

Pizza Business Insights

A Data-Driven Approach with SQL and Power BI

...

About me

My name is Mohammed Mirzan. I have a strong interest in Data science and Machine learning, with experience in SQL, Power BI, Tabulae, and Excel. I am passionate about using data-driven insights to solve real-world problems and enjoy leveraging these tools to uncover meaningful trends and optimize decision-making.



Introduction

This project focuses on analyzing a pizza dataset using SQL for data querying and Power BI for visualization. The dataset contains information about pizza sales, including order details, pizza types, and sales trends.



...

Our Goals

- Analyze pizza sales data to identify top-selling pizzas and sales patterns.
- Understand customer preferences based on pizza types, sizes, and toppings.
- Uncover seasonal or time-based trends in pizza sales.
- Provide data-driven recommendations to optimize inventory and marketing strategies.
- Visualize key insights to support decision-making for business growth.



Dataset Summary

...

- Order Details Dataset
 - (48,620 rows, 4 columns)
 - Contains detailed order information, including order ID, pizza ID, quantity, and quantity of pizzas ordered.
- Orders Dataset
 - (21,350 rows and 3 columns)
 - Includes order-level data with unique order IDs, dates and times.



Dataset Summary Continued

- Pizza Types Dataset:
 - (32 rows, 4 columns)
 - Describes various pizza type IDs, names, categories, and ingredients to understand product offerings.
- Pizzas Dataset:
 - (96 rows and 4 columns)
 - Describes pizzas unique IDs, pizza type id size and price of the pizza



SQL Queries

...

Retrieve the total number of orders placed.

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

total_orders
21350

Calculate the total revenue generated from pizza sales.

```
SELECT
```

```
    ROUND(SUM(pizzas.price * orders_details.quantity),  
          0) AS Grand_total
```

```
FROM
```

```
pizzas
```

```
JOIN
```

```
orders_details ON pizzas.pizza_id = orders_details.pizza_id
```

Grand_total
817860

...

Identify the highest-priced pizza.

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

name	size	price
The Greek Pizza	XXL	35.95

...

Identify the most common pizza size ordered

```
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzas
    JOIN
        orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY size
ORDER BY order_count DESC
LIMIT 1;
```

size	order_count
L	18526

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

...

SELECT

```
    pizza_types.name,  
    SUM(orders_details.quantity) AS total_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.name  
ORDER BY total_quantity DESC  
LIMIT 5;
```

name	total_quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pi...	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(orders_details.quantity) AS quantity  
FROM  
    pizza_types  
        JOIN  
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
        JOIN  
    orders_details ON pizzas.pizza_id = orders_details.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  
ORDER BY order_count DESC
```

hour	order_count
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642
14	1472
15	1468
11	1231
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) AS total_pizza_types  
FROM  
    pizza_types  
GROUP BY category
```

category	total_pizza_types
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(total_orders), 0) AS average_pizza_ordered_per_day
```

FROM

(SELECT

```
orders.order_date,
```

```
SUM(orders_details.quantity) AS total_orders
```

FROM

```
orders
```

```
JOIN orders_details ON orders.order_id = orders_details.order_id
```

```
GROUP BY order_date) AS order_quantity;
```

average_pizza_ordered_per_day

138

...

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

SELECT

```
    pizza_types.name,  
    SUM(pizzas.price * orders_details.quantity) AS revenue
```

FROM

```
pizzas
```

JOIN

```
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
```

JOIN

```
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
```

GROUP BY pizza_types.name

ORDER BY revenue **DESC**

LIMIT 3;

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

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(orders_details.quantity*pizzas.price),2) as total_sales  
from orders_details join pizzas  
on pizzas.pizza_id = orders_details.pizza_id)*100,2) as revenue_percentage  
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 category  
order by revenue_percentage desc
```

category	revenue_percentage
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

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(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;
```

order_date	cum_revenue
2015-01-01	2713.850000000004
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
2015-12-22	803171.6
2015-12-23	805415.9
2015-12-24	807553.75
2015-12-26	809196.8
2015-12-27	810615.8
2015-12-28	812253
2015-12-29	813606.25
2015-12-30	814944.05
2015-12-31	817860.05

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
order by revenue desc) as a) as b
where rn<=3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pi...	42768
The California Chicken Pi...	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	32265.7000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

Power Bi Visualization

- The Power BI visualizations provide a comprehensive overview of pizza sales and customer preferences.
- The Sales Overview Dashboard highlights total revenue and order counts over time, while a line chart identifies the monthly pizza quantities.
- Additionally, a pie chart displays category with sum of quantity , and a slice filters the pizza size and quarterly year.
- Together, these visualizations deliver valuable insights for strategic decision-making in the pizza business.



Power Bi Dashboard



817.86K

Total Revenue

21K

Order Counts

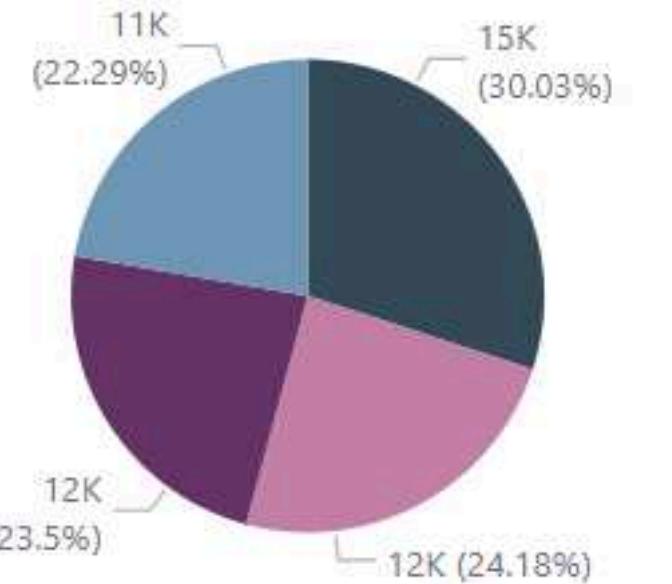
50K

Quantity

Quantity by Month



Sum of quantity by Category



- Category
- Classic
 - Supreme
 - Veggie
 - Chicken

Summary

At 4392, July had the highest Quantity and was 13.11% higher than October, which had the lowest Quantity at 3883.

July accounted for 8.86% of Quantity.

Across all 12 Month, Quantity ranged from 3883 to 4392.

Pizza Type ID	Category	Total Revenue	Quantity
bbq_ckn	Chicken	42,768.00	2432
cali_ckn	Chicken	41,409.50	2370
ckn_alfredo	Chicken	16,900.25	987
ckn_pesto	Chicken	16,701.75	973
southw_ckn	Chicken	34,705.75	1917
thai_ckn	Chicken	43,434.25	2371
big_meat	Classic	22,968.00	1914
classic_dlx	Classic	38,180.50	2453
hawaiian	Classic	32,273.25	2422
ital_cpcllo	Classic	25,094.00	1438
napolitana	Classic	24,087.00	1464
pep_msh_pep	Classic	18,834.50	1359
pepperoni	Classic	30,161.75	2418
the_greek	Classic	28,454.10	1420
brie_carre	Supreme	11,588.50	490
calabrese	Supreme	15,934.25	937
ital_supr	Supreme	33,476.75	1884
peppr_salami	Supreme	25,529.00	1446
prsc_argla	Supreme	24,193.25	1457
sicilian	Supreme	30,940.50	1938
soppressata	Supreme	16,425.75	961
spicy_ital	Supreme	34,831.25	1924
spinach_supr	Supreme	15,277.75	950
five_cheese	Veggie	26,066.50	1409
Total		817,860.05	49574

Conclusion

The analysis of the pizza dataset has provided valuable insights into sales trends, customer preferences, and product performance. By leveraging SQL for data extraction and Power BI for visualization, we identified top-selling pizzas and seasonal patterns that can inform inventory and marketing strategies. This data-driven approach enables the pizza business to optimize operations, enhance customer satisfaction, and drive growth. Moving forward, continuous analysis of sales data will be essential for adapting to market trends and improving overall business performance.



...

Thank You!

