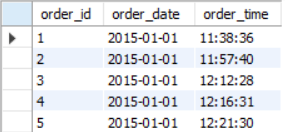
**PizzaBox Sales Analysis**

1. **Viewing the Data**

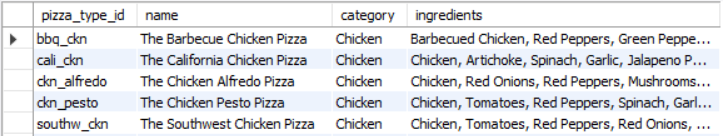
select \* from order\_details;

****

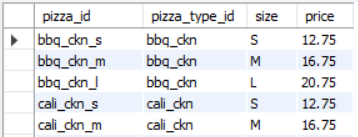
select \* from orders;

****

select \* from pizza\_types;

****

select \* from pizzas;

****

1. **Retrieve the total number of orders placed**

SELECT

COUNT(order\_id) AS total\_orders

FROM

orders;



1. **Calculate the total revenue generated from pizza sales**

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;



1. **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;



1. **Identify the most common pizza size ordered**

SELECT

pizzas.size,

COUNT(order\_details.order\_details\_id) AS order\_count

FROM

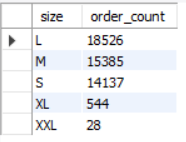
pizzas

JOIN

order\_details ON pizzas.pizza\_id = order\_details.pizza\_id

GROUP BY pizzas.size

ORDER BY order\_count DESC;



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

SELECT

pizza\_types.name, SUM(order\_details.quantity) AS quantity

FROM

pizza\_types

JOIN

pizzas ON pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id

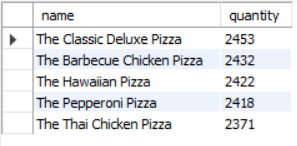
JOIN

order\_details ON order\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.name

ORDER BY quantity DESC

LIMIT 5;



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

SELECT

pizza\_types.category,

SUM(order\_details.quantity) AS quantity

FROM

pizza\_types

JOIN

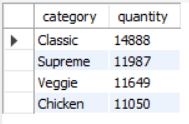
pizzas ON pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id

JOIN

order\_details ON order\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.category

ORDER BY quantity DESC;



1. **Determine the distribution of orders by hour of the day**

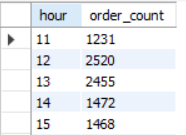
SELECT

HOUR(order\_time) AS hour, COUNT(order\_id) AS order\_count

FROM

orders

GROUP BY HOUR(order\_time);



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

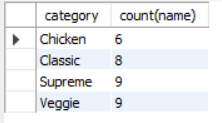
SELECT

category, COUNT(name)

FROM

pizza\_types

GROUP BY category;



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

SELECT

ROUND(AVG(quantity), 0)

FROM

(SELECT

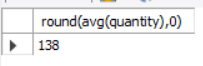
orders.order\_date, SUM(order\_details.quantity) AS quantity

FROM

orders

JOIN order\_details ON orders.order\_id = order\_details.order\_id

GROUP BY orders.order\_date) AS order\_quantity;



1. **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

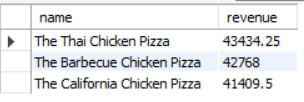
JOIN

order\_details ON order\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.name

ORDER BY revenue DESC

LIMIT 3;



**12)** **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

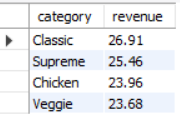
pizzas ON pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id

JOIN

order\_details ON order\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.category

ORDER BY revenue DESC;



**13) Analyse 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

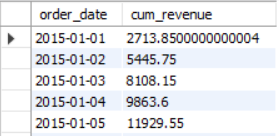
from order\_details join pizzas

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;



**14) 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;

