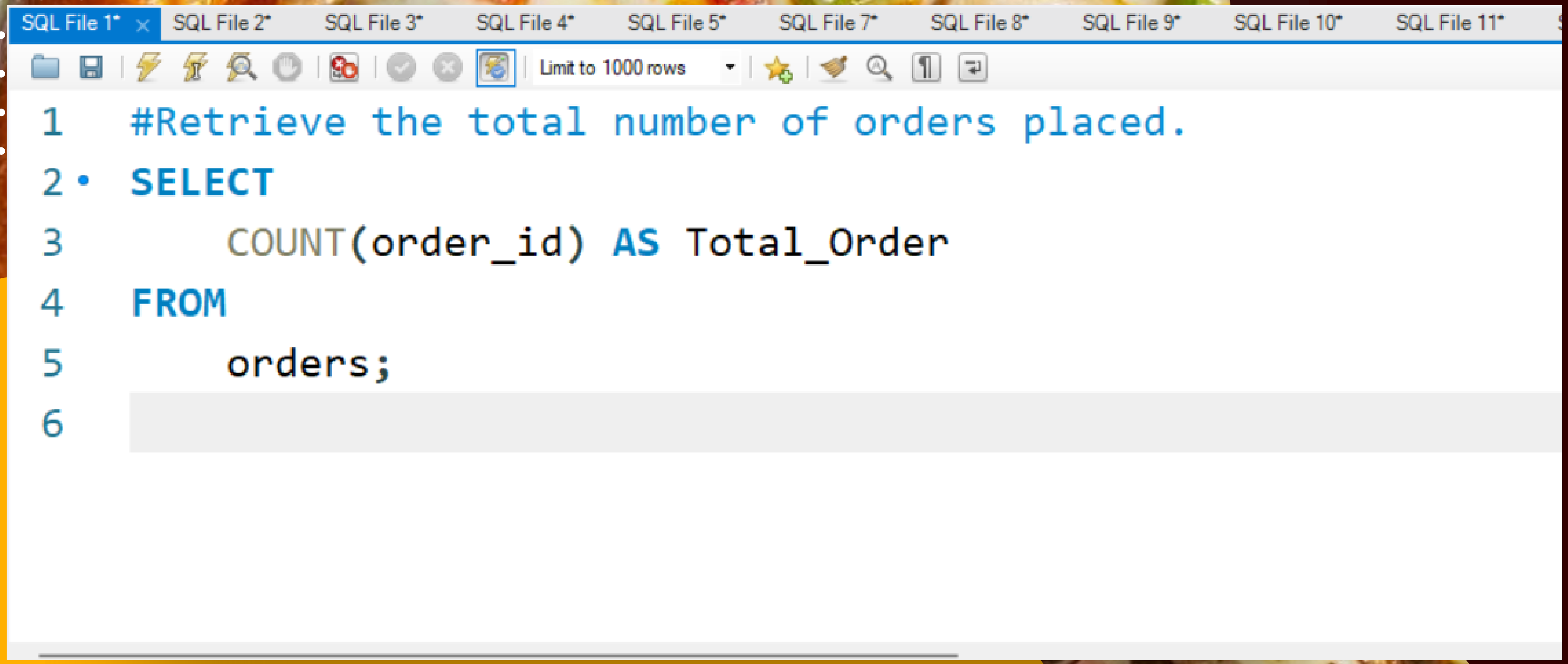
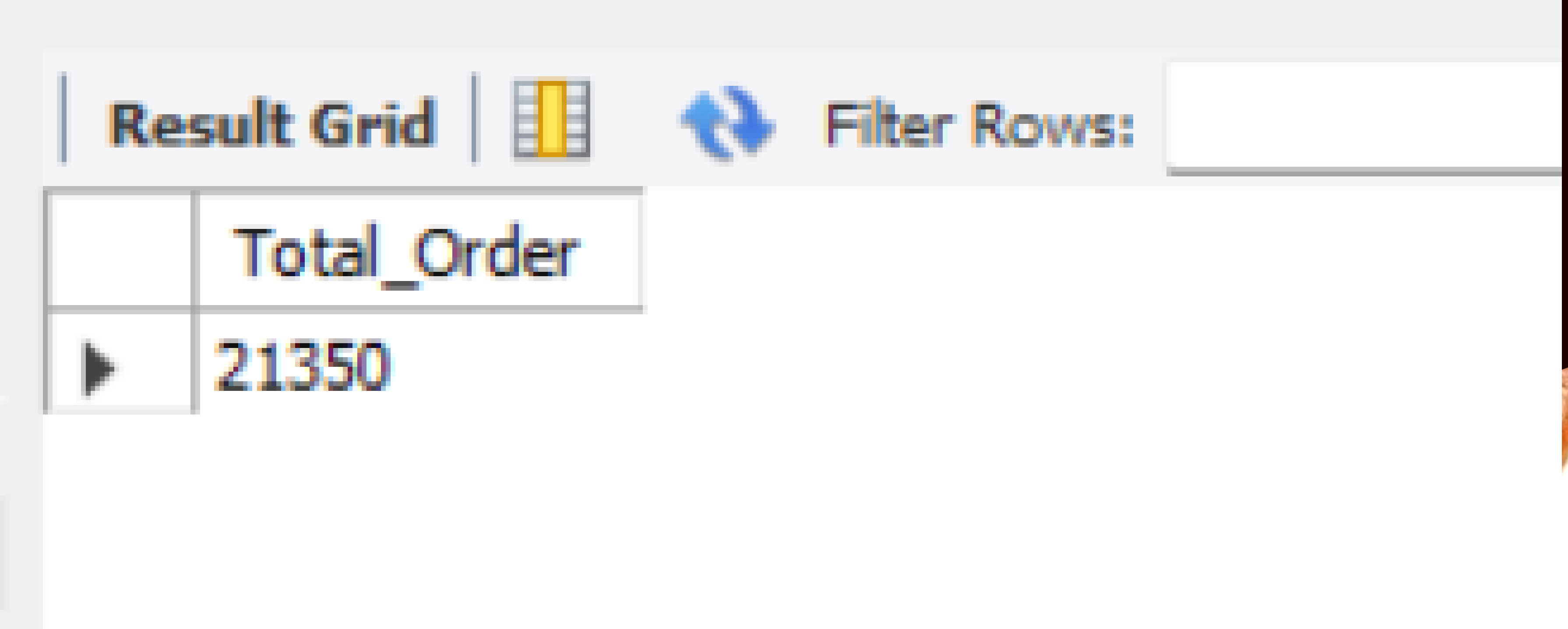
BY: RAJVEER RAJ

PIZZA SALES SQL

#Retrieve the total number of orders placed.

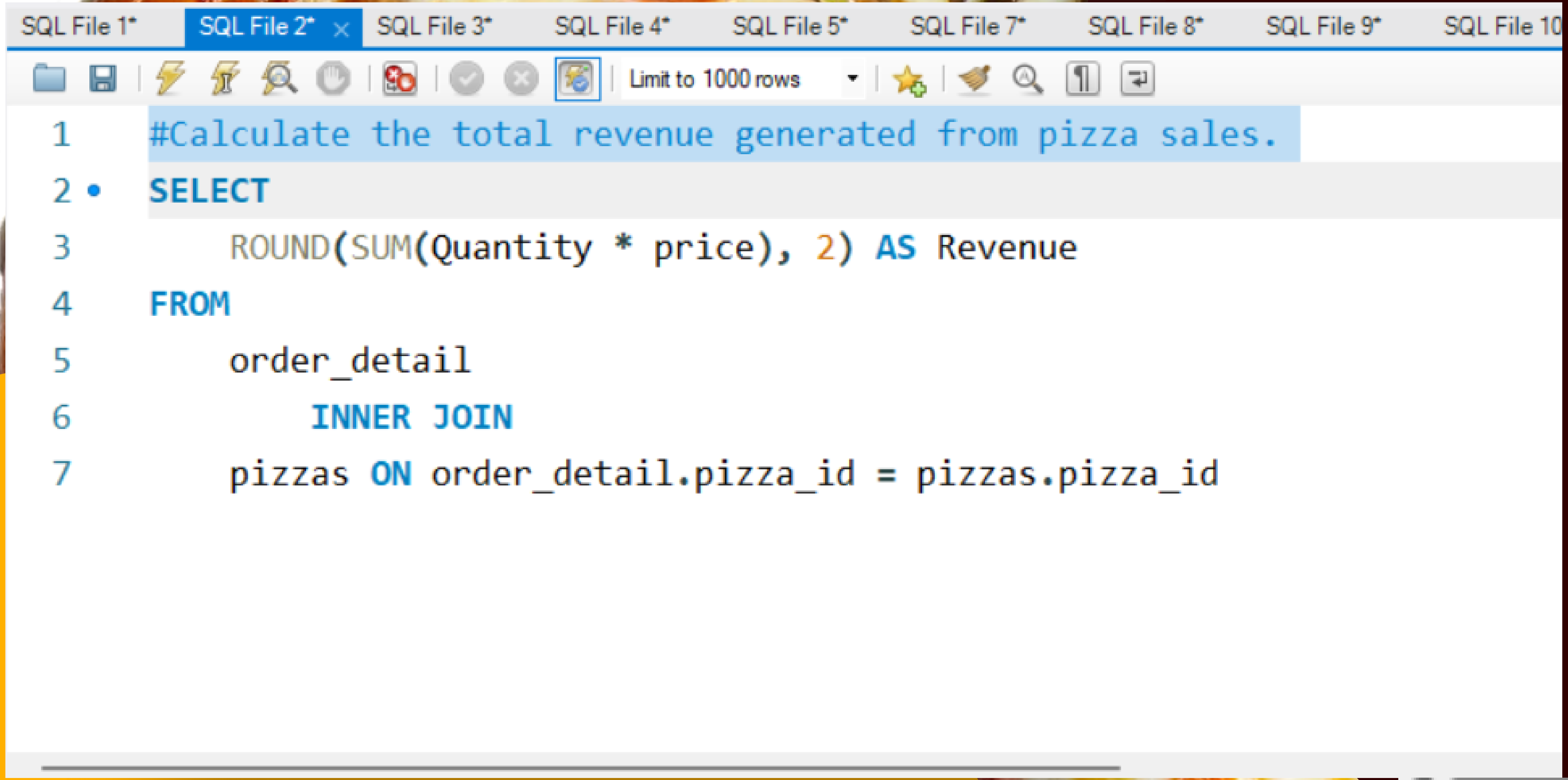


```sql
#Retrieve the total number of orders placed.
SELECT
    COUNT(order_id) AS Total_Order
FROM
    orders;
```

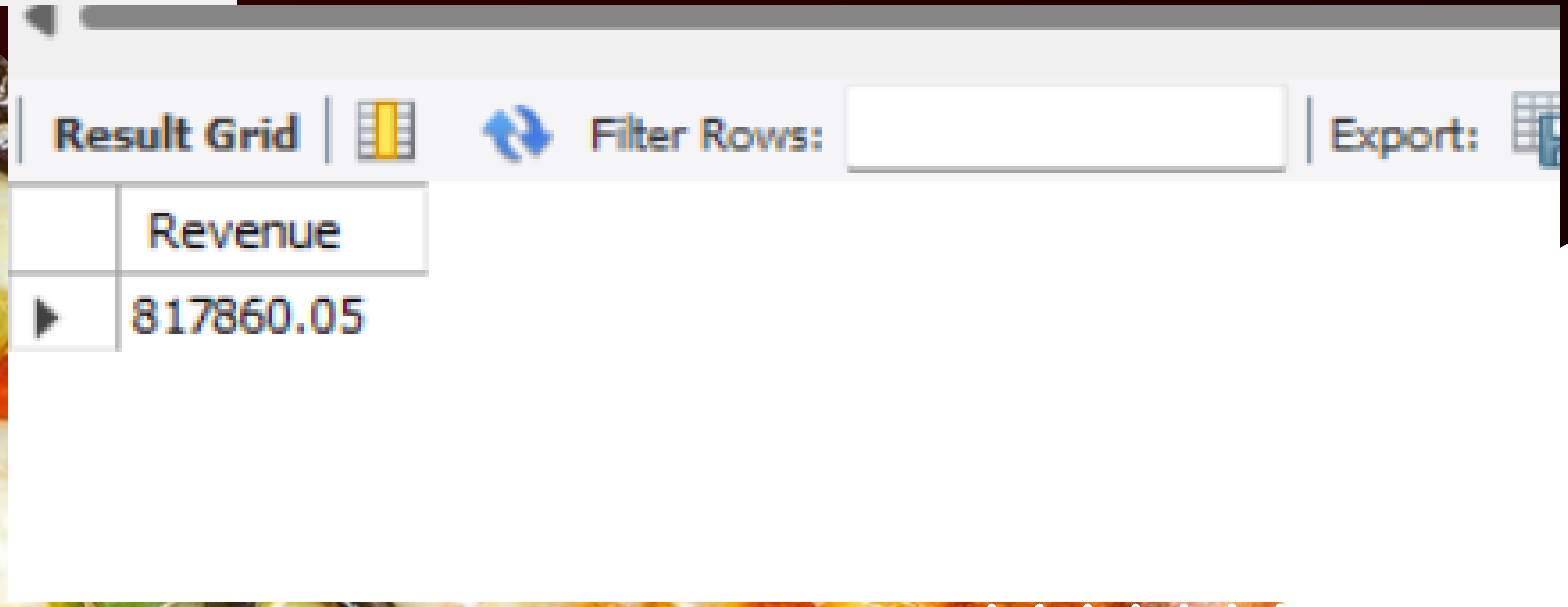


```
Total_Order
21350
```

#Calculate the total revenue generated from pizza sales.

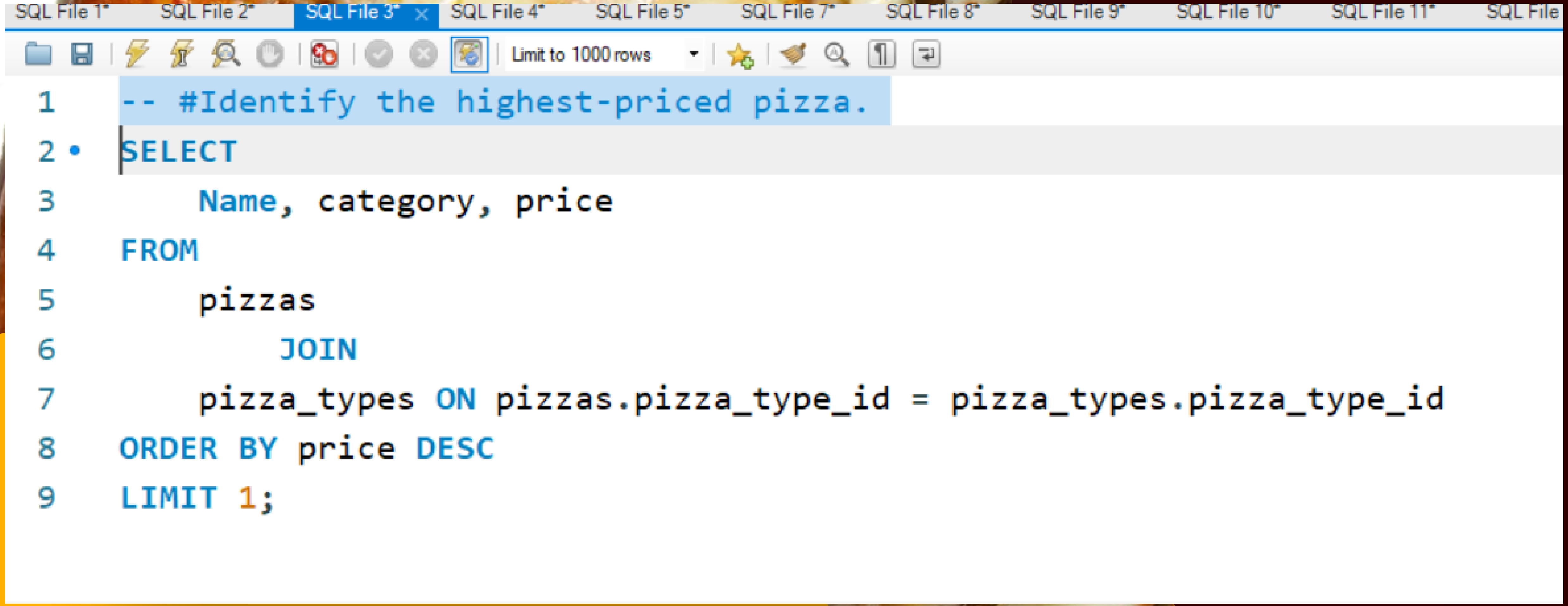


```sql
#Calculate the total revenue generated from pizza sales.
SELECT
    ROUND(SUM(Quantity * price), 2) AS Revenue
FROM
    order_detail
        INNER JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id
```

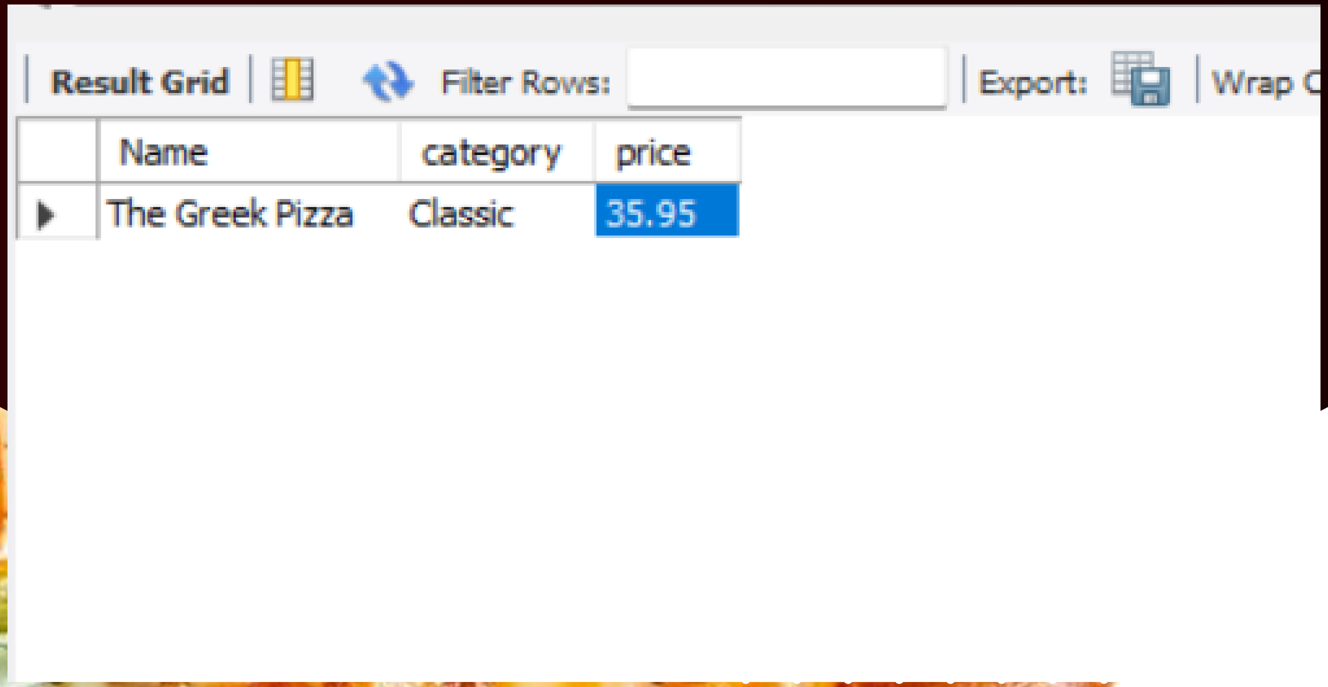


Revenue
817860.05

#Identify the highest-priced pizza.

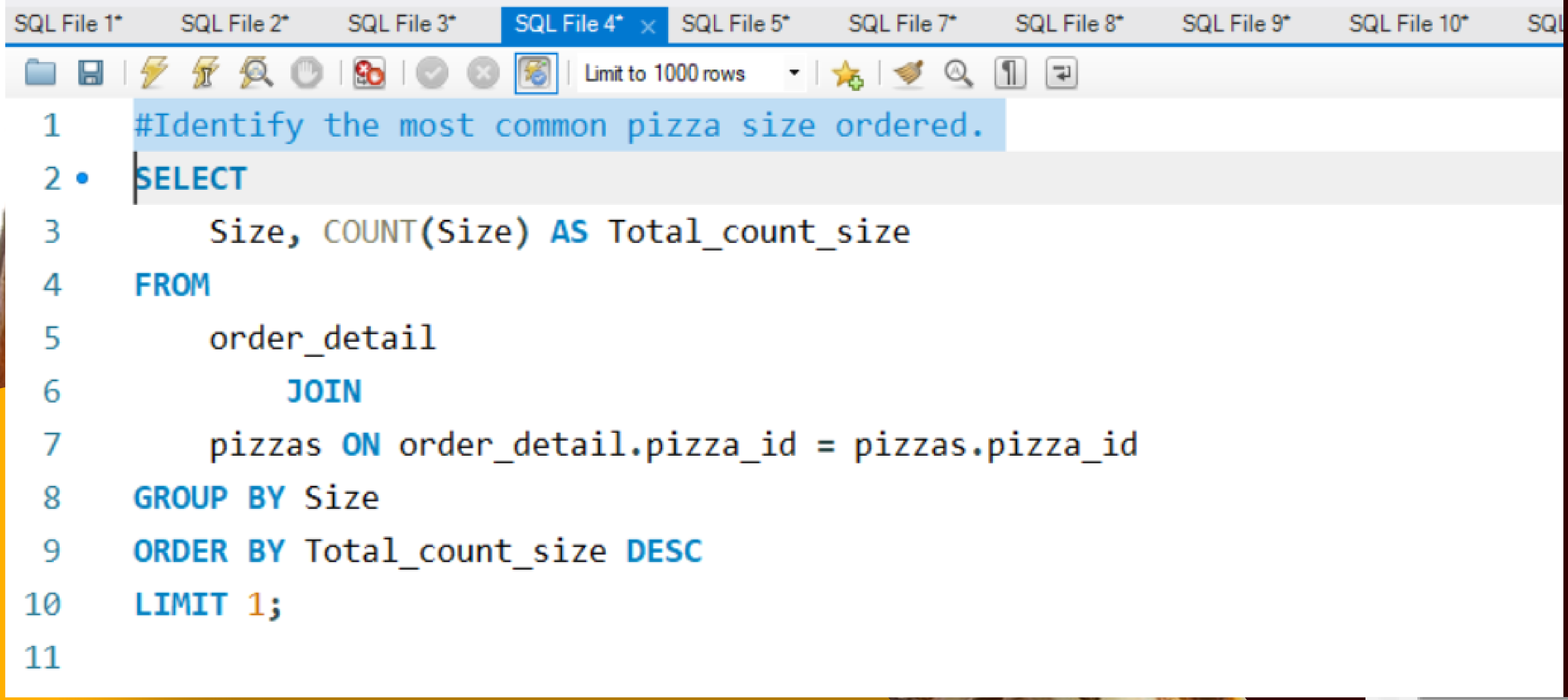


```sql
-- #Identify the highest-priced pizza.
SELECT
    Name, category, price
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

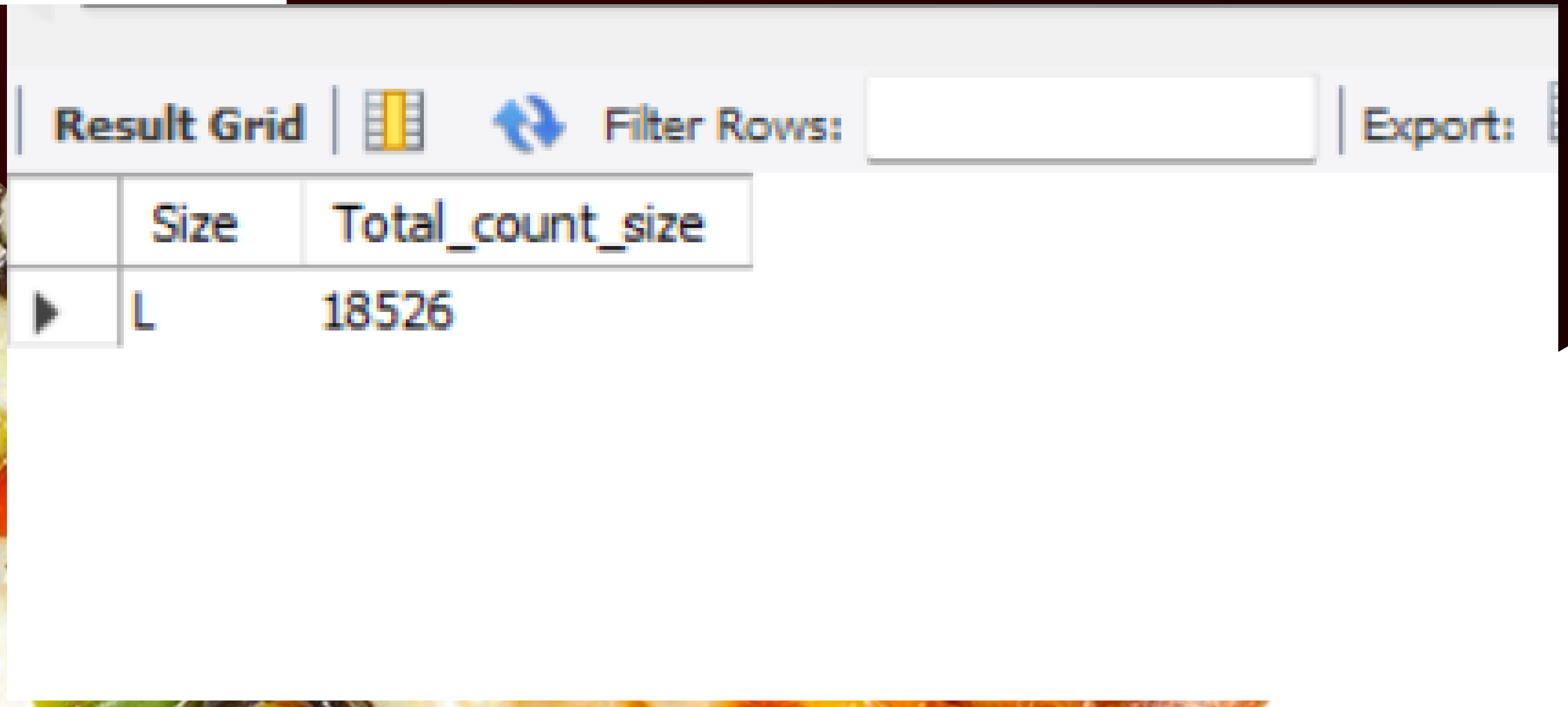


Name	category	price
The Greek Pizza	Classic	35.95

#Identify the most common pizza size ordered.

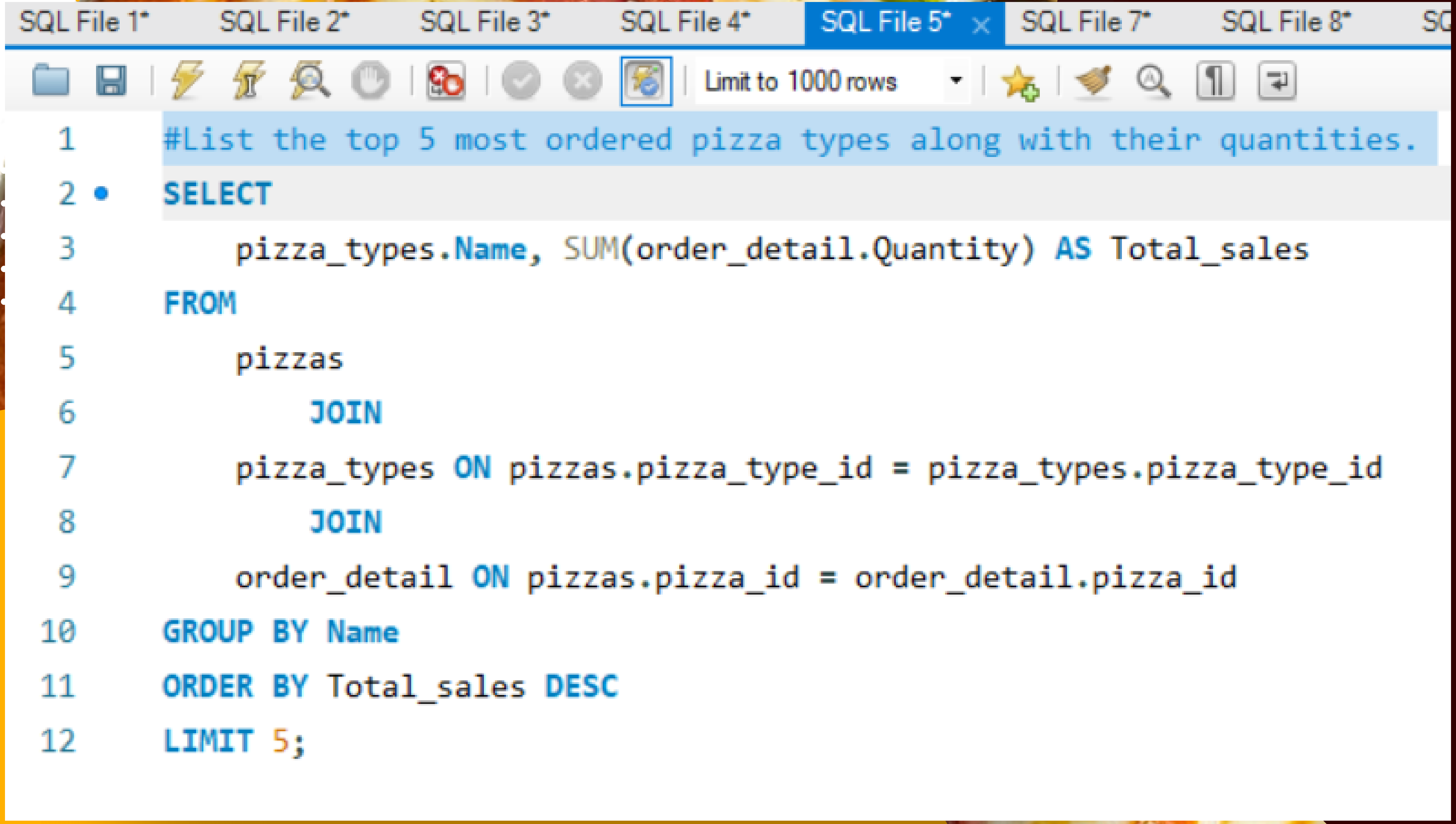


```sql
#Identify the most common pizza size ordered.
SELECT
    Size, COUNT(Size) AS Total_count_size
FROM
    order_detail
        JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY Size
ORDER BY Total_count_size DESC
LIMIT 1;
```

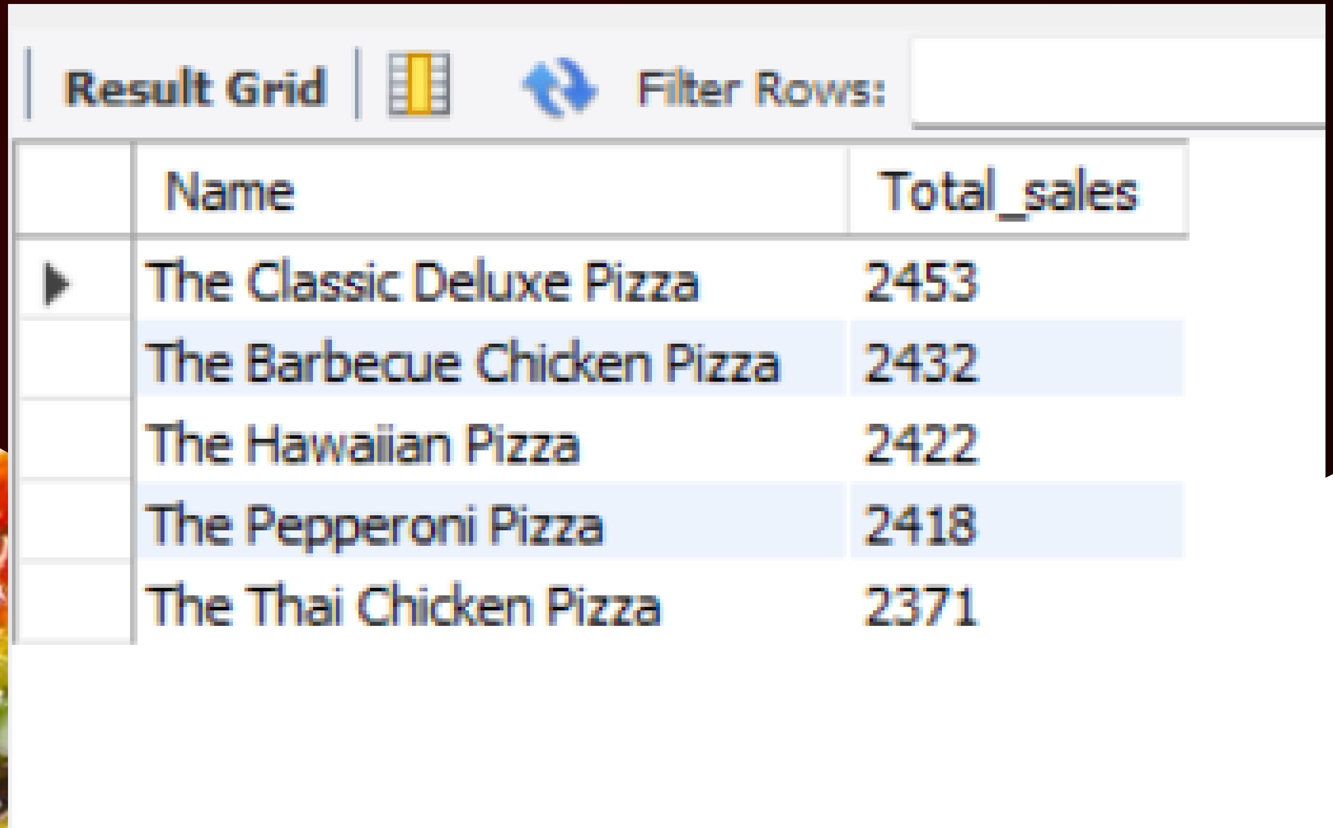


Size	Total_count_size
L	18526

#List the top 5 most ordered pizza types along with their quantities.

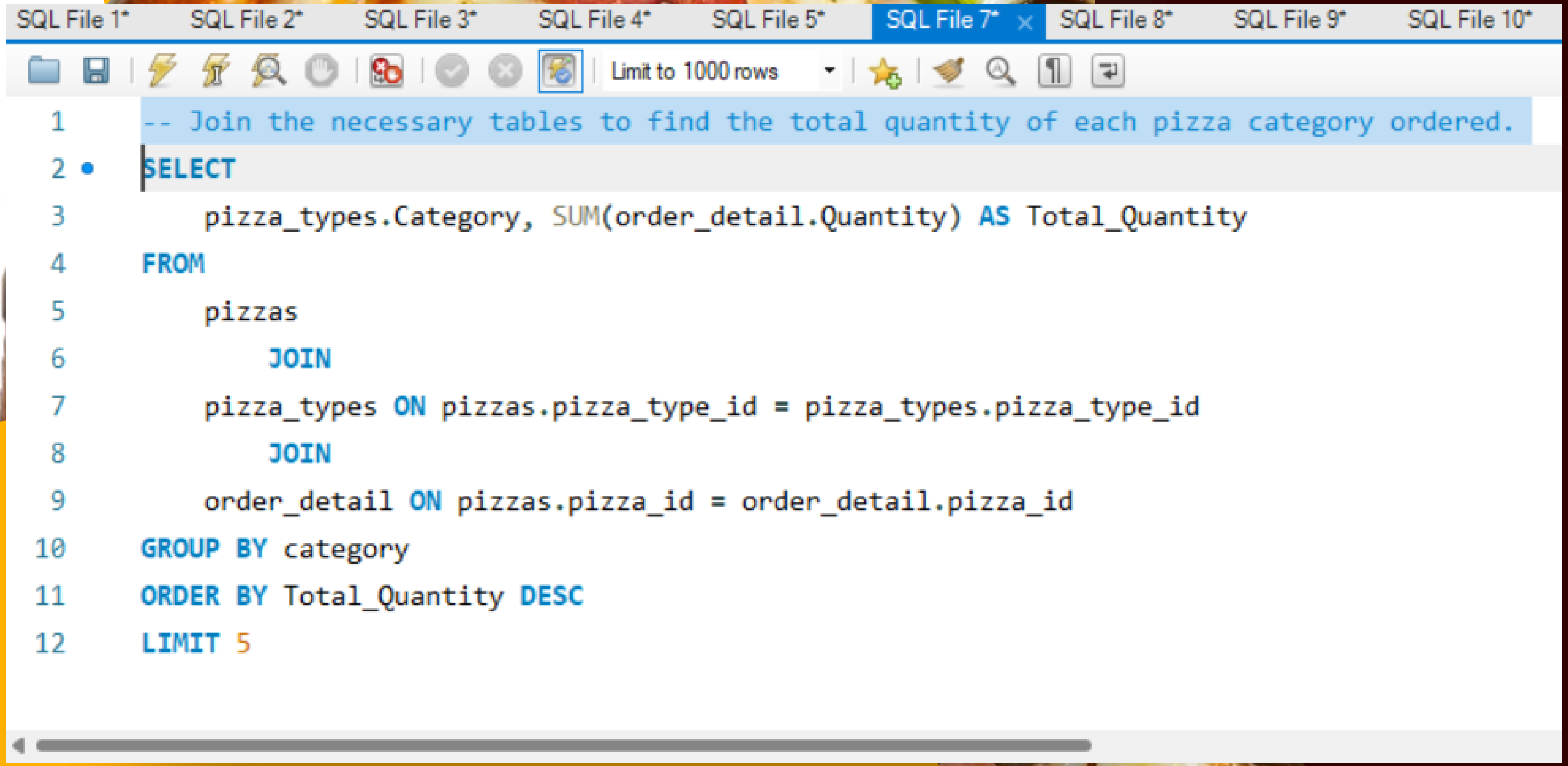


```sql
#List the top 5 most ordered pizza types along with their quantities.
SELECT
    pizza_types.Name, SUM(order_detail.Quantity) AS Total_sales
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_detail ON pizzas.pizza_id = order_detail.pizza_id
GROUP BY Name
ORDER BY Total_sales DESC
LIMIT 5;
```

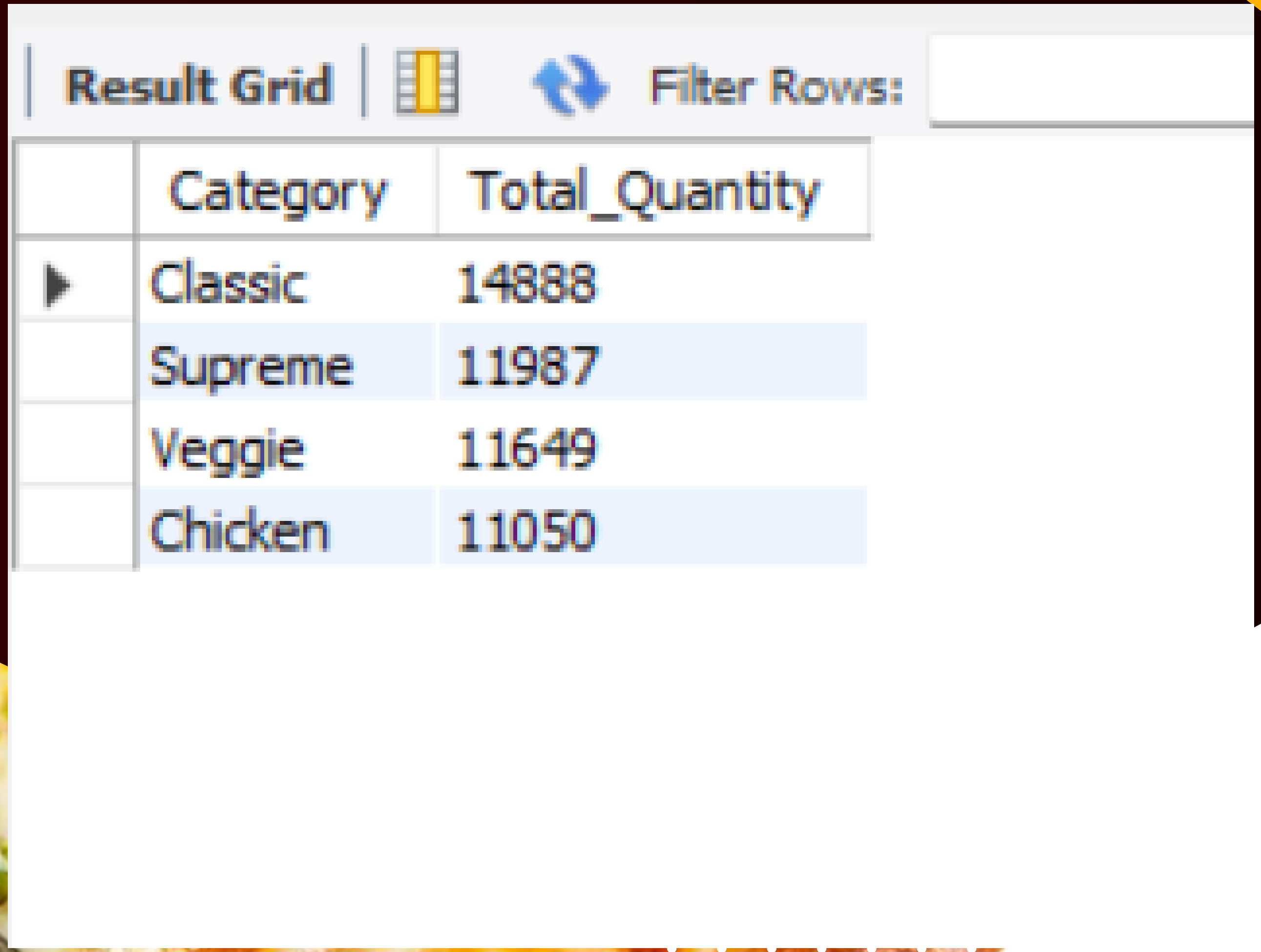


Name	Total_sales
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

#Join the necessary tables to find the total quantity of each pizza category ordered.

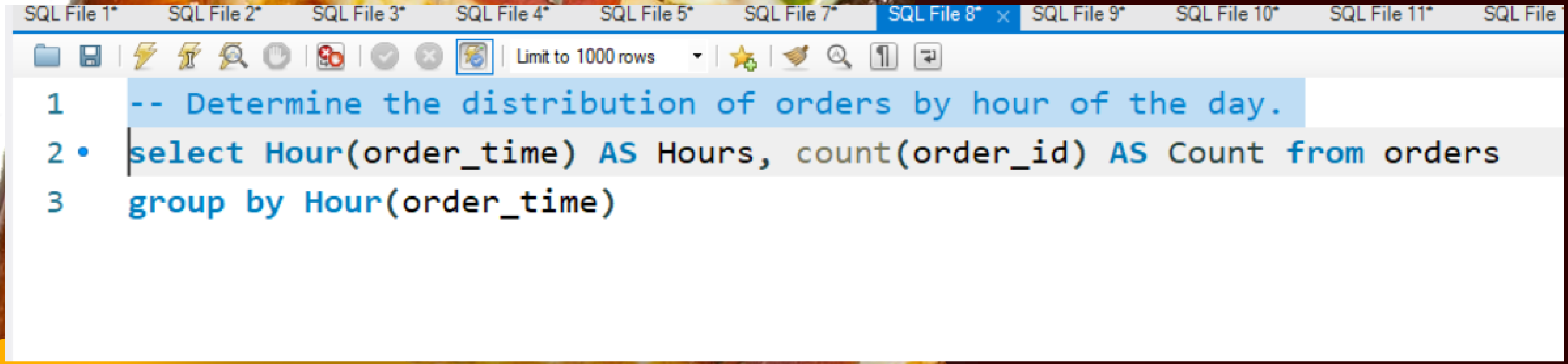


```sql
-- Join the necessary tables to find the total quantity of each pizza category ordered.
SELECT
    pizza_types.Category, SUM(order_detail.Quantity) AS Total_Quantity
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_detail ON pizzas.pizza_id = order_detail.pizza_id
GROUP BY category
ORDER BY Total_Quantity DESC
LIMIT 5
```

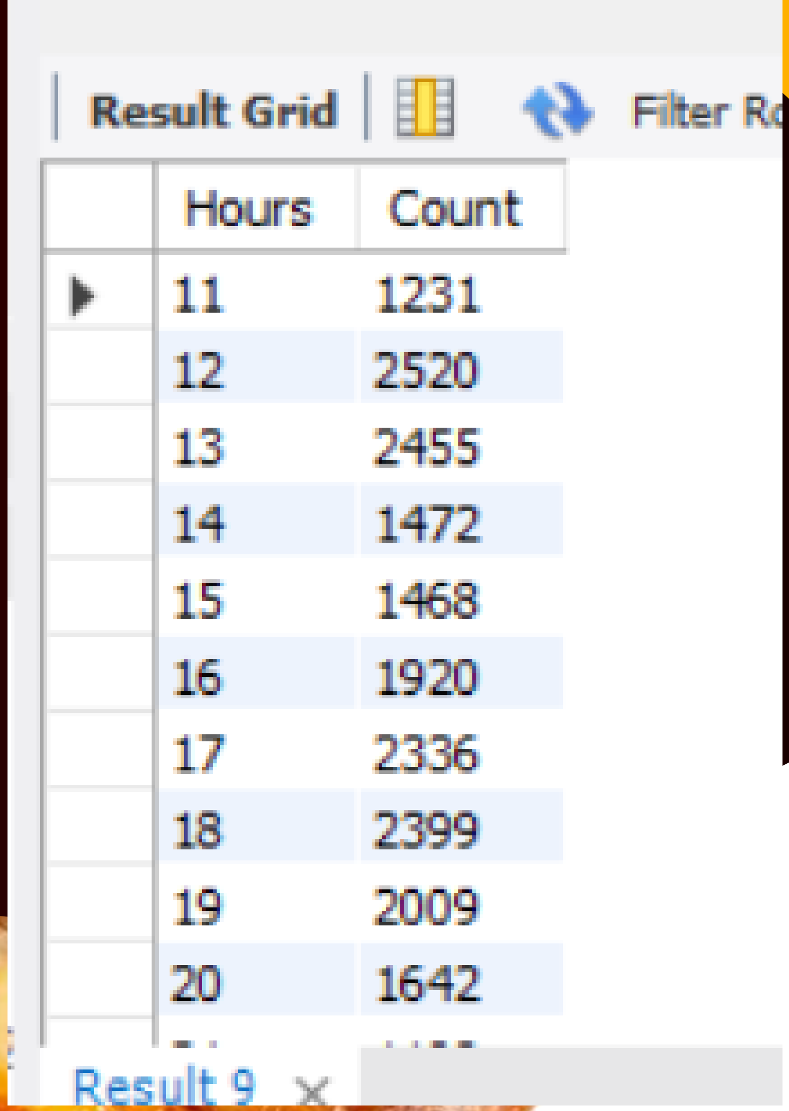


Category	Total_Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

#Determine the distribution of orders by hour of the day.

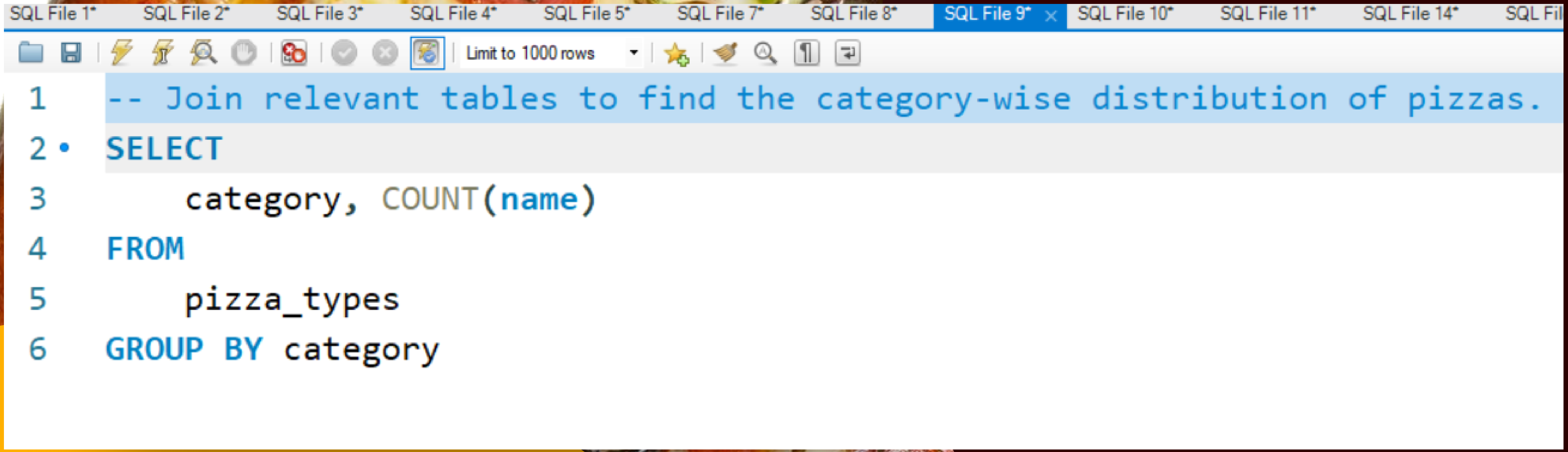


```sql
-- Determine the distribution of orders by hour of the day.
select Hour(order_time) AS Hours, count(order_id) AS Count from orders
group by Hour(order_time)
```

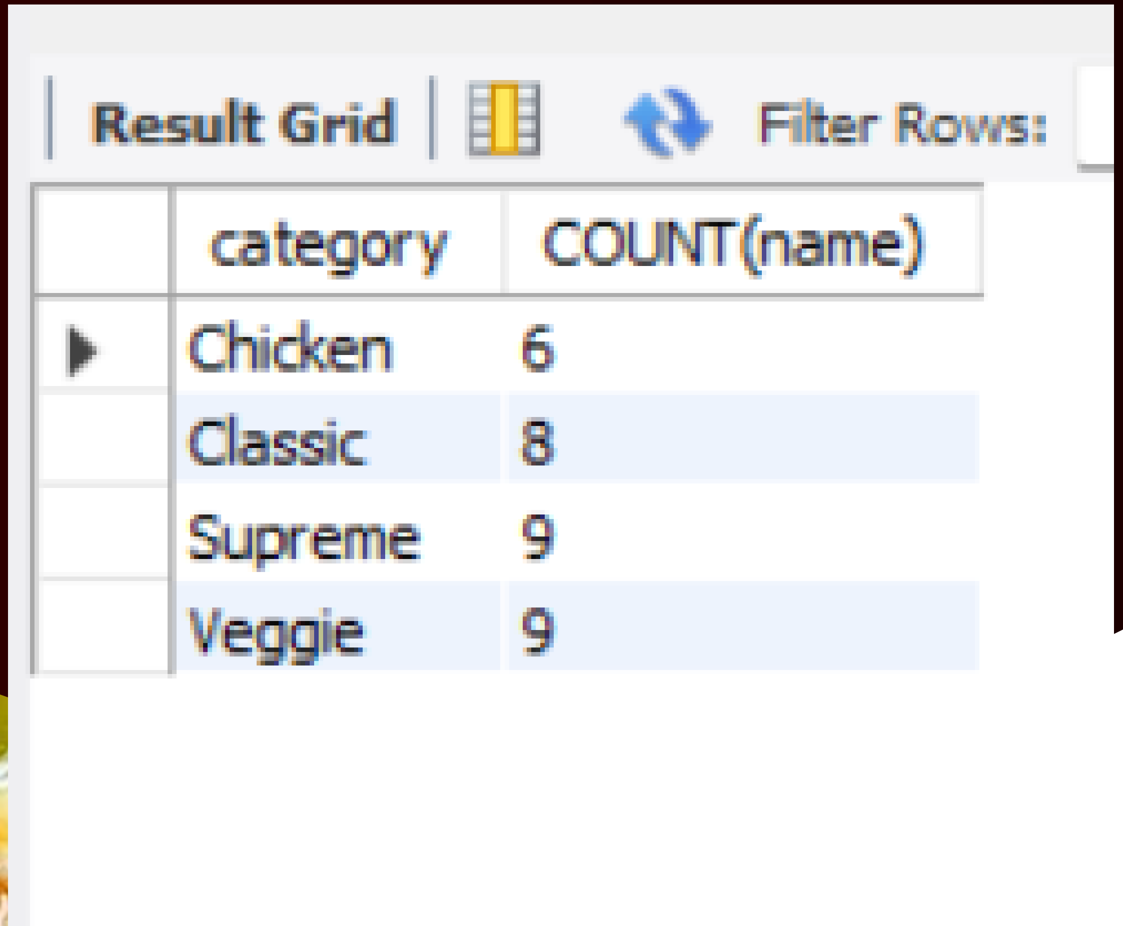


Hours	Count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642

#Join relevant tables to find the category-wise distribution of pizzas.

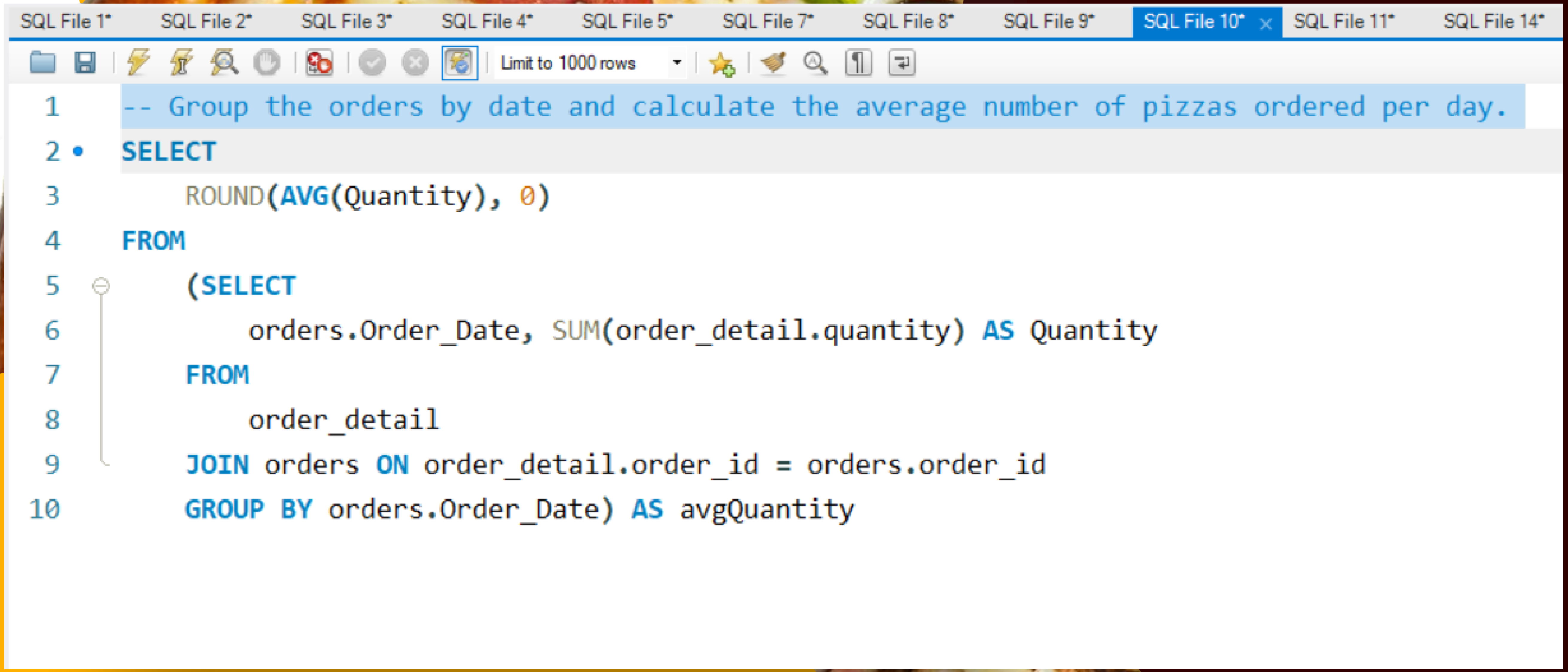


```sql
-- Join relevant tables to find the category-wise distribution of pizzas.
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category
```

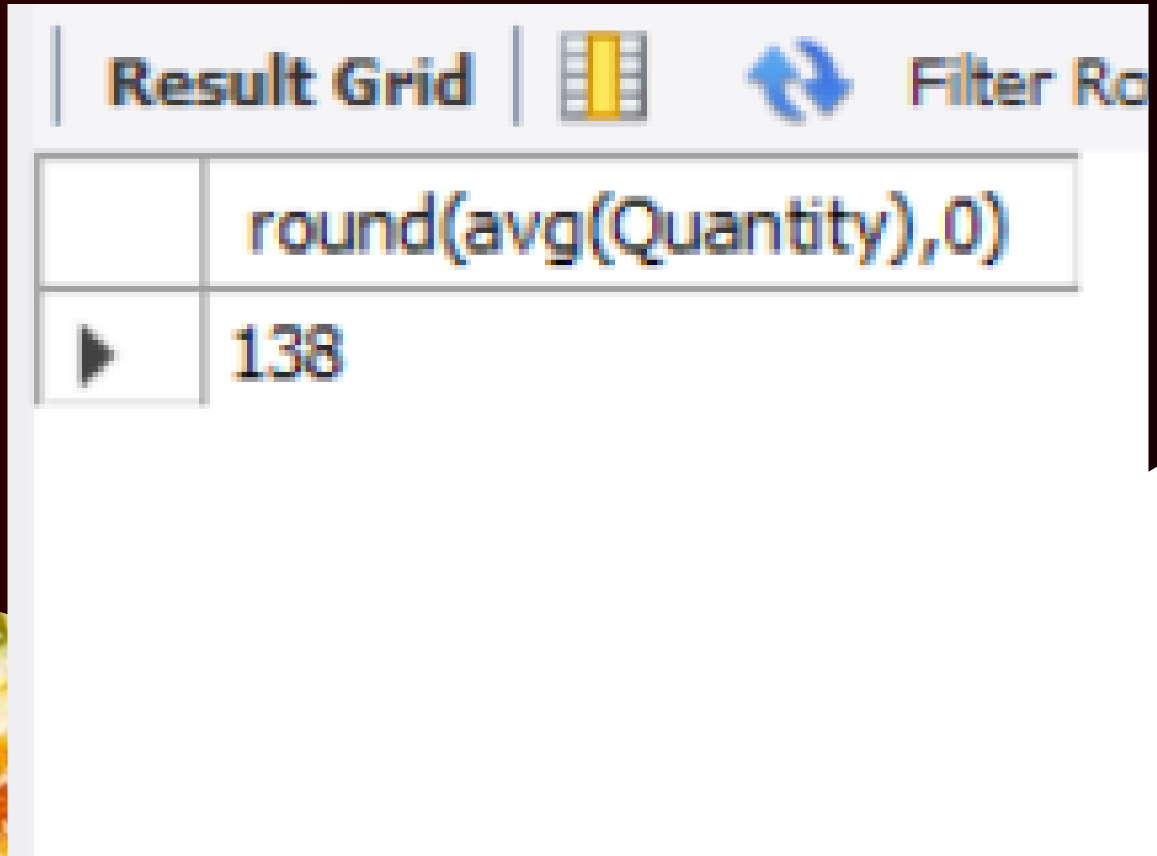


category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

#Group the orders by date and calculate the average number of pizzas ordered per day.

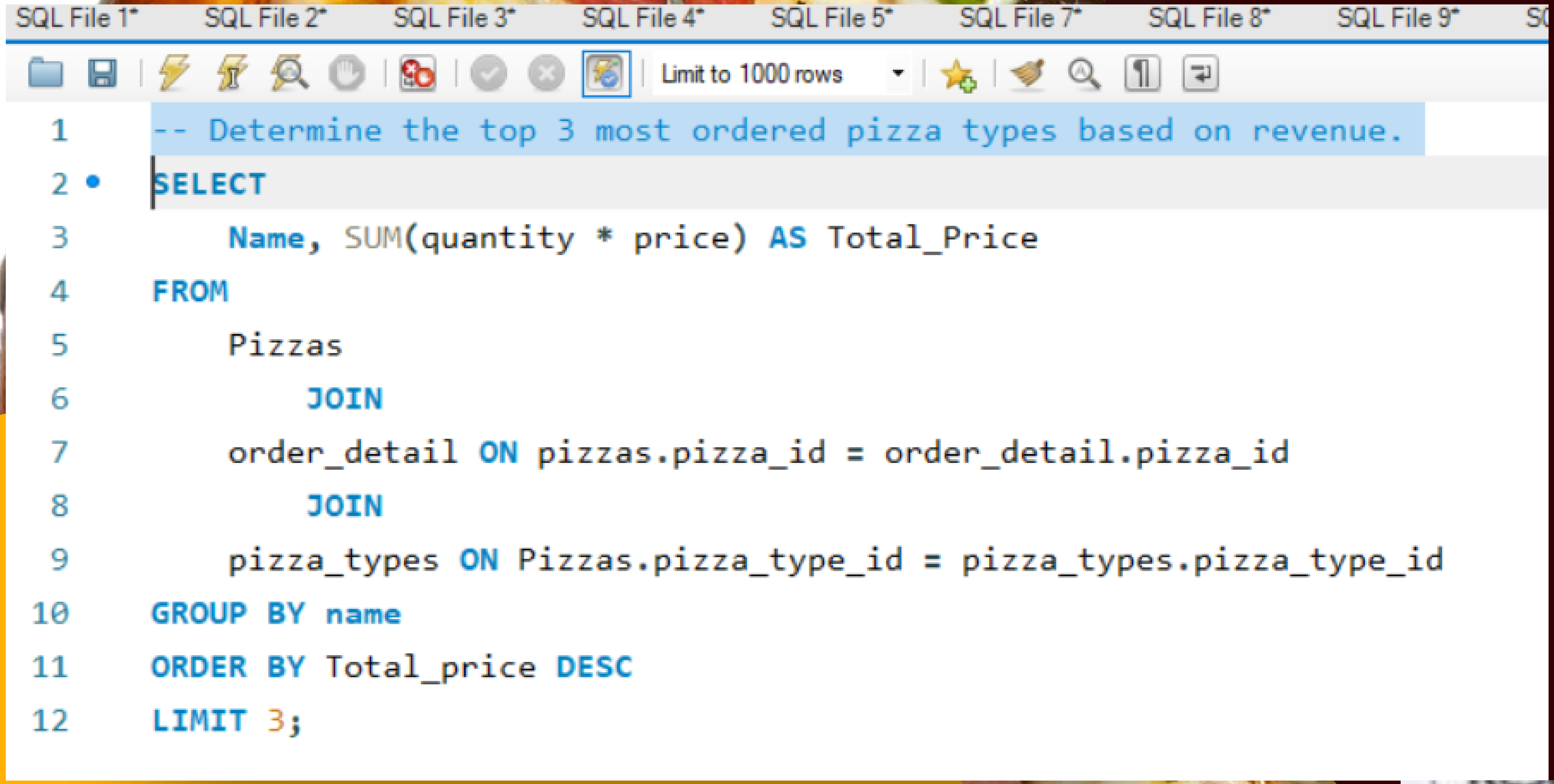


```sql
-- Group the orders by date and calculate the average number of pizzas ordered per day.
SELECT
    ROUND(AVG(Quantity), 0)
FROM
    (SELECT
        orders.Order_Date, SUM(order_detail.quantity) AS Quantity
    FROM
        order_detail
    JOIN orders ON order_detail.order_id = orders.order_id
    GROUP BY orders.Order_Date) AS avgQuantity
```

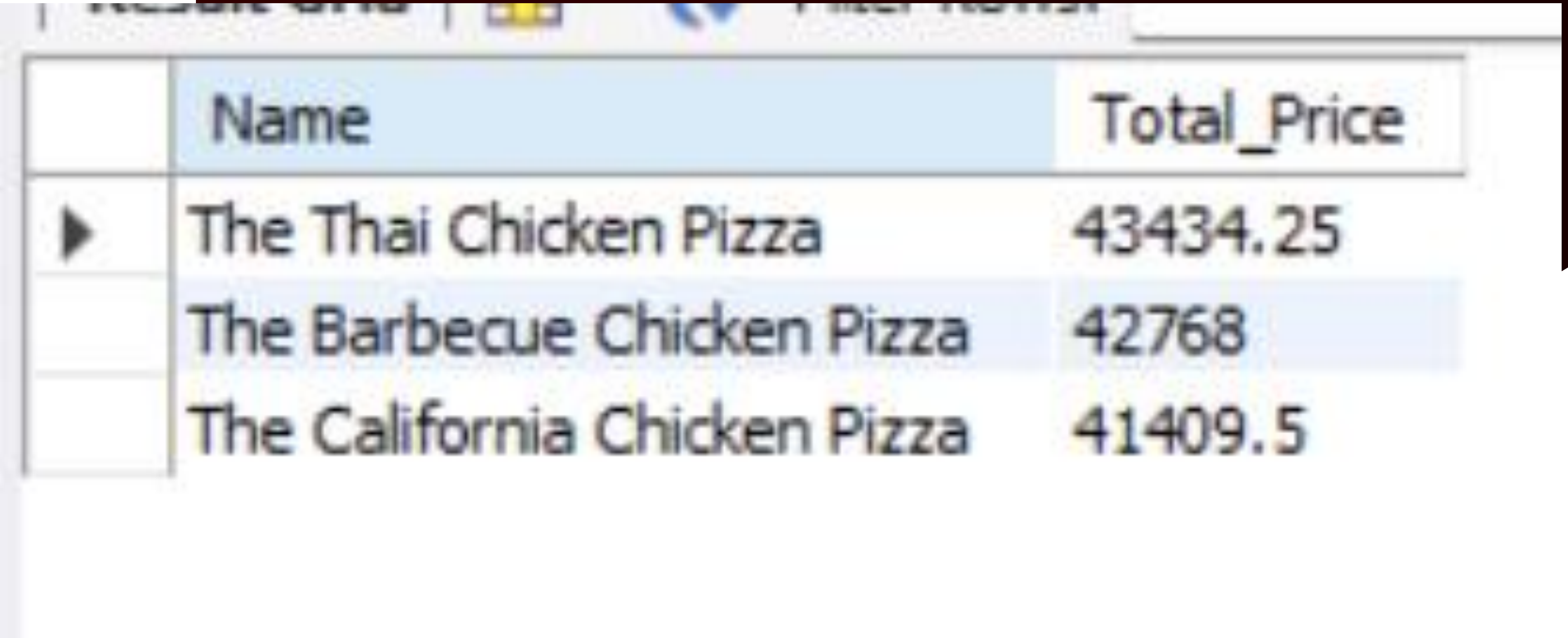


	round(avg(Quantity),0)
▶	138

#Determine the top 3 most ordered pizza types based on revenue.

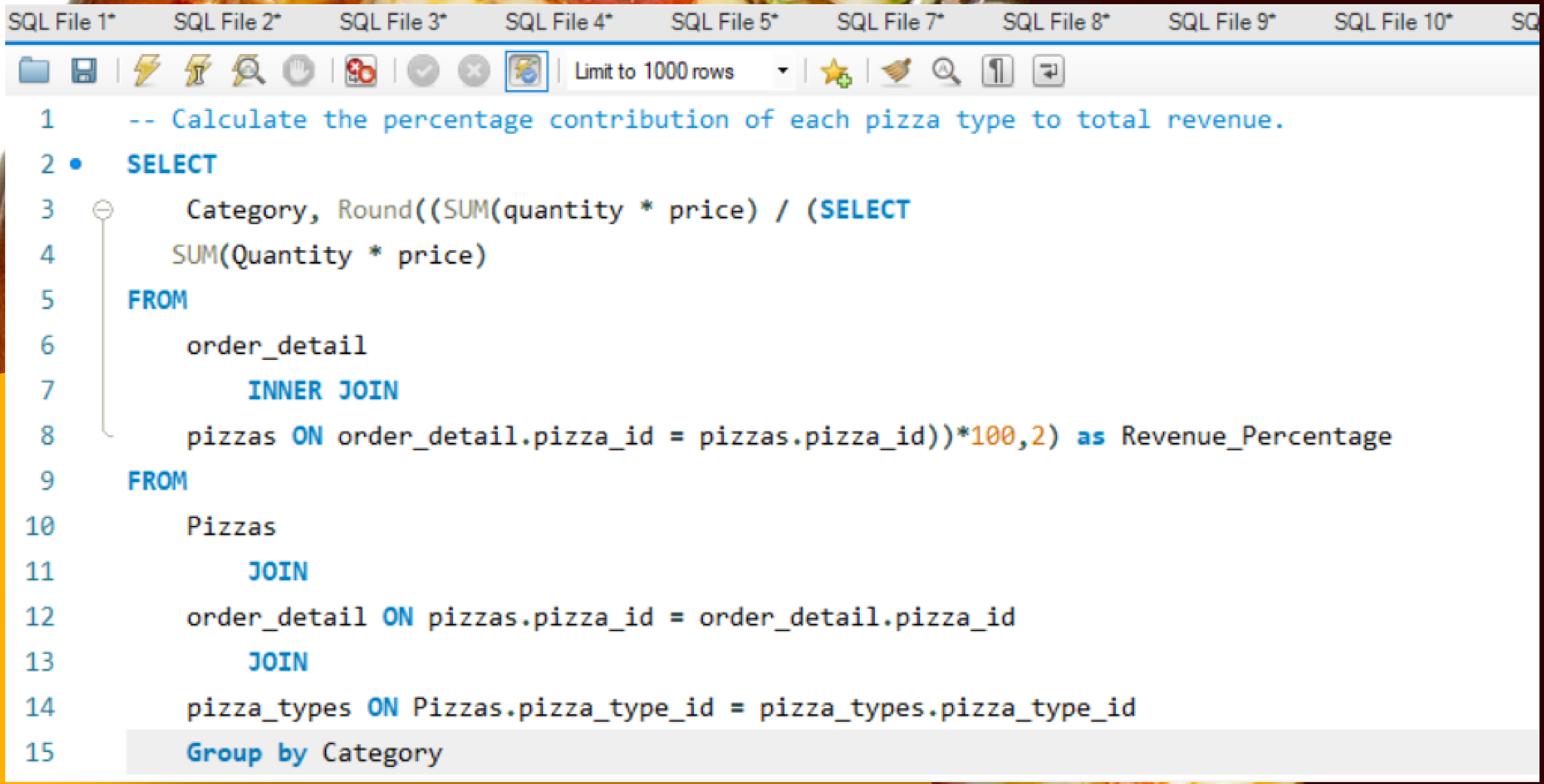


```
-- Determine the top 3 most ordered pizza types based on revenue.
SELECT
    Name, SUM(quantity * price) AS Total_Price
FROM
    Pizzas
        JOIN
    order_detail ON pizzas.pizza_id = order_detail.pizza_id
        JOIN
    pizza_types ON Pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY name
ORDER BY Total_price DESC
LIMIT 3;
```



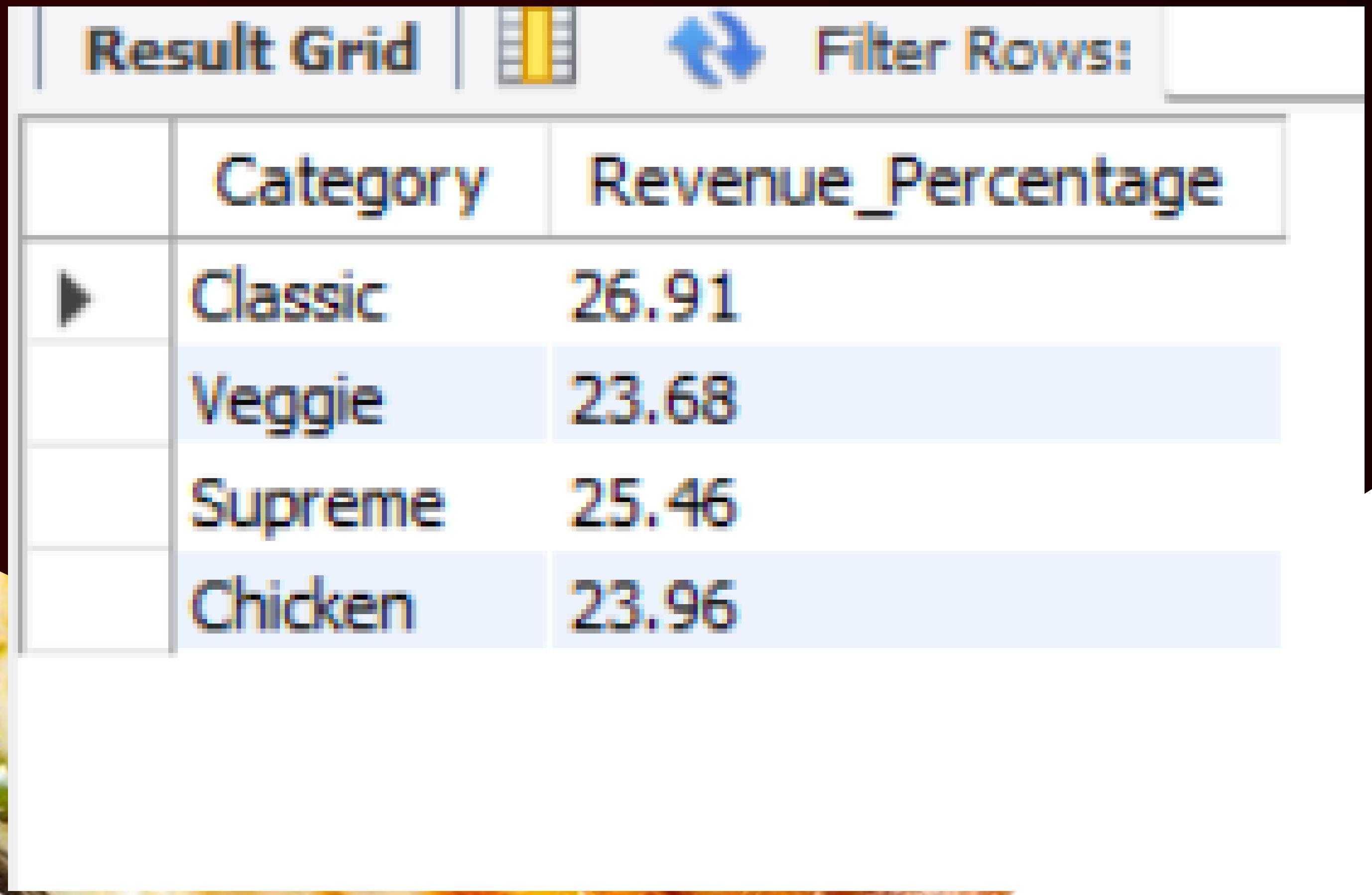
Name	Total_Price
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

#Calculate the percentage contribution of each pizza type to total revenue.



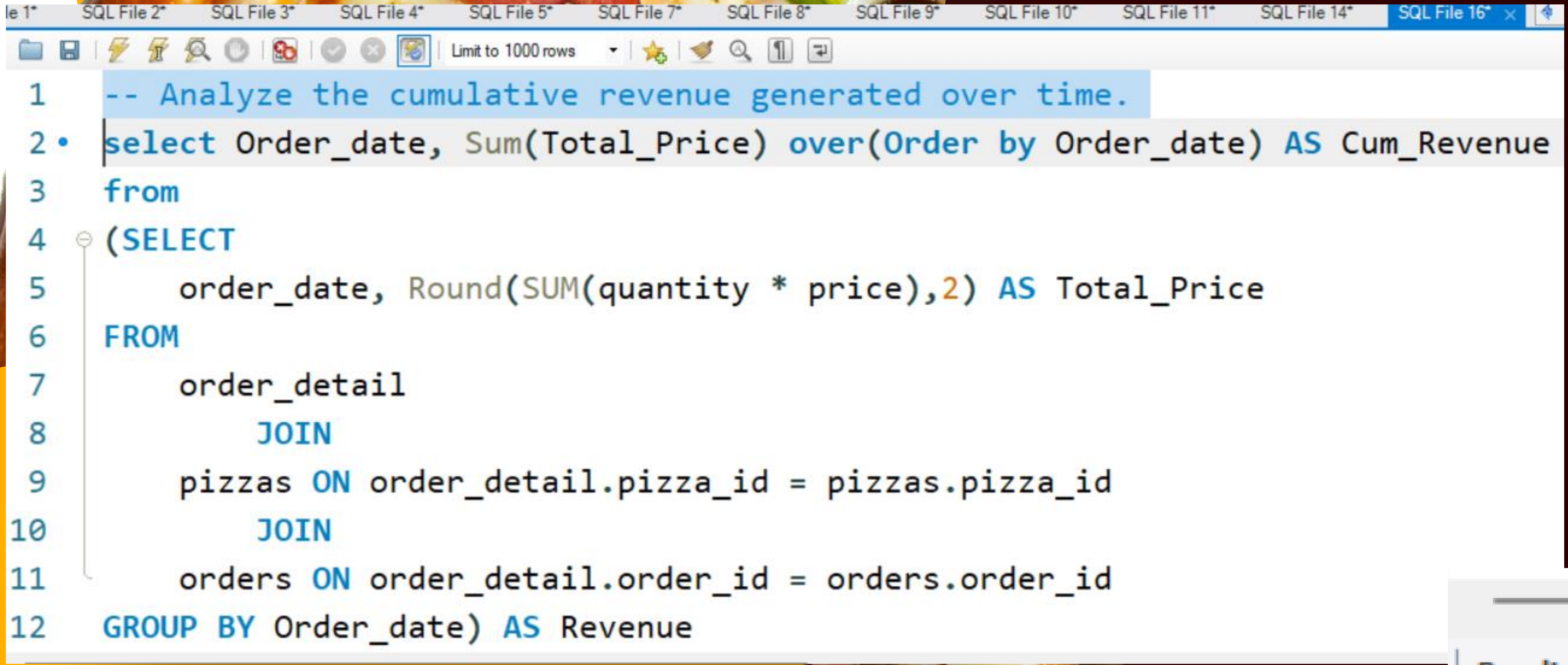
Limit to 1000 rows

```
-- Calculate the percentage contribution of each pizza type to total revenue.
SELECT
    Category, Round((SUM(quantity * price) / (SELECT
   SUM(Quantity * price)
FROM
    order_detail
        INNER JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id))*100,2) as Revenue_Percentage
FROM
    Pizzas
        JOIN
    order_detail ON pizzas.pizza_id = order_detail.pizza_id
        JOIN
    pizza_types ON Pizzas.pizza_type_id = pizza_types.pizza_type_id
    Group by Category
```



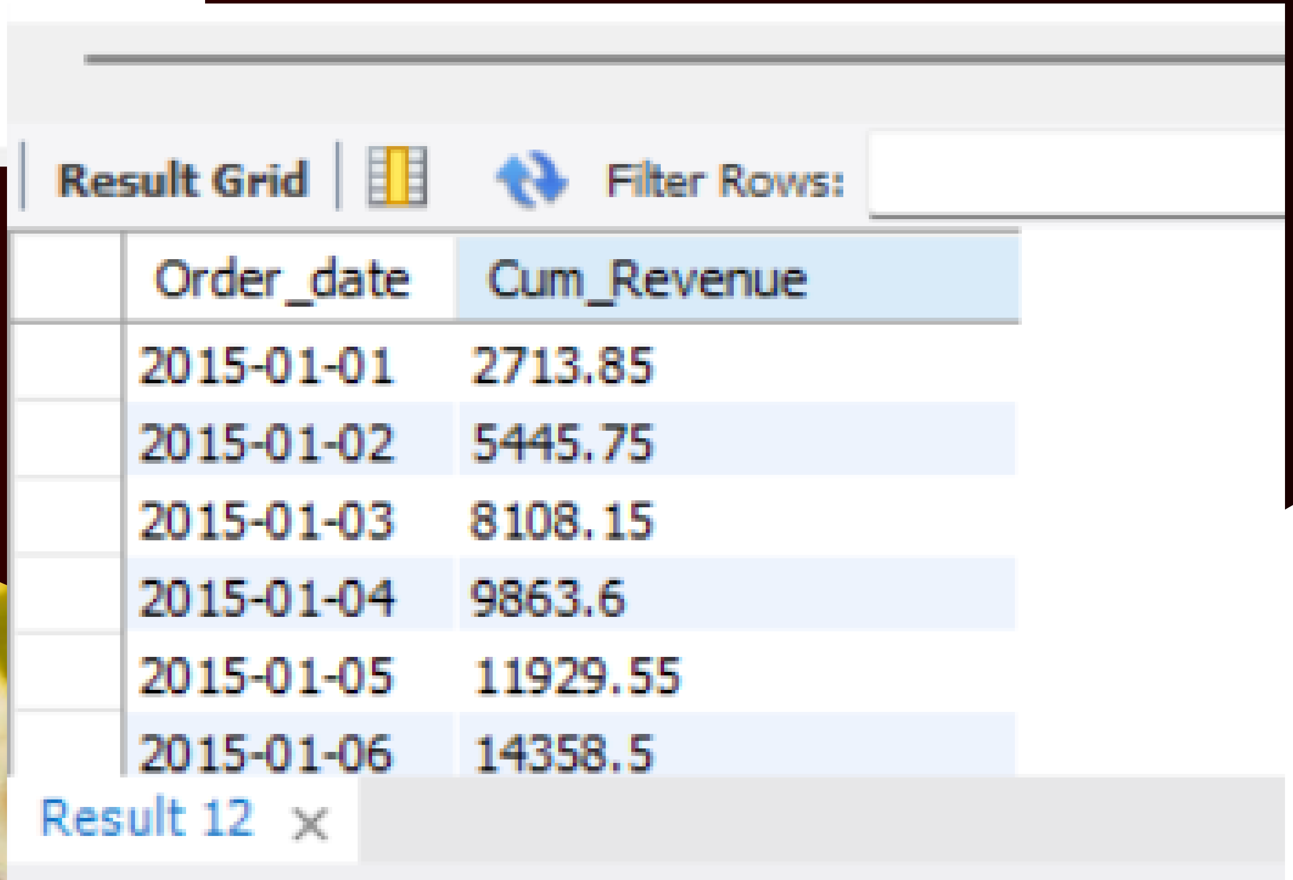
Category	Revenue_Percentage
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96

#Analyze the cumulative revenue generated over time.



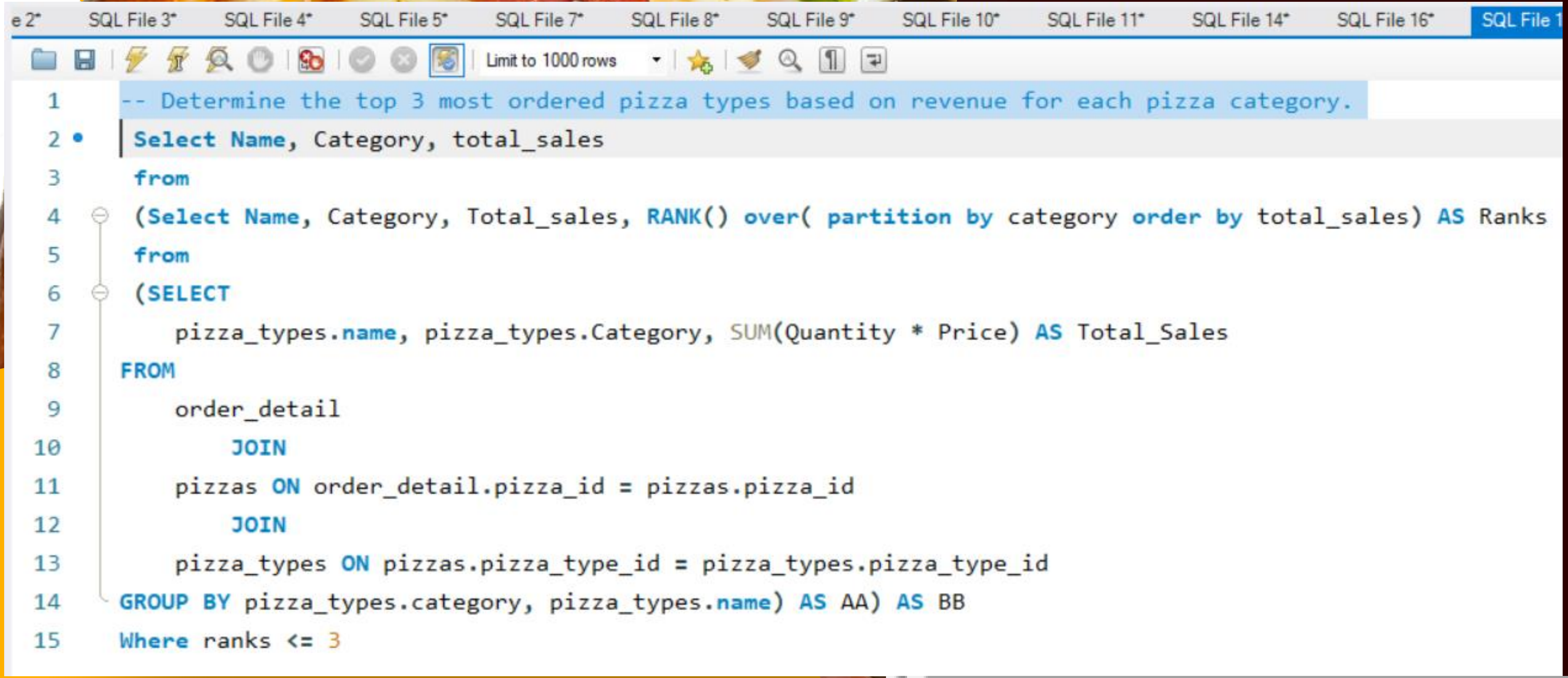
Limit to 1000 rows

```sql
-- Analyze the cumulative revenue generated over time.
select Order_date, Sum(Total_Price) over(Order by Order_date) AS Cum_Revenue
from
(SELECT
    order_date, Round(SUM(quantity * price),2) AS Total_Price
FROM
    order_detail
        JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id
        JOIN
    orders ON order_detail.order_id = orders.order_id
GROUP BY Order_date) AS Revenue
```

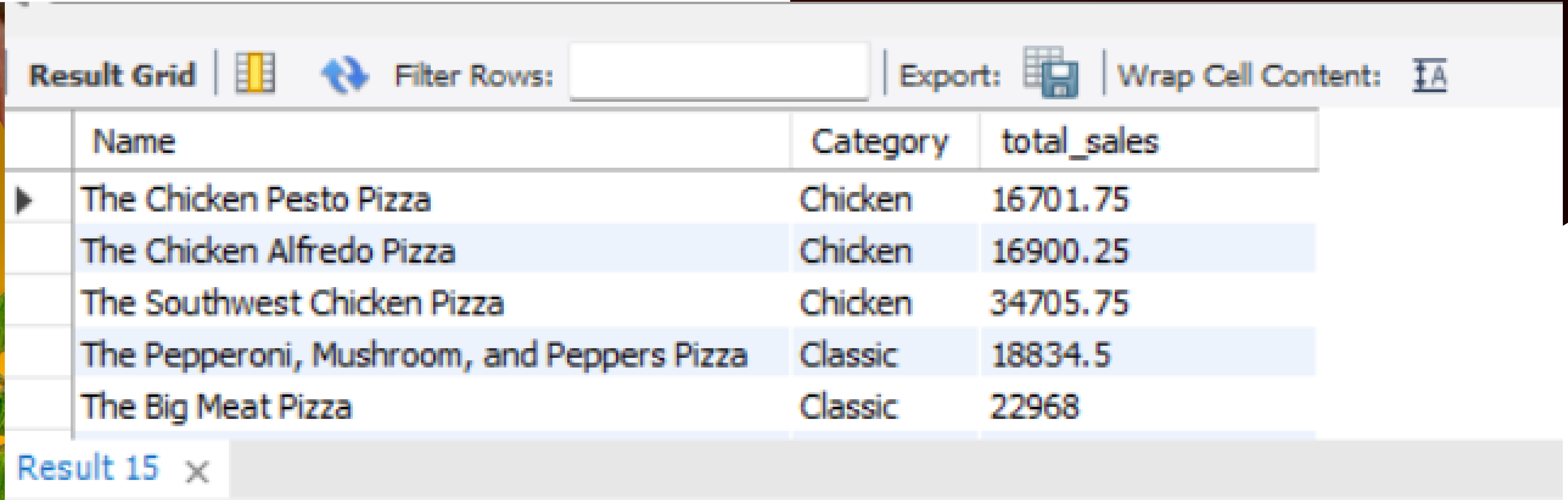


Order_date	Cum_Revenue
2015-01-01	2713.85
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5

#Determine the top 3 most ordered pizza types based on revenue for each pizza category.



```sql
-- Determine the top 3 most ordered pizza types based on revenue for each pizza category.
 Select Name, Category, total_sales
 from
 (Select Name, Category, Total_sales, RANK() over( partition by category order by total_sales) AS Ranks
 from
 (SELECT
    pizza_types.name, pizza_types.Category, SUM(Quantity * Price) AS Total_Sales
FROM
    order_detail
        JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category, pizza_types.name) AS AA) AS BB
Where ranks <= 3
```



Name	Category	total_sales
The Chicken Pesto Pizza	Chicken	16701.75
The Chicken Alfredo Pizza	Chicken	16900.25
The Southwest Chicken Pizza	Chicken	34705.75
The Pepperoni, Mushroom, and Peppers Pizza	Classic	18834.5
The Big Meat Pizza	Classic	22968

Home About Contact

THANK YOU

FOR ATTENTION

● BY: RAJVEER