PIZZA SALES

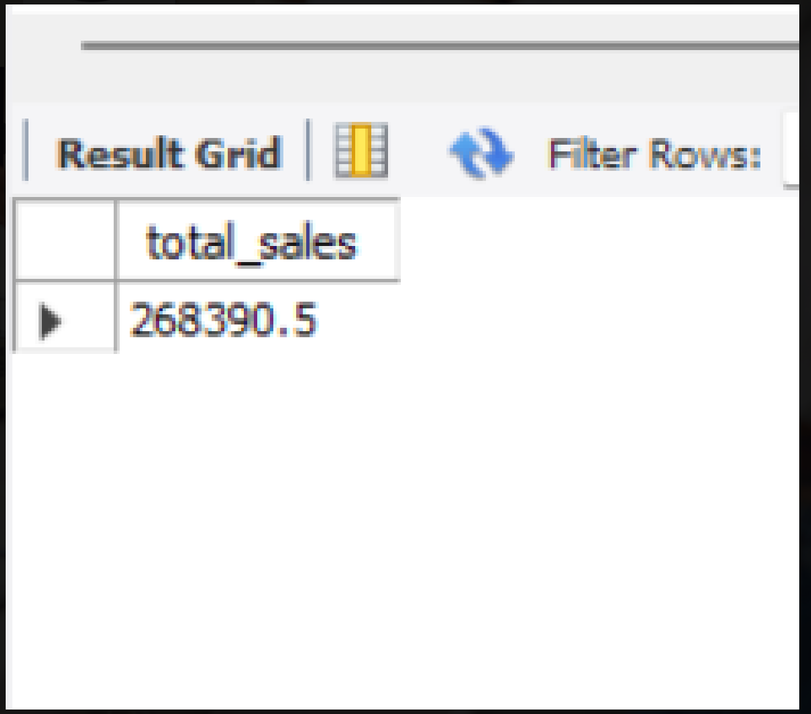
HELLO!

My name is Nikhil Neela. In this project I have utilised sql queries to solve questions that were related to 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
```

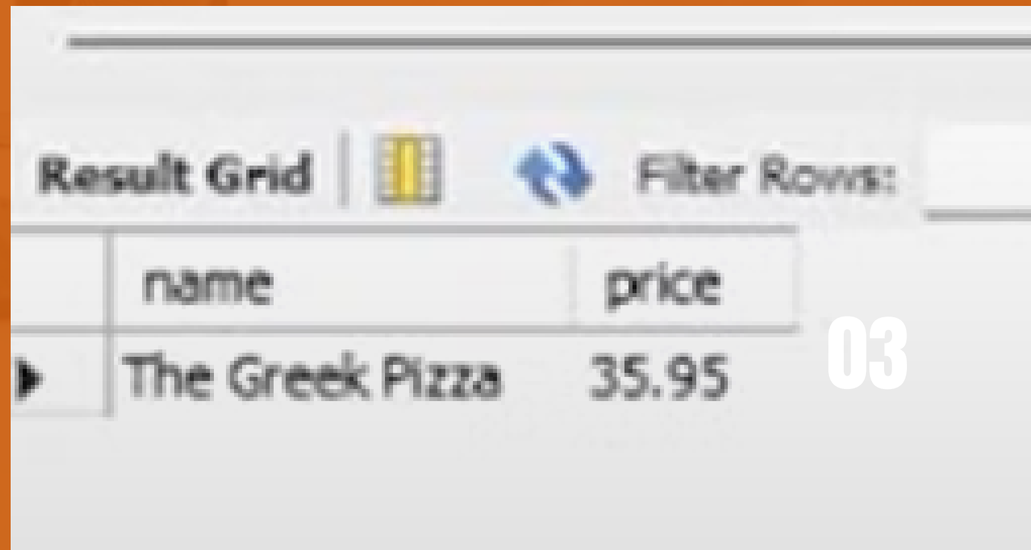
Calculated the total revenue generated from pizza sales by multiplying the quantity of each pizza ordered with its corresponding price and summing the total



Identify the highest-priced pizza.

```sql
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 2;
```

Identified the most expensive pizza from the database.



I dentify the most common pizza size ordered.

```sql
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 limit 3
;
```

Result Grid | Filter Rows:

size	order_count
L	9195
M	4925
S	3057

Identified the most frequently ordered pizza size based on the number of orders.

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
22	663
23	28
10	8

Analyzed the number of orders placed at different hours of the day to identify peak ordering times.

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
3 •    SELECT category, count(name) from pizza_types
4      group by category;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Cor

category	count(name)
ital_supr_l	2
peppr_salami_l	2
prsc_argla_m	2
spicy_ital_l	4
ckn_pesto_l	2
five_cheese_l	4
spinach_supr_l	2
thai_ckn_s	4
cali_ckn_s	2
cali_ckn_m	2
ckn_alfredo_m	2
pep_msh_pep_m	2
peppr_salami_s	2
spicy_ital_s	2
veggie_veg_m	2
hawaiian_s	2
napolitana_m	4
spinach_supr_m	2
pepperoni_l	2
ital_veggie_s	2
prsc_argla_l	2
mediterraneo_m	2
thai_ckn_l	4

DETERMINED THE DISTRIBUTION OF PIZZAS BASED ON THEIR CATEGORIES BY JOINING RELEVANT TABLES.

ANALYZE THE CUMMULATIVE REVENUE GENERATED OVER TIME.

```sql
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;
```

Calculated and analyze cumulative revenue generated over time to track business performance and sales growth.

Result Grid | Filter Rows:

order_date	cum_revenue
2015-01-01	854.25
2015-01-02	2247
2015-01-03	2961
2015-01-04	3751
2015-01-05	4414.75
2015-01-06	5182.75
2015-01-07	6032
2015-01-08	7223
2015-01-09	8083
2015-01-10	9024.5
2015-01-11	9413.25
2015-01-12	9868.25
2015-01-13	10404.5
2015-01-14	11344.25
2015-01-15	12235.25
2015-01-16	12962.25
2015-01-17	13479.75
2015-01-18	14165.75
2015-01-19	14703
2015-01-20	15916
2015-01-21	16510

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

```sql
select category, name, revenue,
rank() over(partition by category order by revenue desc) 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;
```

Identified the top 3 best-selling pizzas per category based on total revenue generated.

Result Grid | Filter Rows:

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

Project Summary: Pizza Sales and Revenue Analysis

OBJECTIVE :

This project aims to analyze pizza sales data to extract insights on order trends, revenue generation, pizza popularity, peak sales hours, and category-wise performance. The analysis provides valuable business intelligence for optimizing inventory, marketing, and operational decisions.

CONCLUSION :

This project provided data-driven insights into sales trends, revenue performance, and customer preferences. Implementing these findings can help optimize marketing strategies, improve inventory planning, and increase overall profitability.