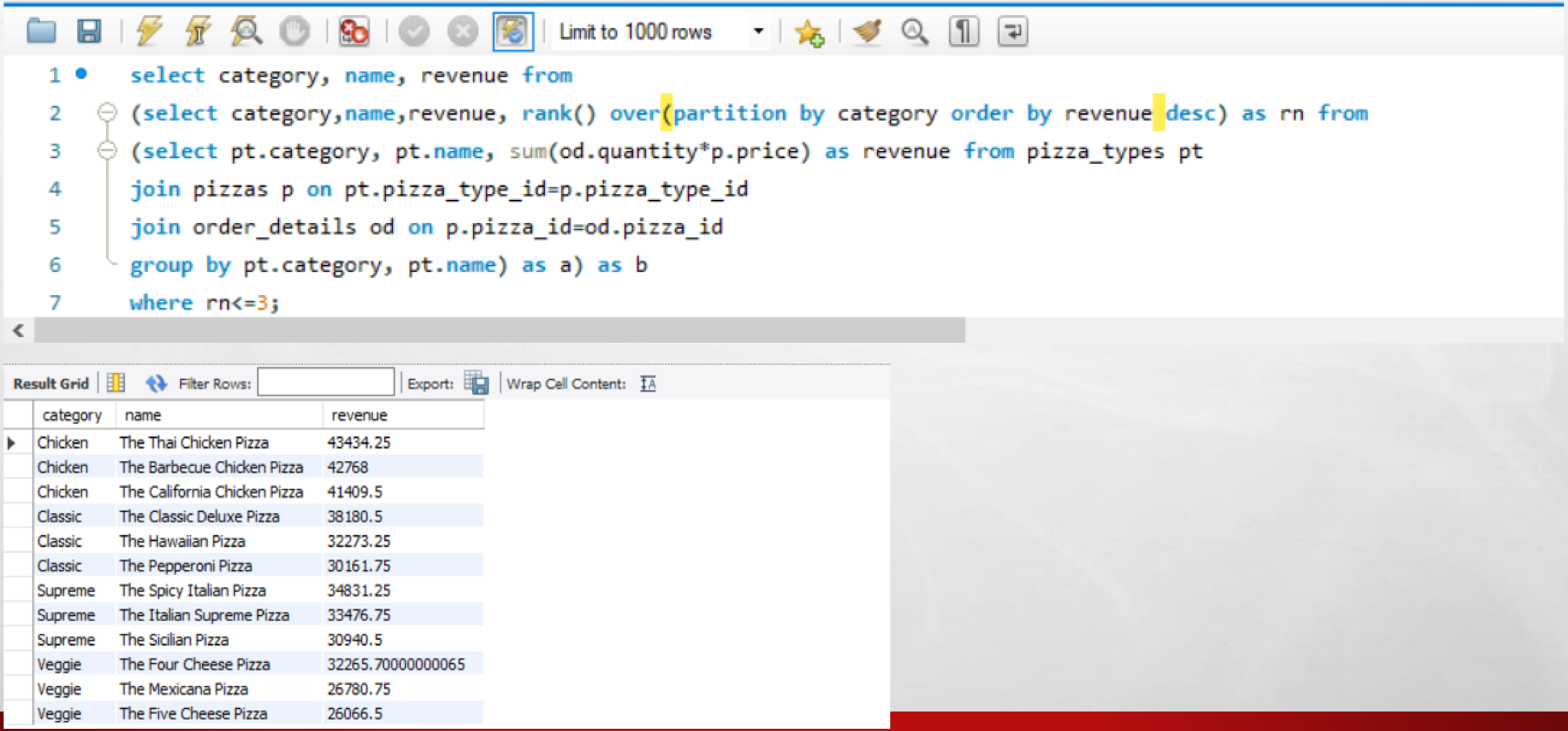
ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.



```sql
select order_date, sum(revenue) over( order by order_date) as cum_revenue from
(select o.order_date, round(sum(od.quantity*p.price),2) as revenue from orders o
Join order_details od on o.order_id=od.order_id
Join pizzas p on od.pizza_id=p.pizza_id
group by o.order_date) as total_revenue;
```

order_date	cum_revenue
2015-01-01	2713.85
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

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



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

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