SQL PROJECT

PIZZA HUT SALES DATA

>> Lalit

ABOUT THE PROJECT

THIS DATASET CONTAINS DETAILED RECORDS FROM PIZZA HUT'S SALES OPERATIONS, OFFERING A COMPREHENSIVE VIEW OF ORDERS, PIZZAS, AND THEIR TYPES.

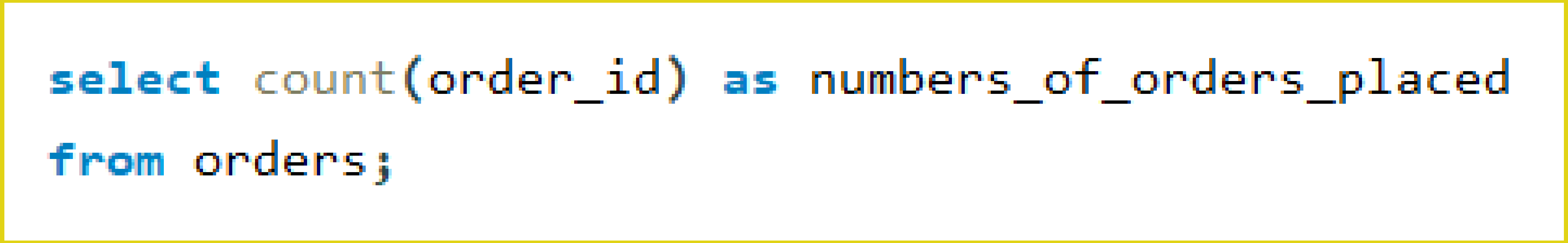
WE HAVE IMPLEMENTED SQL QUERIES RANGING FROM BASIC TO COMPLEX TO EXTRACT VALUABLE INSIGHTS FROM THE DATA.

BASIC

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.

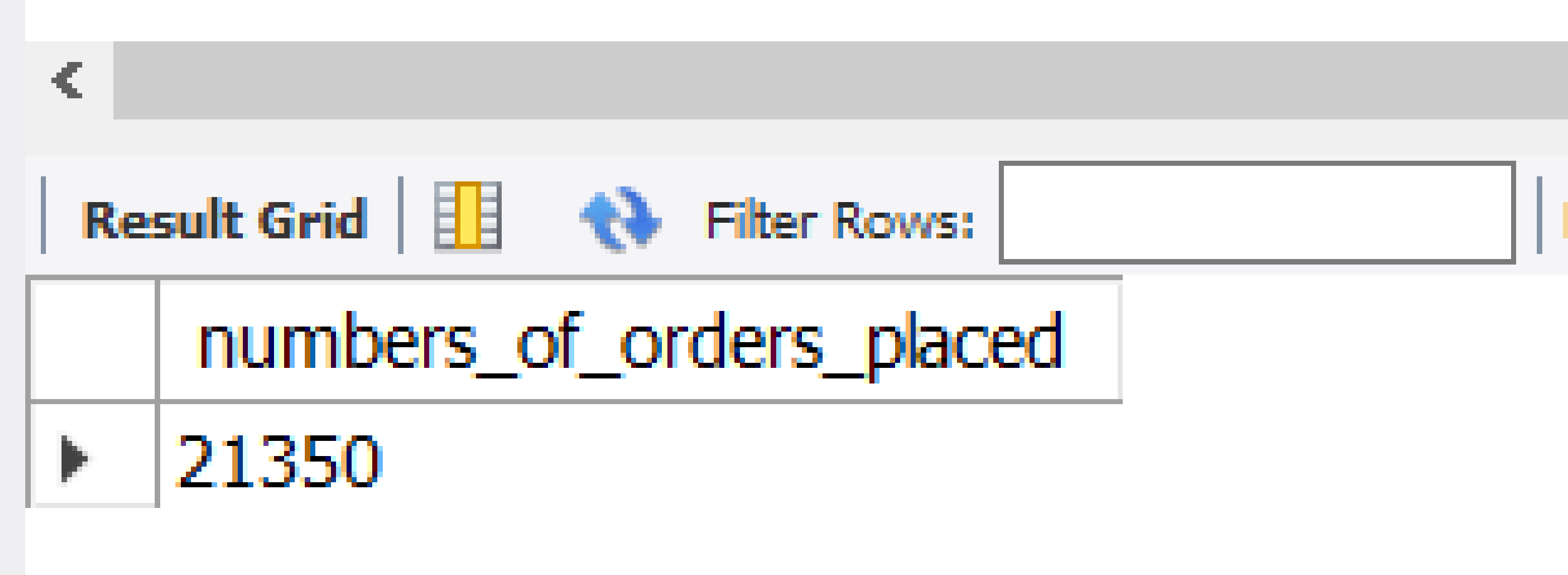
RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

- **Query:**



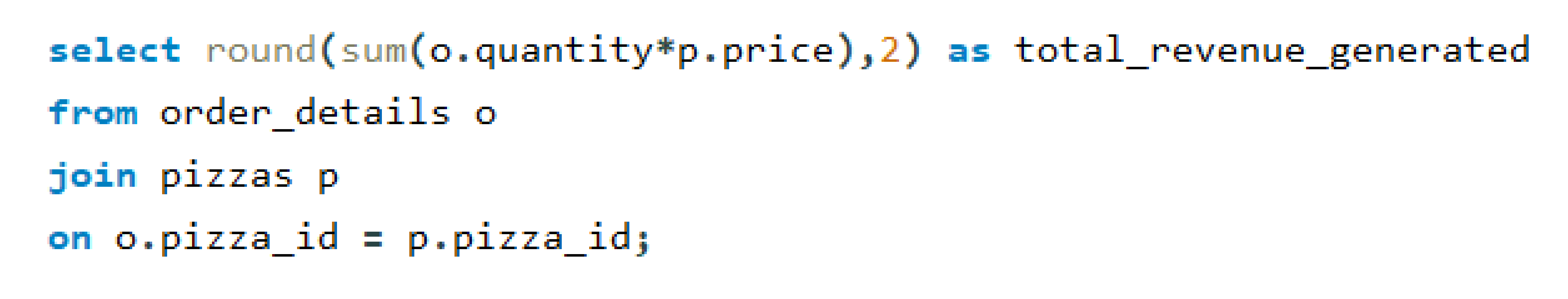
```
select count(order_id) as numbers_of_orders_placed
from orders;
```

- **Output:**

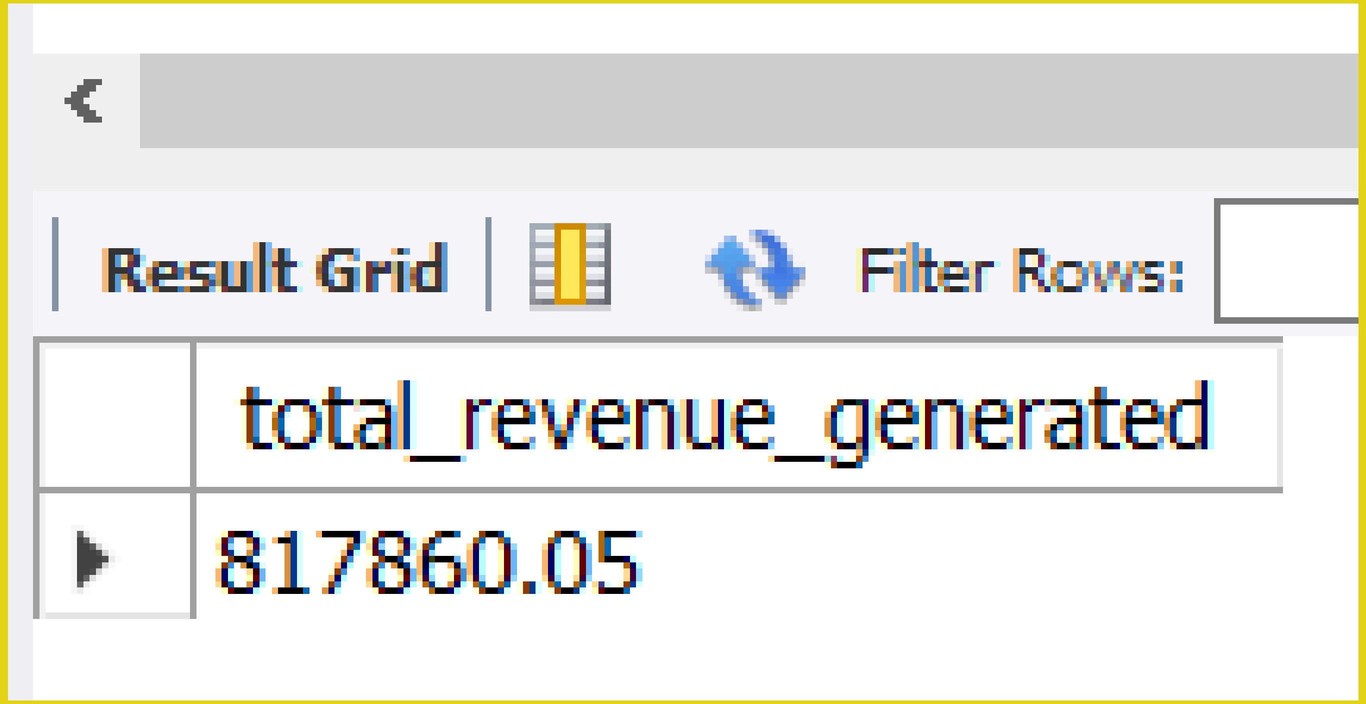


```
numbers_of_orders_placed
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

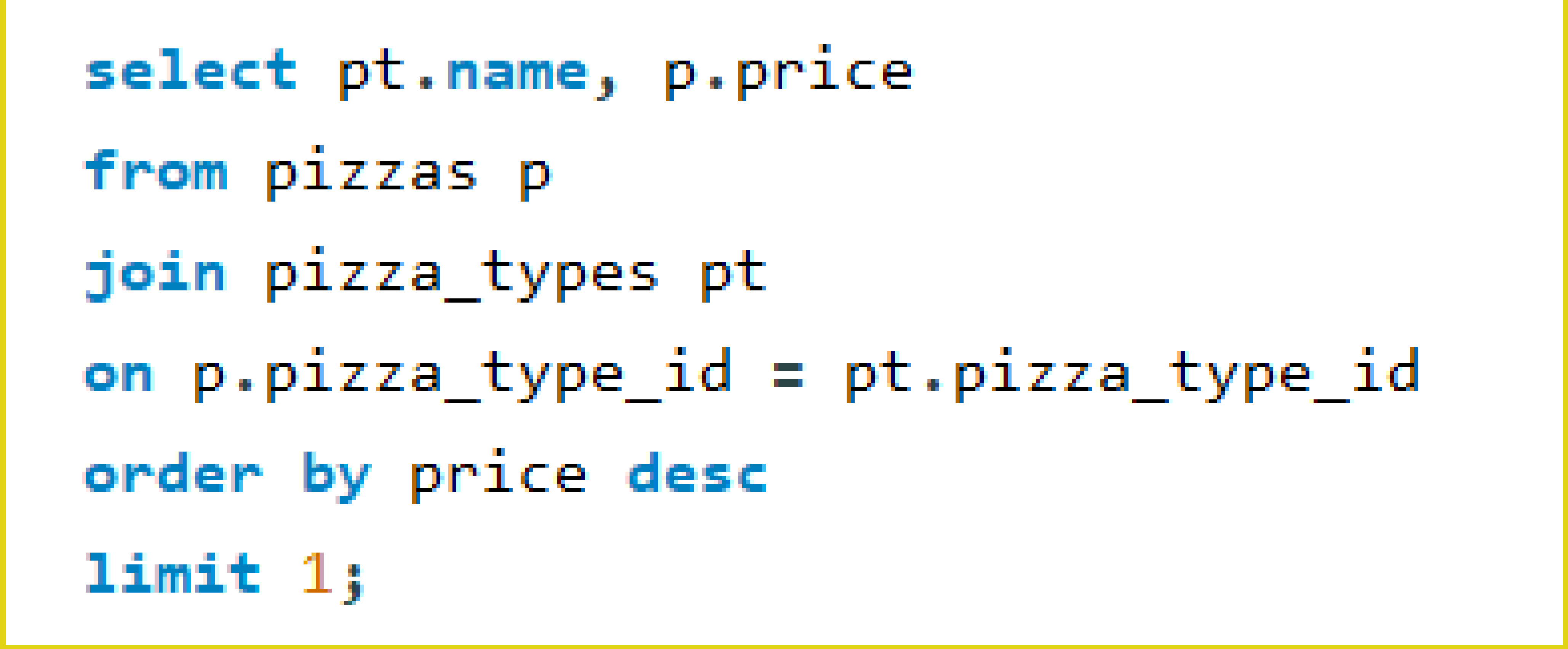


```sql
select round(sum(o.quantity*p.price),2) as total_revenue_generated
from order_details o
join pizzas p
on o.pizza_id = p.pizza_id;
```

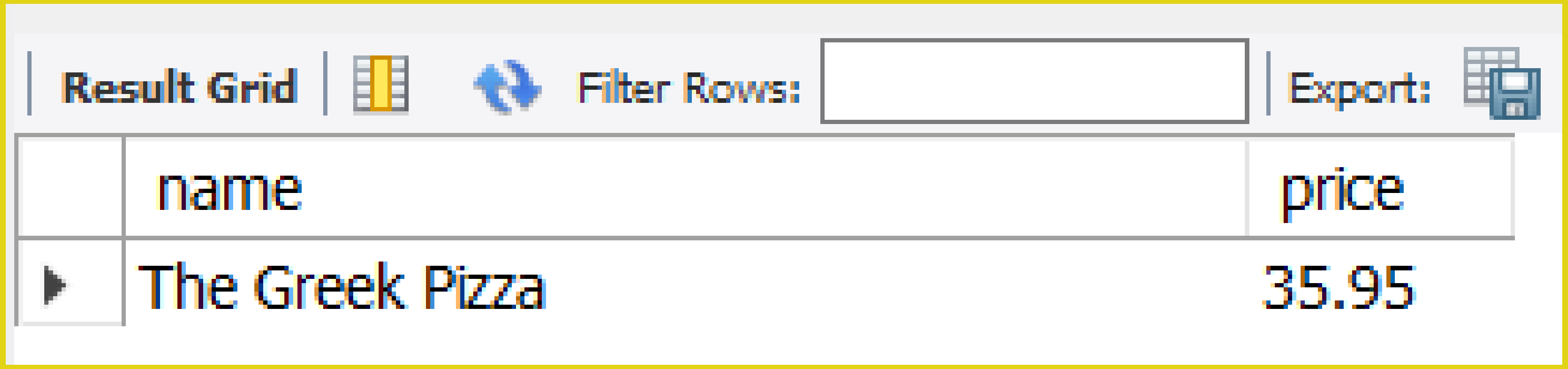


```
total_revenue_generated
817860.05
```

IDENTIFY THE HIGHEST-PRICED PIZZA



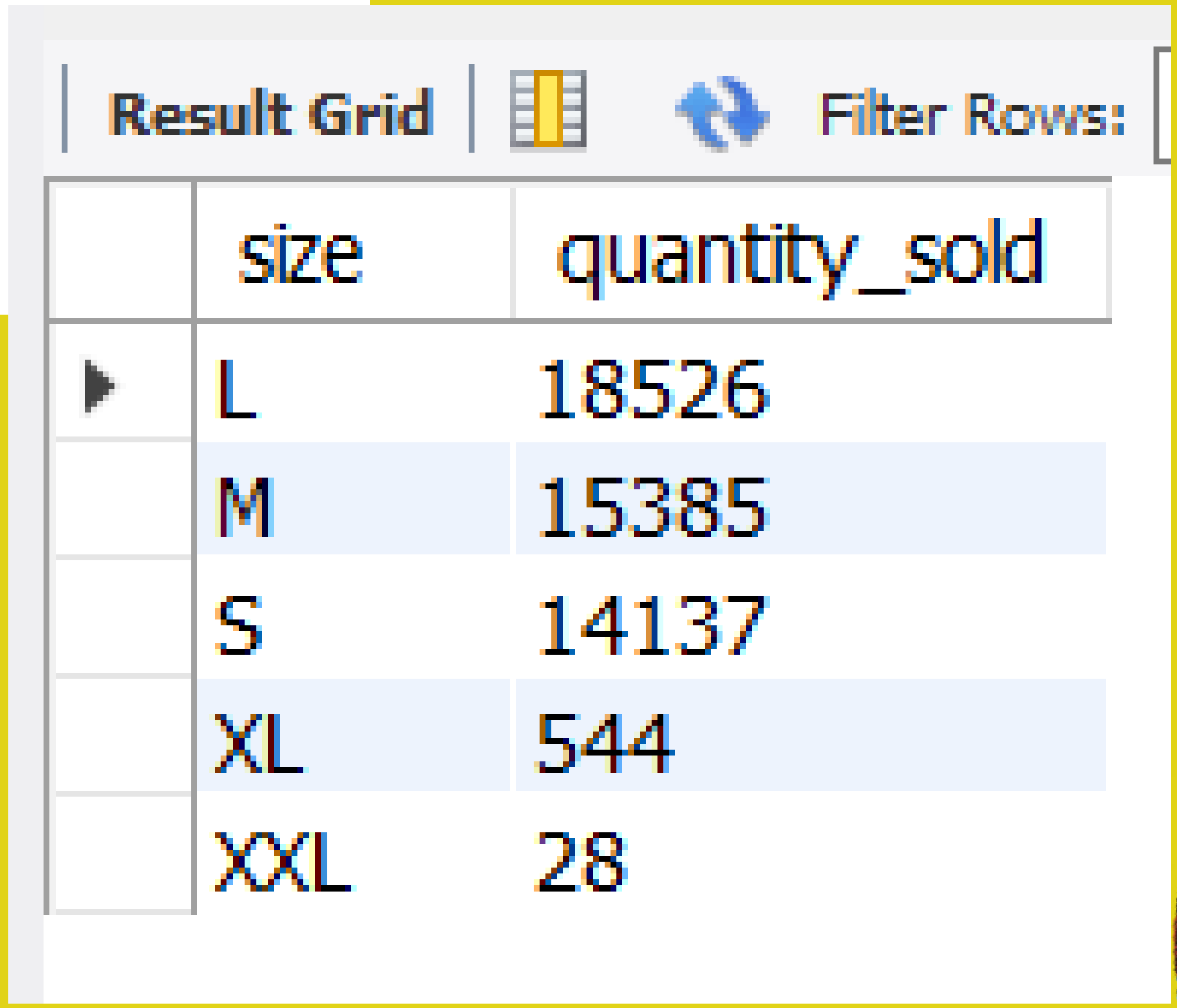
```
select pt.name, p.price
from pizzas p
join pizza_types pt
on p.pizza_type_id = pt.pizza_type_id
order by price desc
limit 1;
```



name	price
The Greek Pizza	35.95

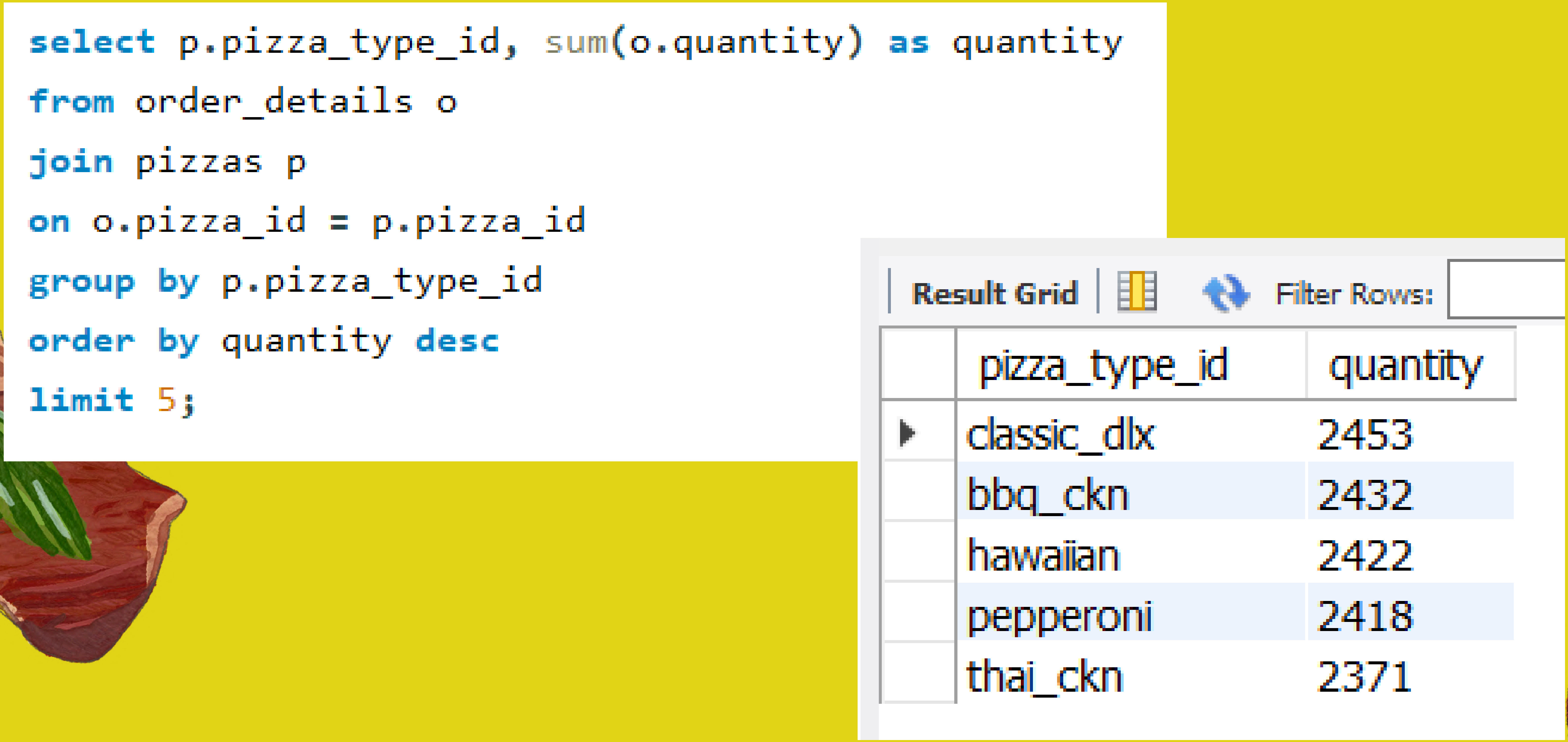
IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
select p.size, count(p.size) as quantity_sold
from order_details o
join pizzas p
on o.pizza_id = p.pizza_id
group by p.size
order by quantity_sold desc;
```



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

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



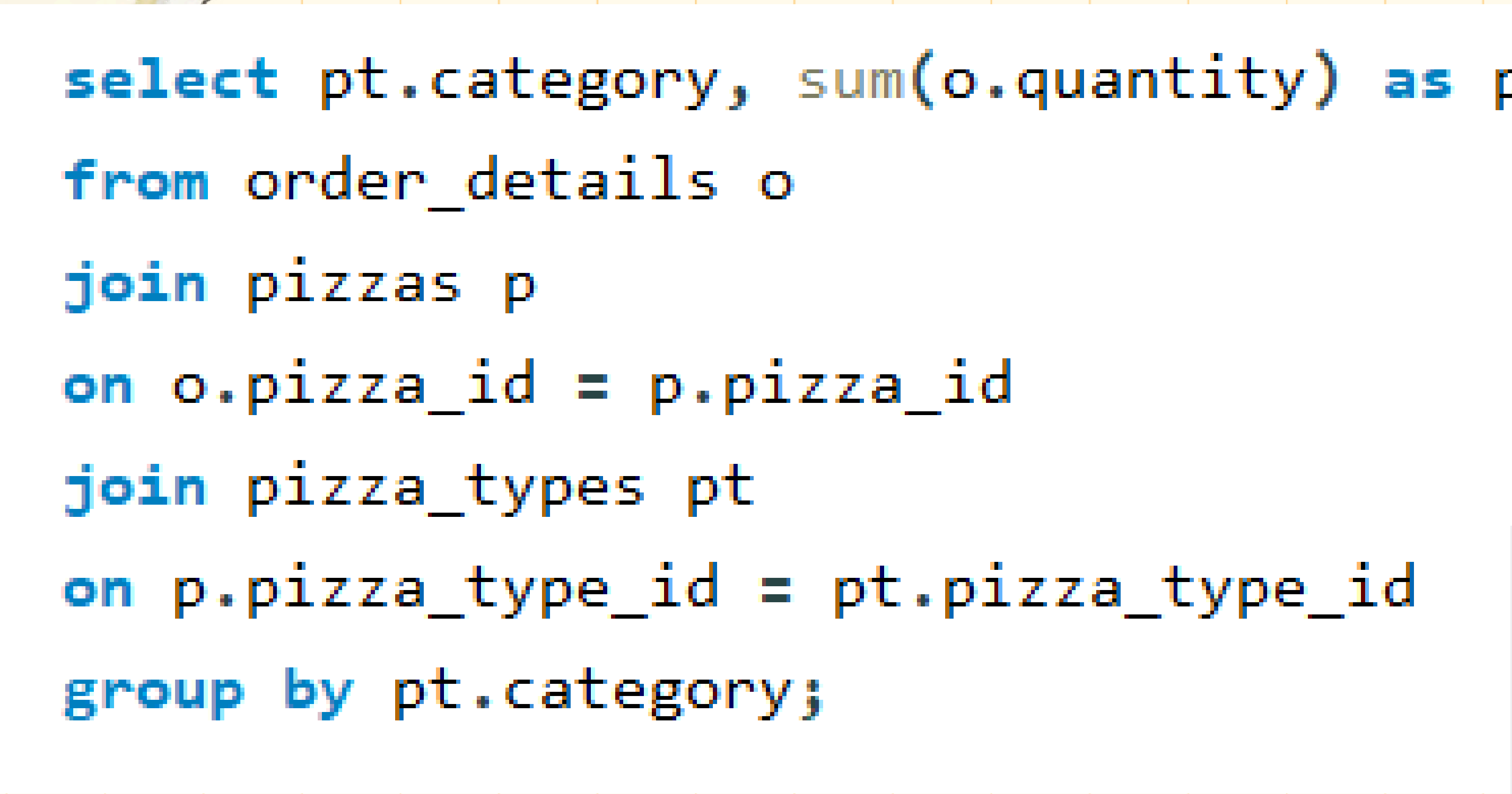
```
select p.pizza_type_id, sum(o.quantity) as quantity
from order_details o
join pizzas p
on o.pizza_id = p.pizza_id
group by p.pizza_type_id
order by quantity desc
limit 5;
```

pizza_type_id	quantity
classic_dlx	2453
bbq_ckn	2432
hawaiian	2422
pepperoni	2418
thai_ckn	2371

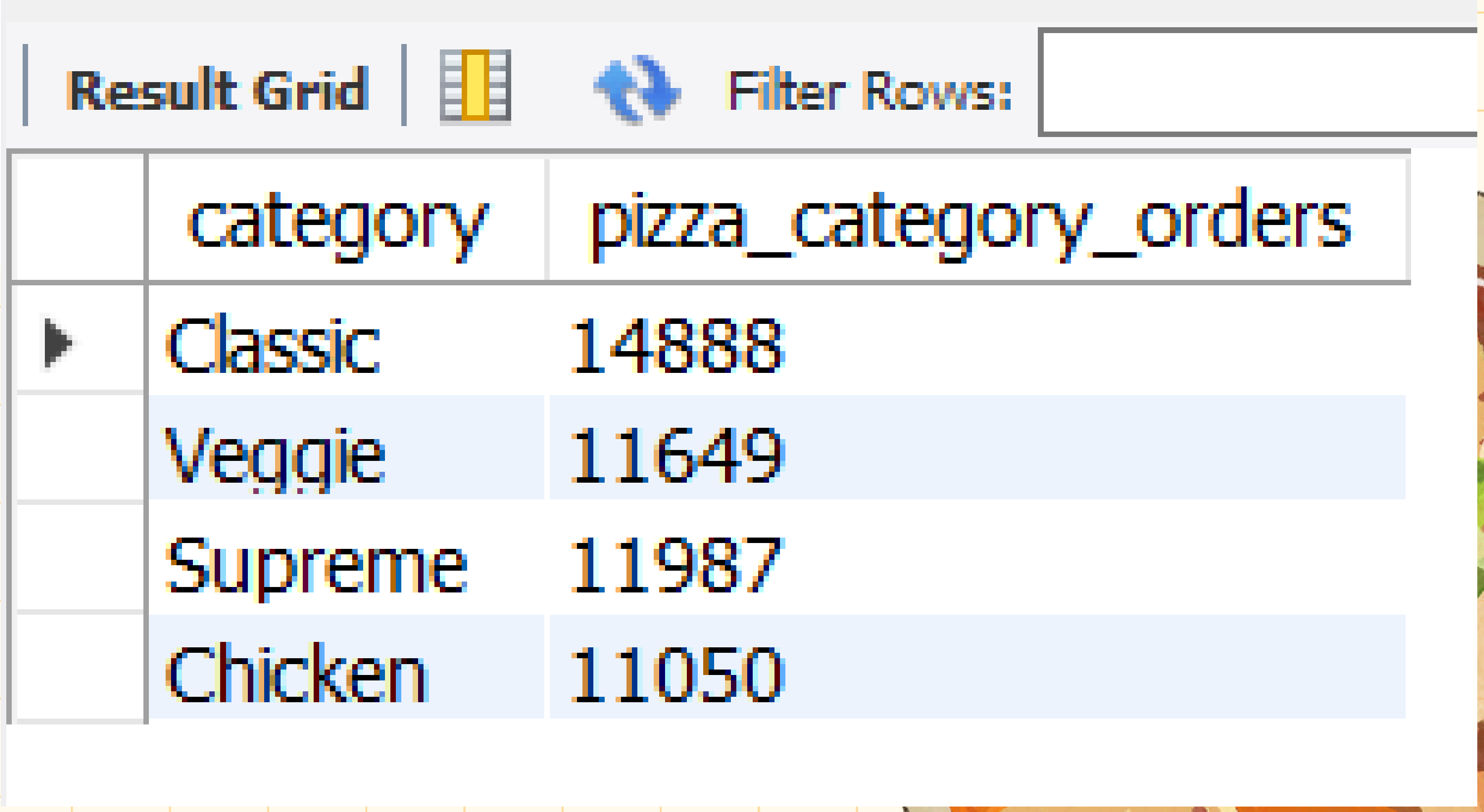
INTERMEDIATE

- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.

FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED



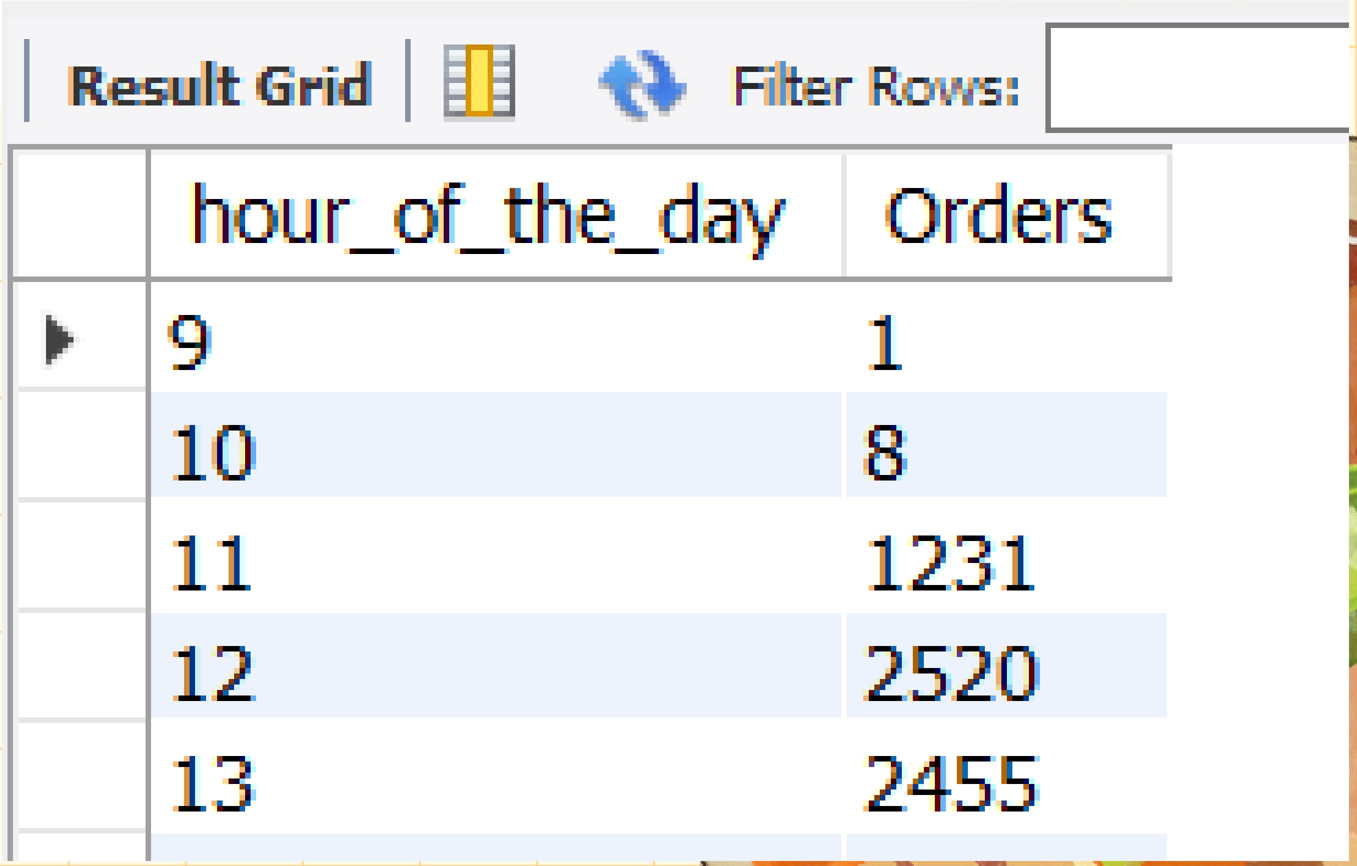
```
select pt.category, sum(o.quantity) as pizza_category_orders
from order_details o
join pizzas p
on o.pizza_id = p.pizza_id
join pizza_types pt
on p.pizza_type_id = pt.pizza_type_id
group by pt.category;
```



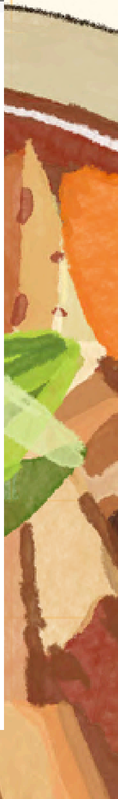
category	pizza_category_orders
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

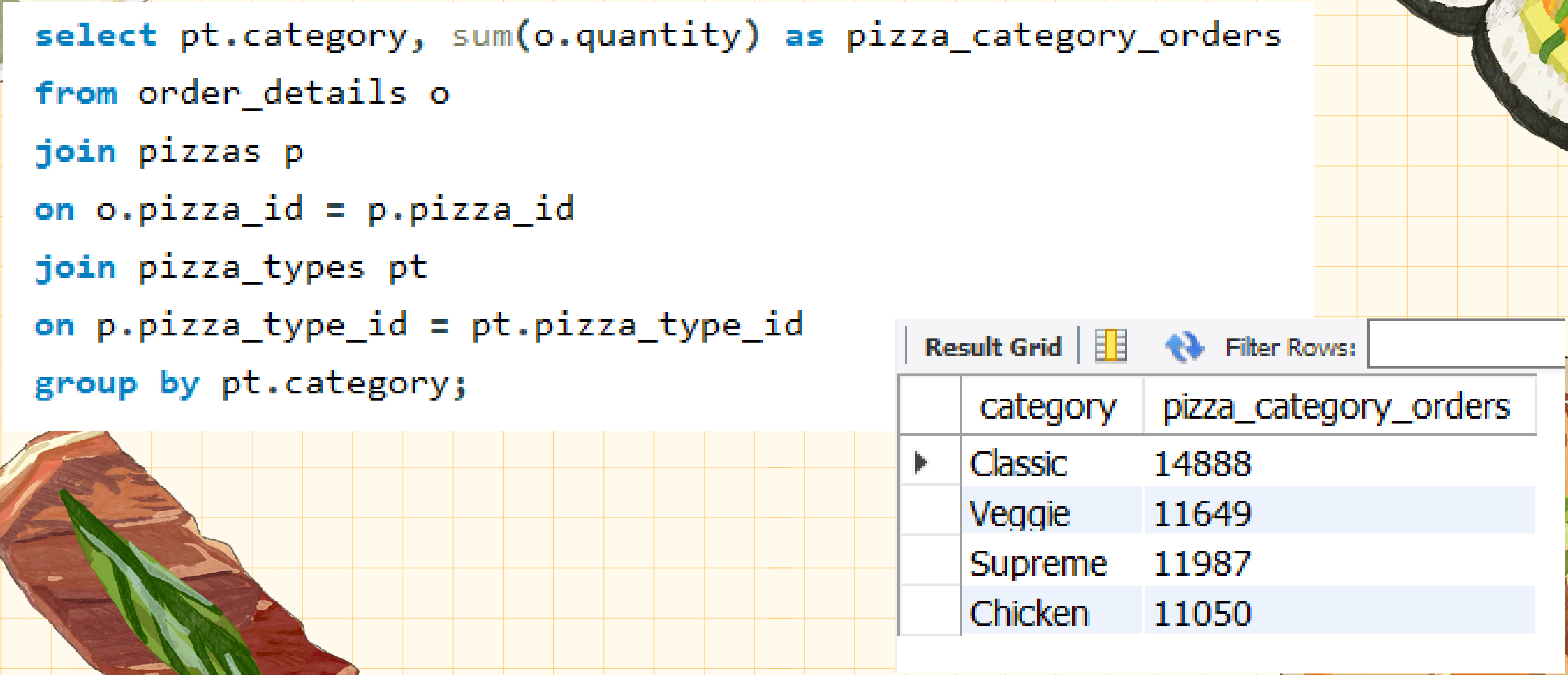
```
select hour(order_time) as hour_of_the_day, count(order_time) as Orders
from orders
group by hour(order_time)
order by hour(order_time);
```



hour_of_the_day	Orders
9	1
10	8
11	1231
12	2520
13	2455



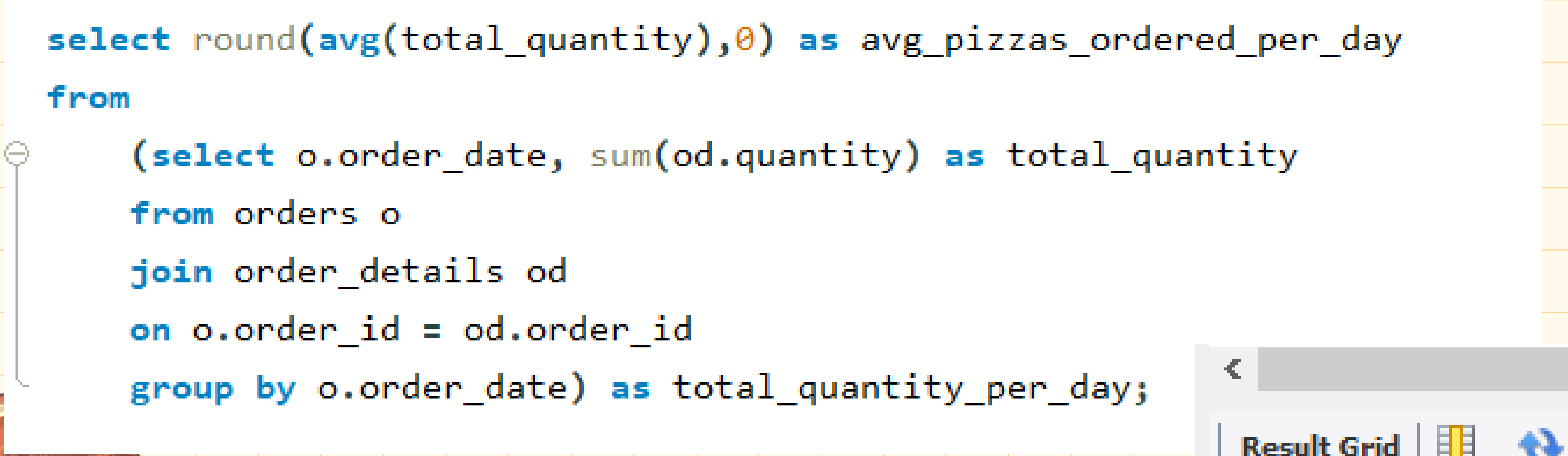
FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS



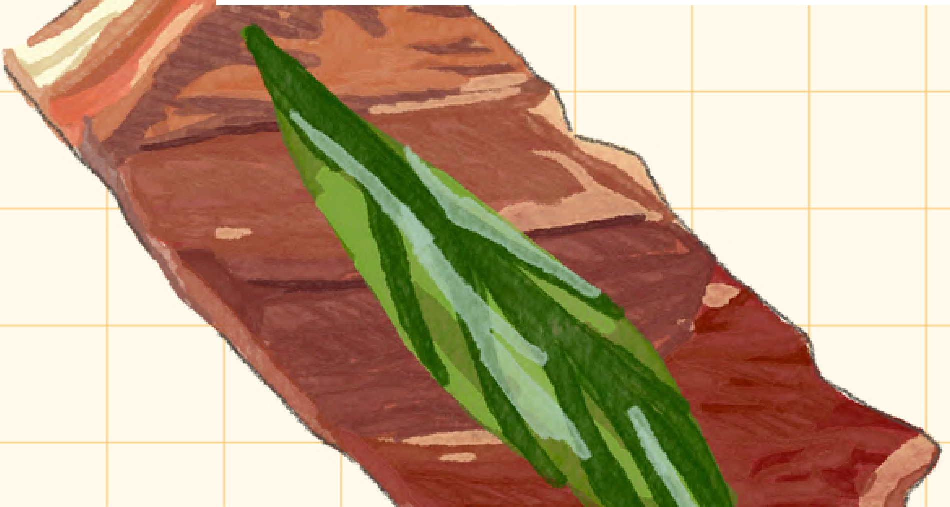
```
select pt.category, sum(o.quantity) as pizza_category_orders
from order_details o
join pizzas p
on o.pizza_id = p.pizza_id
join pizza_types pt
on p.pizza_type_id = pt.pizza_type_id
group by pt.category;
```

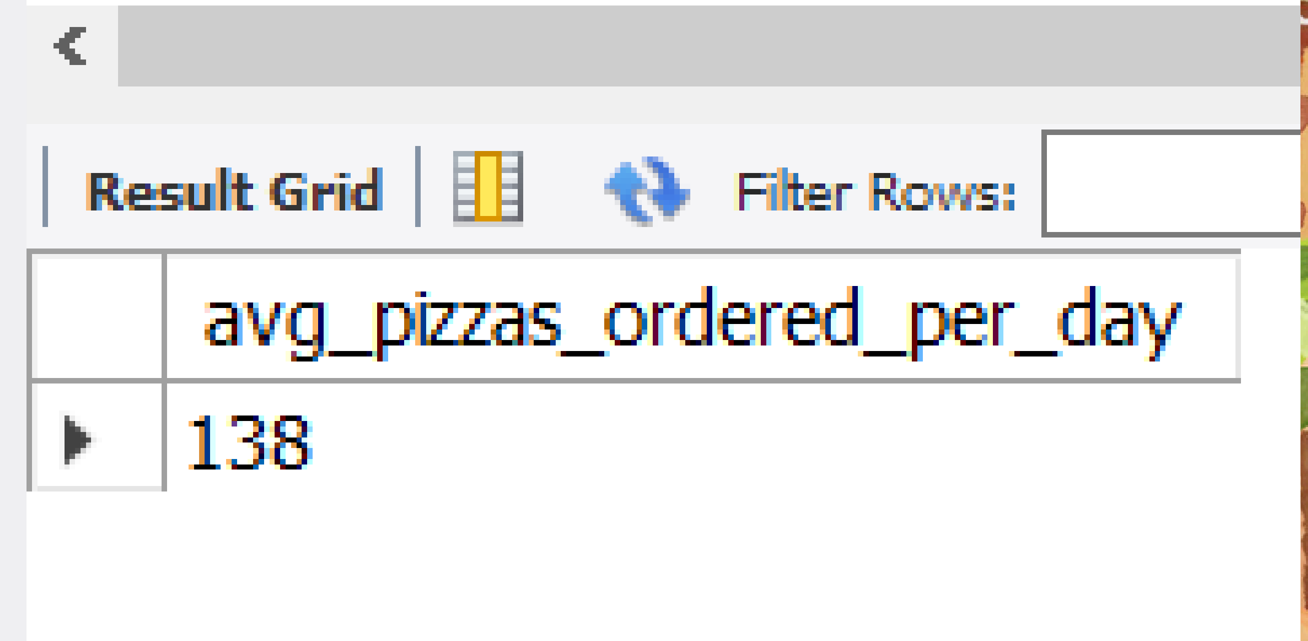
category	pizza_category_orders
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

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

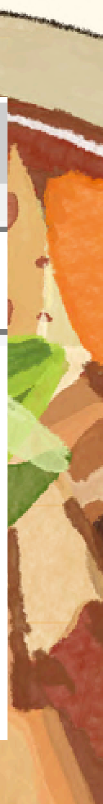


```
select round(avg(total_quantity),0) as avg_pizzas_ordered_per_day
from
    (select o.order_date, sum(od.quantity) as total_quantity
    from orders o
    join order_details od
    on o.order_id = od.order_id
    group by o.order_date) as total_quantity_per_day;
```

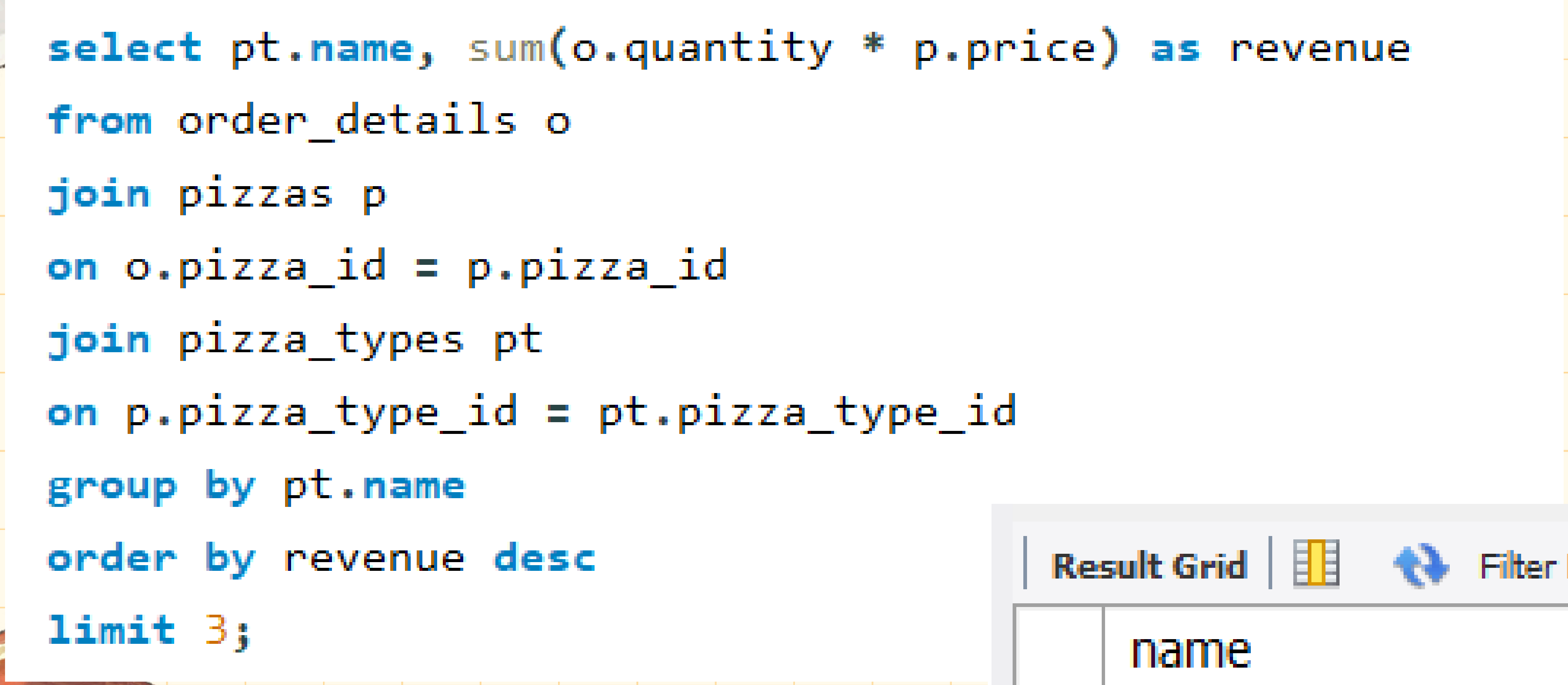




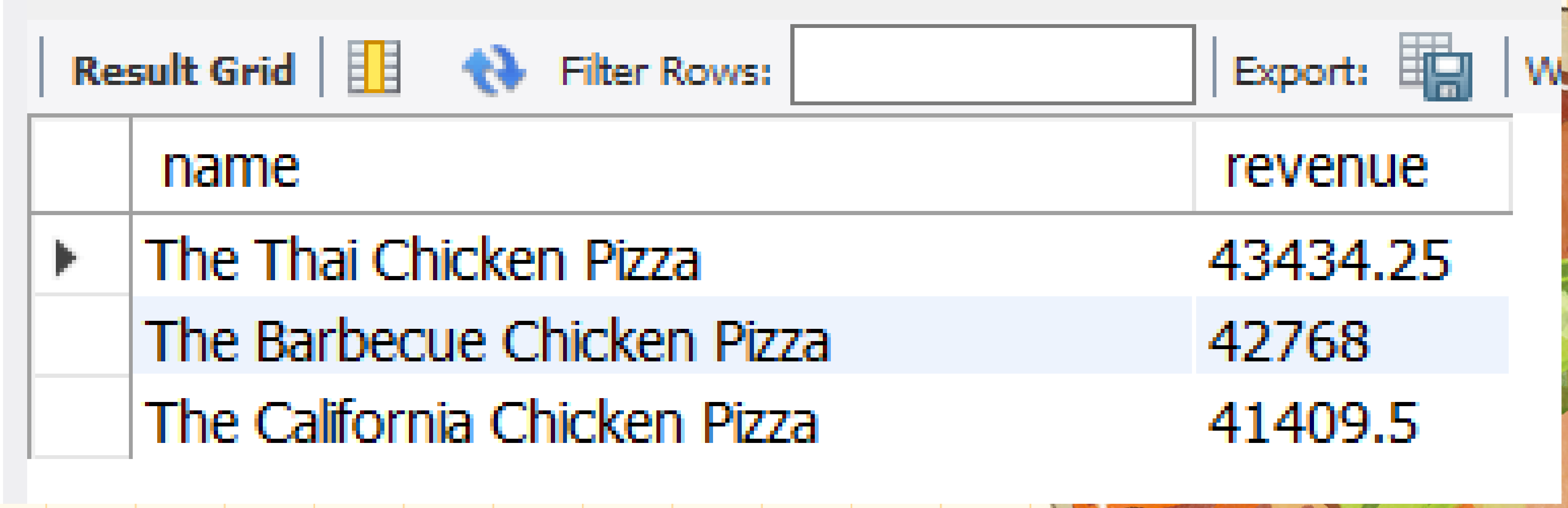
avg_pizzas_ordered_per_day
138



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE



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

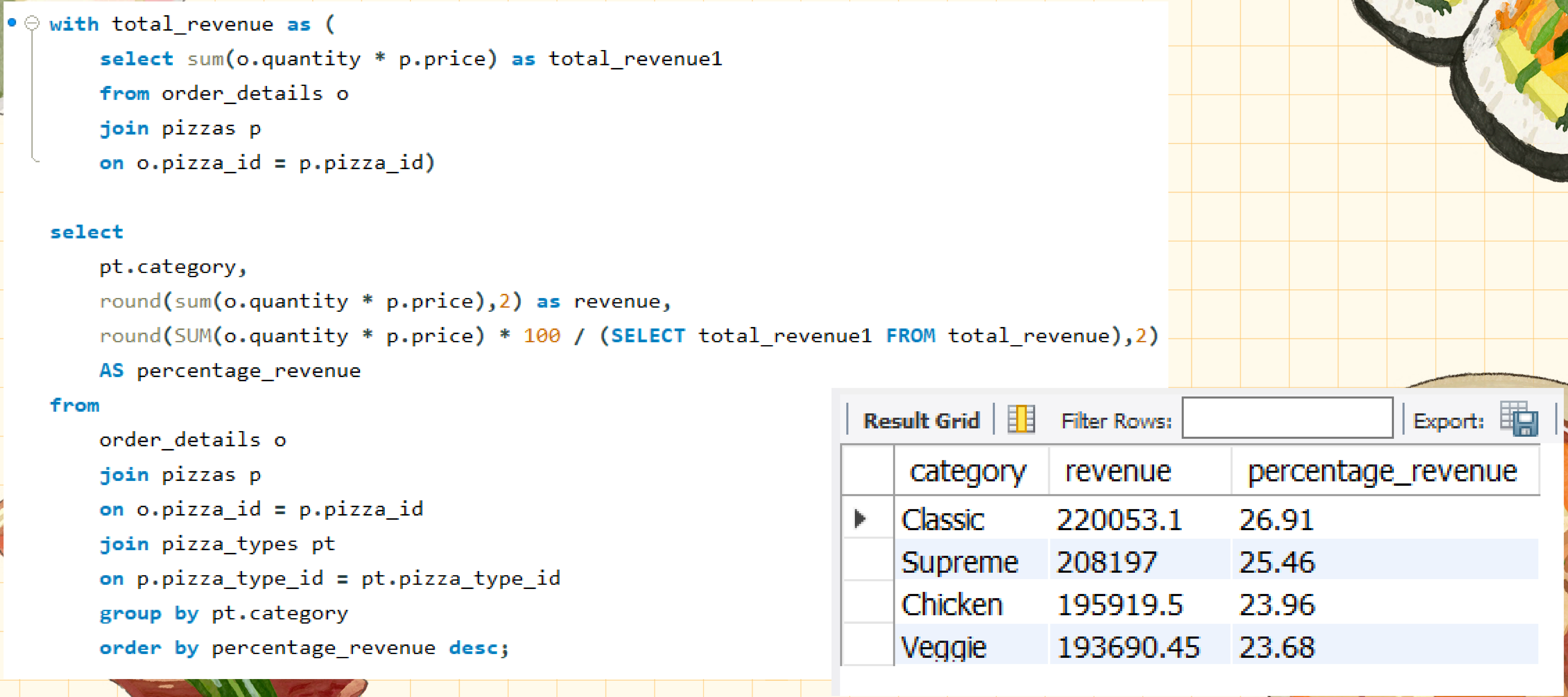


name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

COMPLEX

- Calculate the percentage contribution of each pizza type to total revenue.
    - Using Subquery
    - Using CTE
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

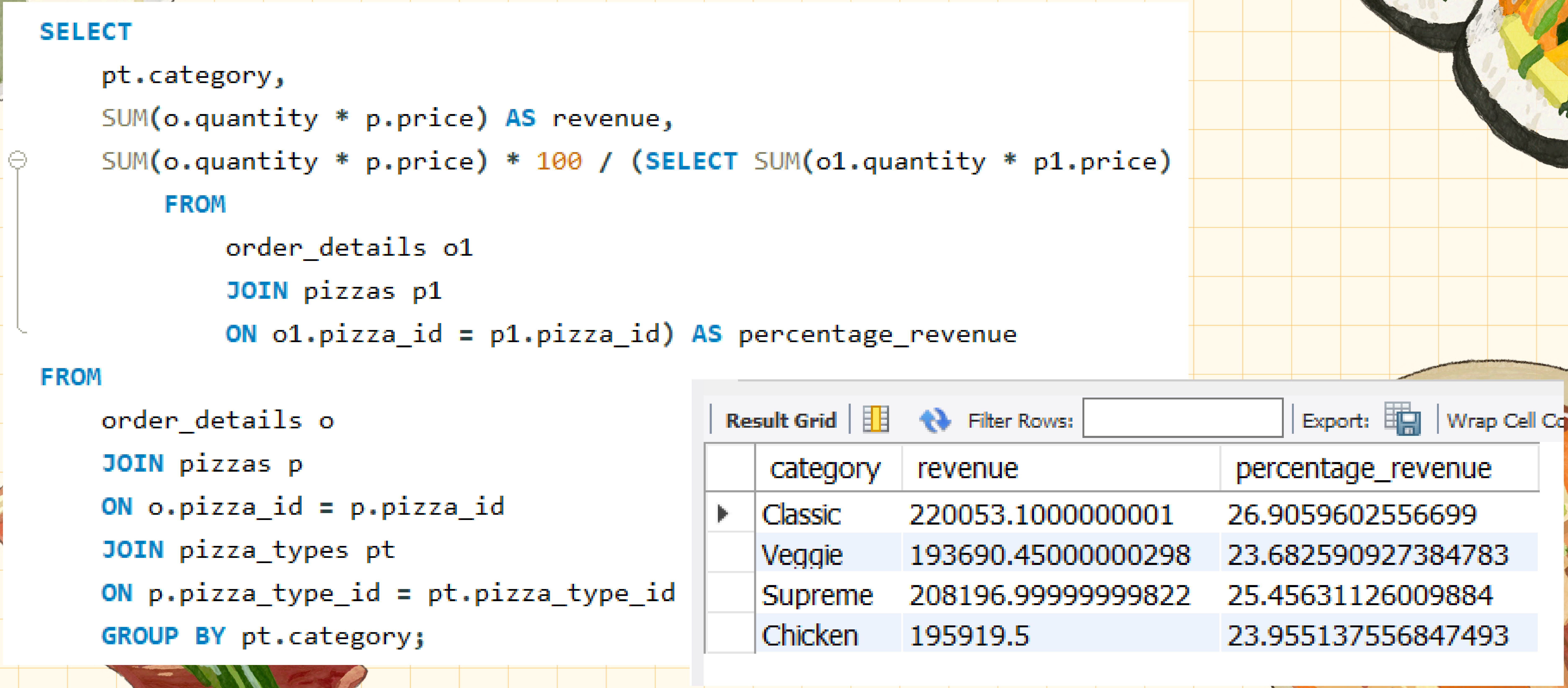


```
with total_revenue as (
    select sum(o.quantity * p.price) as total_revenue1
    from order_details o
    join pizzas p
    on o.pizza_id = p.pizza_id)

select
    pt.category,
    round(sum(o.quantity * p.price),2) as revenue,
    round(SUM(o.quantity * p.price) * 100 / (SELECT total_revenue1 FROM total_revenue),2)
    AS percentage_revenue
from
    order_details o
    join pizzas p
    on o.pizza_id = p.pizza_id
    join pizza_types pt
    on p.pizza_type_id = pt.pizza_type_id
    group by pt.category
    order by percentage_revenue desc;
```

category	revenue	percentage_revenue
Classic	220053.1	26.91
Supreme	208197	25.46
Chicken	195919.5	23.96
Veggie	193690.45	23.68

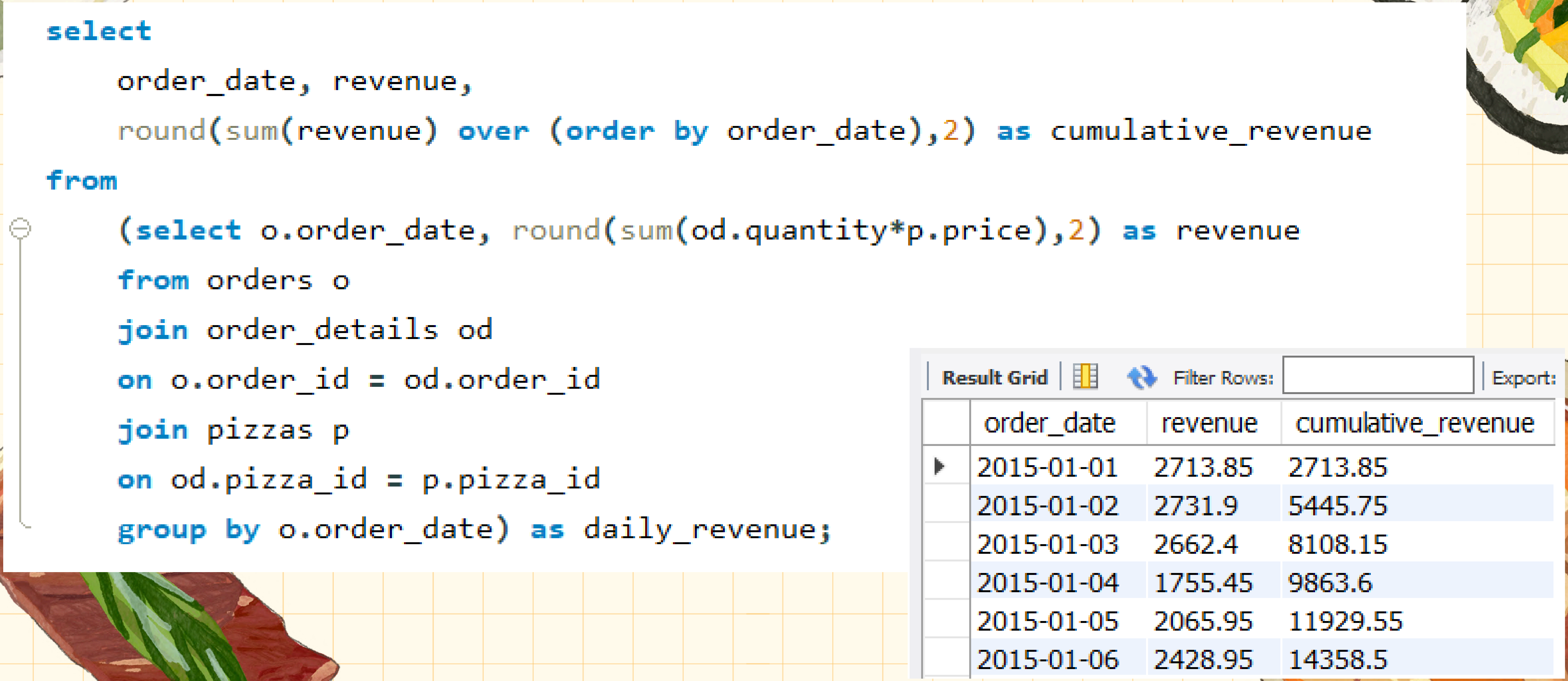
CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE



```
SELECT
    pt.category,
    SUM(o.quantity * p.price) AS revenue,
    SUM(o.quantity * p.price) * 100 / (SELECT SUM(o1.quantity * p1.price)
        FROM
            order_details o1
            JOIN pizzas p1
            ON o1.pizza_id = p1.pizza_id) AS percentage_revenue
FROM
    order_details o
    JOIN pizzas p
    ON o.pizza_id = p.pizza_id
    JOIN pizza_types pt
    ON p.pizza_type_id = pt.pizza_type_id
    GROUP BY pt.category;
```

	category	revenue	percentage_revenue
▶	Classic	220053.1000000001	26.905960255669
	Veggie	193690.45000000298	23.68259092738478
	Supreme	208196.99999999822	25.4563112600988
	Chicken	195919.5	23.9551375568474

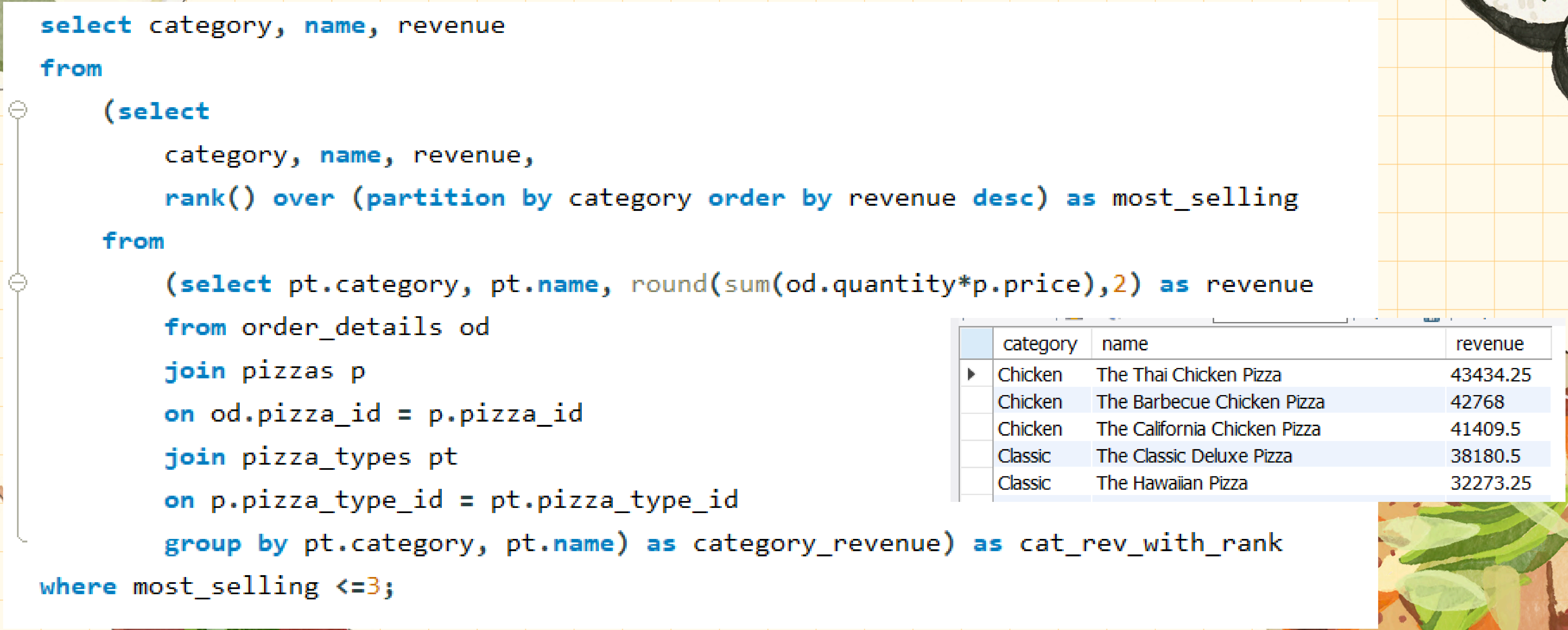
ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME



```sql
select
    order_date, revenue,
    round(sum(revenue) over (order by order_date),2) as cumulative_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 daily_revenue;
```

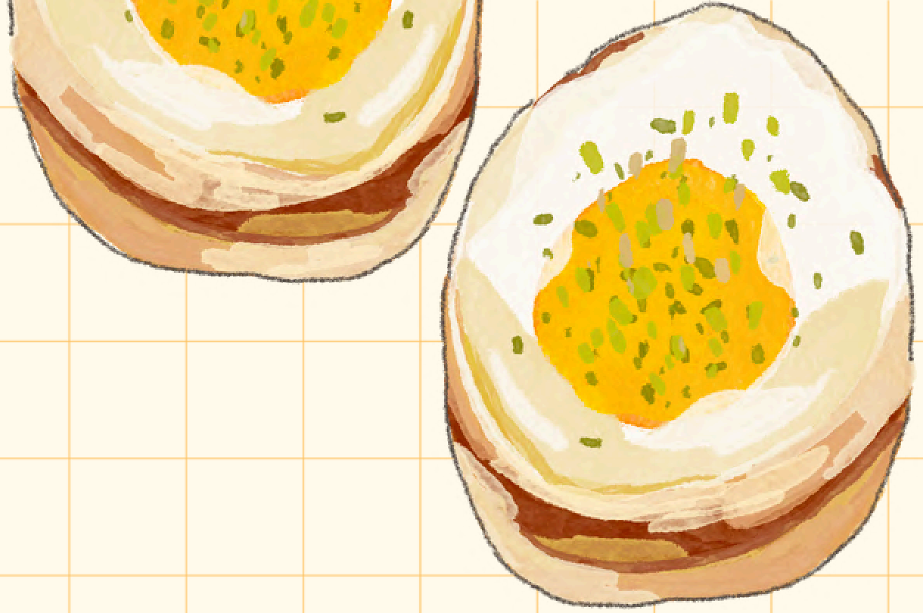
order_date	revenue	cumulative_revenue
2015-01-01	2713.85	2713.85
2015-01-02	2731.9	5445.75
2015-01-03	2662.4	8108.15
2015-01-04	1755.45	9863.6
2015-01-05	2065.95	11929.55
2015-01-06	2428.95	14358.5

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



```
select category, name, revenue
from
    (select
        category, name, revenue,
        rank() over (partition by category order by revenue desc) as most_selling
    from
        (select pt.category, pt.name, round(sum(od.quantity*p.price),2) as revenue
        from order_details od
        join pizzas p
        on od.pizza_id = p.pizza_id
        join pizza_types pt
        on p.pizza_type_id = pt.pizza_type_id
        group by pt.category, pt.name) as category_revenue) as cat_rev_with_rank
where most_selling <=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



GRAZIE!