Pizza Orders Data Analysis using SQL

**Q1) Retrieve the total number of orders placed**

SQL Query:

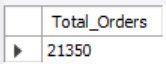
SELECT

COUNT(order\_id) AS Total\_Orders

FROM

orders;

Result:



**Q2) Calculate the total revenue generated from pizza sales**

SQL Query:

SELECT

ROUND(SUM(orders\_details.quantity \* pizzas.price),

2) AS total\_sales

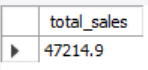
FROM

orders\_details

JOIN

pizzas ON pizzas.pizza\_id = orders\_details.pizza\_id;

Result:



**Q3) Identify the highest priced prizza**

SQL Query:

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;

Result:

**Q4) Identify the most common pizza size ordered**

SQL Query:

SELECT

pizzas.size,

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

FROM

pizzas

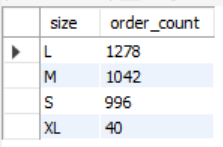
JOIN

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

GROUP BY pizzas.size

ORDER BY order\_count DESC;

Result:



**Q5) List the top 5 most ordered pizza types along with their quantities**

SQL Query:

SELECT

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

FROM

pizza\_types

JOIN

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

JOIN

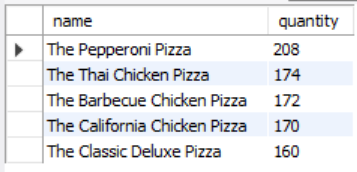
orders\_details ON orders\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.name

ORDER BY quantity DESC

LIMIT 5;

Result:



**Q6) Join the necessary tables to find the total quantity of each pizza category ordered**

SQL Query:

SELECT

pizza\_types.category,

SUM(orders\_details.quantity) AS quantity

FROM

pizza\_types

JOIN

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

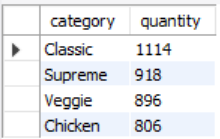
JOIN

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

GROUP BY pizza\_types.category

ORDER BY quantity DESC;

Result:



**Q7) Determine the distribution of orders by hour of the day**

SELECT

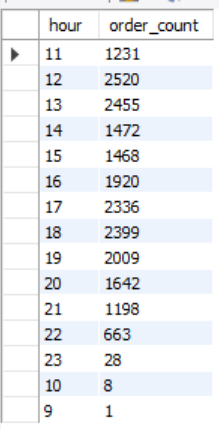
HOUR(order\_time) AS hour, COUNT(order\_id) AS order\_count

FROM

orders

GROUP BY HOUR(order\_time);

Result:



**Q8) Join relevant tables to find the category-wise distribution of pizzas**

SQL Query:

SELECT

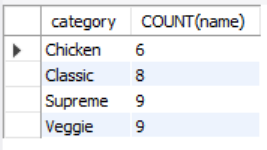
category, COUNT(name)

FROM

pizza\_types

GROUP BY category;

Result:



**Q9) Group the orders by date and calculate the average number of pizzas ordered per day**

SQL Query:

SELECT

ROUND(AVG(quantity), 0)

FROM

(SELECT

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

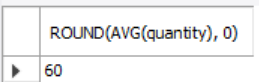
FROM

orders

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

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

Result:



**Q10) Determine the top three most ordered pizza types based on revenue**

SQL Query:

SELECT

pizza\_types.name,

SUM(orders\_details.quantity \* pizzas.price) AS revenue

FROM

pizza\_types

JOIN

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

JOIN

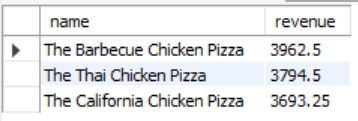
orders\_details ON orders\_details.pizza\_id = pizzas.pizza\_id

GROUP BY pizza\_types.name

ORDER BY revenue DESC

LIMIT 3;

Result:

****

**Q11) Calculate the percentage contribution of each pizza type to total revenue**

SQL Query:

SELECT

pizza\_types.category,

ROUND(SUM(orders\_details.quantity \* pizzas.price) / (SELECT

ROUND(SUM(orders\_details.quantity \* pizzas.price),

2) AS total\_sales

FROM

orders\_details

JOIN

pizzas ON pizzas.pizza\_id = orders\_details.pizza\_id) \* 100,

2) AS revenue

FROM

pizza\_types

JOIN

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

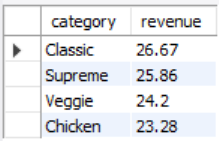
JOIN

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

GROUP BY pizza\_types.category

ORDER BY revenue DESC;

Result:

****

**Q12) Analyze the cumulative revenue generated over time**

SQL Query:

select order\_date,

sum(revenue) over(order by order\_date) as cum\_revenue

from

(select orders.order\_date,

sum(orders\_details.quantity\*pizzas.price) as revenue

from orders\_details join pizzas

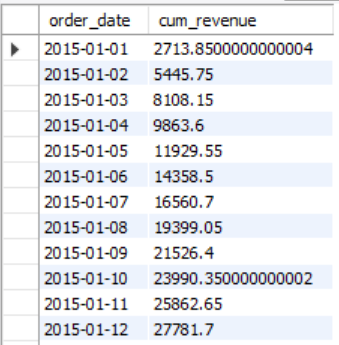
on orders\_details.pizza\_id=pizzas.pizza\_id

join orders

On orders.order\_id=orders\_details.order\_id

group by orders.order\_date) as sales;

Result:

****

**Q13) Determine the top 3 most ordered pizza types based on revenue for each pizza category**

SQL Query:

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((orders\_details.quantity)\*pizzas.price) as revenue

from pizza\_types join pizzas

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

join orders\_details

on orders\_details.pizza\_id=pizzas.pizza\_id

group by pizza\_types.category,pizza\_types.name) as a) as b

where rn<=3;

Result:

