

Pizza Sales SQL Project

Suraj Kumar Singh



This project aims to analyze the sales data of pizzas to uncover trends and insights related to different sizes, months, and orders. By leveraging SQL, we have queried the database to answer specific questions about pizza sales performance. This analysis provides valuable information for understanding customer preferences, seasonal demand, and overall sales efficiency.

Name- Suraj Kumar Singh

Email- Suraj.KumarS@outlook.com

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

```
1 • Create database Pizzahut;  
2  
3     -- Q1    Retrieve the total number of orders placed.  
4  
5  
6 • use pizzahut;  
7  
8 • Select count(order_id) as totalorder from orders  
9
```

Result Grid		Filter Rows:	Export:
	totalorder		
▶	21350		

Q2 Calculate the total revenue generated from pizza sales.

```
1      -- Q2  Calculate the total revenue generated from pizza sales.  
2 •  SELECT  
3   ⌂    ROUND(SUM(order_details.quantity * pizzas.price),  
4           2) AS total_sales  
5  FROM  
6    order_details  
7    JOIN  
8    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

Result Grid		Filter Rows:	Export:
total_sales			
▶ 817860.05			

Q3. Identify the highest-priced pizza.

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

```
Select pizza_types.name, pizzas.price  
from pizza_types join pizzas  
on pizzas.pizza_type_id = pizza_types.pizza_type_id  
order by pizzas.price desc Limit 1
```

Result Grid | Filter Rows: Export:

	name	price
▶	The Greek Pizza	35.95

Q 4. IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
Query 1 SQL File 3* SQL File 4* SQL File 5* x
2 -- Q4 Identify the most common pizza size ordered.
3
4 • SELECT
5     pizzas.size,
6     COUNT(order_details.order_details_id) AS order_count
7 FROM
8     pizzas
9     JOIN
10    order_details ON pizzas.pizza_id = order_details.pizza_id
11 GROUP BY pizzas.size
12 ORDER BY order_count DESC;
```



Result Grid | Filter Rows: Export:

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

Q5. List the top 5 most ordered pizza types along with their quantities.

```
4      -- Q5 List the top 5 most ordered pizza types along with their quantities.  
5  
6 • SELECT  
7      pizza_types.name, SUM(order_details.quantity) AS quantity  
8  FROM  
9      pizza_types  
10     JOIN  
11      pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
12     JOIN  
13      order_details ON order_details.pizza_id = pizzas.pizza_id  
14  GROUP BY pizza_types.name  
15  ORDER BY quantity DESC  
16  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

Q6 . Join the necessary tables to find the total quantity of each pizza category ordered.

```
1 -- Join the necessary tables to find the total quantity of each pizza category ordered.  
2 • SELECT  
3     pizza_types.category,  
4     SUM(order_details.quantity) AS quantity  
5 FROM  
6     pizza_types  
7     JOIN  
8     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
9     JOIN  
10    order_details ON order_details.pizza_id = pizzas.pizza_id  
11   GROUP BY pizza_types.category  
12   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.

```
1 -- Determine the distribution of orders by hour of the day.  
2 • SELECT  
3     hour(orders_time) AS hour, COUNT(order_id) AS order_count  
4 FROM  
5     orders  
6 GROUP BY HOUR(orders_time);
```

-- JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
1      -- Join relevant tables to find the category-wise distribution of pizzas.  
2  
3 •  SELECT  
4      Category, COUNT(name)  
5  FROM  
6      pizza_types  
7  GROUP BY CATEGORY;
```

Result Grid | Filter Rows: Export:

	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.

```
1      -- Group the orders by date and calculate the average number of pizzas ordered per day.  
2  
3 •  select  
4      round(avg(QUANTITY),0) as avg_pizza_ordered_per_day  
5      from  
6      (select orders.date, Sum ( order_details.quantity) as quantity  
7      from orders join order_details on orders.order_id = order_details.order_id  
8      group by orders.order_date) as order_quantity;  
9
```

-- DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
1      -- Determine the top 3 most ordered pizza types based on revenue.  
2 •  SELECT  
3      pizza_types.name,  
4      SUM(order_details.quantity * pizzas.price) AS revenue  
5  FROM  
6      pizza_types  
7      JOIN  
8      pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
9      JOIN  
10     order_details ON order_details.pizza_id = pizzas.pizza_id  
11    GROUP BY pizza_types.name  
12    ORDER BY revenue DESC  
13    LIMIT 3;
```

Result Grid | Filter Rows: Export:

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