

# PIZZA SALES ANALYSIS USING MYSQL

# Pizza Sales Overview

Power BI Desktop

pizzaCategory

Chicken

Classic

Supreme

Veggie

date

01-01-2015



31-12-2015



Total Sales

817.86K

OrdersPerMonth

1935

1853

1845

1841

1840

1799

1792

1773

1685

1680

1661

1646

July

May

January

August

March

April

November

June

February

December

September

October

Top Selling Pizza

The Classic Deluxe Pizza

2453

The Barbecue Chicken Pizza

2432

The Hawaiian Pizza

2422

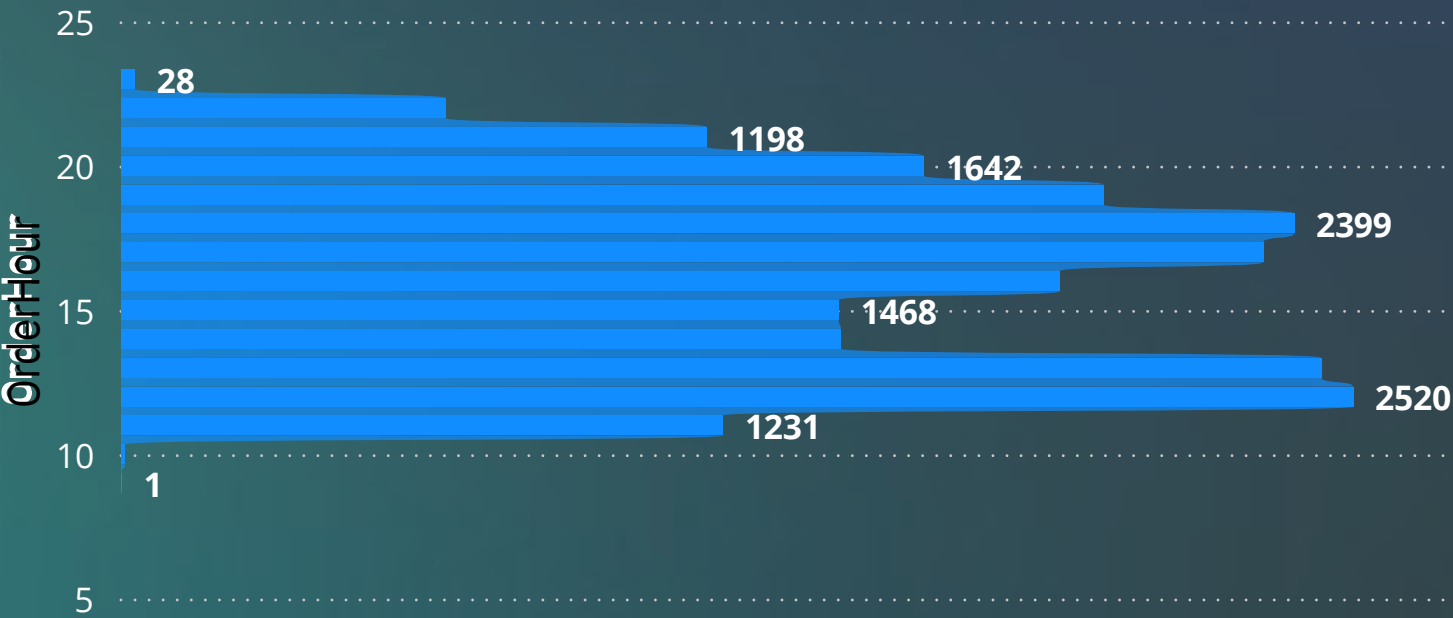
The Pepperoni Pizza

2418

The Thai Chicken Pizza

2371

Orders Per Hour






OrdersPerHour

# There are four tables



1. **order\_detail**: It has four columns: 'order\_details\_id', 'order\_id', 'pizza\_id', 'quantity'.
2. **orders**: It has three columns: 'order\_id', 'ored\_date', and 'order\_time'.
3. **pizza\_types**: It has four columns: 'pizza\_type\_id', 'name', 'category', 'ingredients'.
4. **pizzas**: it has four columns: 'pizza\_id', 'pizza\_type\_id', 'size', 'price'.

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

```
select count(order_id) as total_orders from orders;
```

Result Grid			 Filter Rows: <input type="text"/>
	total_orders		
	21350		

```
-- Calculate the total revenue generated from pizza sales
SELECT
    SUM((order_detail.quantity * pizzas.price)) AS revenue
FROM
    order_detail
    JOIN
    pizzas ON order_detail.pizza_id = pizzas.pizza_id;
```

Result Grid			 Filter Rows: <input data-bbox="1512 1046 1739 1185" type="text"/>
	revenue		
▶	817860.0499999993		

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

```
SELECT
```



```
    *
```

```
FROM
```

```
    pizzas
```

```
ORDER BY price DESC
```

```
LIMIT 1;
```

Result Grid				
  Filter Rows: <input type="text"/>				
	pizza_id	pizza_type_id	size	price
▶	the_greek_xxl	the_greek	XXL	35.95

```
-- 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 |      |             | Filter Rows: |  |
|-------------|------|-------------|--------------|--|
|             | size | order_count |              |  |
| ▶           | L    | 18526       |              |  |
|             | M    | 15385       |              |  |
|             | S    | 14137       |              |  |
|             | XL   | 544         |              |  |
|             | XXL  | 28          |              |  |

```
-- 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;
```

| Result Grid |                            |          | Filter Rows: |  | Export: |
|-------------|----------------------------|----------|--------------|--|---------|
|             | 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;
```

| Result Grid |          |          | Filter Rows: |  |
|-------------|----------|----------|--------------|--|
|             | 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(order_time) order by hour;`

| 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        |              |
|             | 18   | 2399        |              |
|             | 19   | 2222        |              |

-- Join relevant tables to find the category-wise distribution of pizzas.67

- ```
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	

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

```
SELECT
```

```
    AVG(avg_order)
```

```
FROM
```

```
    (SELECT
```

```
        orders.ored_date, SUM(order_detail.quantity) AS avg_order
```

```
    FROM
```

```
        orders
```

```
    JOIN order_detail ON orders.order_id = order_detail.order_id
```

```
    GROUP BY orders.ored_date) AS order_quantity;
```

Result Grid



Filter Rows

	AVG(avg_order)
▶	138.4749

-- 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 pizzas.pizza_type_id = pizza_types.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 |                              |          | Filter Rows: |
|-------------|------------------------------|----------|--------------|
|             | 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)
    FROM
        order_detail
        JOIN
        pizzas ON pizzas.pizza_id = order_detail.pizza_id))*100,2) as revenue
FROM
    pizzas
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN
    order_detail ON order_detail.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

Result Grid



Filter Rows:

|   | category | revenue |
|---|----------|---------|
| ▶ | Classic  | 26.91   |
|   | Supreme  | 25.46   |
|   | Chicken  | 23.96   |
|   | Veggie   | 23.68   |

[illegible]

-- 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 rn from
(select pizza_types.category, pizza_types.name, sum(order_detail.quantity* pizzas.price) as revenue from pizzas
join pizza_types 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				Filter Rows:	Export:	Wrap Cell Content
	name	category	revenue			
▶	The Thai Chicken Pizza	Chicken	43434.25			
	The Barbecue Chicken Pizza	Chicken	42768			
	The California Chicken Pizza	Chicken	41409.5			
	The Classic Deluxe Pizza	Classic	38180.5			
	The Hawaiian Pizza	Classic	32273.25			
	The Pepperoni Pizza	Classic	30161.75			
	The Spicy Italian Pizza	Supreme	34831.25			
	The Italian Supreme Pizza	Supreme	33476.75			
	The Sicilian Pizza	Supreme	30940.5			
	The Four Cheese Pizza	Veggie	32265.700000000065			
	The Mexicana Pizza	Veggie	26780.75			
	The Five Cheese Pizza	Veggie	26066.5			