PIZZA SALES PROJECT

INTRODUCTION

Hello everyone, Im Shahil Raj. In this project, Ive used SQL queries to analyze and address various questions related to pizza sales. My goal through this analysis is to uncover key insights and trends within the data, which can guide decision-making and help optimize sales strategies.

INTRODUCTION

Hello everyone, I'm Shahil Raj. In this project, I've used SQL queries to analyze and address various questions related to pizza sales. My goal through this analysis is to uncover key insights and trends within the data, which can guide decision-making and help optimize sales strategies.

Project Overview:

In this project, I conducted a comprehensive analysis of pizza sales data using SQL queries, structured into three levels of complexity:

Basic Analysis:

- Retrieved the total number of orders placed.
- Calculated the total revenue generated from pizza sales.
- Identified the highest-priced pizza.
- Determined the most commonly ordered pizza size

Intermediate Analysis

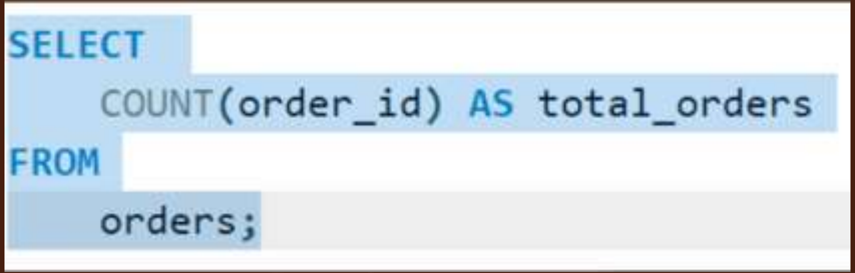
- Analyzed the distribution of orders by hour of the day.
- Joined relevant tables to find the category-wise distribution of pizzas.
- Grouped orders by date and calculated the average number of pizzas ordered per day.
- Identified the top 3 most ordered pizza types based on revenue.

Advanced Analysis

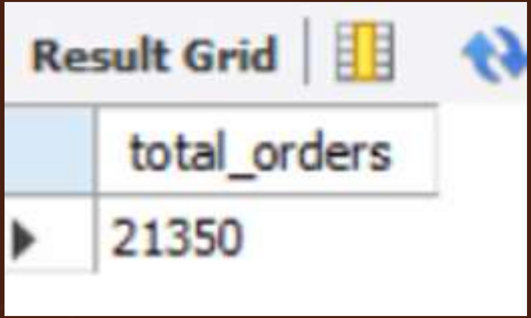
- Calculated the percentage contribution of each pizza type to total revenue.
- Analyzed the cumulative revenue generated over time.
- Determined the top 3 most ordered pizza types based on revenue for each pizza category.

PIZZA SALES

Retrieve the total number of orders placed.



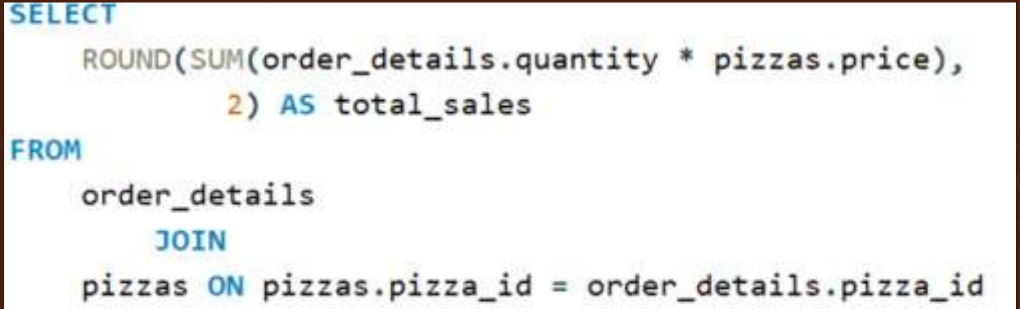
```
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```



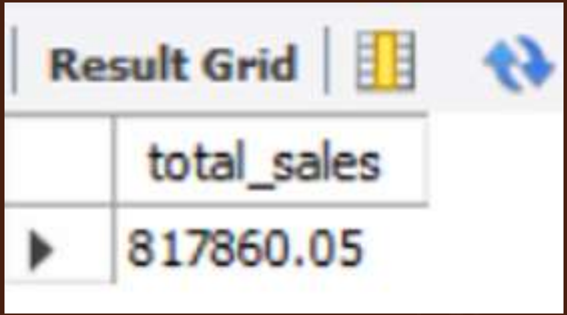
total_orders
21350

PIZZA SALES

Calculate the total revenue generated from pizza sales..



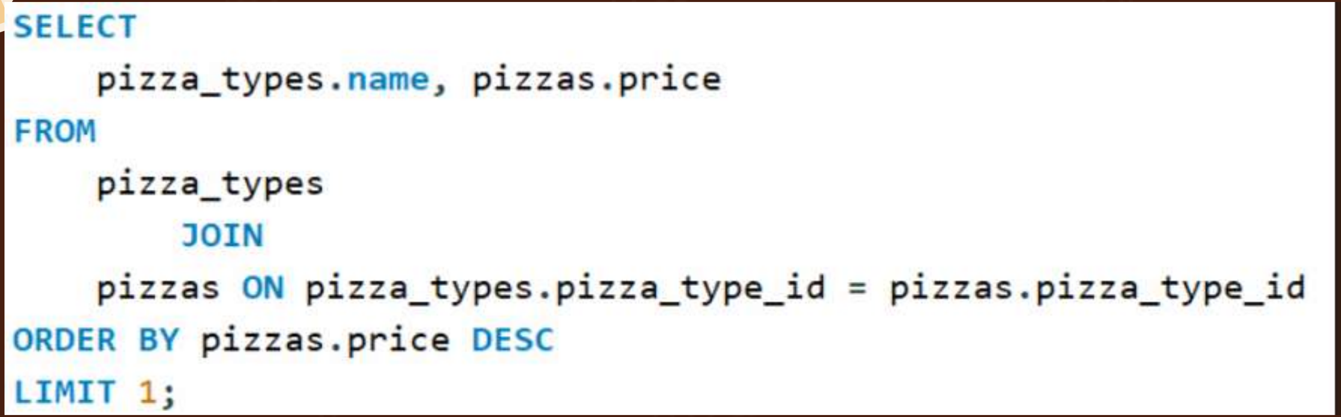
```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```



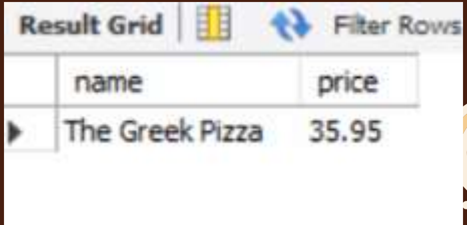
```
total_sales
817860.05
```

PIZZA SALES

Identify the highest priced-pizza.



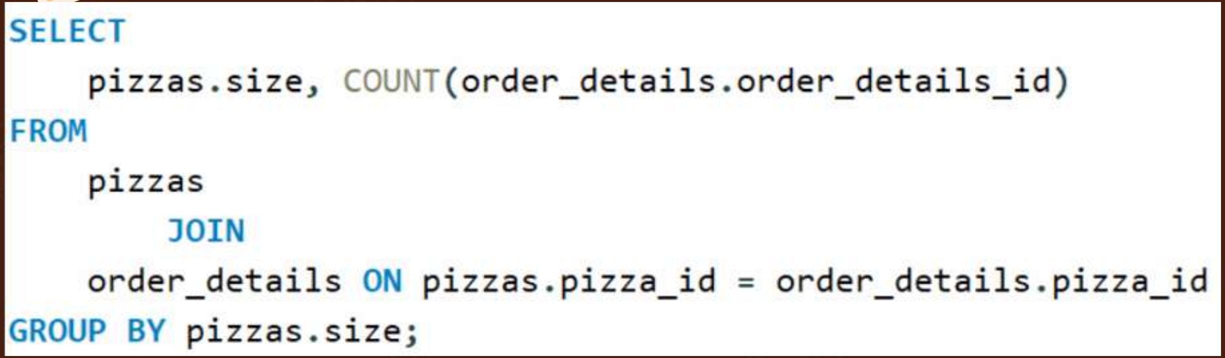
```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```



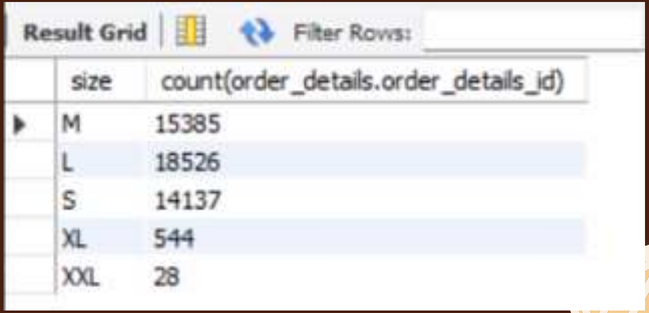
```
name             price
The Greek Pizza  35.95
```

PIZZA SALES

Identify the most common pizza size ordered..



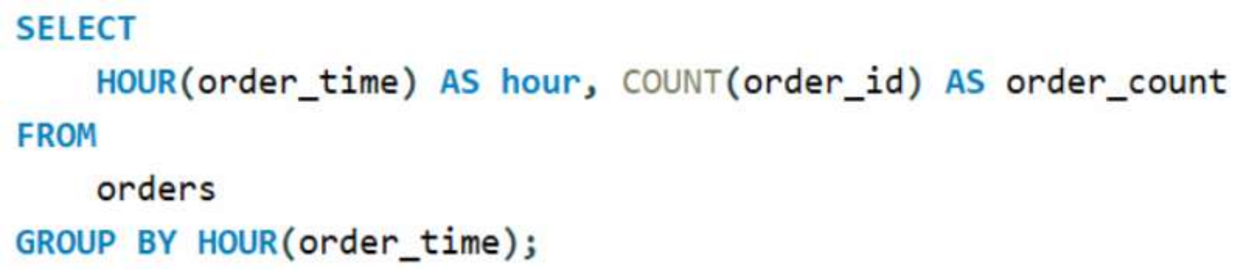
```
SELECT
    pizzas.size, COUNT(order_details.order_details_id)
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size;
```



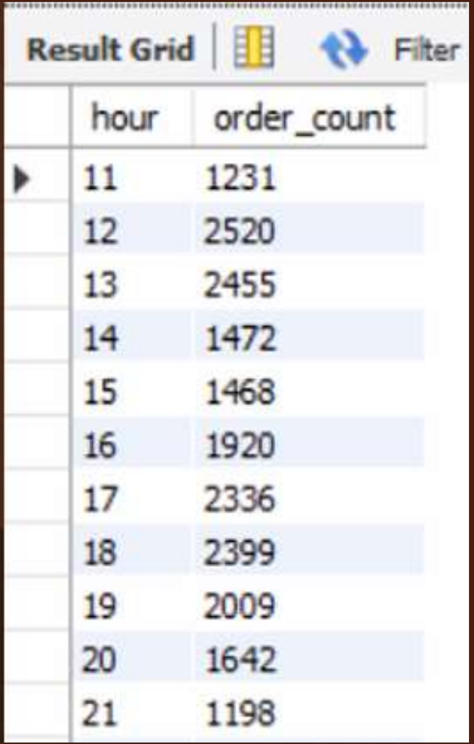
size	count(order_details.order_details_id)
M	15385
L	18526
S	14137
XL	544
XXL	28

PIZZA SALES

Determine the distribution of orders by hour of the day.



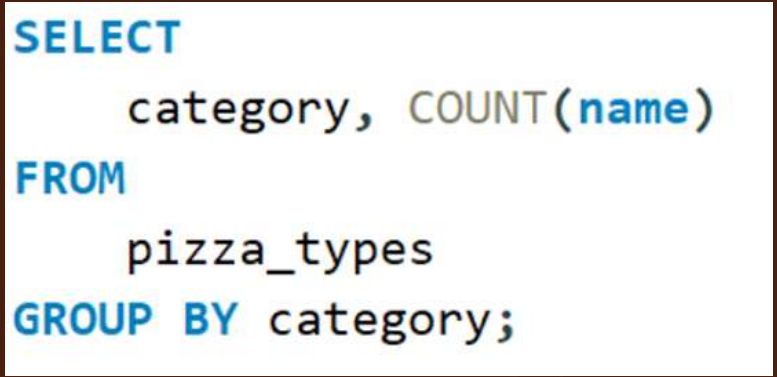
```sql
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```



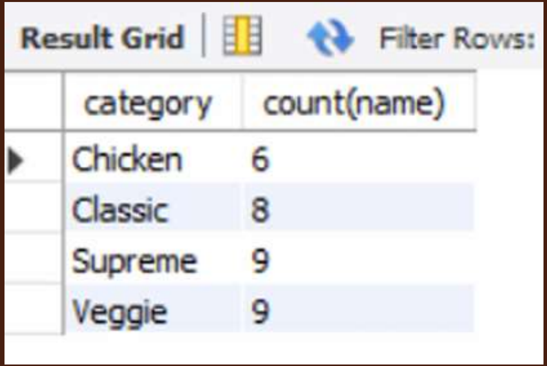
hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198

PIZZA SALES

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



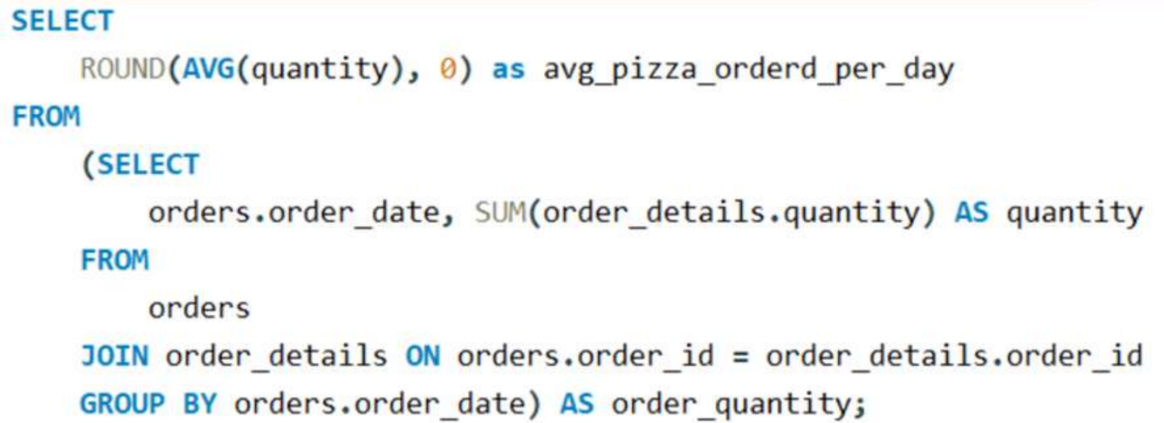
```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```



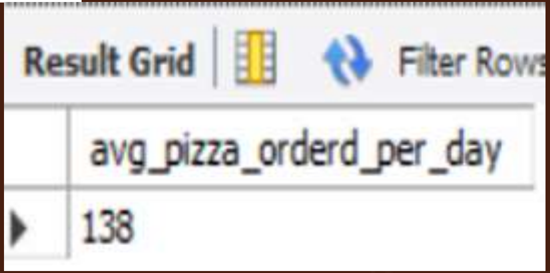
category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

PIZZA SALES

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



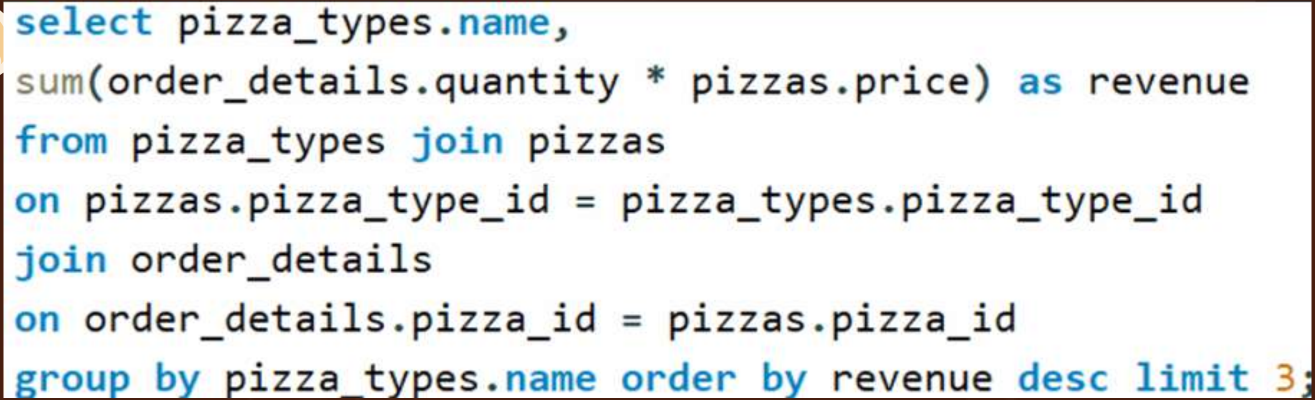
```sql
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_orderd_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```



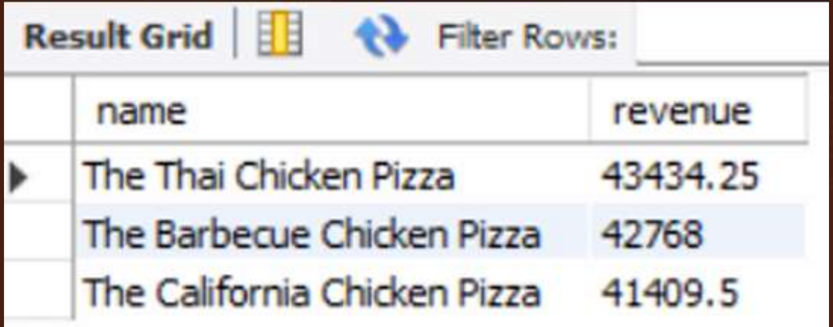
avg_pizza_orderd_per_day
138

PIZZA SALES

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



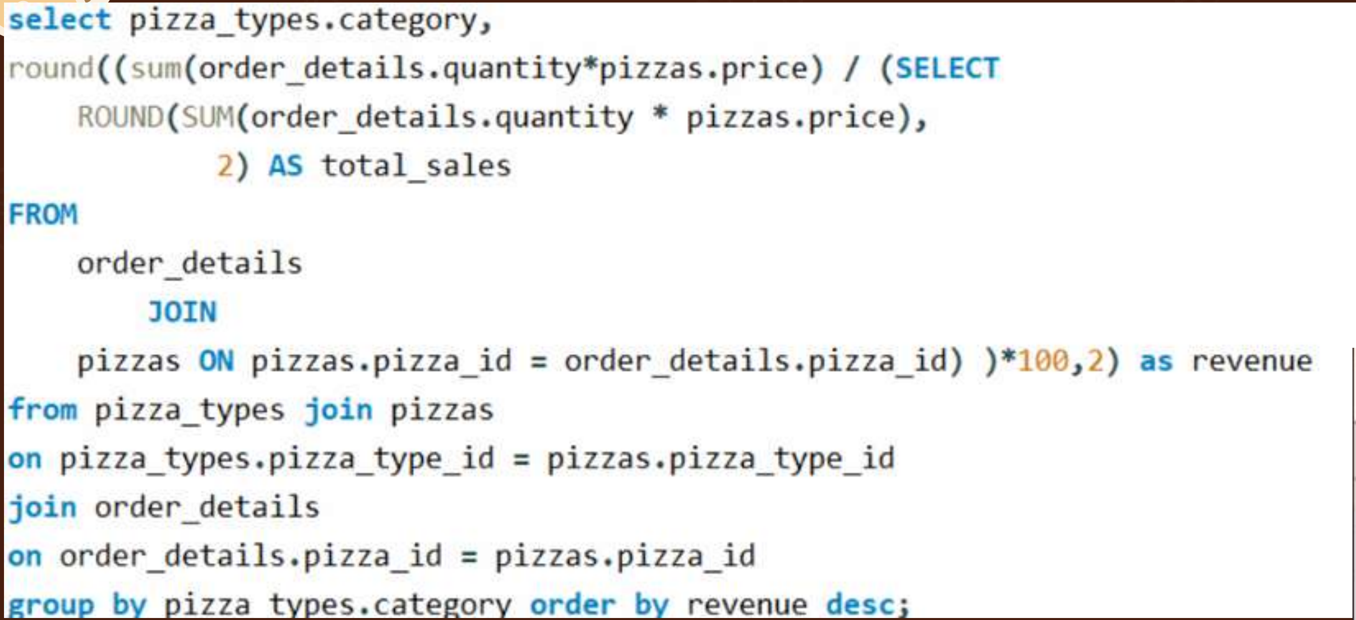
```
select pizza_types.name,
sum(order_details.quantity * pizzas.price) as revenue
from pizza_types join pizzas
on pizzas.pizza_type_id = pizza_types.pizza_type_id
join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.name order by revenue desc limit 3;
```



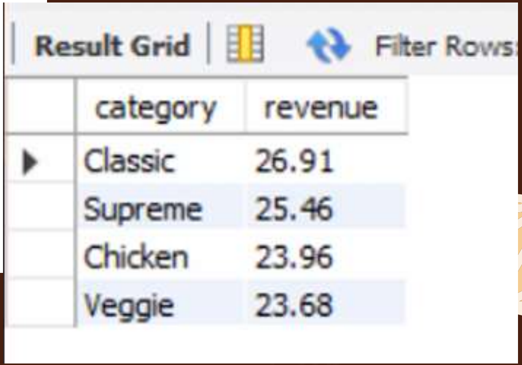
name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

PIZZA SALES

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



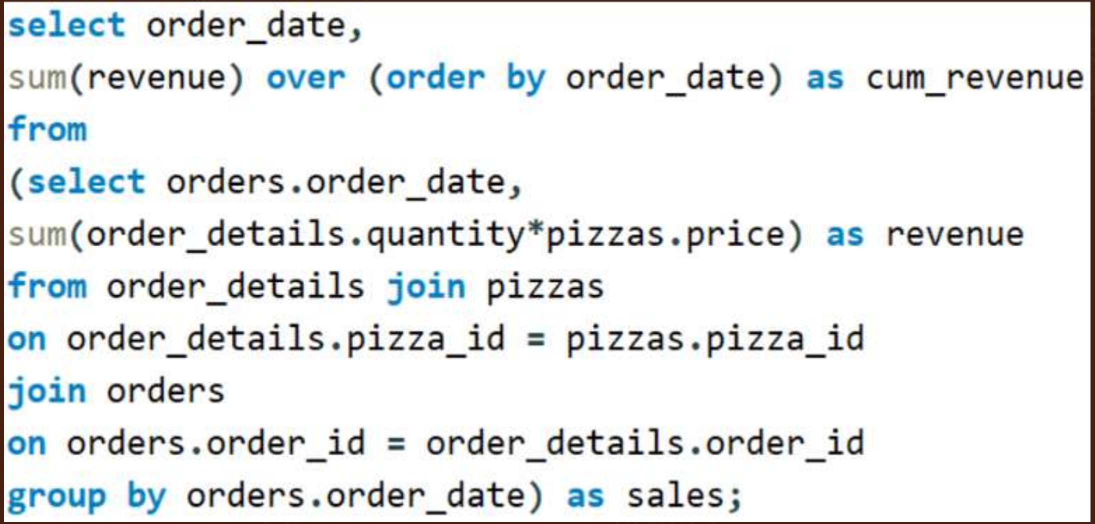
```
select pizza_types.category,
round((sum(order_details.quantity*pizzas.price) / (SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id) )*100,2) as revenue
from pizza_types 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 revenue desc;
```



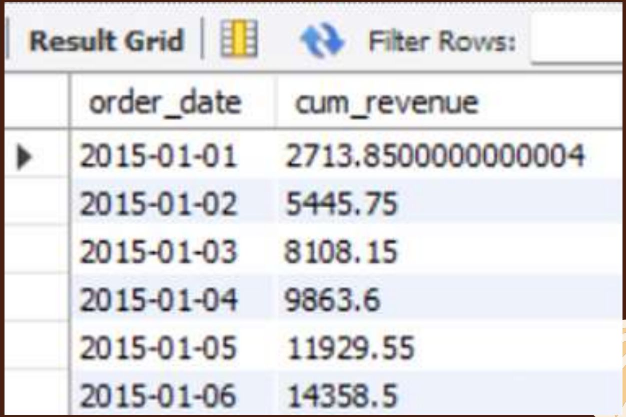
category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

PIZZA SALES

Analyze the cumulative revenue generated over time..



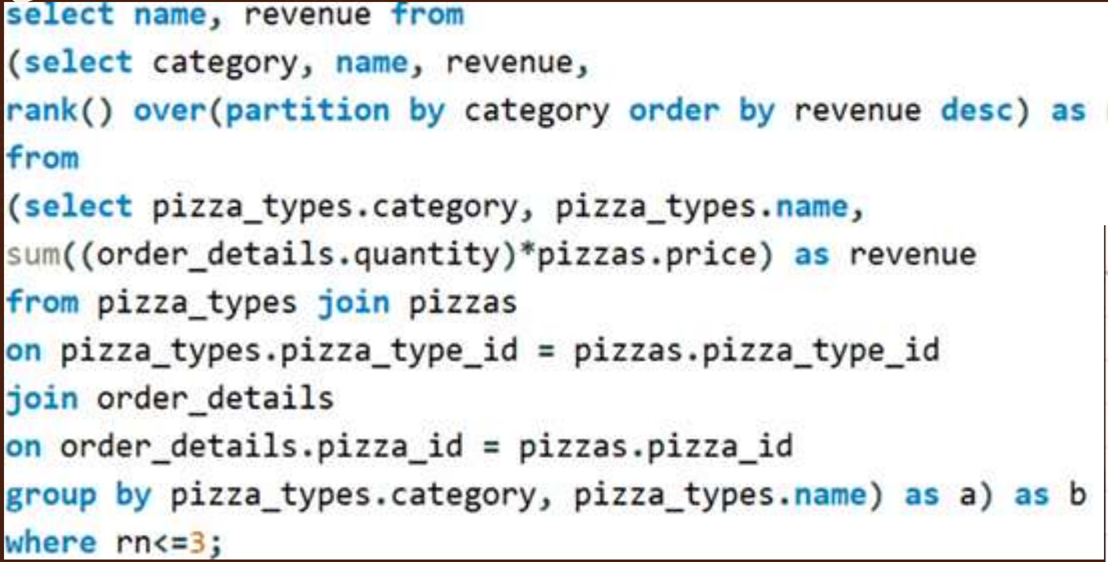
```
select order_date,
sum(revenue) over (order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity*pizzas.price) as revenue
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by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



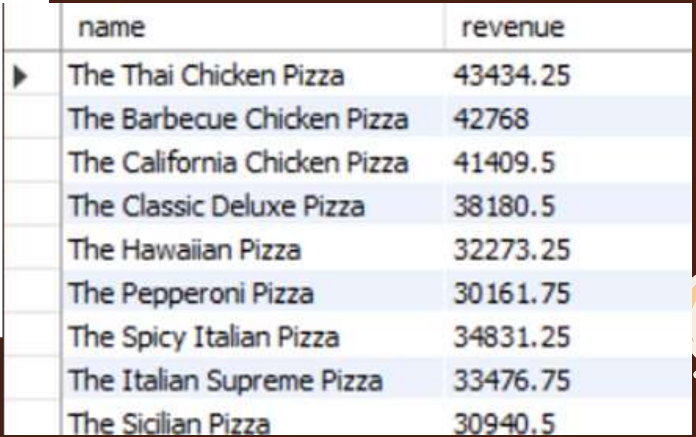
order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5

PIZZA SALES

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 rn
from
(select pizza_types.category, pizza_types.name,
sum((order_details.quantity)*pizzas.price) as revenue
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, pizza_types.name) as a) as b
where rn<=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

THANK YOU

SHAHIL RAJ

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**