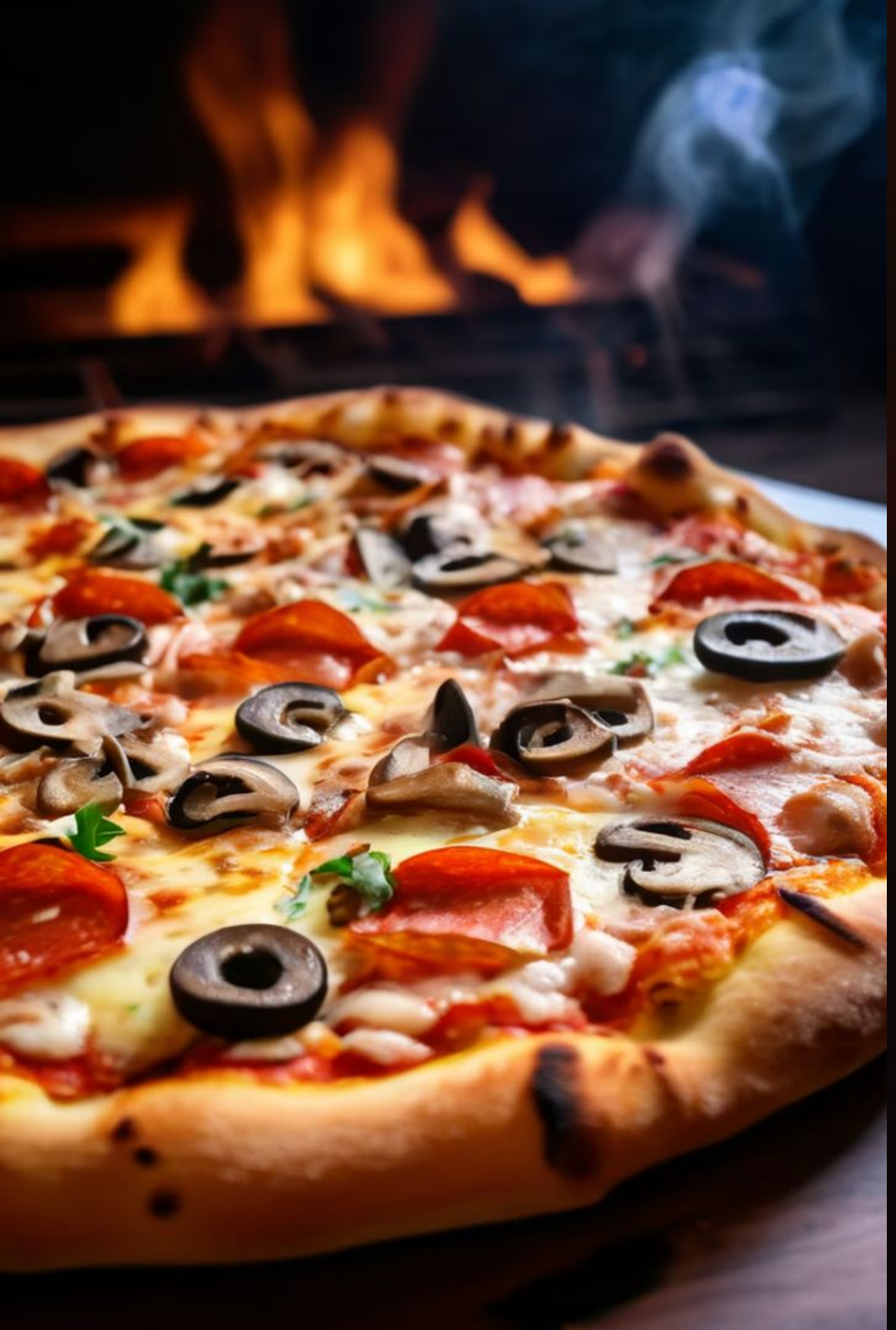


Prime Pizza Analytics:

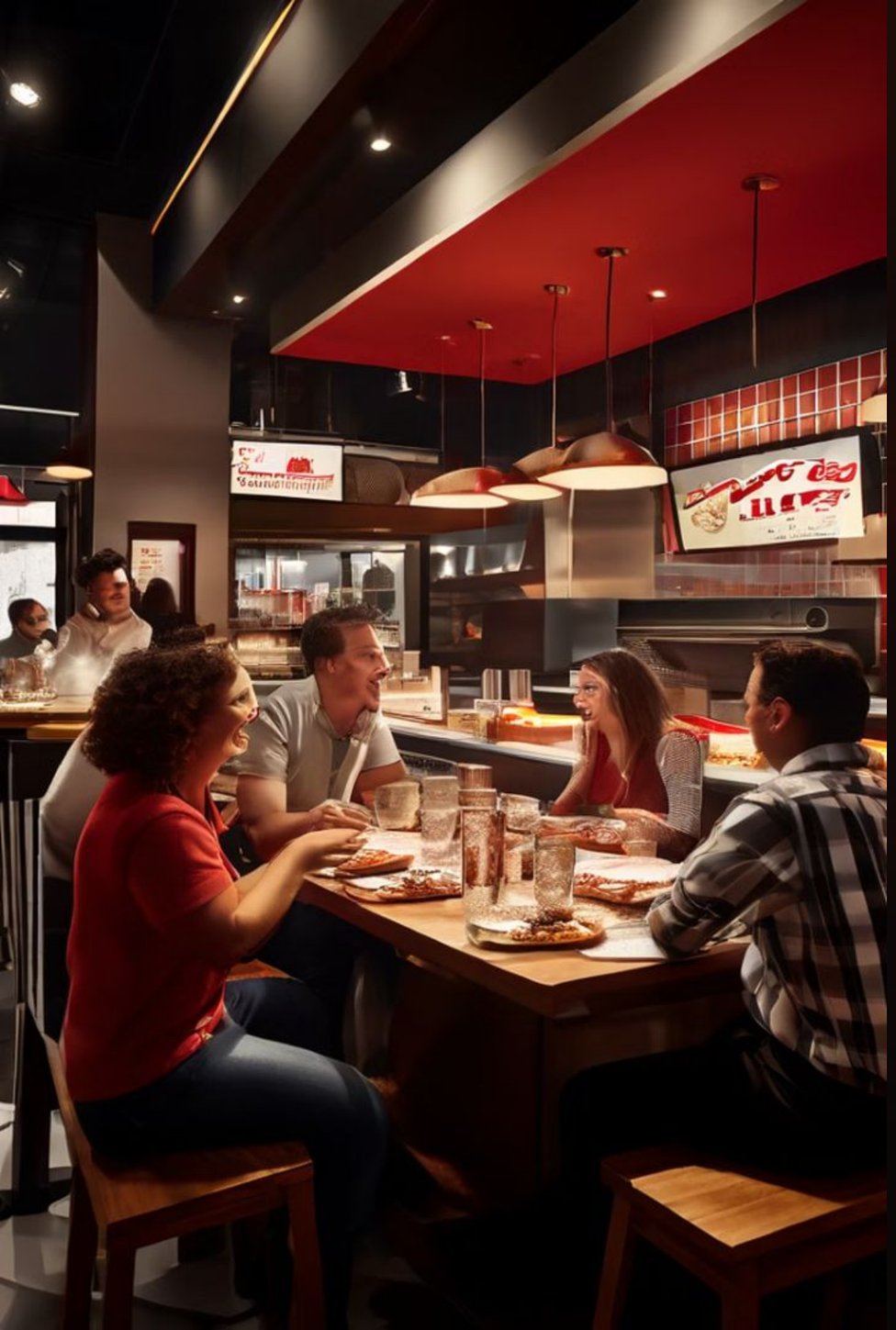
A MySQL Project on Pizza Sales

Dive into the world of Prime Pizza and explore its sales data through a powerful MySQL project. Analyze pizza orders, identify sales trends, and discover actionable insights.

Presented by: Syed Ahsan Hussain

Role: Data Analyst

https://www.linkedin.com/in/syed-ahsan-sa01/

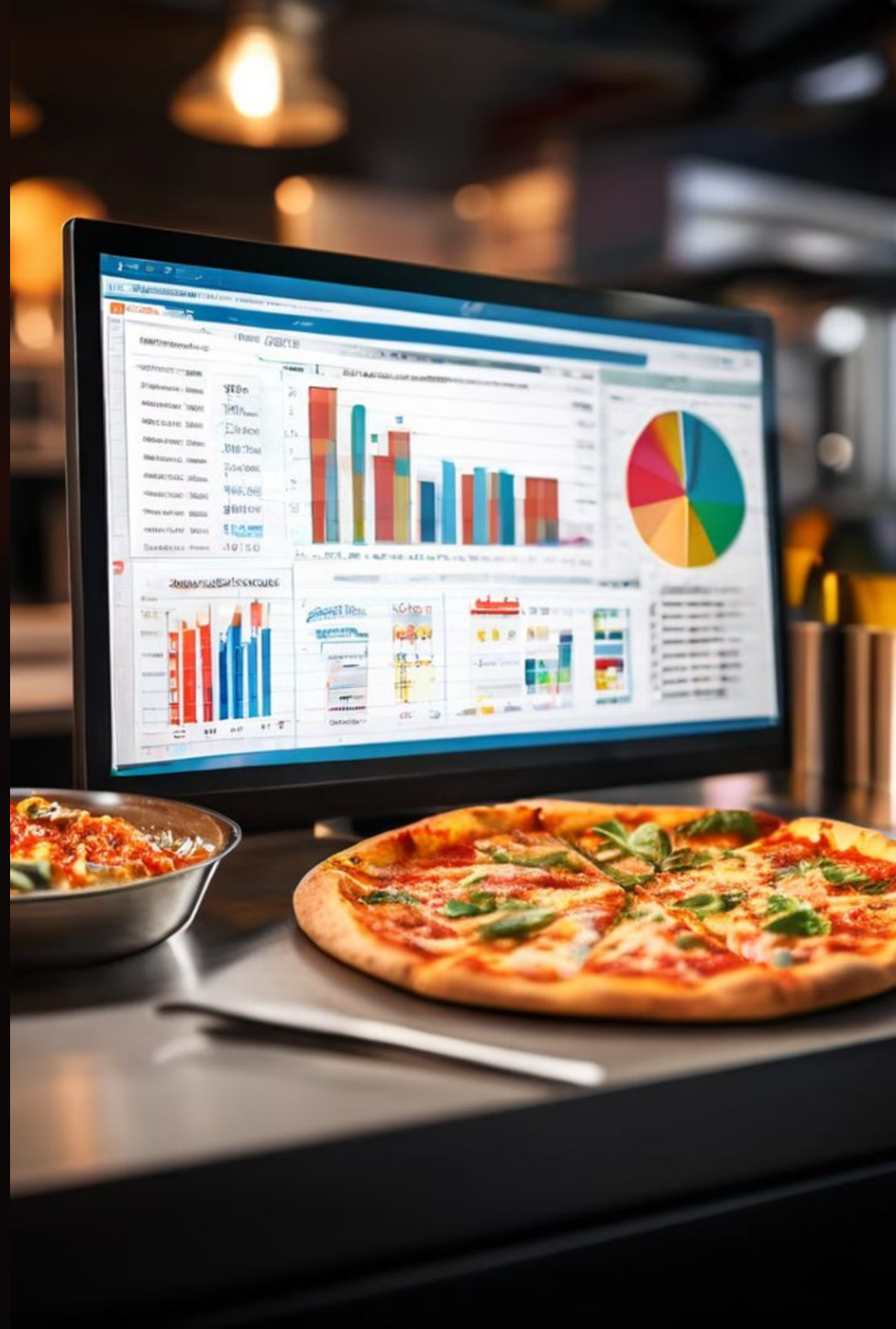


Introduction to Prime Pizza

Welcome to **Prime Pizza Analytics**, a comprehensive MySQL-based project led by **Syed Ahsan Hussain**, focused on unraveling the key insights from pizza sales data. The goal of this analysis is to provide actionable insights that can help optimize inventory, enhance customer satisfaction, and ultimately boost sales for a pizza restaurant. Whether you're looking to understand your best-selling pizzas or discover revenue trends, **Prime Pizza Analytics** offers the data-driven insights you need to make informed business decisions.

Data Overview

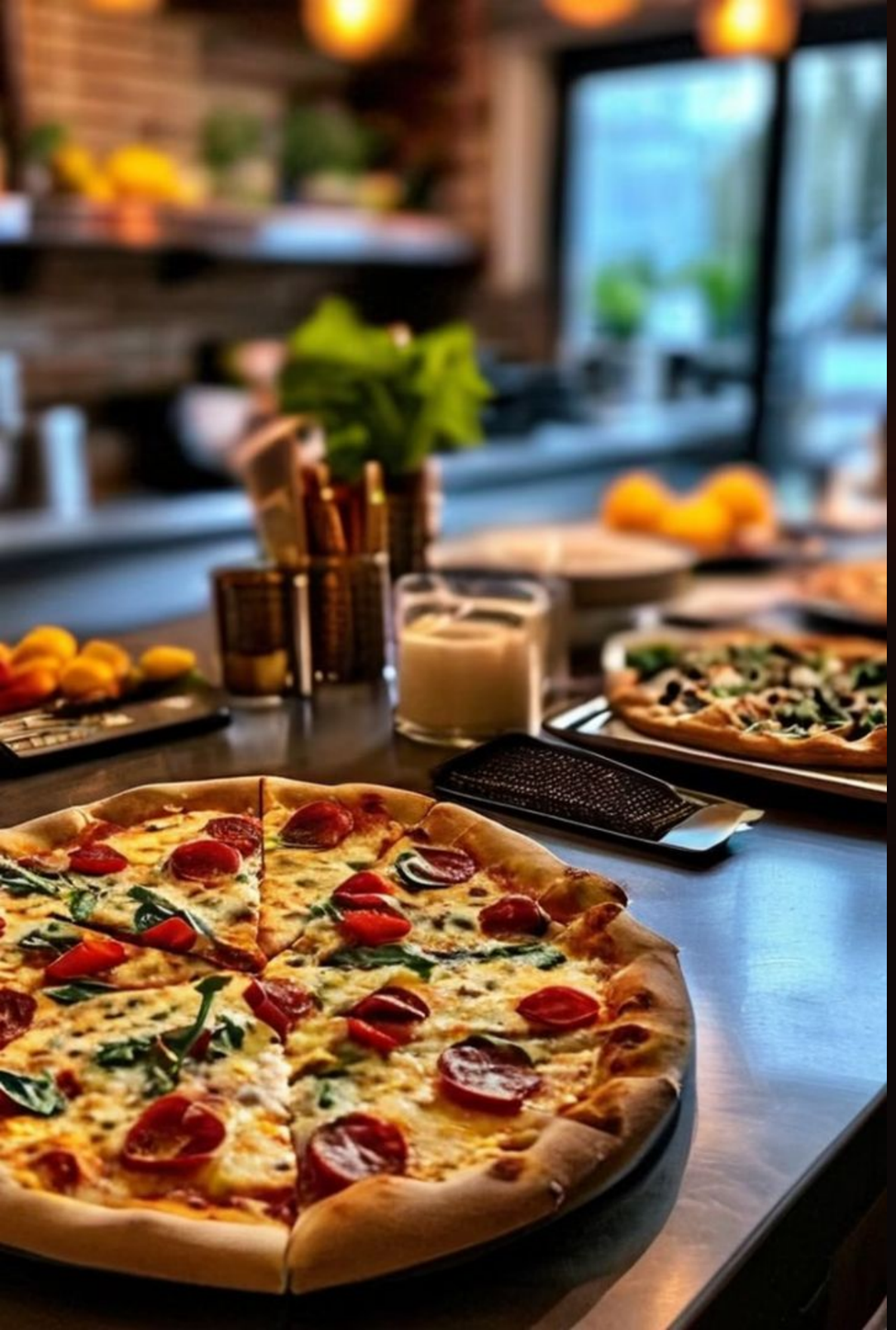
The dataset used in **Prime Pizza Analytics** provides a comprehensive overview of the restaurant's sales activities. It includes key attributes such as **order IDs, pizza types, sizes, prices, quantities ordered, and timestamps.** Each entry represents a unique order, capturing critical details that enable the analysis of sales patterns and customer preferences. The data is structured in relational tables, making it easy to perform various SQL queries for in-depth exploration. This dataset serves as the backbone of the project, allowing for insights into revenue generation, popular pizza choices, and ordering trends, all of which are crucial for optimizing business strategies.



Database Design and Schema

A well-designed database schema is crucial for organizing and managing the sales data effectively. The schema defines tables, columns, data types, and relationships between different entities.

Table	Columns	Data Types
Pizzas	Pizza ID, Pizza Type ID, Size, Price	INT, ,INT, VARCHAR, DECIMAL
Pizza Types	Pizza Type ID, Name, Category, Ingredients	INT, VARCHAR, VARCHAR, VARCHAR
Orders	Order ID, Order Date, Order Time	INT, DATE, TIME
Order Details	Order ID, Pizza ID, Quantity, Order Details ID	INT, INT, INT



Total Numbers of Orders Placed

To understand the overall sales volume, it is important to analyze the total number of orders placed over a specific period. This information provides a baseline for further analysis.

SQL Query

```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

Result

Result Grid

	total_orders
▶	21350

Total revenue Generated from Pizza Sales

The total revenue generated from pizza sales provides a clear measure of the restaurant's financial performance, reflecting the income from all pizza orders placed during the analyzed period.

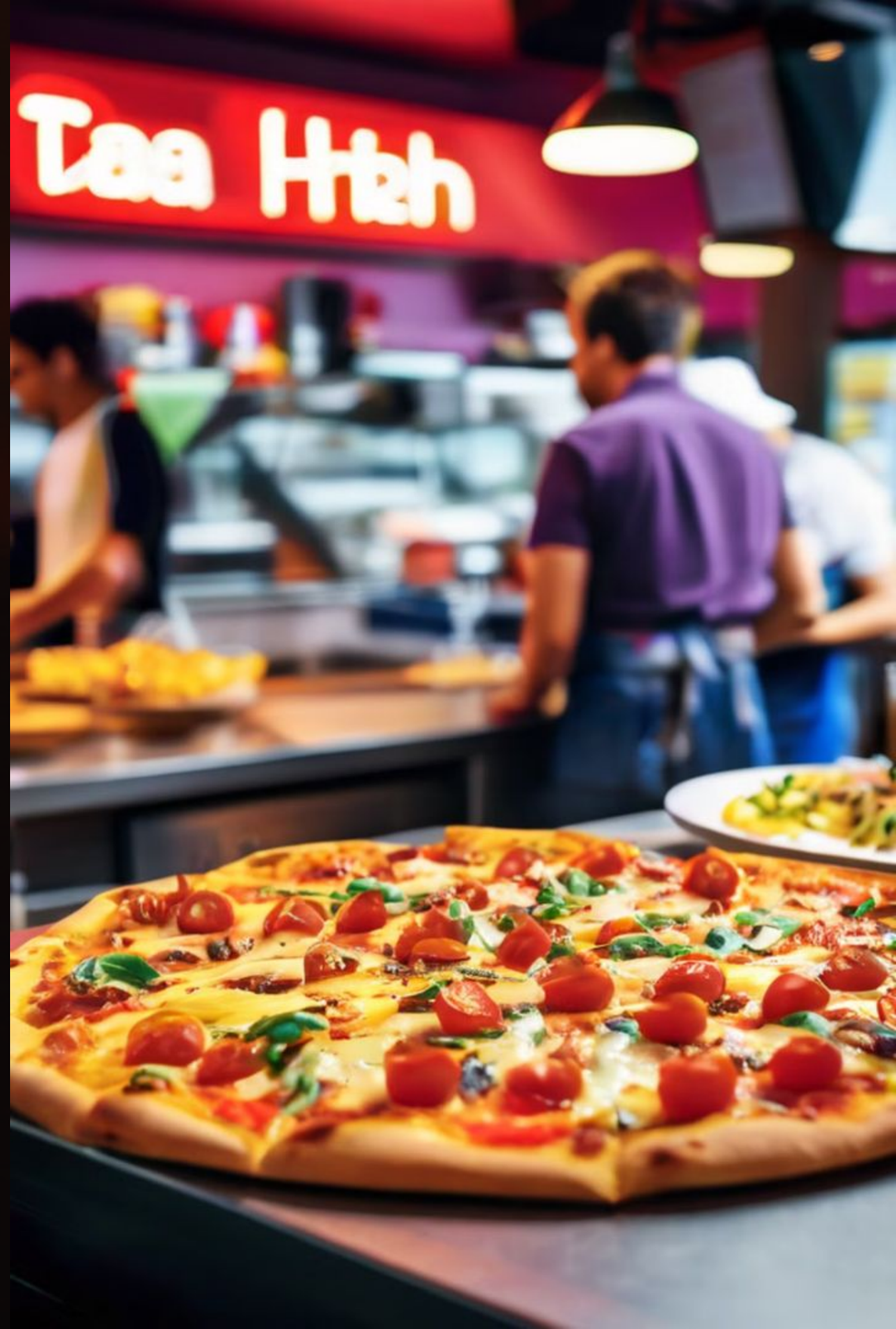
SQL Query

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

Result

Result Grid

total_revenue
786457.1
30209.2
1014.75
179



Highest Priced Pizza

This query highlights the most expensive pizza in the menu, often distinguished by gourmet ingredients, a specialty recipe, or an extra-large size, catering to customers seeking a more indulgent experience.

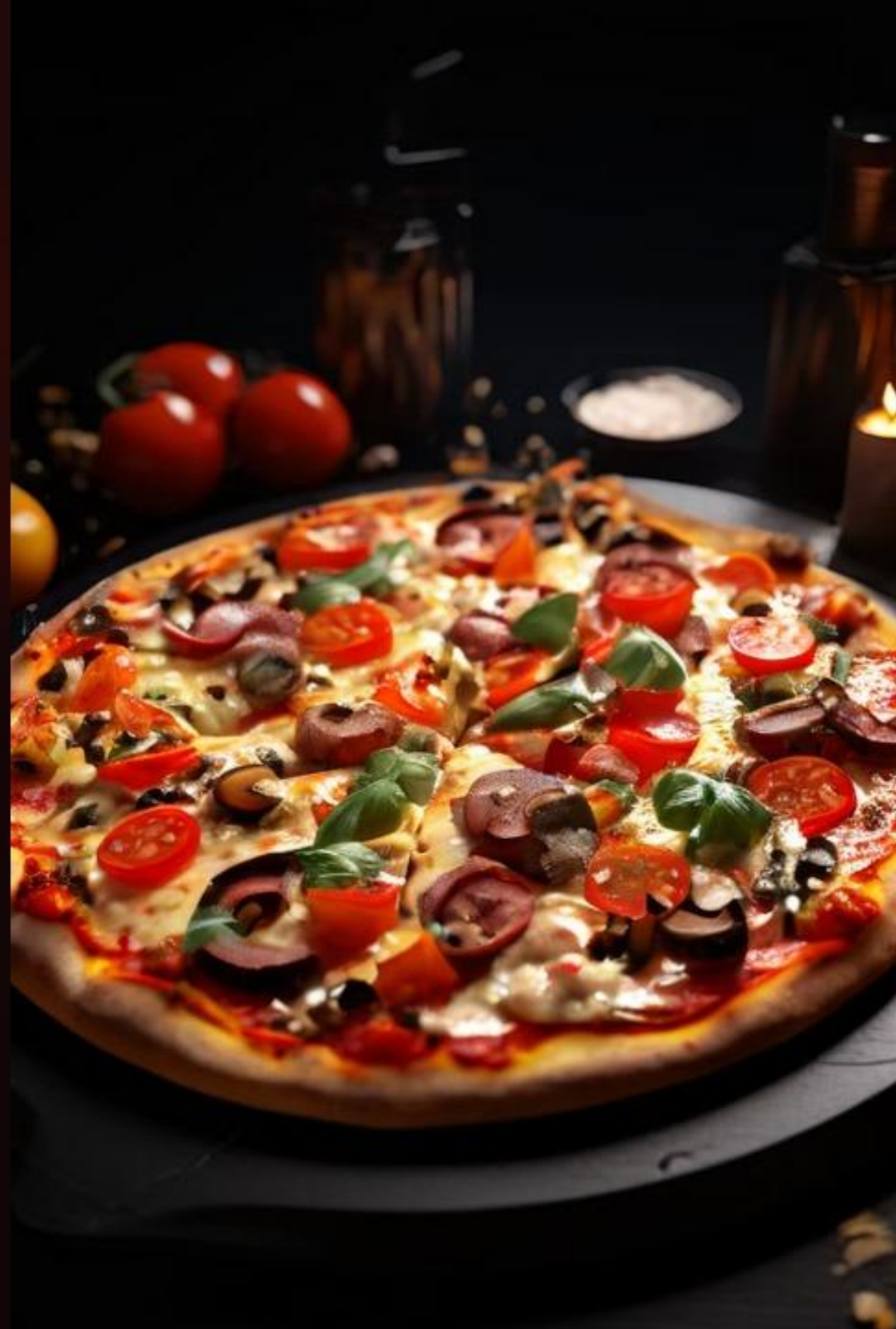
SQL Query

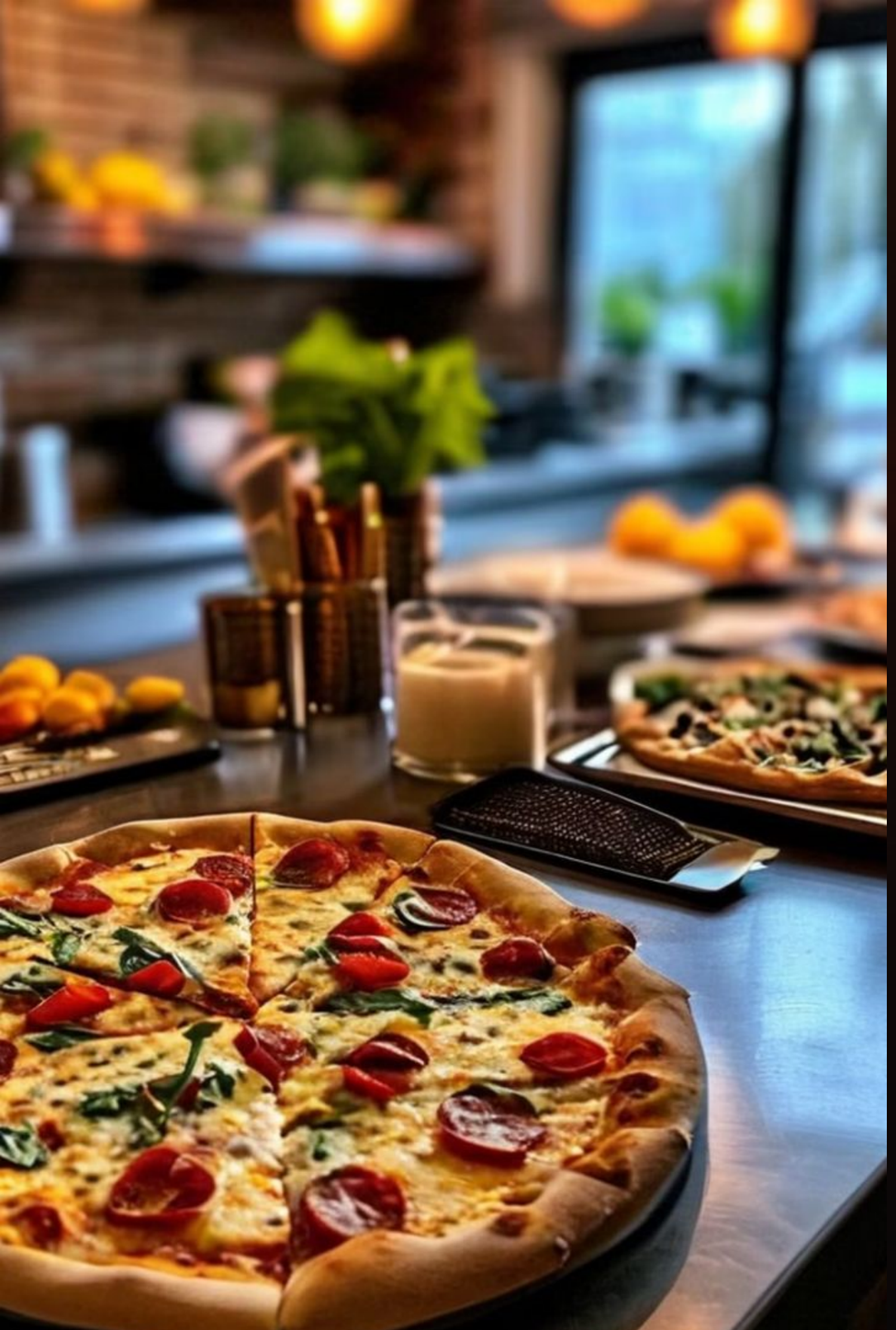
```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

Result

Result Grid | Filter Rows:

name	price
The Greek Pizza	35.95





The Most Common Pizza Size Ordered

The most common pizza size ordered represents the preferred choice among customers, indicating the size that is most frequently selected, often reflecting popular demand and customer preferences.

SQL Query

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

Result

Result Grid

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

Top 5 Most Ordered Pizza Types

The top 5 most ordered pizza types are the most popular choices among customers, representing the pizzas that are ordered the most frequently and reflect prevailing customer preferences.

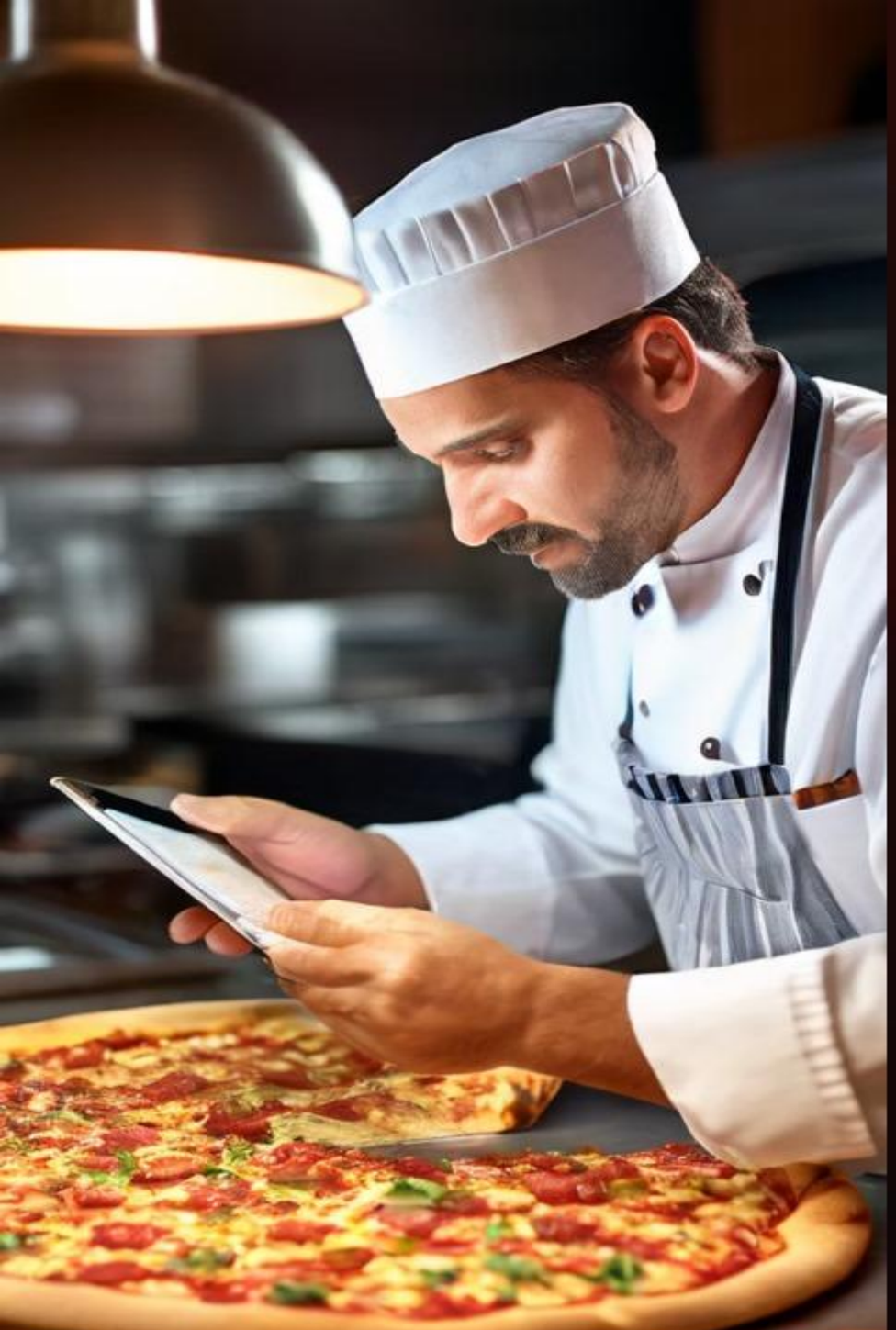
SQL Query

```sql
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;
```

Result

Result Grid | Filter Rows:

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



Total Quantity of Each Pizza Category Ordered

The total quantity of each pizza category ordered indicates the overall demand for each type of pizza, summarizing how many of each category were sold during the analyzed period.

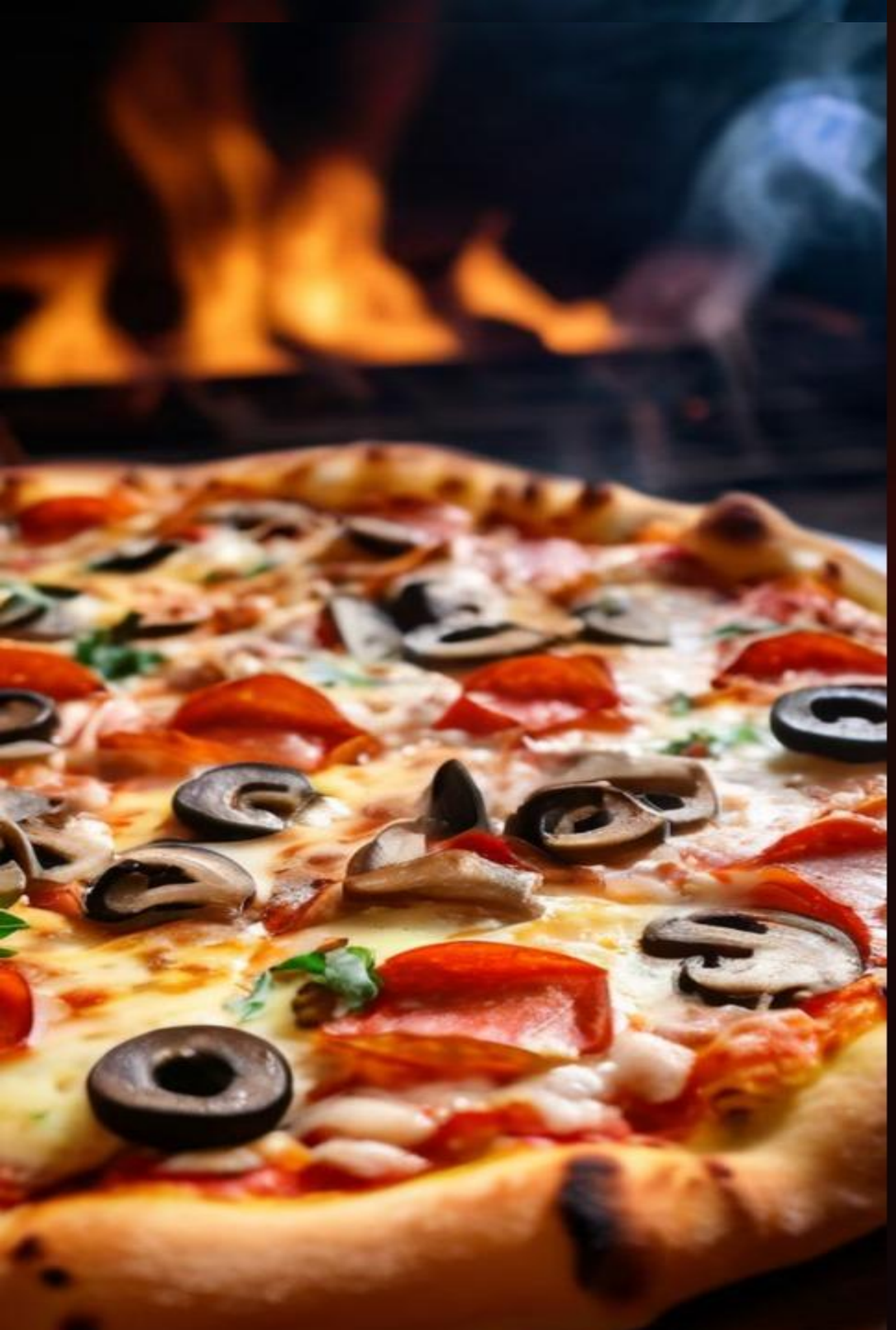
SQL Query

```sql
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;
```

Result

Result Grid | Filter Rows:

category	quantity
Chicken	11050
Veggie	11649
Supreme	11987
Classic	14888



Distribution of orders by hour (Day)

The distribution of orders by hour of the day illustrates how pizza orders vary throughout the day, highlighting peak times and trends in customer activity across different hours.

SQL Query

```
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time)
ORDER BY hour ASC;
```

Result

Result Grid | Filter Rows:

hour	order_count
9	1
10	8
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336

Average number of Pizzas Ordered Per Day

The average number of pizzas ordered per day calculates the mean daily order volume over a specified period, providing insight into typical daily sales and customer demand.

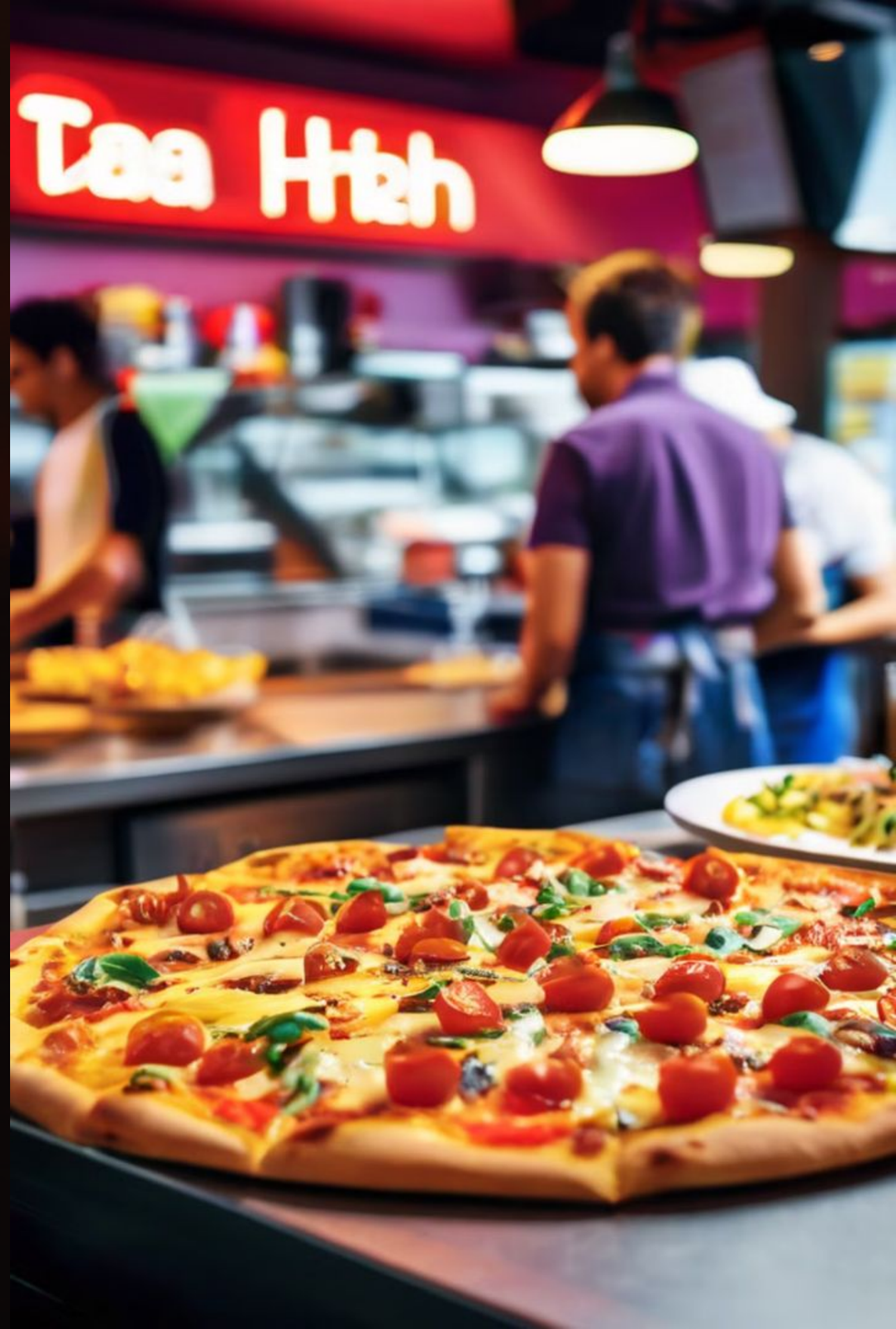
SQL Query

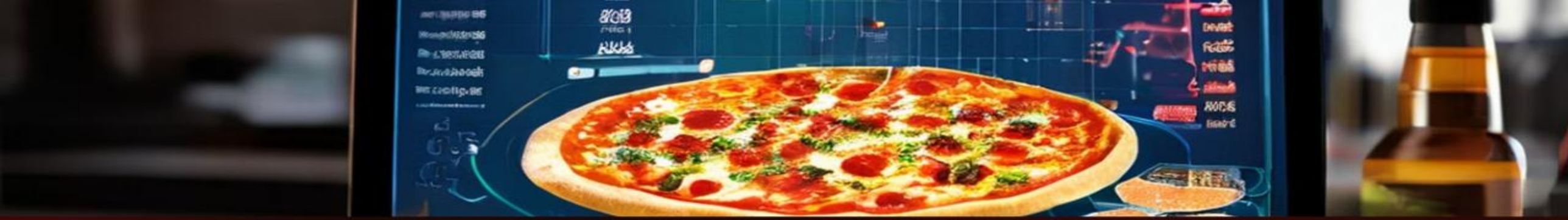
```sql
SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity
```

Result

Result Grid | Filter Rows:

avg_pizza_ordered
138





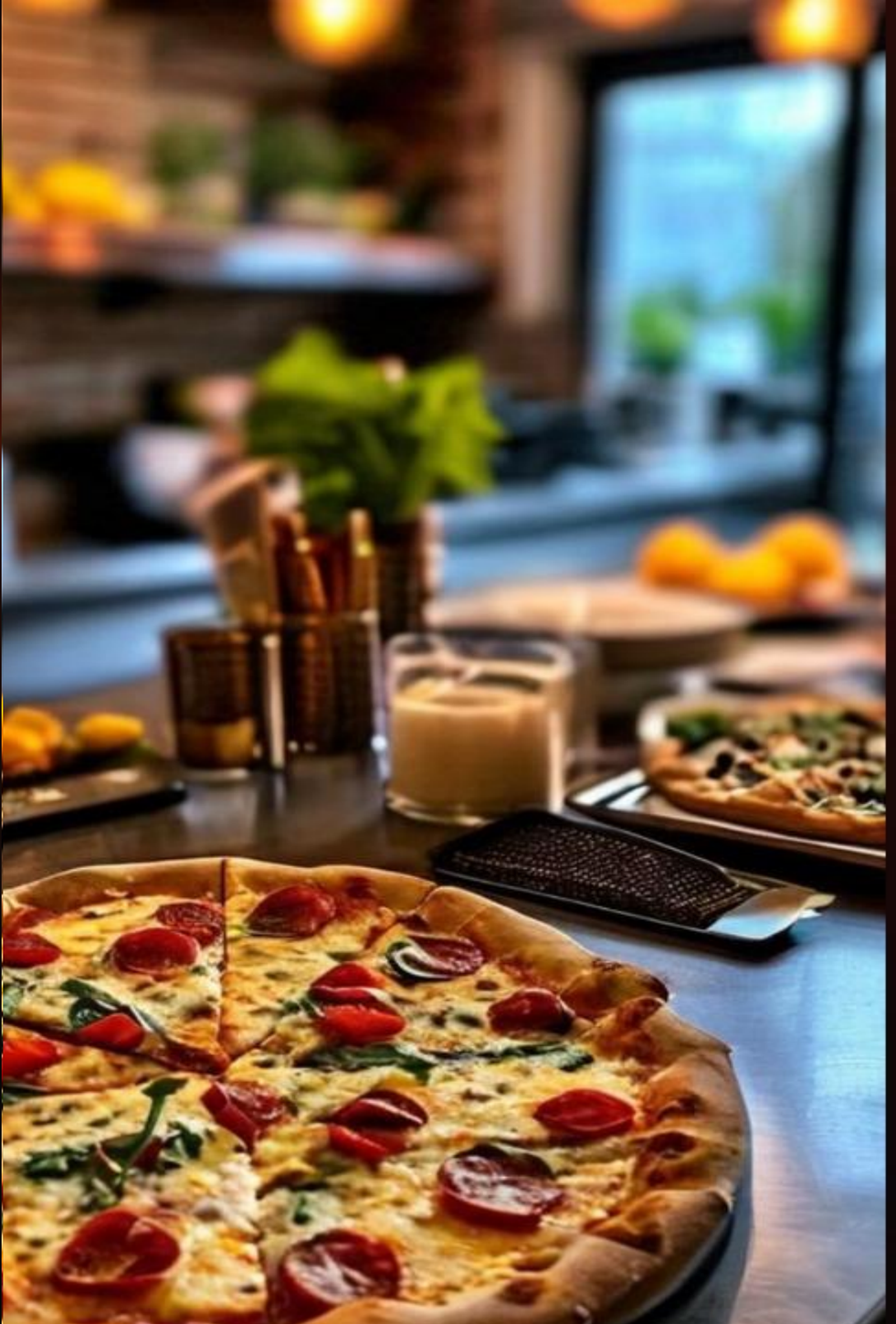
Category-Wise Distribution of Pizzas

The category-wise distribution of pizzas breaks down the sales by different pizza categories, such as toppings, sizes, or styles, showing the popularity and demand within each category.

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

Result Grid | Filter Rows:

category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9



Top 3 Most Ordered Pizza Types Based on Revenue

The top 3 most ordered pizza types based on revenue identify the three pizza varieties that generate the highest total revenue, reflecting both their popularity and pricing impact on the restaurant's financial performance.

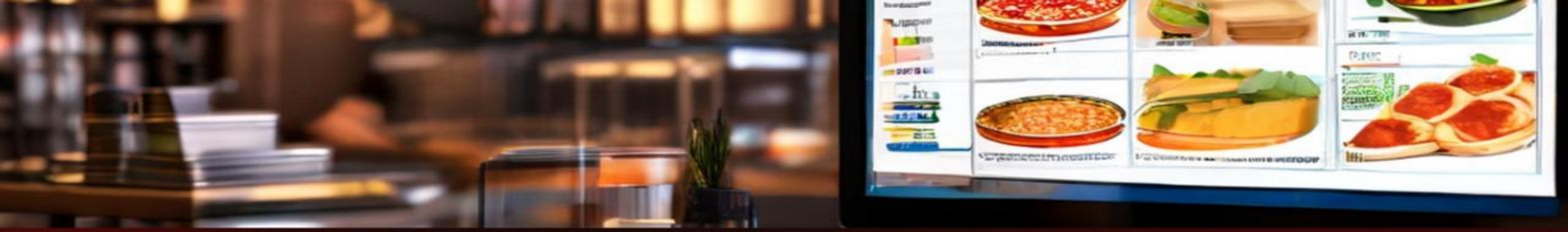
SQL Query

```sql
SELECT
    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.name
ORDER BY revenue DESC
LIMIT 3;
```

Result

Result Grid | Filter Rows:

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



Percentage Contribution of Each Pizza Type to Total Revenue

The percentage contribution of each pizza type to total revenue shows how much each type contributes to the overall revenue, highlighting the relative financial impact of each pizza variety.

SQL Query

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

Result

Result Grid | Filter Rows:

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Cumulative Revenue Generated Over Time

Cumulative revenue generated over time tracks the total revenue accumulated from sales from the beginning of the period up to a specific point, providing insights into revenue growth trends and long-term financial performance.

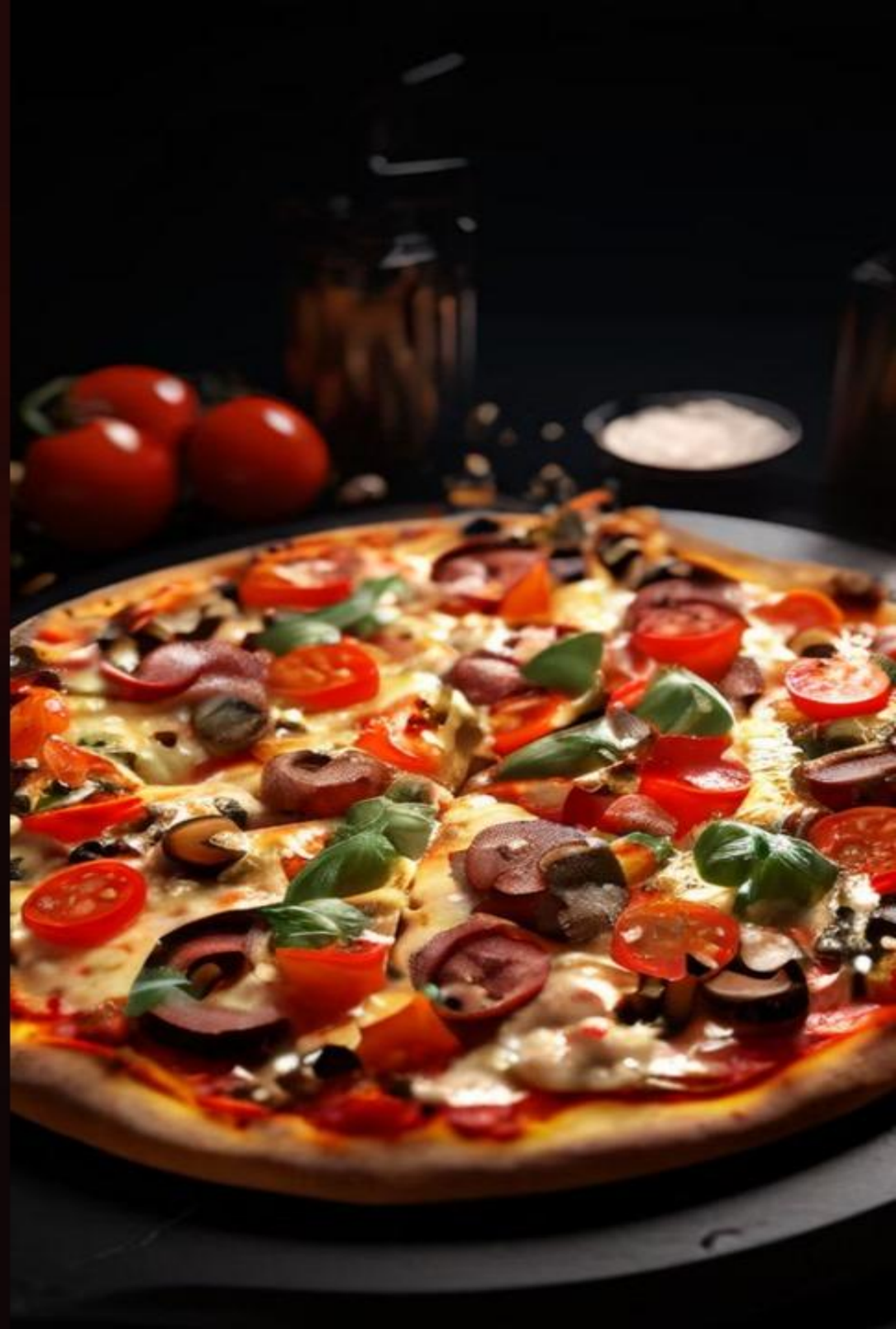
SQL Query

```sql
select order_date,
Round(sum(revenue) over(order by order_date),0) as cumulative_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;
```

Result

Result Grid | Filter Rows:

order_date	cumulative_revenue
2015-01-01	2714
2015-01-02	5446
2015-01-03	8108
2015-01-04	9864
2015-01-05	11930
2015-01-06	14358



Top 3 Most Ordered Pizza Types Based on Revenue

The top 3 most ordered pizza types based on revenue identify the three pizza varieties that generate the highest total revenue, reflecting both their popularity and pricing impact on the restaurant's financial performance.

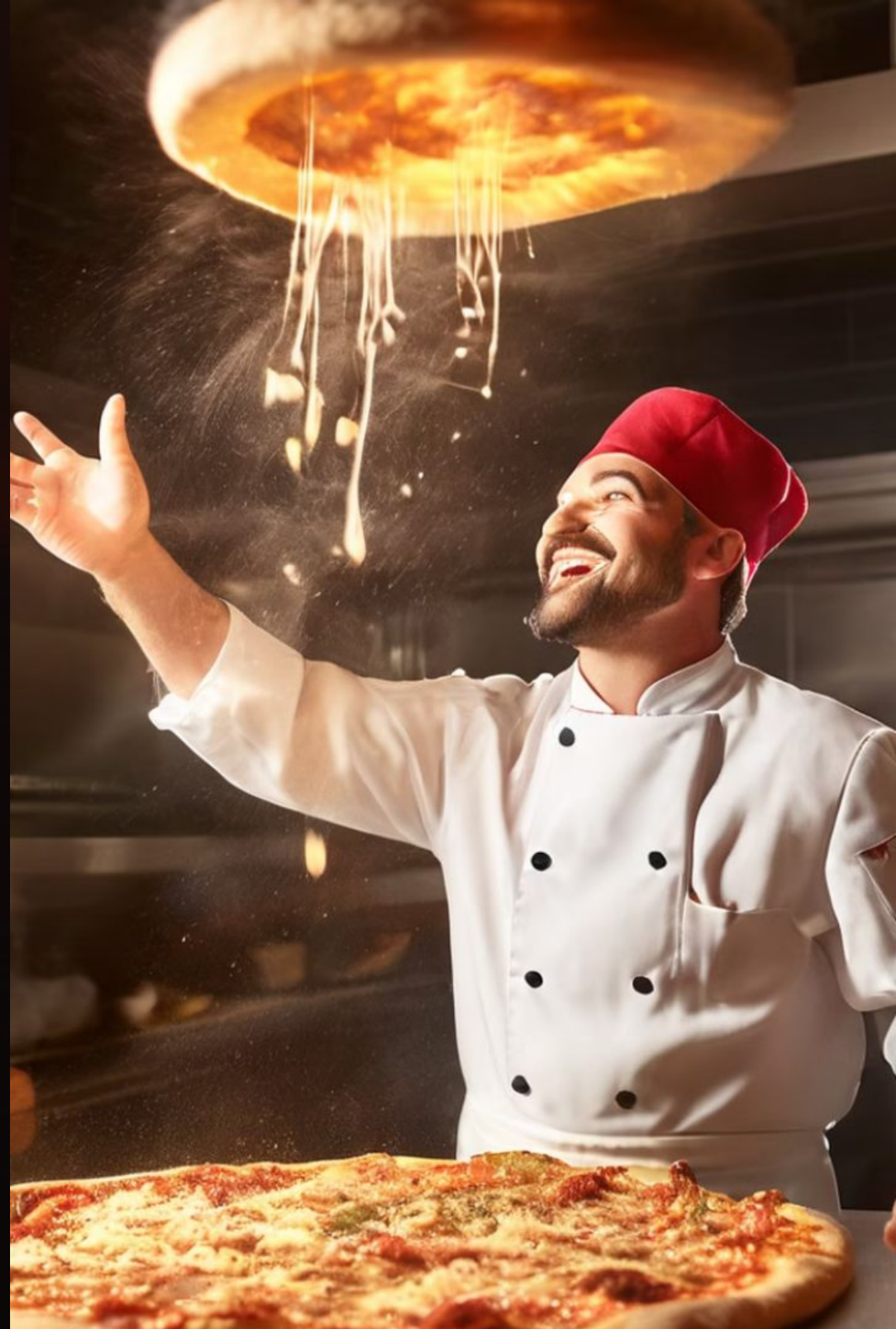
SQL Query

```sql
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;
```

Result

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75



Data Analysis and Insights

Once the data is stored in the database, it can be analyzed using SQL queries to generate valuable insights. This involves exploring patterns, trends, and relationships within the data.

1 Top-Selling Pizzas

Identifying the most popular pizzas based on order frequency.

2 Customer Segmentation

Grouping customers based on purchase history and preferences.

3 Peak Hours and Days

Analyzing sales volume during different times of the day and week.

4 Delivery vs. Pickup

Assessing the ratio of delivery orders to pickup orders.

Recommendations and Future Improvements

Based on the analysis of pizza sales data, recommendations can be made for future improvements. This may involve implementing new strategies or adjusting existing ones.

1 Marketing Campaigns

Launching targeted marketing campaigns based on customer preferences and demographics.

2 Menu Optimization

Adding new pizza options based on popular toppings or customer requests.

3 Loyalty Programs

Implementing a loyalty program to incentivize repeat customers and boost sales.

4 Online Ordering System

Improving the online ordering system for a seamless customer experience.

Conclusion and Key Takeaways

This MySQL project on Prime Pizza sales provides valuable insights into customer behavior, sales trends, and areas for improvement. The analysis reveals actionable strategies for enhancing customer satisfaction and increasing profitability.

1 **Data-Driven Decisions**

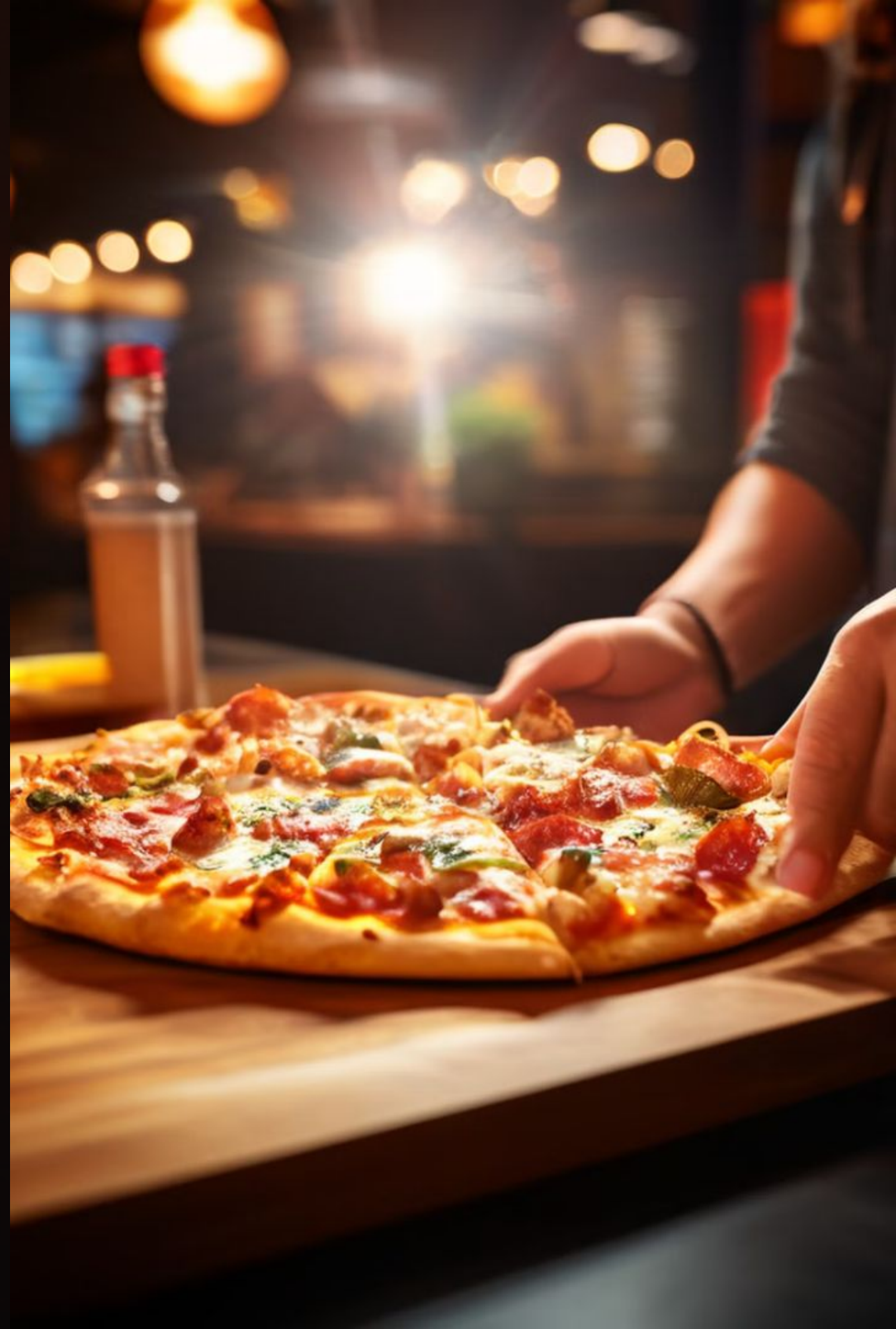
The importance of data-driven decision-making for optimizing business operations.

2 **Customer Understanding**

The value of understanding customer preferences and demographics.

3 **Continuous Improvement**

The need for continuous improvement and adaptation based on data insights.



Thank You!