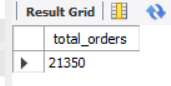
**Pizza sales SQL project**

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

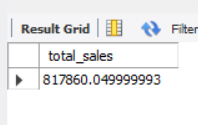
SUM(orders\_details.quantity \* pizzas.price) AS total\_sales

FROM

orders\_details

JOIN

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



1. Identify the highest-priced pizza.

* SELECT

pizza\_types.name, pizzas.price

FROM

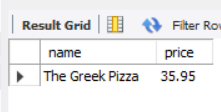
pizza\_types

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

FROM

pizzas

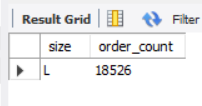
JOIN

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

GROUP BY pizzas.size

ORDER BY order\_count DESC

LIMIT 1;



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

* SELECT

pizza\_types.name,

SUM(orders\_details.quantity) AS order\_quantity

FROM

pizza\_types

JOIN

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

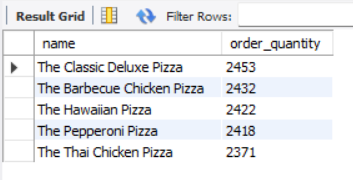
JOIN

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

GROUP BY pizza\_types.name

ORDER BY order\_quantity DESC

LIMIT 5;



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

* SELECT

pizza\_types.category,

SUM(orders\_details.quantity) AS category\_quantity

FROM

pizza\_types

JOIN

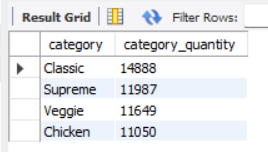
pizzas ON pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id

JOIN

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

GROUP BY pizza\_types.category

ORDER BY category\_quantity DESC;



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

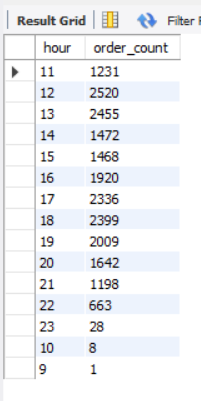
* SELECT

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

FROM

orders

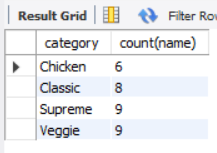
GROUP BY HOUR(order\_time);



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) as avg\_pizza\_ordered\_per\_day

FROM

(SELECT

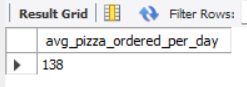
orders.order\_date, SUM(orders\_details.quantity) AS quantity

FROM

orders

JOIN orders\_details ON orders.order\_id = orders\_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(ROUND((orders\_details.quantity \* pizzas.price),

0)) AS revenue

FROM

pizza\_types

JOIN

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

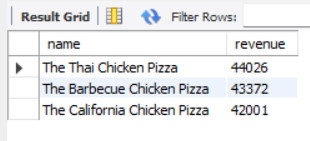
JOIN

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

GROUP BY pizza\_types.name

ORDER BY revenue DESC

LIMIT 3;



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

SELECT

pizza\_types.category,

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

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

FROM

orders\_details

JOIN

pizzas ON orders\_details.pizza\_id = pizzas.pizza\_id) \* 100, 3)

AS contribution\_of\_each\_pizza\_type

FROM

pizza\_types

JOIN

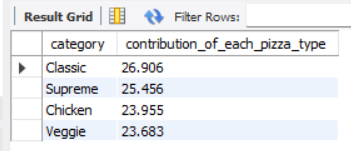
pizzas ON pizza\_types.pizza\_type\_id = pizzas.pizza\_type\_id

JOIN

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

GROUP BY pizza\_types.category

ORDER BY contribution\_of\_each\_pizza\_type DESC;



1. Analyze the cumulative revenue generated over time.

* select order\_date,

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

from

(select orders.order\_date,

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

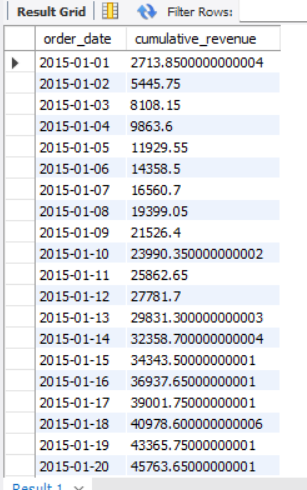
from orders\_details join pizzas

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;



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

* select name, category, revenue

from

(select category, name, revenue,

rank() over(partition by category order by revenue desc) as rnk

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 rnk <= 3;

