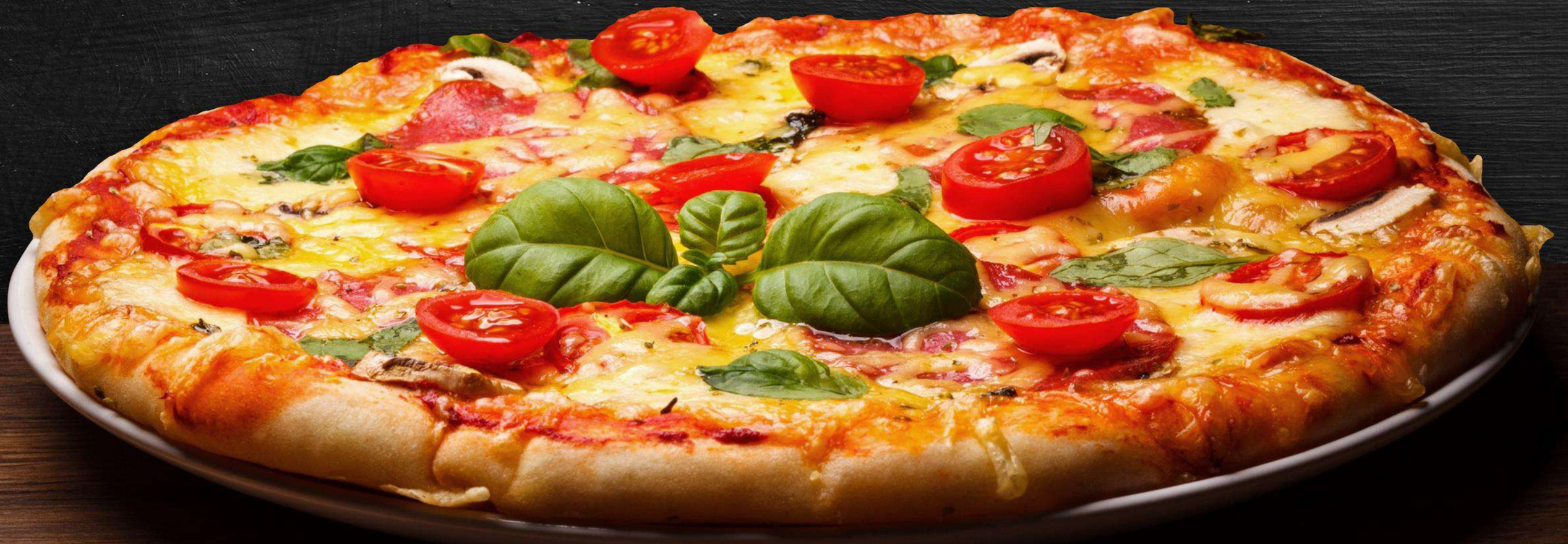
Pizza Sales Analysis Using SQL

Uncovering Key Insight



WHAT ARE WE ANALYZING?

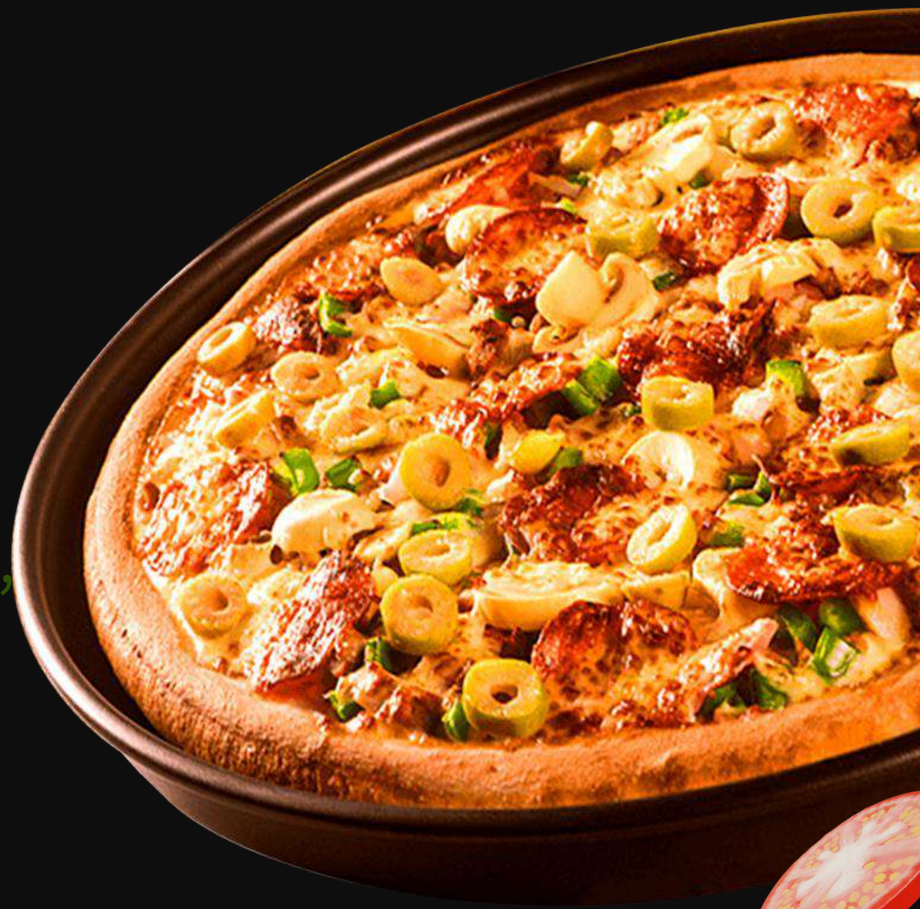
Goals & Key Business Questions

Objective:

- Analyze pizza sales data and uncover key trends, popular pizzas, and sales performance.

Business Task:

- Identify best-selling pizzas, most popular sizes, and peak sales times.
- Recommend strategies based on insights from the data.



OUR DATA SOURCE

UNDERSTANDING THE PIZZA SALES DATASET

Dataset Description

Key tables: orders, orders_details, pizzas, pizza_types, etc.

Data cleaning steps

Handled null values, corrected data types, etc.

```sql
-- Creates the 'pizzasales' database

create database pizzasales;
create table orders(
order_id int primary key,
order_date date not null,
order_time time not null
);

create table orders_details(
order_details_id int primary key,
order_id int,
pizza_id text,
quantity int not null);
```

BASIC SQL ANALYSIS

HOW MANY ORDERS WERE PLACED?

Key Sales Metrics

```sql
-- Q1.Retrieve the total number of orders placed.

SELECT
    COUNT(*) as total_number_of_orders_placed
FROM
    orders AS total_orders;
```

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

total_number_of_orders_placed
21350

HOW MUCH REVENUE DID PIZZA SALES GENERATE

```sql
-- Q2.Calculate the total revenue generated from pizza sales.
SELECT
    ROUND(SUM(o.quantity * p.price), 2) AS total_sales
FROM
    orders_details o
        JOIN
    pizzas p ON o.pizza_id = p.pizza_id;
```

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

total_sales
817860.05

WHICH PIZZA COSTS THE MOST?

PREMIUM MENU INSIGHTS

```sql
-- Q3.Identify the highest-priced pizza.

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch

name	highest_priced_pizza
The Greek Pizza	35.95

WHAT'S THE PREFERRED PIZZA SIZE?

CUSTOMER ORDERING TRENDS

```sql
-- Q4.Identify the most common pizza size ordered.
 SELECT
    p.size, COUNT(o.order_details_id) AS size_order
FROM
    orders_details o
        JOIN
    pizzas p ON o.pizza_id = p.pizza_id
GROUP BY p.size
ORDER BY size_order DESC ;
```

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

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

WHICH PIZZAS ARE THE CROWD FAVORITES?

```sql
-- Q5.List the top 5 most ordered pizza types along with their quantities.
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS quantity
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 quantity DESC
LIMIT 5;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch 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

INTERMEDIATE SQL ANALYSIS

WHICH PIZZA CATEGORIES ARE ORDERED THE MOST

```sql
-- Q6.Join the necessary tables to find the total quantity of each pizza category ordered.
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 Grid | Filter Rows: | Export: | Wrap Cell Content:

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

WHEN DO CUSTOMERS ORDER THE MOST?

```sql
-- 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 Grid | Filter Rows: | Export: | Wrap Cell Content:

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
9	1

WHICH PIZZA CATEGORIES ARE THE MOST POPULAR

```sql
-- Q8.Join relevant tables to find the category-wise distribution of pizzas.
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

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

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

Daily Demand: How Many Pizzas Are Sold on Average?

```sql
-- Q9.Group the orders by date and calculate the average number of pizzas ordered per day.

SELECT
    ROUND(AVG(quantity), 0) AS average_pizzas_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;
```

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

average_pizzas_ordered_per_day
138

WHICH PIZZAS GENERATE THE MOST REVENUE?

```sql
-- Q10.Determine the top 3 most ordered pizza types based on revenue.
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 Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

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

WHICH PIZZA CATEGORIES DRIVE THE HIGHEST SALES?

```sql
-- Q11. Calculate the percentage contribution of each pizza type to total revenue.
select pizza_types.category,
round(sum(orders_details.quantity*pizzas.price) /(SELECT
    ROUND(SUM(o.quantity * p.price), 2) AS total_sales
FROM
    orders_details o
        JOIN
    pizzas p ON o.pizza_id = p.pizza_id)*100 ,2)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.category order by revenue desc ;
```

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

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

HOW HAS REVENUE GROWN OVER TIME?

```sql
-- Q12.Analyze the cumulati  r v  ue  e erate  over time.

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 Grid | Filter Rows: | Export: | Wrap Cell Content:

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004

BEST-SELLING PIZZAS IN EACH CATEGORY

```sql
-- Q13.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((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;
```

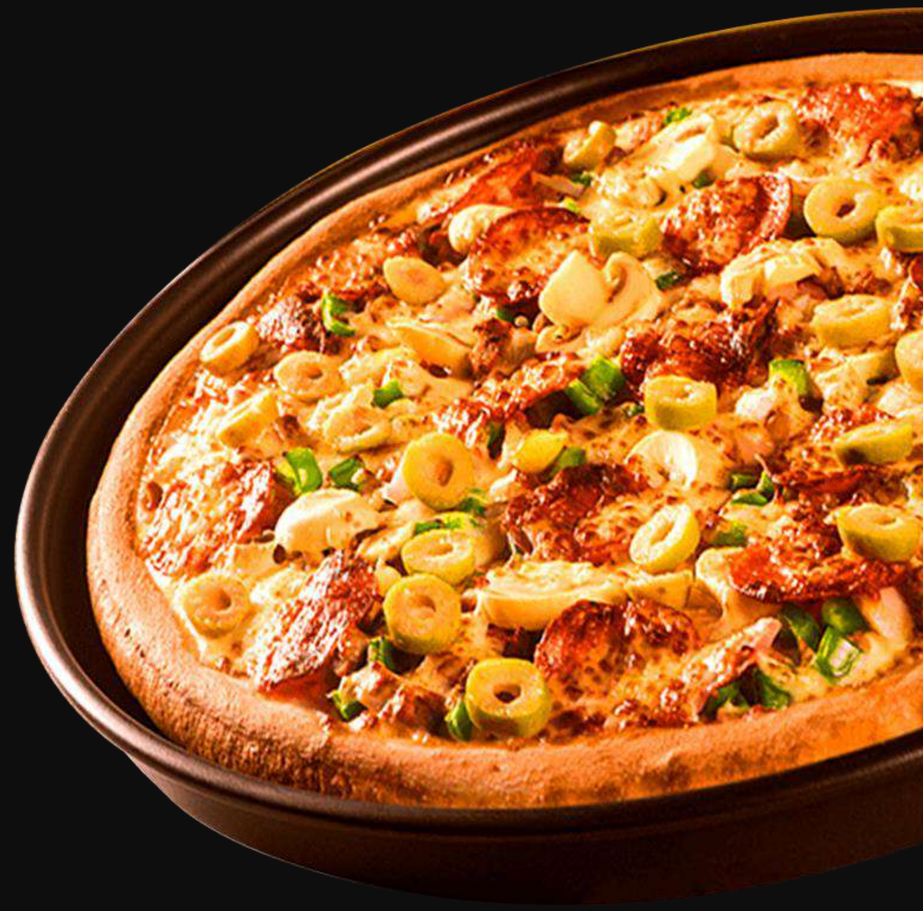
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

KEY TAKEAWAYS

💰 SALES & REVENUE PERFORMANCE

- 🚀 21,350 orders | 💰 $817,860.05 revenue | 🕛 Peak: 12-1 PM → Opportunity for lunchtime promotions.



🍕 Customer Preferences

- 🔥 **Best-Seller:** The Classic Deluxe Pizza (2,453 orders) → Feature in "Best-Seller" promotions.
- 📊 **Preferred Size:** Large (18,526 orders) → Upsell XL/XXL sizes for higher revenue.

💡 Growth Opportunities

- 📌 **Top Revenue Pizza:** Thai Chicken Pizza ($43,434.25) → Bundle in meal deals.
- 📊 **Category Share:** Classic (26.91%), Supreme (25.46%), Chicken (23.96%), Veggie (23.68%) → Promote underperforming categories.