



PIZZA

SALES

**SQL
PROJECT**





VIEW NOW







Overview


Hello , My name is Shweta Rathod, and I am excited to present my project on Pizza Sales Analysis using SQL. This project focuses on analyzing pizza sales data for the year 2015. Through this analysis, I have explored various aspects of pizza sales, including the number of orders, revenue generation, and other insightful metrics. By utilizing SQL, I have been able to extract meaningful insights from the dataset, which consists of detailed information on pizza types, prices, and order quantities. This analysis aims to provide a comprehensive understanding of the sales dynamics and performance of different pizza offerings over the course of the year. Join me as I delve into the data and uncover key trends and patterns that can help drive business decisions and strategies for optimizing pizza sales.

Tables Used For Project

	order_details
▼	 Columns
◆	order_details_id
◆	order_id
◆	pizza_id
◆	quantity

	orders
▼	 Columns
◆	order_id
◆	order_date
◆	order_time

	pizza_types
▼	 Columns
◆	pizza_type_id
◆	name
◆	category
◆	ingredients

	pizzas
▼	 Columns
◆	pizza_id
◆	pizza_type_id
◆	size
◆	price

Questions

1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.
3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.
5. List the top 5 most ordered pizza types along with their quantities.
6. Join the necessary tables to find the total quantity of each pizza category ordered.
7. Determine the distribution of orders by hour of the day.
8. Join relevant tables to find the category-wise distribution of pizzas.
9. Group the orders by date and calculate the average number of pizzas ordered per day.
10. Determine the top 3 most ordered pizza types based on revenue.
11. Calculate the percentage contribution of each pizza type to total revenue.
12. Analyze the cumulative revenue generated over time.
13. Determine the top 3 most ordered pizza types based on revenue for each pizza category.

Creating Database and Table

```
1 ● create database pizzahut;  
2  
3 ● ⊖ create table orders(  
4     order_id int not null,  
5     order_date date not null,  
6     order_time time not null,  
7     primary key(order_id) );
```

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

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

	total_orders
▶	21350

```
-- 2. Calculate the total revenue generated from pizza sales.
```

```
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;
```

	total_sales
▶	817860.05

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

```
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;
```

	name	price
▶	The Greek Pizza	35.95

```
-- 4. Identify the most common pizza size ordered.
```

```
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;
```

	size	order_count
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28


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

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;
```

	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

```
-- 6.join the necessary table to find the total quantity of pizza orderd.
```

```
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;
```

	category	quantity
►	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

```
-- 7.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);
```

	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

```
-- 8.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

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

```
SELECT  
    round( AVG(quantity),0) as avg_pizza_ordered_per_day  
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;
```

	avg_pizza_ordered_per_day
▶	138

```
-- 10. Determine the top 3 most ordered pizza types based on revenue.
```

```
select pizza_types.name,  
sum(order_details.quantity * pizzas.price) as revenue  
from pizza_types join pizzas  
on pizzas.pizza_type_id = pizza_types.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;
```

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

```
-- 11.Calculate the percentage contribution of each pizza type to total revenue.
```

```
select pizza_types.category,  
round(sum(order_details.quantity * pizzas.price) / (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)  
*100,2) as revenue  
from pizza_types join pizzas  
on pizzas.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.category order by revenue desc;
```

	category	revenue
►	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

```
-- 12. Analyze the cumulative revenue generate over time.
```

```
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;
```

	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

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

select name, revenue,category, ranks from
(select category, name, revenue,
rank() over (partition by category order by revenue desc)
as ranks
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 ranks<=3;
```

	name	revenue	category	ranks
►	The Thai Chicken Pizza	43434.25	Chicken	1
	The Barbecue Chicken Pizza	42768	Chicken	2
	The California Chicken Pizza	41409.5	Chicken	3
	The Classic Deluxe Pizza	38180.5	Classic	1
	The Hawaiian Pizza	32273.25	Classic	2
	The Pepperoni Pizza	30161.75	Classic	3
	The Spicy Italian Pizza	34831.25	Supreme	1
	The Italian Supreme Pizza	33476.75	Supreme	2
	The Sicilian Pizza	30940.5	Supreme	3
	The Four Cheese Pizza	32265.700000000065	Veggie	1
	The Mexicana Pizza	26780.75	Veggie	2
	The Five Cheese Pizza	26066.5	Veggie	3

Key Indicators

- ✓ The most ordered pizza size was Large, with an order quantity of 18,956, while the XXL size is underperforming.
- ✓ The Classic pizza category had the highest order quantity at 14,888, whereas the Chicken category had the lowest at 11,050 orders.
- ✓ Most orders occur during afternoon lunch hours (12:00-13:00) and evening hours (17:00-19:00).
- ✓ The average pizza order quantity is 138.
- ✓ The top 7 pizzas based on order quantity are:
 - The Classic Deluxe Pizza
 - The Barbecue Chicken Pizza
 - The Hawaiian Pizza
 - The Pepperoni Pizza
 - The Thai Chicken Pizza
 - The California Chicken Pizza
 - The Sicilian Pizza
- ✓ The Classic category generates the highest revenue percentage at 26.91%.

THANK YOU !!