

Pizza Sales

Data

(Dashboard creation in Microsoft Excel and
verification of the dashboard using queries in
Microsoft SQL Server)

Problem Statement

As a data analyst, I have been tasked with creating a comprehensive pizza sales dashboard to provide actionable insights and improve decision-making for a pizza company. The company's management recognizes the need to leverage data-driven strategies to optimize sales performance and enhance customer satisfaction. The goal is to develop a user-friendly and visually appealing pizza sales dashboard .By leveraging data visualization techniques, the dashboard will enable stakeholders to track sales performance, identify popular pizza flavours and toppings, analyse customer preferences and identify opportunities for growth and improvement.

KPI requirement

We need to analyse key indicators for our pizza sales data to gain insights into our business performance.

Specifically we want to calculate the following metrics:-

1. **Total revenue**: sum of total price of all the pizza orders
2. **Average Order value**: average amount spent per order, calculated by dividing the total revenue by the total number of orders
3. **Total pizzas sold**: sum of the quantities of all the pizza sold
4. **Total orders**: total number of orders placed
5. **Average pizzas per order**: average number of pizzas sold per order, calculated by dividing the total number of pizzas sold by the total number of orders

Charts requirement

We would like to visualize various aspects of our pizza sales data to gain insights and understand key trends.

We have identified the following requirements for creating charts:-

1. **Daily trend for total orders** : created a bar chart that displays the daily trend of total orders over a specific time period. This chart will help us identify any fluctuations in order volumes on a daily basis.
2. **Hourly trend for total orders**: created a line chart that displays the hourly trend of total orders throughout the day. This chart will allow us to identify peak hours or periods of high order activity.
3. **Percentage of sales by pizza category**: created a doughnut chart that shows the distribution of sales across different pizza categories which provides an insight into the popularity of various pizza categories and their contribution to overall sales.
4. **Percentage of sales by pizza size**: generated a pie chart that represents the percentage of sales by different pizza sizes which helps to understand customer preferences for pizza sizes and their impact on sales.
5. **Total pizzas sold by pizza category**: created a funnel chart that represents the total number of pizzas sold for each pizza category. This chart will allow us to compare the sales performance of different pizza categories.
6. **Top 5 best sellers by total pizzas sold**: Generated a stacked bar chart highlighting the top 5 best – selling pizzas based on the total number of pizzas sold which identified the most popular pizza options.
7. **Bottom 5 worst sellers by total pizzas sold**: Generated a bar chart showing the bottom 5 worst – selling pizzas based on the total number of pizzas sold which identified the less popular pizza options.

PIZZA SALES

Total Revenue
\$8,17,860

Total Orders
21350

Total Pizza sold
49574

Average Order Value
\$38.31

Average pizza per order
2.32

Busiest Days and Times

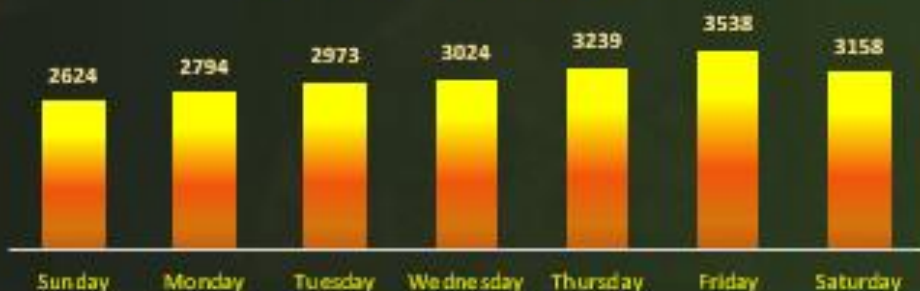
DAYS

Orders are highest on Friday and Saturday

TIMES

Maximum orders are placed from 12pm-1pm and 4pm-7pm

Daily Trend for Total Order



Hourly Trend for Total Order



Sales by Category & Size

CATEGORY

Classic category contributes to maximum sales(26.91%)

SIZE

Large pizza size contributes to maximum sales(45.89%)

Percentage of sales by pizza category



Percentage of sales by pizza size



Total pizza sold by pizza category



Best and Worst sellers

Best ones

The Classic Deluxe and The Barbecue Chicken are the bestsellers and revenue

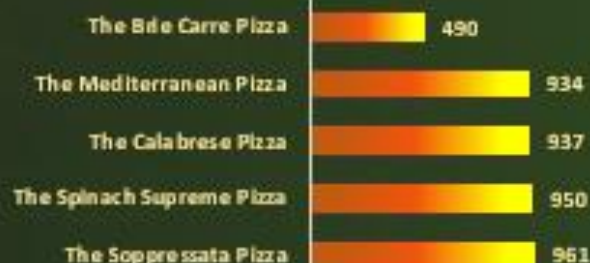
Worst ones

The Brie Carre Pizza is at the bottom in both orders and revenue

Top 5 best selling pizza



Bottom 5 worst selling pizza



order_date

All Periods

MONTHS

2015

JAN

FEB

MAR

APR

MAY

JUN

Problem statement

As a data analyst, my objective is to verify the accuracy and integrity of the pizza sales dashboard that I have made for the pizza company using SQL queries. It is crucial to ensure that the data is reliable and trustworthy to support informed decision-making and analysis. However, there are concerns regarding potential data inconsistencies, errors, or discrepancies within the database that may impact the validity of the sales data. I will import the pizza sales dataset in Microsoft SQL Server and verify the values are correct or not so that the user should only get the correct insights.

1. What is the Total Revenue generated?

```
SELECT SUM(total price) AS Total_Revenue FROM pizzadb.dbo.pizza_sales ;
```

Results		Messages	
	Total_Revenue		
1	817860.05083847		

2. What is the Average Order Value?

```
SELECT (SUM(total_price) / COUNT(DISTINCT order_id)) AS Avg_order_Value  
FROM pizzadb.dbo.pizza_sales;
```

Results		Messages	
	Avg_order_Value		
1	38.3072623343546		

3. What is the Total number of Pizzas Sold?

```
SELECT SUM(quantity) AS Total_pizza_sold FROM pizzadb.dbo.pizza_sales;
```

Results		Messages	
	Total_pizza_sold		
1	49574		

4. What are the Total number Orders placed?

```
SELECT COUNT(DISTINCT order_id) AS Total_Orders FROM pizzadb.dbo.pizza_sales;
```

Results		Messages	
	Total_Orders		
1	21350		

5. What is the Average number of Pizzas Per Order?

```
SELECT CAST(CAST(SUM(quantity) AS DECIMAL(10,2)) /  
CAST(COUNT(DISTINCT order_id) AS DECIMAL(10,2)) AS DECIMAL(10,2)) AS Avg_Pizzas_per_order  
FROM pizzadb.dbo.pizza_sales;
```

Results		Messages	
		Avg_Pizzas_per_order	
1		2.32	

6. Daily Trend for Total Orders

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders
FROM pizzadb.dbo.pizza_sales
GROUP BY DATENAME(DW, order_date);
```

Results			Messages		
	order_day	total_orders			
1	Saturday	3158			
2	Wednesday	3024			
3	Monday	2794			
4	Sunday	2624			
5	Friday	3538			
6	Thursday	3239			
7	Tuesday	2973			

7. Hourly Trend for Orders

```
SELECT DATEPART(HOUR, order_time) as order_hours, COUNT(DISTINCT order_id) as total_orders FROM  
pizzadb.dbo.pizza_sales  
GROUP BY DATEPART(HOUR, order_time) ORDER BY DATEPART(HOUR, order_time);
```

Results Messages		
	order_hours	total_orders
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28

8. What is the total percentage of Sales by Pizza Category?

```
SELECT pizza_category, CAST(SUM(total_price) AS DECIMAL(10,2)) as total_revenue,  
CAST(SUM(total_price) * 100 / (SELECT SUM(total_price) from pizzadb.dbo.pizza_sales) AS DECIMAL(10,2))  
AS PCT  
FROM pizzadb.dbo.pizza_sales GROUP BY pizza_category;
```

Results		Messages	
	pizza_category	total_revenue	PCT
1	Classic	220053.10	26.91
2	Chicken	195919.50	23.96
3	Veggie	193690.45	23.68
4	Supreme	208197.00	25.46

9. What is the total percentage of Sales by Pizza Size?

```
SELECT pizza_size, CAST(SUM(total_price) AS DECIMAL(10,2)) as total_revenue,  
CAST(SUM(total_price) * 100 / (SELECT SUM(total_price) from pizzadb.dbo.pizza_sales) AS DECIMAL(10,2))  
AS PCT FROM pizzadb.dbo.pizza_sales  
GROUP BY pizza_size ORDER BY pizza_size;
```

	pizza_size	total_revenue	PCT
1	L	375318.70	45.89
2	M	249382.25	30.49
3	S	178076.50	21.77
4	XL	14076.00	1.72
5	XXL	1006.60	0.12

10. What is the Total number of Pizzas Sold by Pizza Category?

```
SELECT pizza_category, SUM(quantity) as Total_Quantity_Sold FROM pizzadb.dbo.pizza_sales  
WHERE MONTH(order_date) = 2  
GROUP BY pizza_category ORDER BY Total_Quantity_Sold DESC;
```

Results Messages		
	pizza_category	Total_Quantity_Sold
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

11. Top 5 Best Sellers by Total Pizzas Sold

SELECT Top 5 pizza_name, **SUM**(quantity) **AS** Total_Pizza_Sold **FROM** pizzadb.dbo.pizza_sales
GROUP BY pizza_name **ORDER BY** Total_Pizza_Sold **DESC**;

	pizza_name	Total_Pizza_Sold
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

12. Bottom 5 Worst Sellers by Total Pizzas Sold

```
SELECT TOP 5 pizza_name, SUM(quantity) AS Total_Pizza_Sold FROM pizzadb.dbo.pizza_sales  
GROUP BY pizza_name ORDER BY Total_Pizza_Sold ASC;
```

Results Messages		
	pizza_name	Total_Pizza_Sold
1	The Brie Carre Pizza	490
2	The Mediterranean Pizza	934
3	The Calabrese Pizza	937
4	The Spinach Supreme Pizza	950
5	The Soppressata Pizza	961

NOTE:

If you want to apply the Month, Quarter, Week filters to the above queries you can use WHERE clause. Follow some of below examples

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders
FROM pizzadb.dbo.pizza_sales
WHERE MONTH(order_date) = 1
GROUP BY DATENAME(DW, order_date)
```

***Here MONTH(order_date) = 1 indicates that the output is for the month of January. MONTH(order_date) = 4 indicates output for Month of April.**

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders
FROM pizzadb.dbo.pizza_sales
WHERE DATEPART(QUARTER, order_date) = 1
GROUP BY DATENAME(DW, order_date)
```

***Here DATEPART(QUARTER, order_date) = 1 indicates that the output is for the Quarter 1. MONTH(order_date) = 3 indicates output for Quarter 3.**