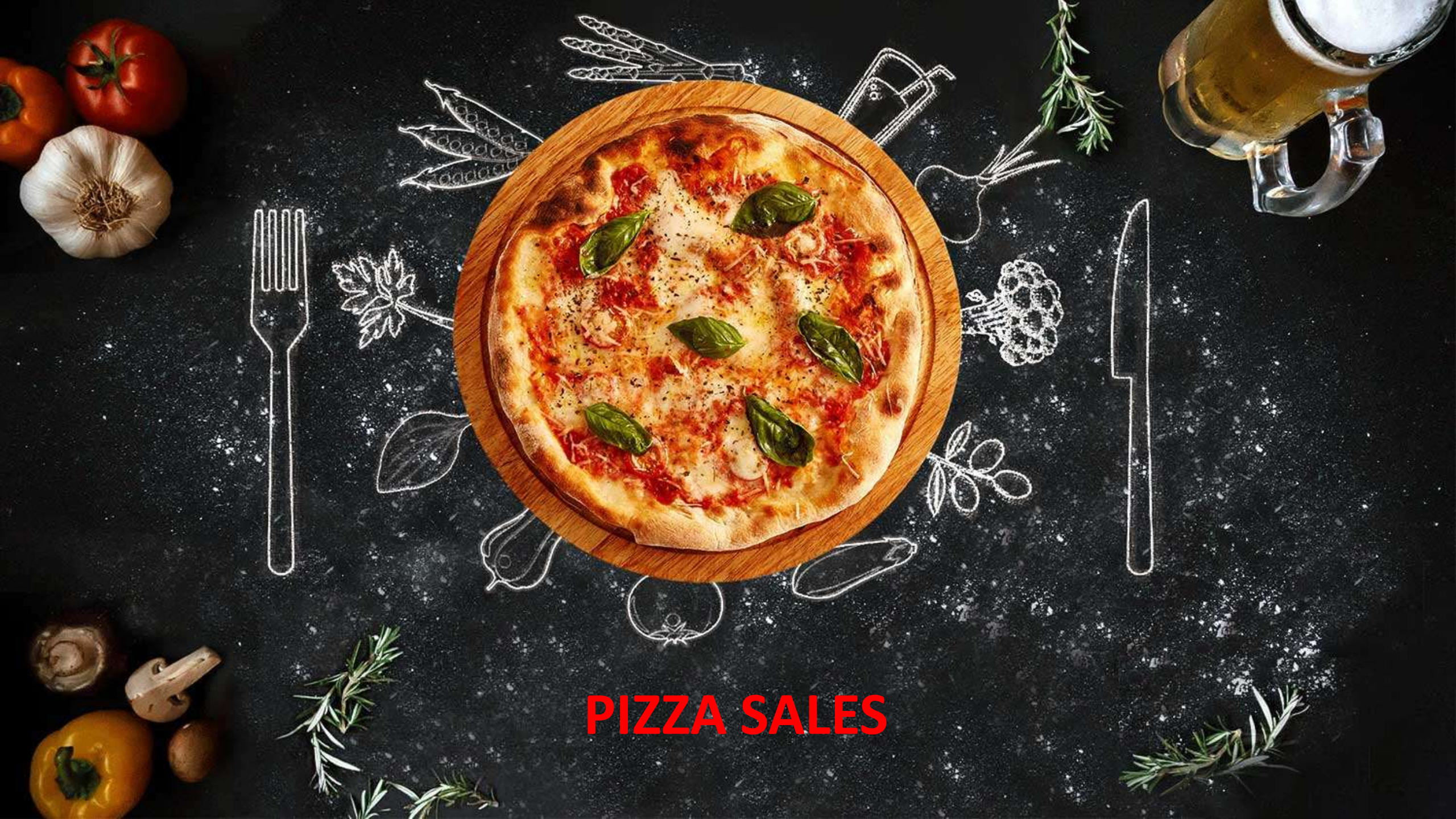
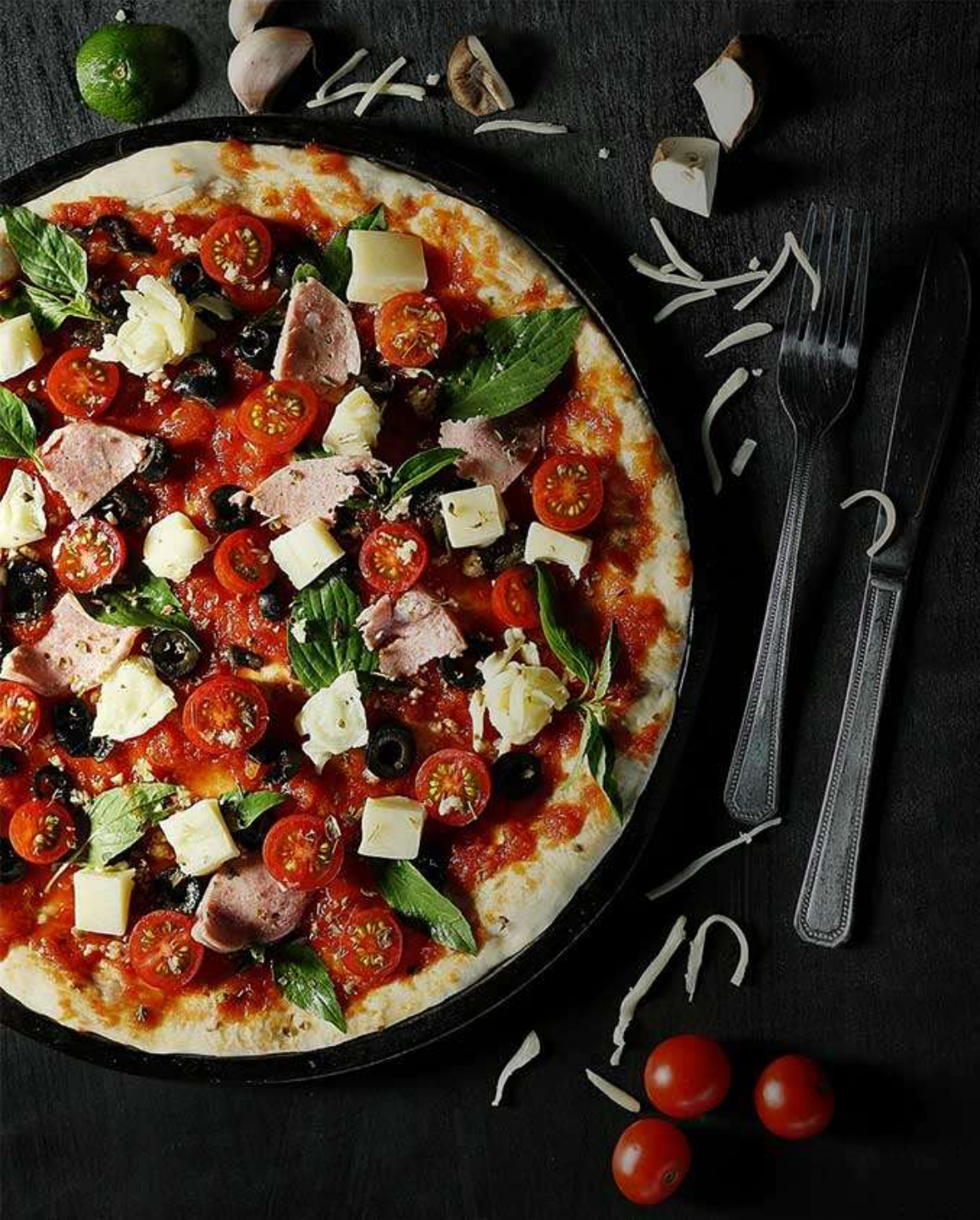
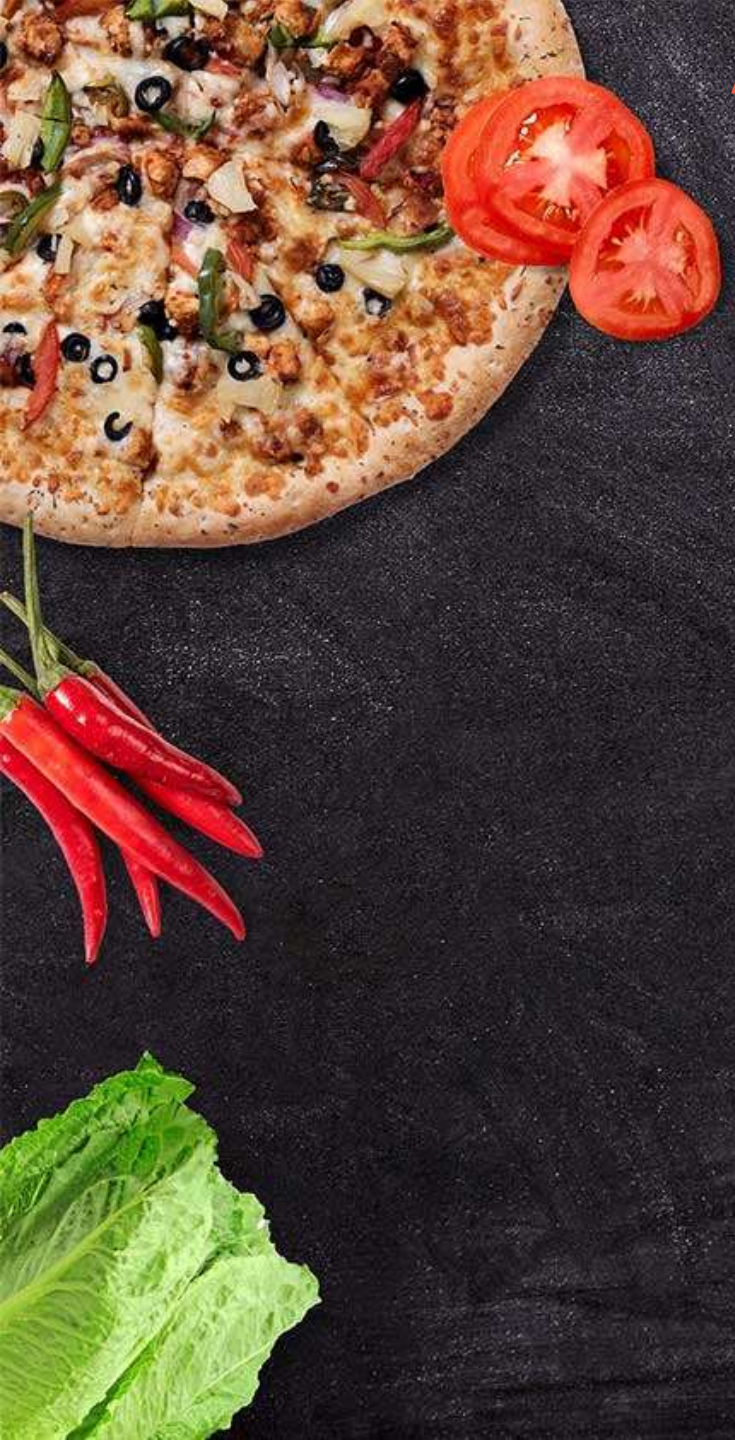

PIZZA SALES



Introduction :

Hello, My name is Saumyasuteshnu Behera. In this project I have utilized SQL queries to solve the questions that are related to Pizza sales.



All questions I have solved in this project through SQL queries

1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.
3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.
5. List the top 5 most ordered pizza types along with their quantities.
6. Join the necessary tables to find the total quantity of each pizza category ordered.
7. Determine the distribution of orders by hour of the day.
8. Join relevant tables to find the category-wise distribution of pizzas.
9. Group the orders by date and calculate the average number of pizzas ordered per day.
10. Determine the top 3 most ordered pizza types based on revenue.
11. Calculate the percentage contribution of each pizza type to total revenue.
12. Analyze the cumulative revenue generated over time.
13. Determine the top 3 most ordered pizza types based on revenue for each pizza category.

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 ?

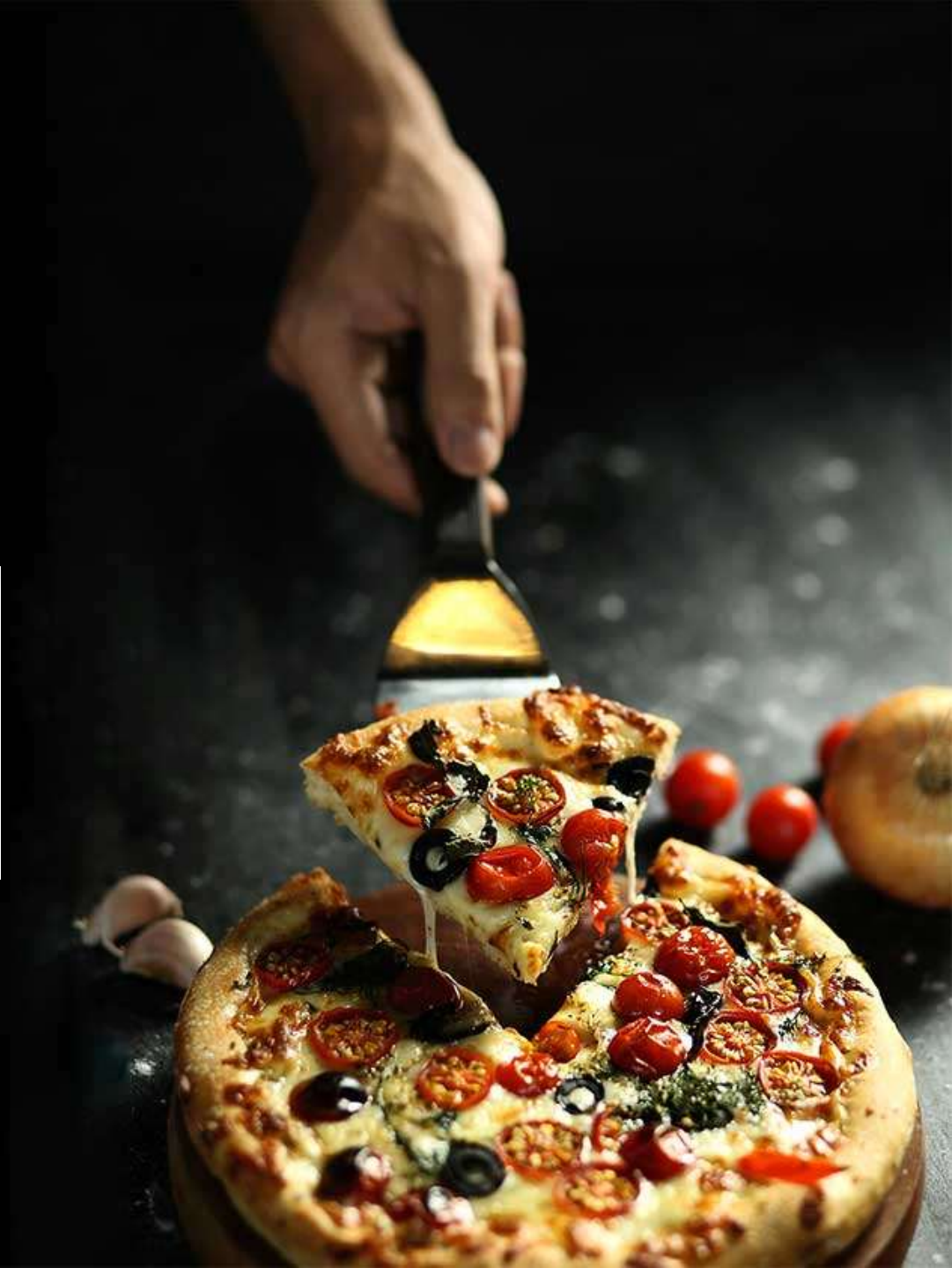
```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

	total_revenue
▶	817860.05

Identify the highest-priced pizza ?

```sql
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 ?

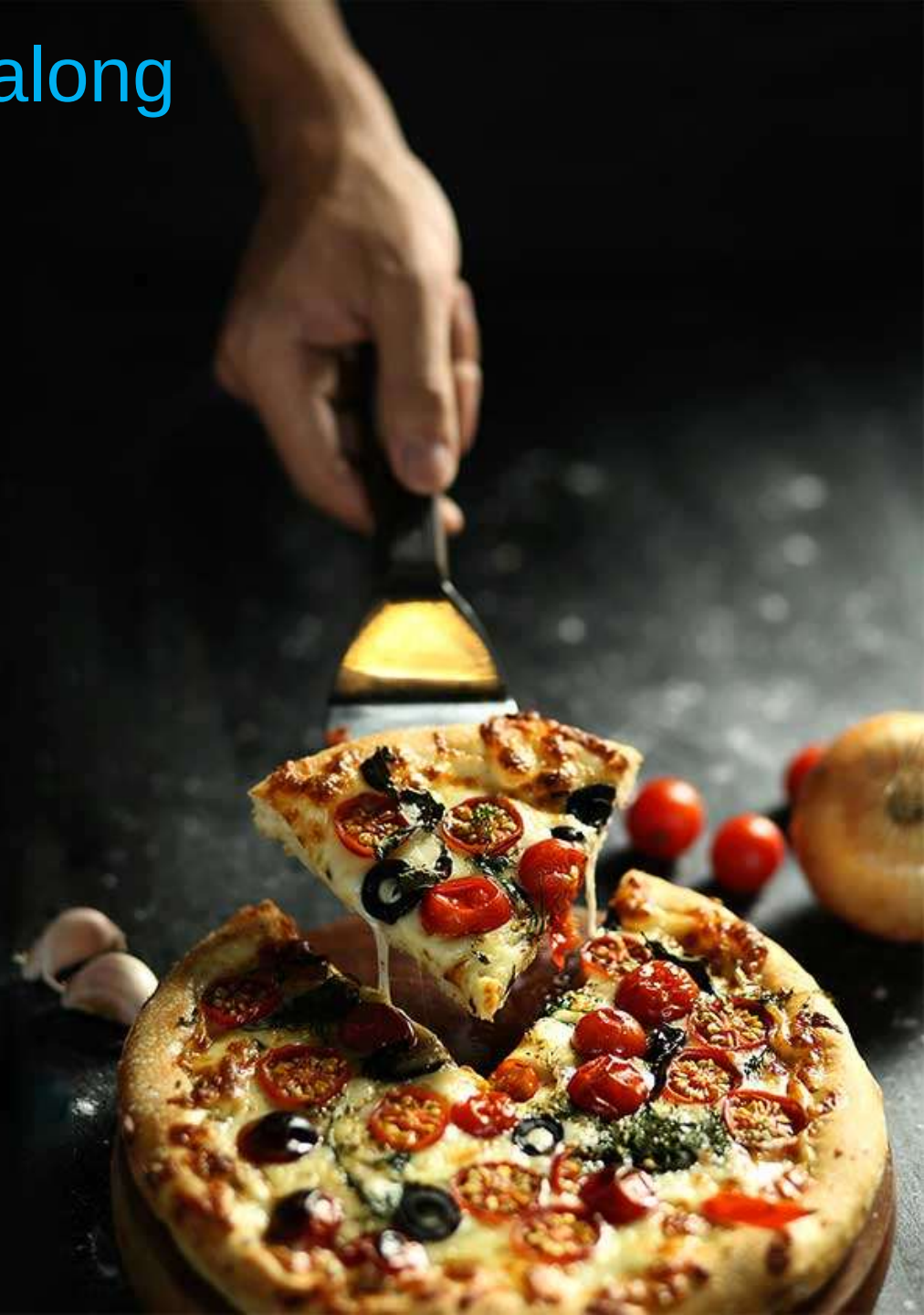
```sql
SELECT
    p.size, COUNT(od.order_details_id) AS total_order
FROM
    pizzas p
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY p.size
ORDER BY COUNT(od.order_details_id) DESC;
```

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

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

```sql
SELECT
    pt.name, SUM(od.quantity) AS 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
GROUP BY pt.name
ORDER BY COUNT(od.quantity) DESC
LIMIT 5;
```

name	quantity
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 ?

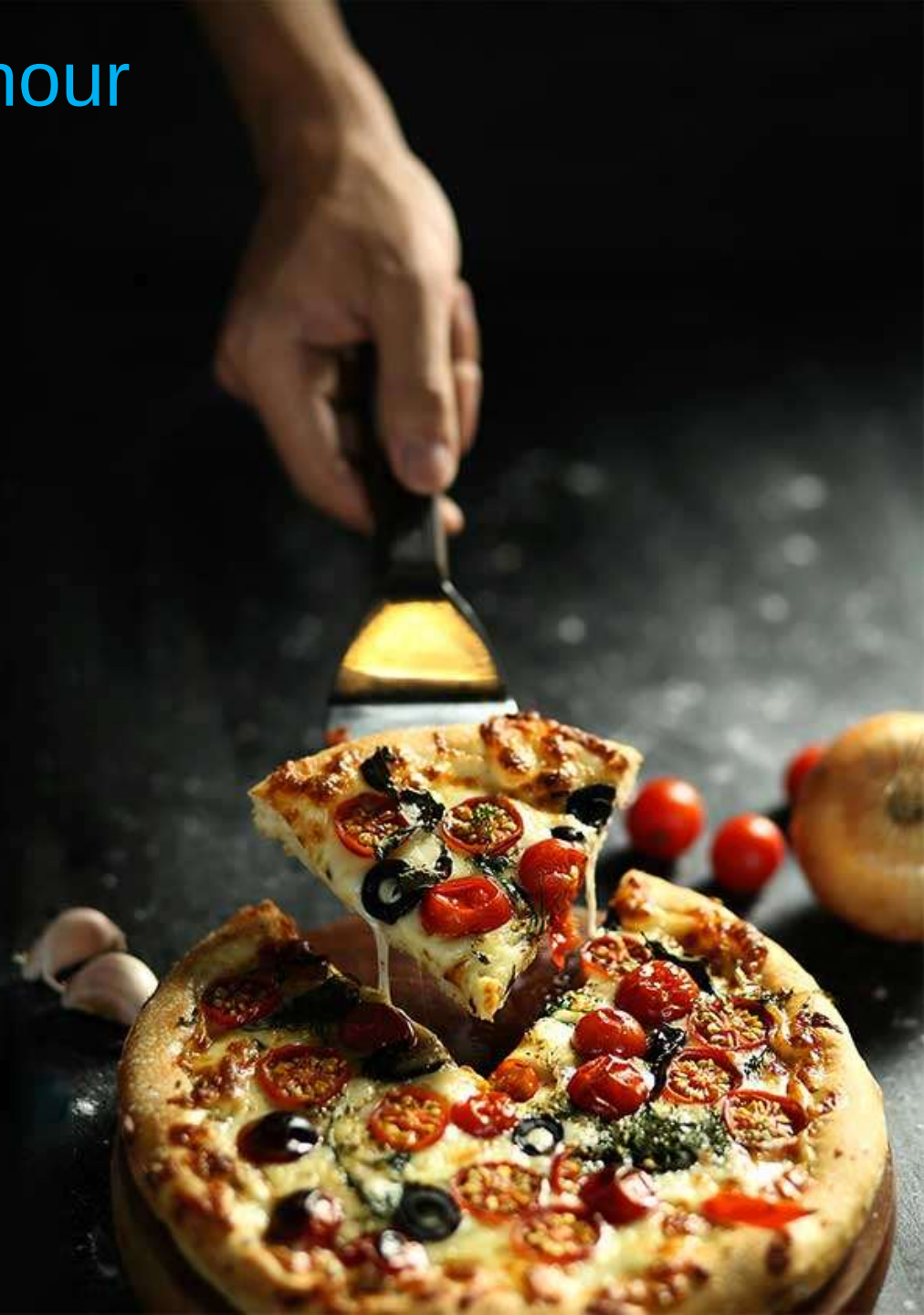
```sql
SELECT
    pt.category, SUM(od.quantity) AS total_quantity
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY pt.category
ORDER BY total_quantity DESC;
```

	category	total_quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

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)
ORDER BY HOUR(order_time);
```

hour	order_count
9	1
10	8
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



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

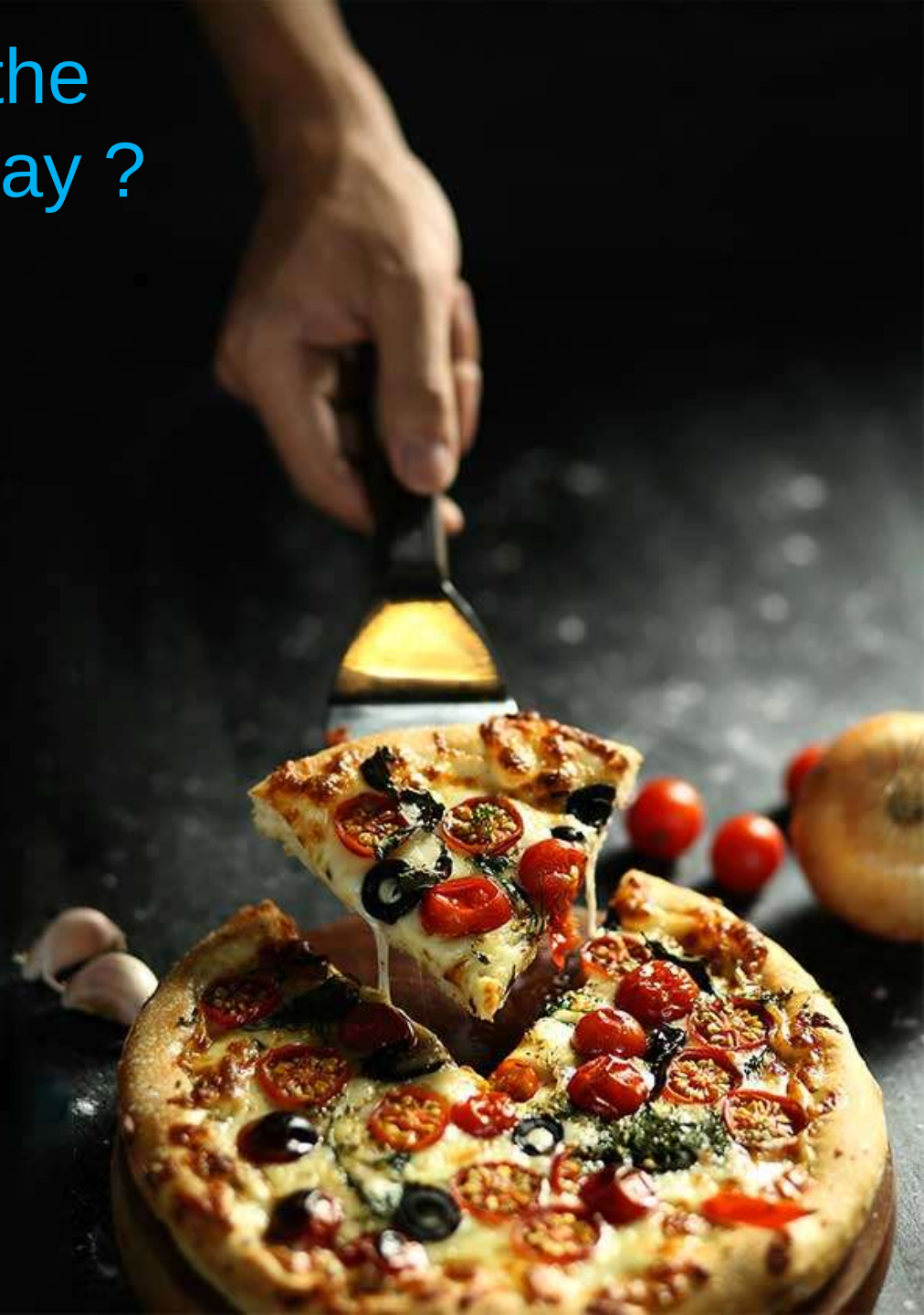
```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

	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
SELECT
    ROUND(AVG(total_quantity), 0) AS avg_pizza_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 quantity_per_day;
```

	avg_pizza_ordered_per_day
▶	138



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

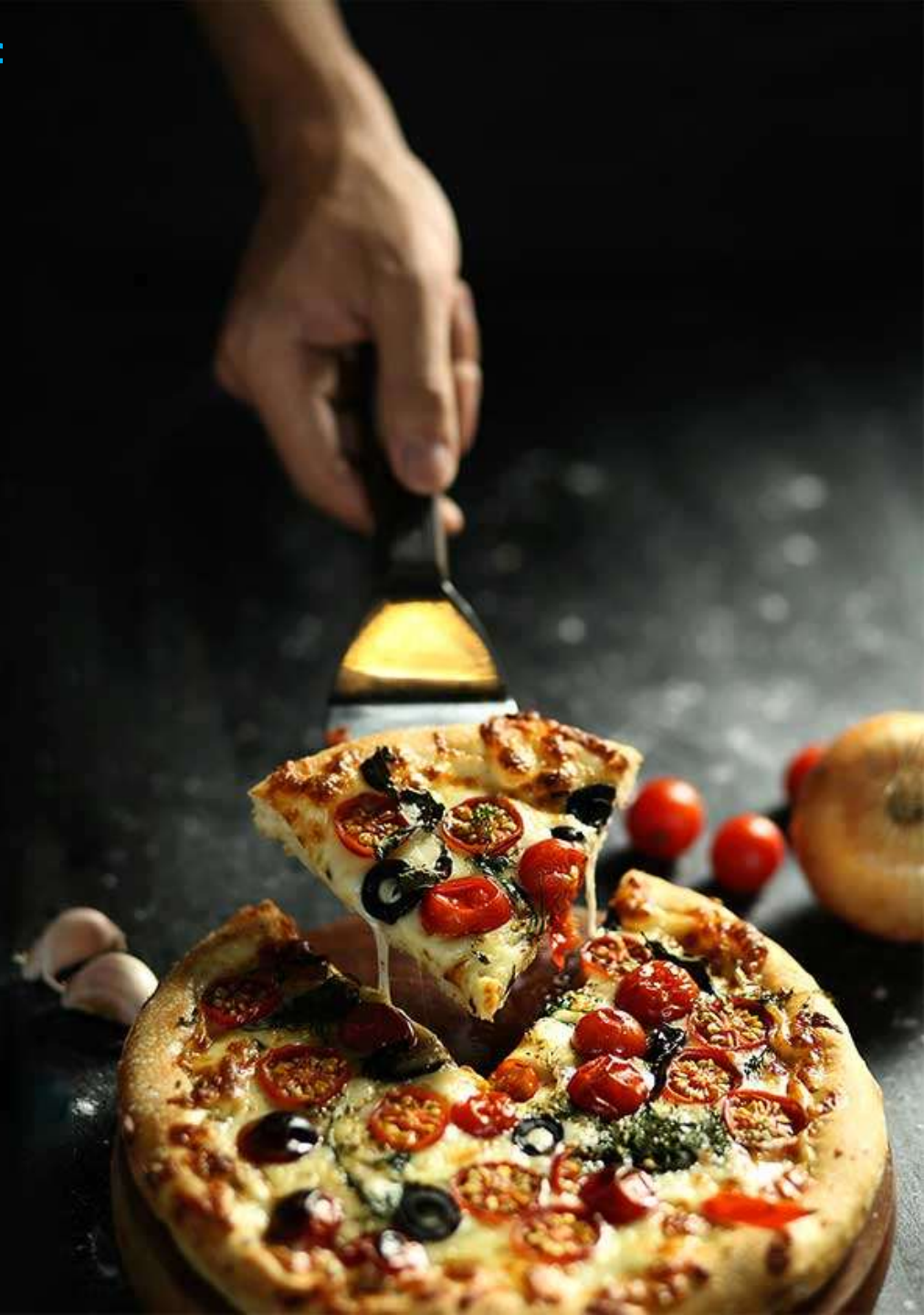
```sql
SELECT
    pt.name, SUM(p.price * od.quantity) AS revenue
FROM
    pizzas p
        JOIN
    pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY pt.name
ORDER BY revenue DESC
LIMIT 3;
```

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 ?

```sql
SELECT
    pt.category,
    ROUND(SUM(p.price * od.quantity) / (SELECT
                    ROUND(SUM(p.price * od.quantity), 2) AS 'Total revenue'
                FROM
                    pizzas p
                        JOIN
                    order_details od ON p.pizza_id = od.pizza_id) * 100,
            2) AS revenue
FROM
    pizzas p
        JOIN
    pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY pt.category
ORDER BY revenue DESC
```

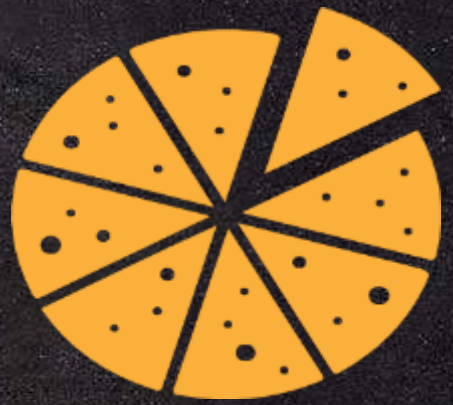
category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68



Analyze the cumulative revenue generated over time ?

```sql
select order_date,
sum(revenue) over (order by order_date) as cumulative_revenue
from
(select o.order_date,
round(sum(p.price*od.quantity),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 revenue_by_date
```

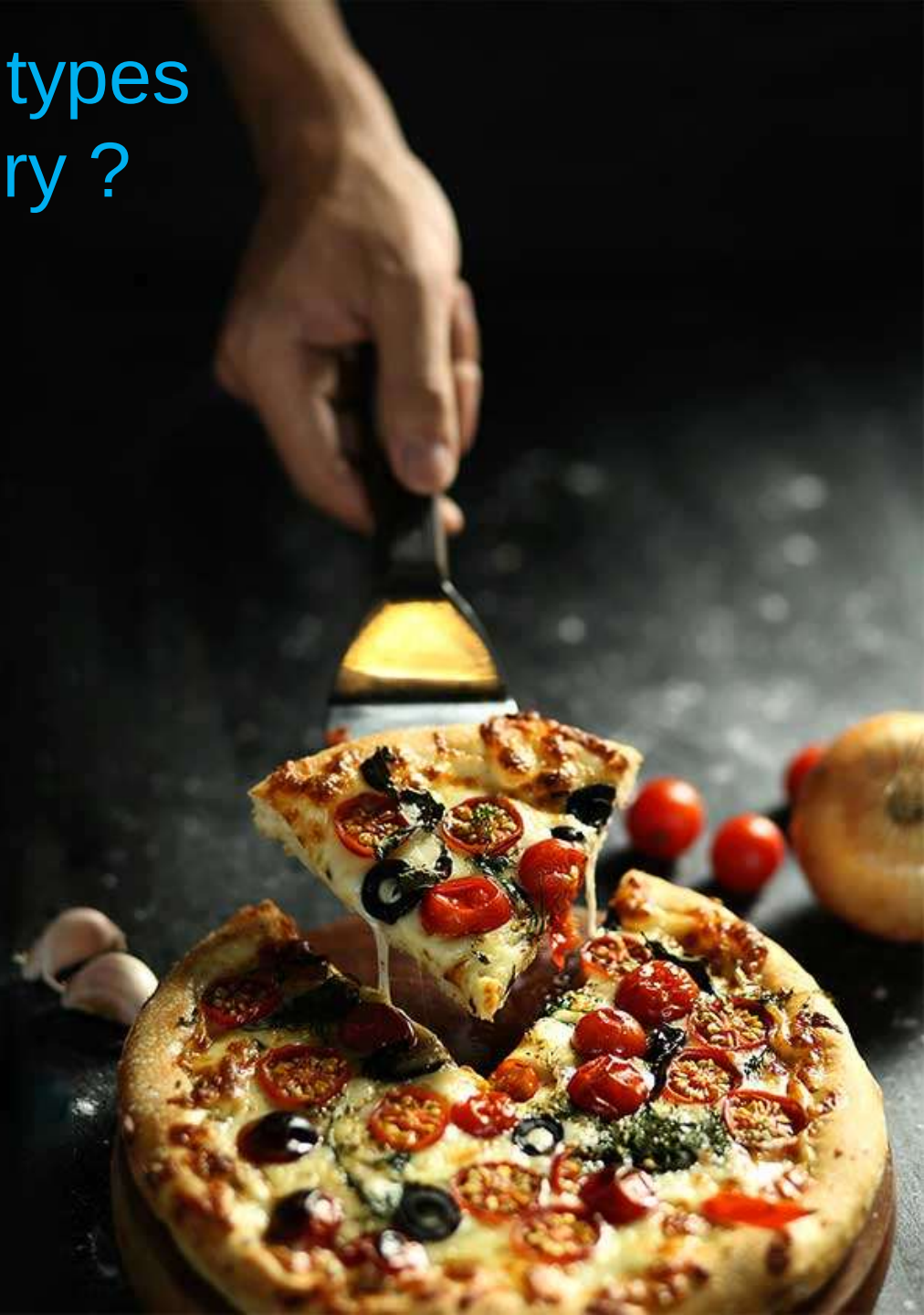
	order_date	cumulative_revenue
▶	2015-01-01	2713.85
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55
	2015-01-06	14358.5
	2015-01-07	16560.7
	2015-01-08	19399.05

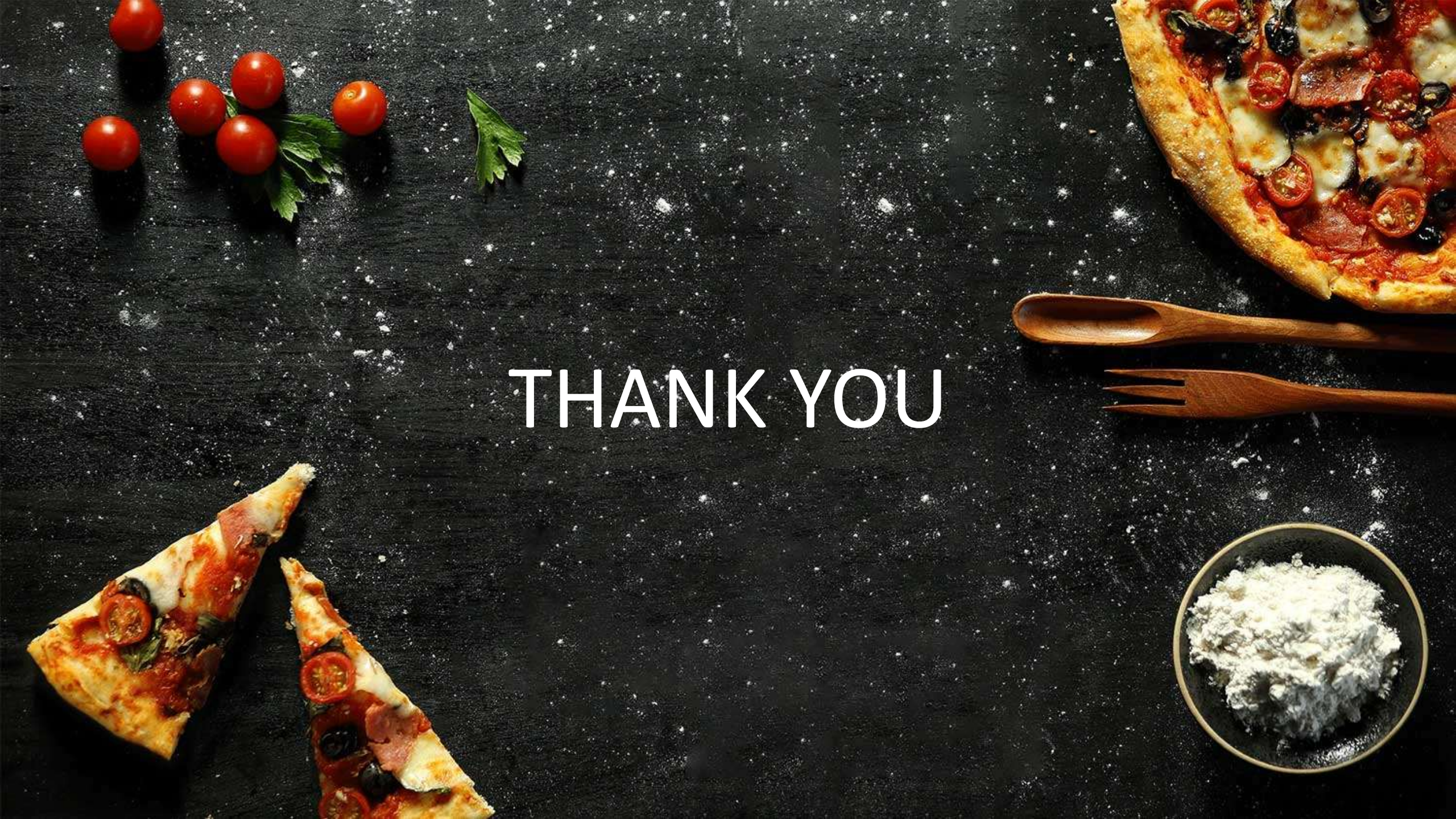


Determine the top 3 most ordered pizza types based on revenue for each pizza category ?

```sql
select name,revenue,rn 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	rn
▶	The Thai Chicken Pizza	43434.25	1
	The Barbecue Chicken Pizza	42768	2
	The California Chicken Pizza	41409.5	3
	The Classic Deluxe Pizza	38180.5	1
	The Hawaiian Pizza	32273.25	2
	The Pepperoni Pizza	30161.75	3
	The Spicy Italian Pizza	34831.25	1
	The Italian Supreme Pizza	33476.75	2
	The Sicilian Pizza	30940.5	3
	The Four Cheese Pizza	32265.70000000065	1
	The Mexicana Pizza	26780.75	2
	The Five Cheese Pizza	26066.5	3




THANK YOU