PIZZA SALES ANALYSIS USING SQL

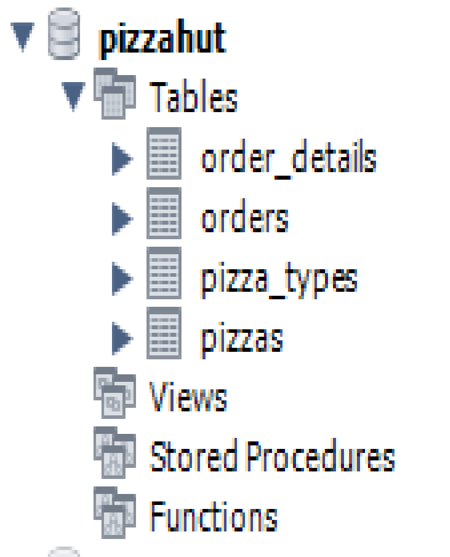
INTRODUCTION

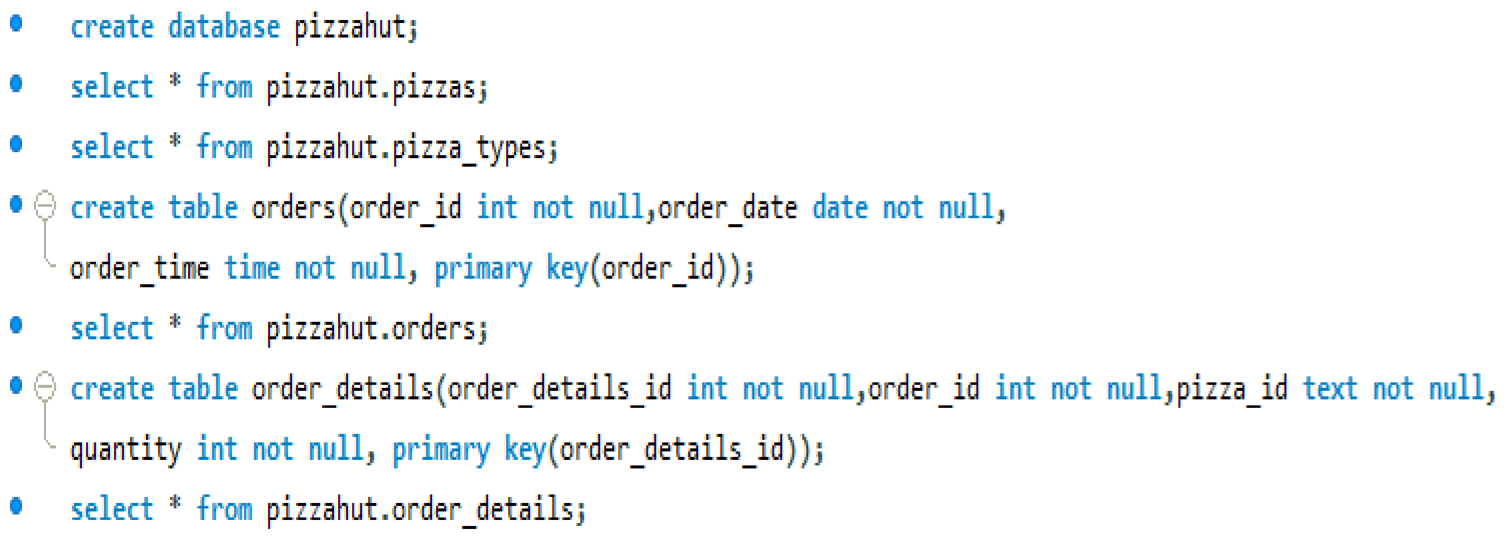
The Pizza Sales Project harnesses the power of SQL to transform raw sales data into meaningful insights. This project not only helps in understanding current sales dynamics but also equips the restaurant with data-driven strategies to boost future performance. Through meticulous data analysis, we aim to elevate the overall customer experience and drive sustained business growth.

AGENDA

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

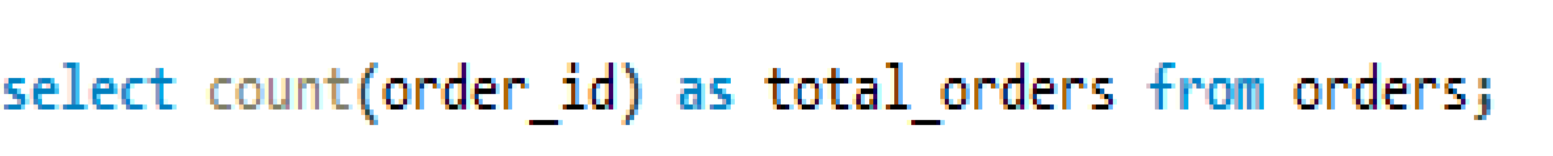
PIZZA SALES DATABASE



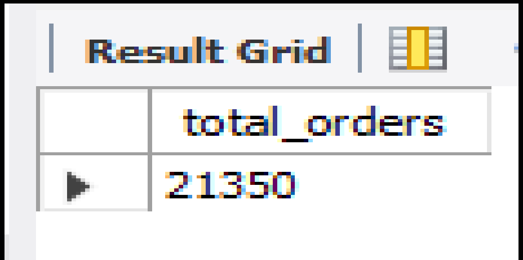


```
create database pizzahut;
select * from pizzahut.pizzas;
select * from pizzahut.pizza_types;
create table orders(order_id int not null,order_date date not null,
order_time time not null, primary key(order_id));
select * from pizzahut.orders;
create table order_details(order_details_id int not null,order_id int not null,pizza_id text not null,
quantity int not null, primary key(order_details_id));
select * from pizzahut.order_details;
```

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

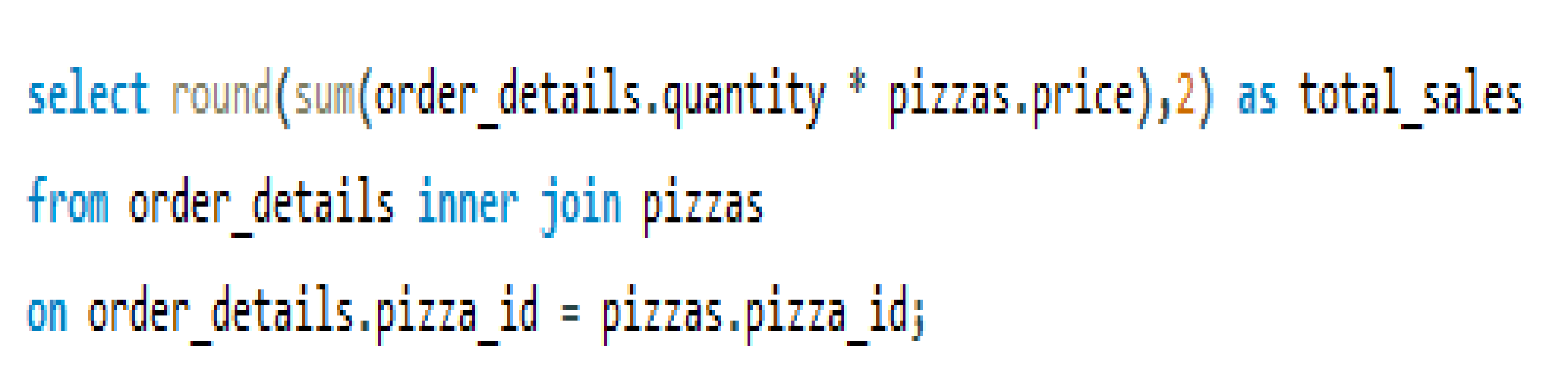


```sql
select count(order_id) as total_orders from orders;
```

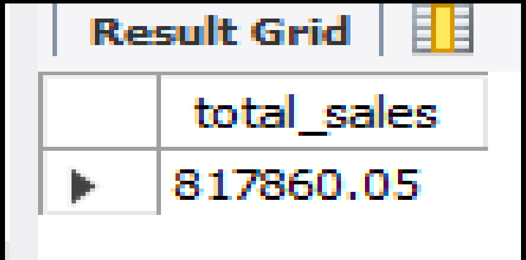


	total_orders
▶	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

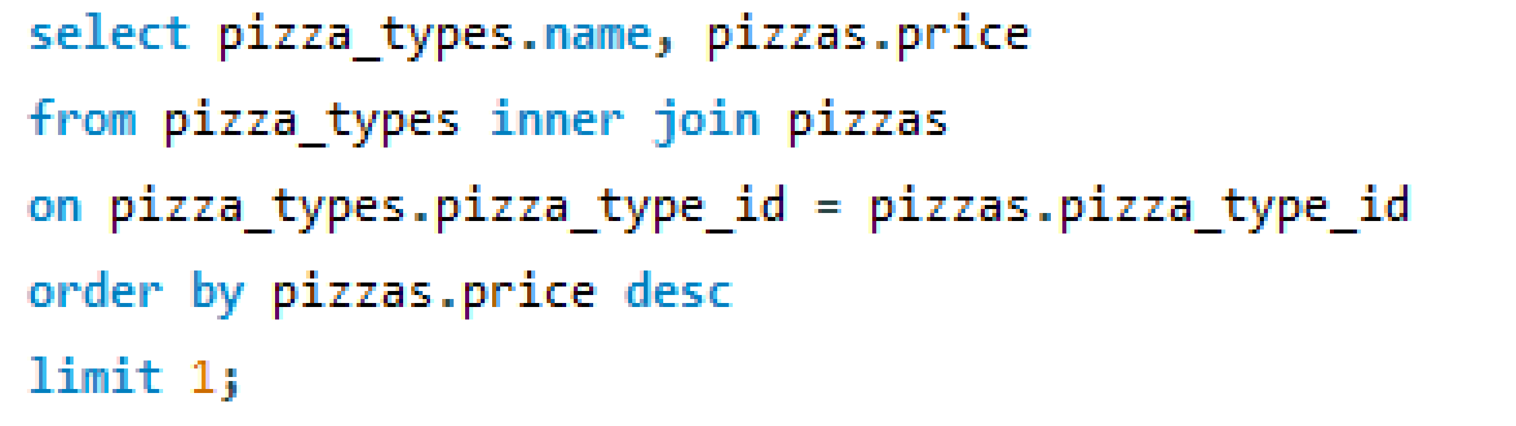


```sql
select round(sum(order_details.quantity * pizzas.price),2) as total_sales
from order_details inner join pizzas
on order_details.pizza_id = pizzas.pizza_id;
```

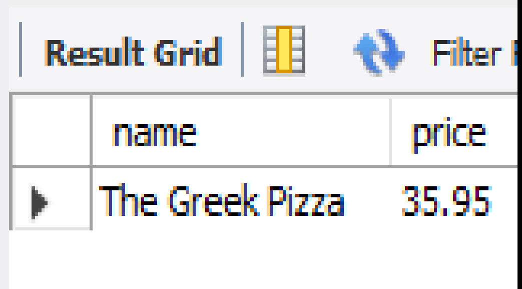


```
total_sales
817860.05
```

IDENTIFY THE HIGHEST-PRICED PIZZA

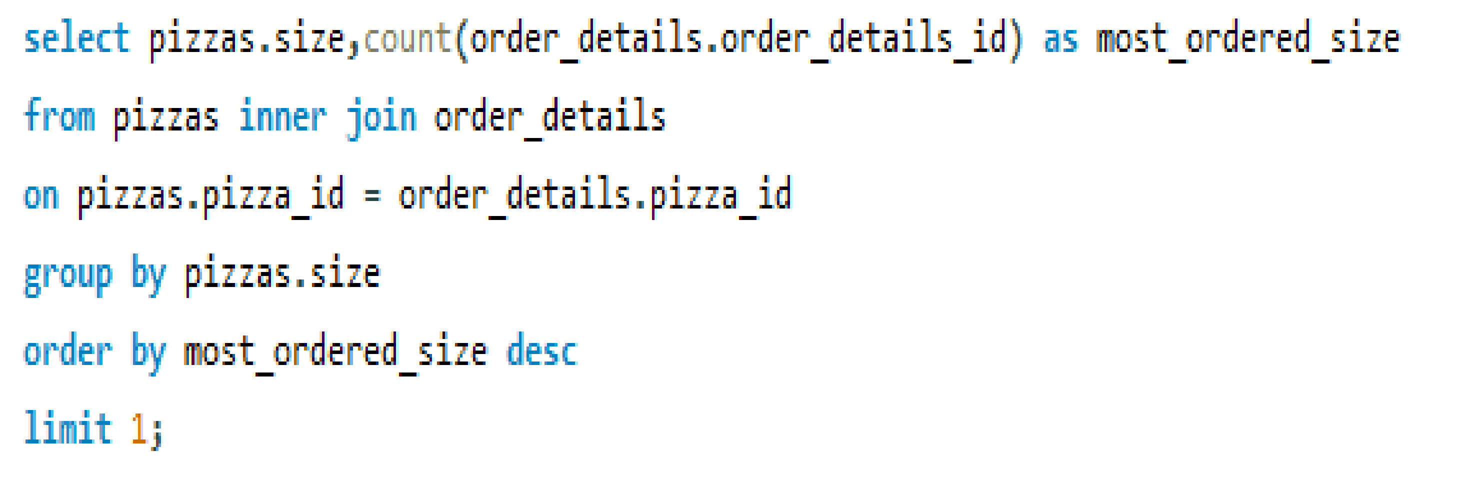


```
select pizza_types.name, pizzas.price
from pizza_types inner join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
order by pizzas.price desc
limit 1;
```

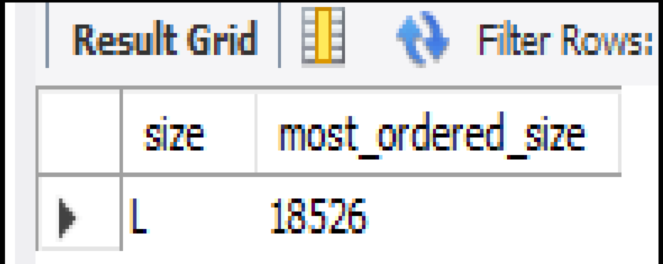


name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

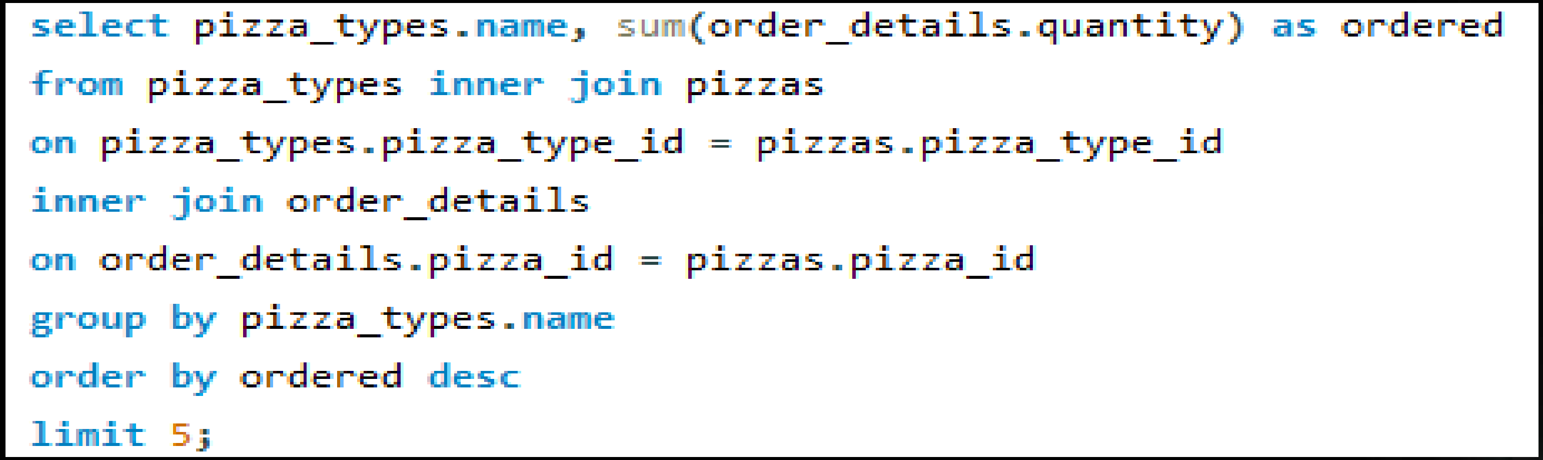


```
select pizzas.size,count(order_details.order_details_id) as most_ordered_size
from pizzas inner join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizzas.size
order by most_ordered_size desc
limit 1;
```

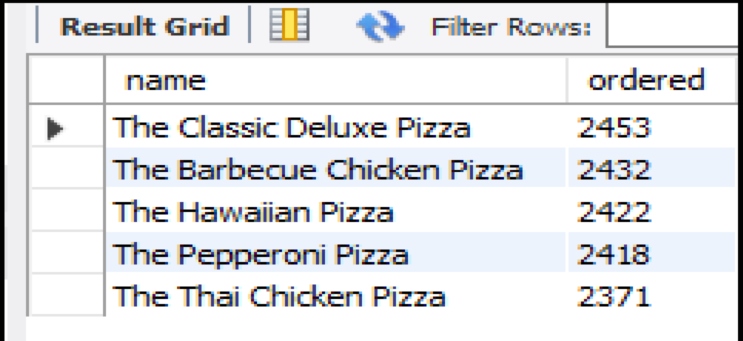


```
size    most_ordered_size
L       18526
```

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

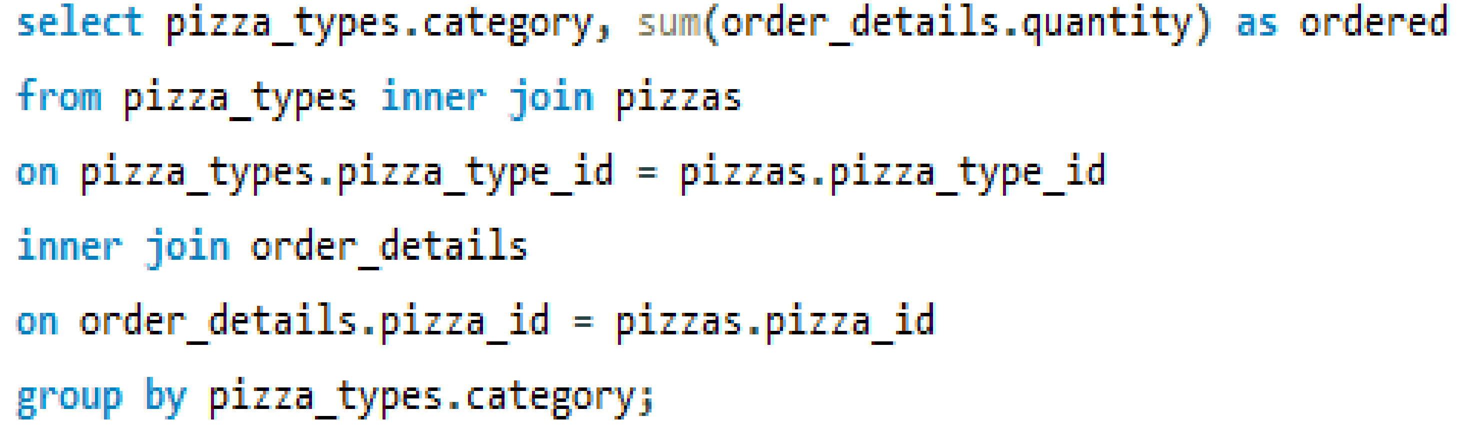


```
select pizza_types.name, sum(order_details.quantity) as ordered
from pizza_types inner join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
inner join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.name
order by ordered desc
limit 5;
```

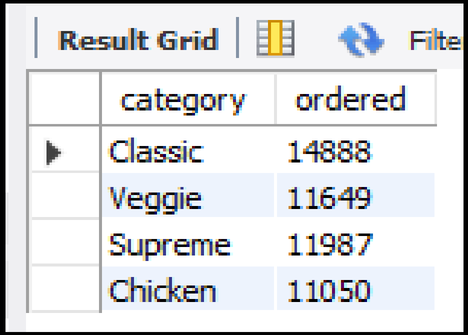


name	ordered
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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

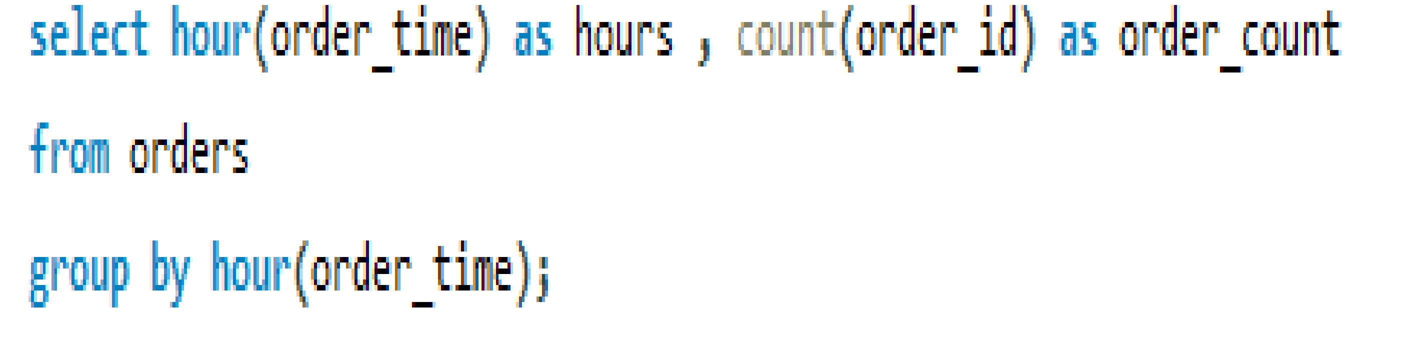


```
select pizza_types.category, sum(order_details.quantity) as ordered
from pizza_types inner join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
inner join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category;
```

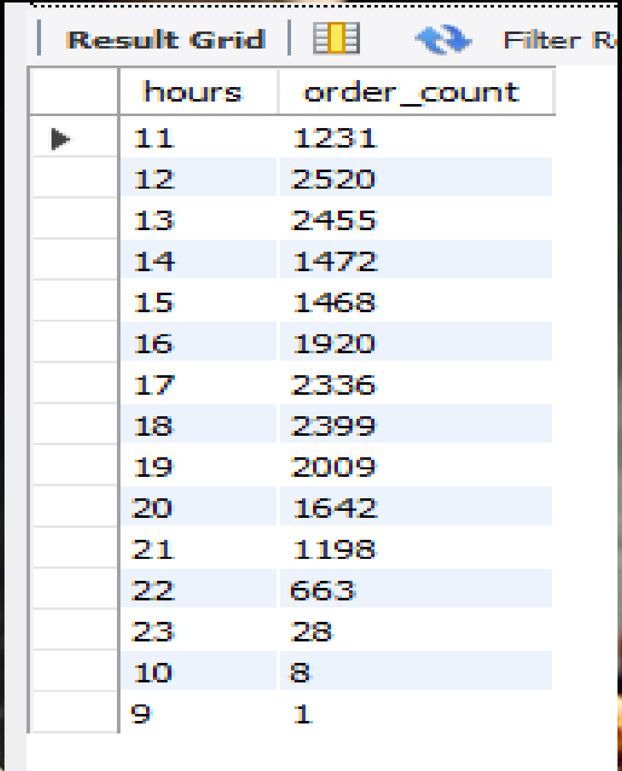


category	ordered
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

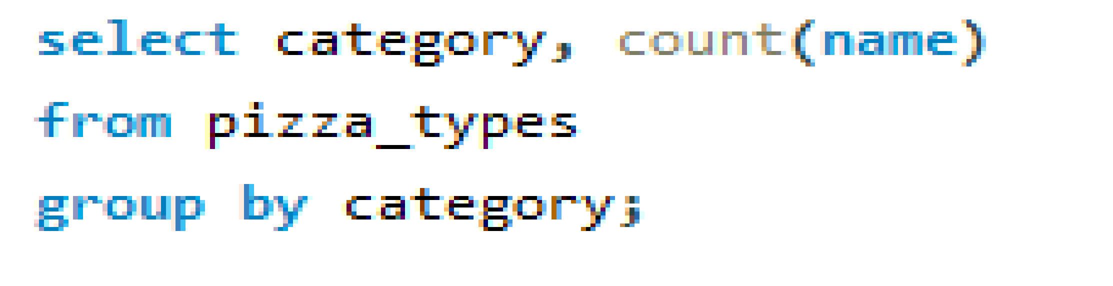


```
select hour(order_time) as hours , count(order_id) as order_count

from orders

group by hour(order_time);
```

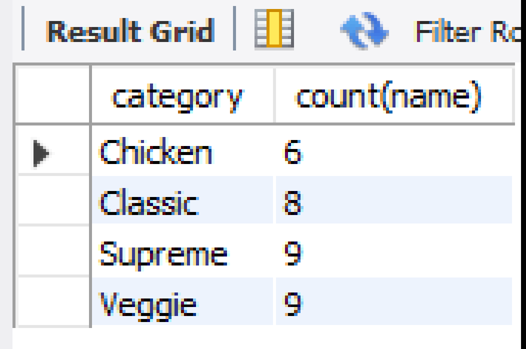


hours	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

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

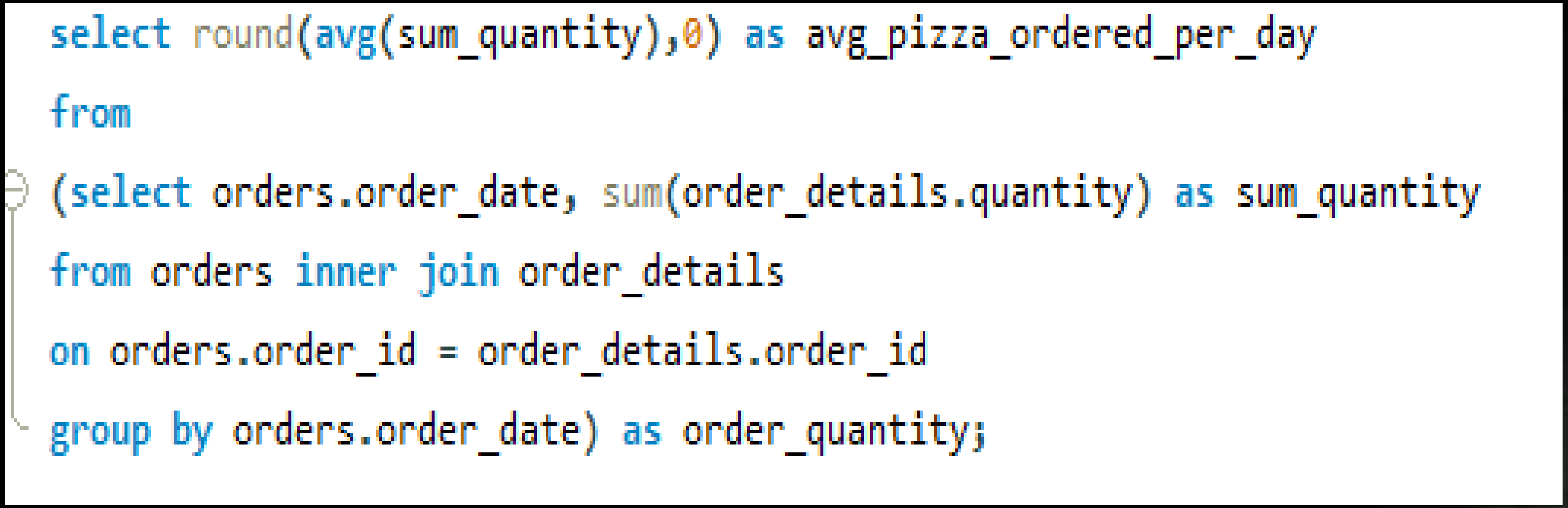


```
select category, count(name)
from pizza_types
group by category;
```

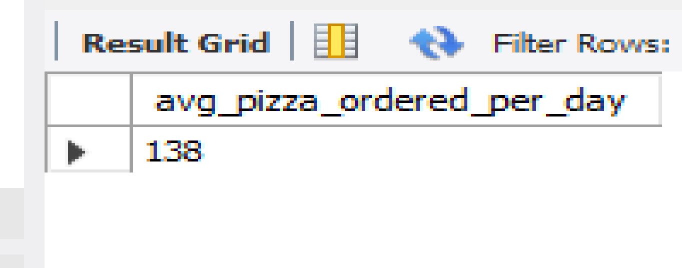


category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

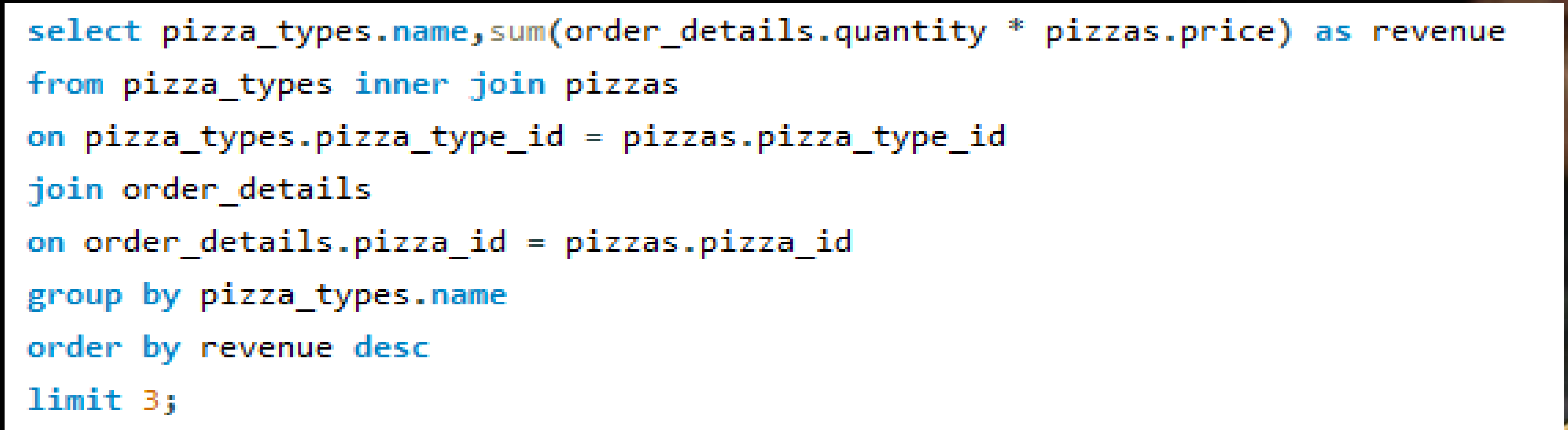


```
select round(avg(sum_quantity),0) as avg_pizza_ordered_per_day
from
(select orders.order_date, sum(order_details.quantity) as sum_quantity
from orders inner join order_details
on orders.order_id = order_details.order_id
group by orders.order_date) as order_quantity;
```

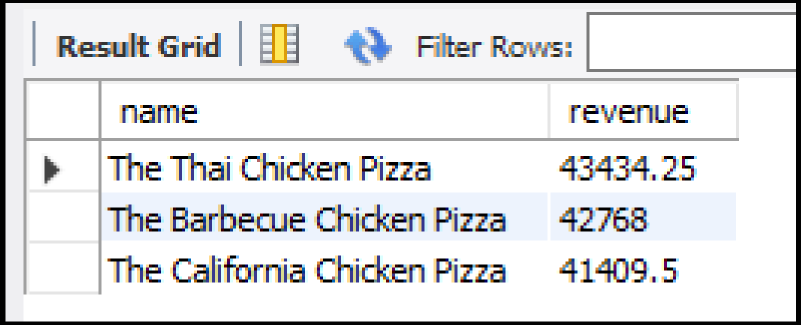


```
avg_pizza_ordered_per_day
138
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

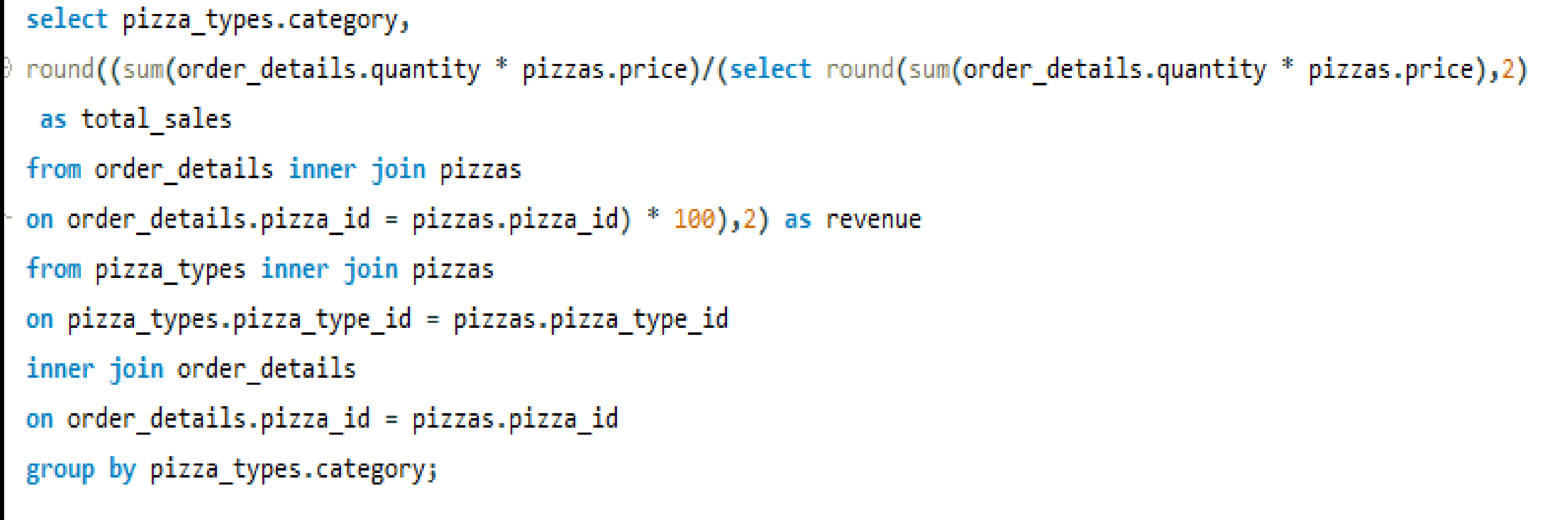


```sql
select pizza_types.name,sum(order_details.quantity * pizzas.price) as revenue
from pizza_types inner join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.name
order by revenue desc
limit 3;
```

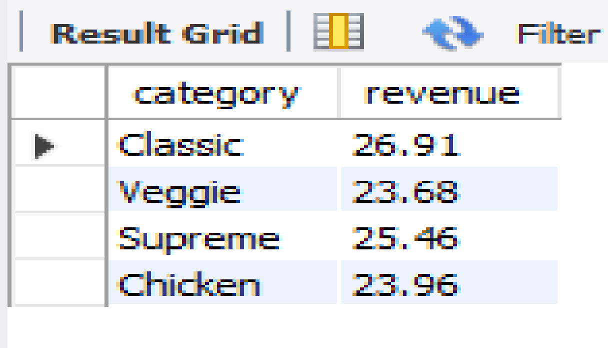


name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

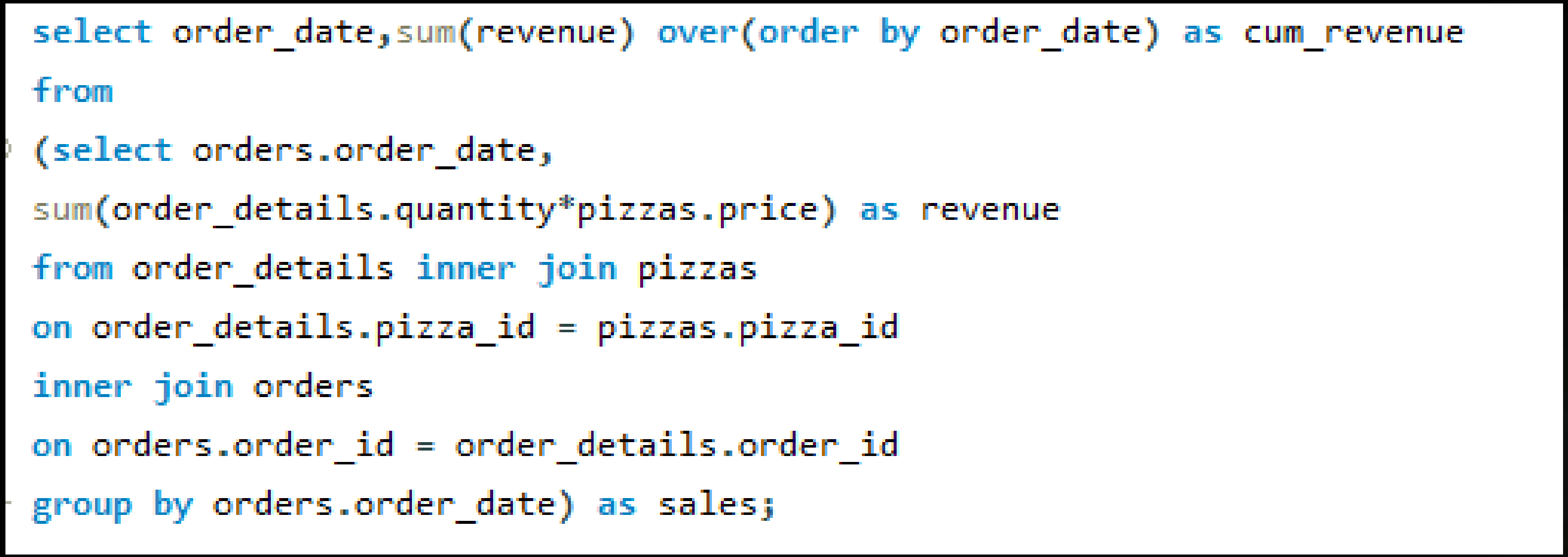


```
select pizza_types.category,
round((sum(order_details.quantity * pizzas.price)/(select round(sum(order_details.quantity * pizzas.price),2)
 as total_sales
from order_details inner join pizzas
on order_details.pizza_id = pizzas.pizza_id) * 100),2) as revenue
from pizza_types inner join pizzas
on pizza_types.pizza_type_id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

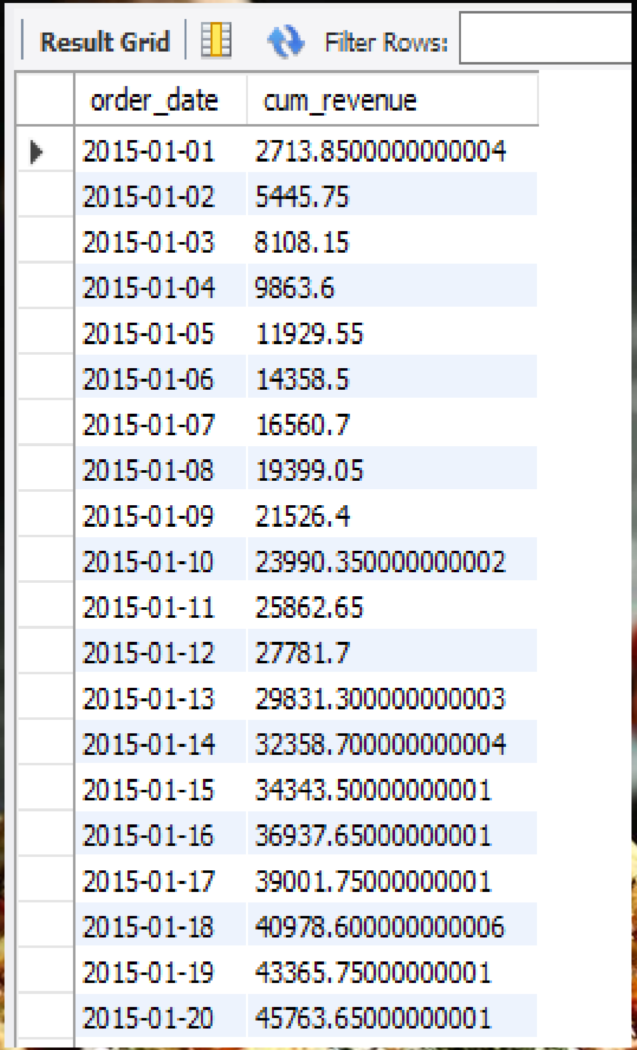


category	revenue
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

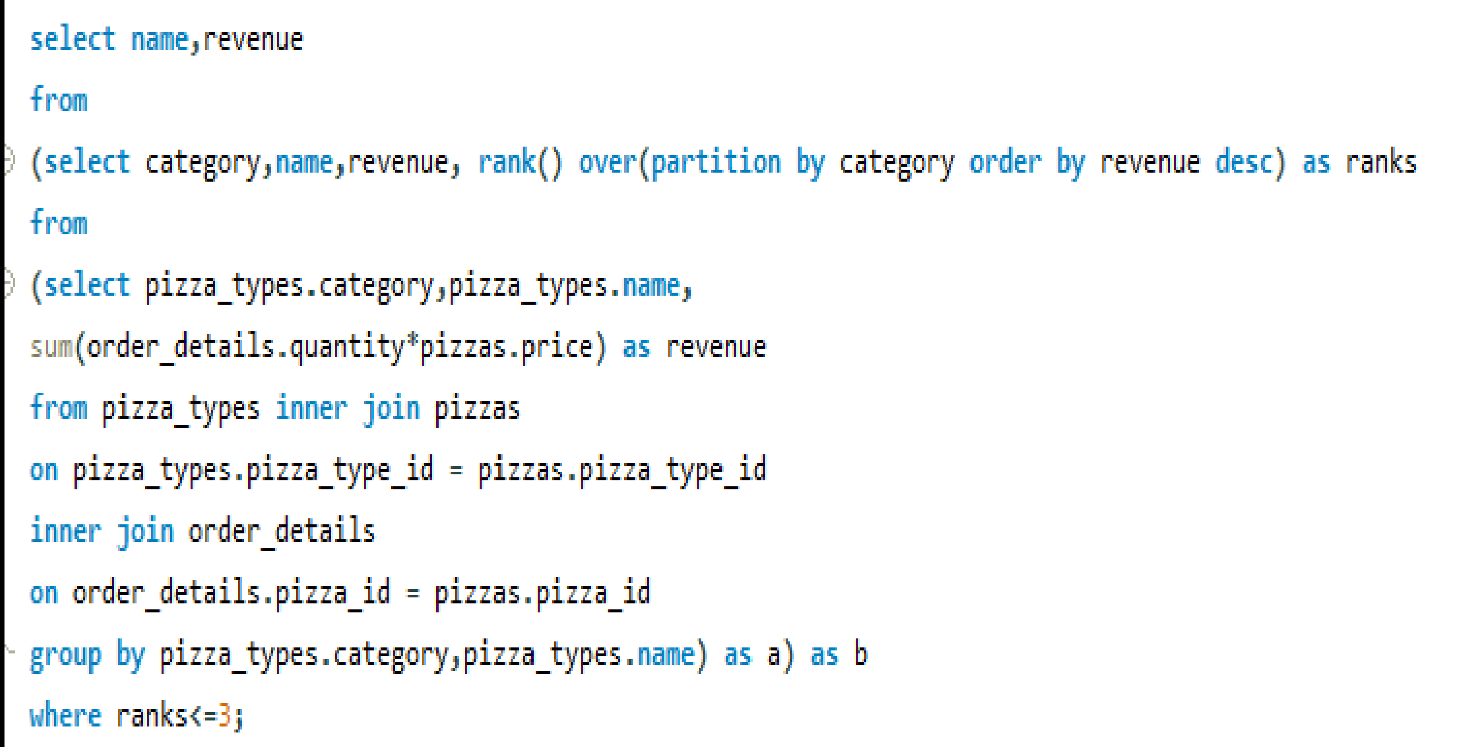


```
select order_date,sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity*pizzas.price) as revenue
from order_details inner join pizzas
on order_details.pizza_id = pizzas.pizza_id
inner join orders
on orders.order_id = order_details.order_id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

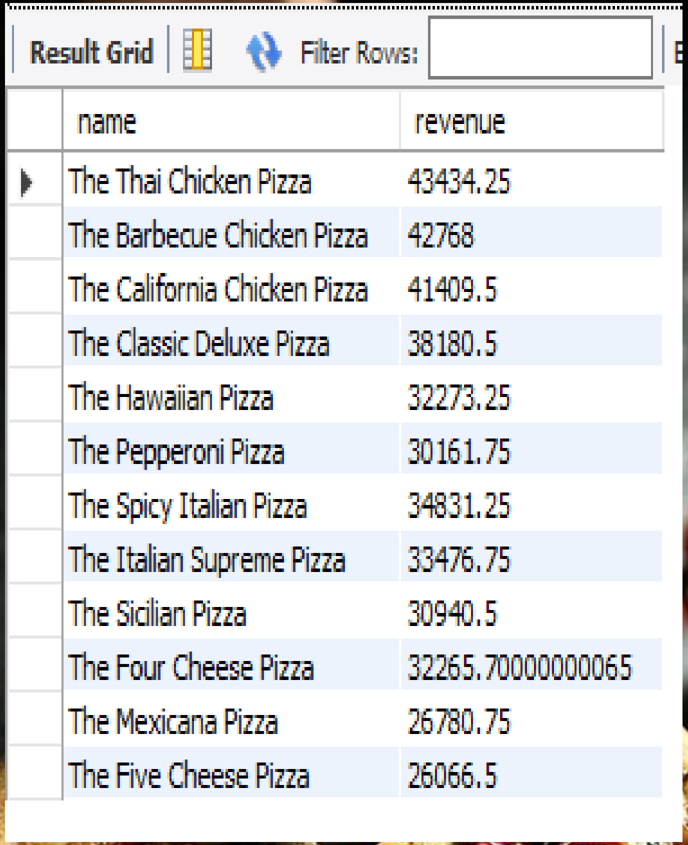


order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006
2015-01-19	43365.75000000001
2015-01-20	45763.65000000001

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 ranks
from
(select pizza_types.category,pizza_types.name,
sum(order_details.quantity*pizzas.price) as revenue
from pizza_types inner join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
inner join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category,pizza_types.name) as a) as b
where ranks<=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

SUMMARY

❑ INCREASED SALES AND PROFITABILITY: "BY IDENTIFYING THE MOST POPULAR PIZZA TYPES AND HIGHEST-PRICED PIZZAS, I CAN RECOMMEND TARGETED PROMOTIONS AND UPSELLING STRATEGIES TO MAXIMIZE REVENUE.“

❑ IMPROVED OPERATIONAL EFFICIENCY: "ANALYZING ORDER PATTERNS BY TIME OF DAY CAN HELP OPTIMIZE STAFFING LEVELS AND ENSURE EFFICIENT ORDER FULFILLMENT DURING PEAK HOURS."

❑ ENHANCED CUSTOMER EXPERIENCE: "UNDERSTANDING CATEGORY-WISE PIZZA DISTRIBUTION ALLOWS COGNIZANT TO TAILOR MENUS AND PROMOTIONS TO SPECIFIC CUSTOMER PREFERENCES, LEADING TO A MORE SATISFYING DINING EXPERIENCE."

THANK YOU!!!