

# Pizza Sales Data Analysis Using SQL

This project is based on analyzing pizza sales data using SQL to derive useful insights for business understanding and decision-making. I created a database named `pizzahut` and used various SQL queries to explore the data such as total orders placed, revenue generated, most common pizza sizes, top-selling pizza types, and category-wise sales. I also analyzed the time-

based trends like orders by hour and daily averages. Techniques like joins, aggregations, window functions, and ranking helped uncover patterns in customer preferences and revenue distribution. Overall, this project helped me practice real-world SQL skills and understand how data can guide business strategies in the food industry.



Retrieve the total number of orders placed.

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

| Result Grid |              |
|-------------|--------------|
|             | total_orders |
| ▶           | 21350        |

-- Calculate the total revenue generated from pizza sales.

```
SELECT
    ROUND(SUM(order_detail.quantity * pizzas.price),
          2) AS total_sales
FROM
    order_detail
    JOIN
    pizzas ON pizzas.pizza_id = order_detail.pizza_id
```

| Result Grid |             |
|-------------|-------------|
|             | total_sales |
| ▶           | 817860.05   |

# Identify the most common pizza size ordered.

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

| Result Grid |      |             |
|-------------|------|-------------|
|             | size | order_count |
| ▶           | L    | 18526       |
|             | M    | 15385       |
|             | S    | 14137       |
|             | XL   | 544         |
|             | XXL  | 28          |

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

```
SELECT
    pizza_types.name, SUM(order_detail.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_detail ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

|   | 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     |

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

```
SELECT
    pizza_types.category, SUM(order_detail.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_detail ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

|   | category | quantity |
|---|----------|----------|
| ▶ | Classic  | 14888    |
|   | Supreme  | 11987    |
|   | Veggie   | 11649    |
|   | Chicken  | 11050    |

# 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;
```

|   | 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           |

# Join relevant tables to find the category-wise distribution of pizzas

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

|   | category | count(name) |
|---|----------|-------------|
| ▶ | Chicken  | 6           |
|   | Classic  | 8           |
|   | Supreme  | 9           |
|   | Veggie   | 9           |

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

```
SELECT
    ROUND(AVG(quantity), 0) AS average_pizza_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_detail.quantity) AS quantity
    FROM
        orders
    JOIN order_detail ON orders.order_id = order_detail.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

| Result Grid |                               |
|-------------|-------------------------------|
|             | average_pizza_ordered_per_day |
| ▶           | 138                           |

# Determine the top 3 most ordered pizza types based on revenue

```
SELECT
    pizza_types.name,
    SUM(order_detail.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_detail ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

| Result Grid |                              |          |
|-------------|------------------------------|----------|
|             | name                         | revenue  |
| ▶           | The Thai Chicken Pizza       | 43434.25 |
|             | The Barbecue Chicken Pizza   | 42768    |
|             | The California Chicken Pizza | 41409.5  |

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

```
SELECT
    pizza_types.category,
    round((SUM(order_detail.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_detail.quantity * pizzas.price),
        2) AS total_sales
    FROM
        order_detail
        JOIN
            pizzas ON pizzas.pizza_id = order_detail.pizza_id)),2) * 100 AS revenue
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
        order_detail ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

| category | revenue |
|----------|---------|
| Classic  | 27      |
| Supreme  | 25      |
| Veggie   | 24      |
| Chicken  | 24      |

# Analyze the cumulative revenue generated over time.

```
select order_date,sum(revenue) over(order by order_date) as cum_revenue  
from  
  
(select orders.order_date, sum(order_detail.quantity*pizzas.price) as revenue  
from order_detail  
join pizzas  
on order_detail.pizza_id=pizzas.pizza_id  
join orders  
on orders.order_id=order_detail.order_id  
group by orders.order_date) as sales;
```

| Result Grid |                    | Filter Rows: |
|-------------|--------------------|--------------|
|             | order_date         | cum_revenue  |
| 2015-06-10  | 368866.6500000002  |              |
| 2015-06-11  | 371517.1500000002  |              |
| 2015-06-12  | 373655.7500000002  |              |
| 2015-06-13  | 376164.6500000002  |              |
| 2015-06-14  | 378023.6500000002  |              |
| 2015-06-15  | 380619.2500000002  |              |
| 2015-06-16  | 382517.55000000016 |              |
| 2015-06-17  | 384654.65000000014 |              |
| 2015-06-18  | 386639.15000000014 |              |
| 2015-06-19  | 389432.60000000015 |              |
| 2015-06-20  | 391493.2000000001  |              |
| 2015-06-21  | 393418.40000000014 |              |
| 2015-06-22  | 395737.7000000001  |              |
| 2015-06-23  | 397780.4500000001  |              |
| 2015-06-24  | 400107.9500000001  |              |
| 2015-06-25  | 402507.10000000015 |              |
| 2015-06-26  | 405252.6000000002  |              |
| 2015-06-27  | 408065.5000000002  |              |
| 2015-06-28  | 409635.2000000002  |              |

# 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((order_detail.quantity)*pizzas.price) as revenue
from pizza_types
join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_detail
on order_detail.pizza_id=pizzas.pizza_id
group by pizza_types.category,pizza_types.name) as a) as b
where rn <=3;
```

| Result Grid |                              |                   |
|-------------|------------------------------|-------------------|
|             | 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           |





