**Portfolio Project on**

**Pizza Sales Analysis using**

**SQL & Power BI**

**BY  
Jeevitha G.**

# Project Overview:

The Pizza Sales Analysis Project is designed to explore and interpret key performance indicators (KPIs) of pizza sales data through structured SQL queries and advanced Power BI visualizations. The project focuses on uncovering insights related to sales performance, customer purchasing trends, and product popularity across various categories and time periods. By combining SQL-based analytical computation with Power BI's dynamic visualization capabilities, this project provides actionable business intelligence that can help stakeholders optimize menu offerings, pricing strategies, and promotional campaigns.

# Objectives of the Project:

The primary objective of the Pizza Sales Analysis Project is to evaluate sales performance across multiple dimensions — revenue, order trends, category preferences, and time-based behavior. The project aims to support data-driven decision-making and operational efficiency improvements for business stakeholders. Key objectives include:

1) Identify the total revenue, total orders, and total pizzas sold to assess business performance.  
2) Analyze average order value and average pizzas sold per order.  
3) Examine daily and monthly order trends to identify peak sales periods.  
4) Evaluate the sales contribution by pizza category and pizza size.  
5) Determine the top and bottom performing pizzas by revenue, quantity, and total orders.  
6) Derive insights to support marketing, inventory, and production strategies.

# Dataset Overview and Description:

The dataset used in this project contains detailed transactional data for pizza sales. Each record represents a pizza sold within a specific order and includes attributes such as order date, time, category, size, quantity, and price. The data provides a comprehensive view of customer behavior and business performance.

Key Columns Include:

- pizza\_id: Unique identifier for each pizza item.  
- order\_id: Unique identifier for each customer order.  
- pizza\_name\_id: Encoded ID for each pizza type.  
- quantity: Number of pizzas sold per order.  
- order\_date and order\_time: Timestamp of the transaction.  
- unit\_price and total\_price: Price per unit and total transaction amount.  
- pizza\_size and pizza\_category: Attributes describing pizza size and type.  
- pizza\_ingredients and pizza\_name: Descriptive details of the pizza.

# Data Loading Process:

The dataset was imported into SQL Server Management Studio (SSMS) for data validation and aggregation. A new database named ‘pizza\_sales’ was created, and the CSV file was imported using the SQL Server Import Wizard. After confirming schema accuracy, SQL queries were developed to calculate KPIs and generate analytical summaries.

The cleaned SQL table was then connected to Power BI through the SQL Server connector. This direct connection allowed Power BI to refresh the data dynamically and perform further transformations within Power Query Editor.

# Data Preparation and Cleaning:

Data preparation involved several transformation and validation steps to ensure data accuracy and consistency. These steps were executed both in SQL Server and Power BI Power Query Editor.

Key steps included:  
- Removing duplicate and null records.  
- Ensuring correct data types for date, time, and numeric fields.  
- Creating additional calculated columns for Month Name, Day Name, and Day Number.  
- Verifying category and size consistency to prevent data mismatches.  
- Adding new DAX measures to compute sales-related KPIs.

# SQL Analysis and Key Metrics (KPIs):

SQL queries were used to compute essential KPIs that serve as the foundation for visualization in Power BI. The following metrics were calculated:

1) Total Revenue: SUM(total\_price)  
2) Average Order Value: SUM(total\_price) / COUNT(DISTINCT order\_id)  
3) Total Pizzas Sold: SUM(quantity)  
4) Total Orders: COUNT(DISTINCT order\_id)  
5) Average Pizzas per Order: SUM(quantity) / COUNT(DISTINCT order\_id)

These queries enabled efficient aggregation and data retrieval, setting the groundwork for Power BI dashboards.

# Power BI Dashboard Development:

The Power BI dashboard was designed to visually communicate the results of the SQL analysis through interactive and intuitive charts. The dashboard includes multiple sections focusing on order trends, category distribution, and performance analysis.

Visual Components Include:  
1) KPI Cards: Displaying Total Revenue, Total Orders, Total Pizzas Sold, Average Order Value, and Average Pizzas per Order.  
2) Daily Trend (Bar Chart): Shows order volumes across weekdays, highlighting increased activity on weekends.  
3) Monthly Trend (Line Chart): Depicts order trends throughout the year, identifying July and January as high-performing months.  
4) Category-wise Sales (Pie Chart): Represents the sales distribution among pizza categories.  
5) Size-wise Sales (Pie Chart): Displays contribution to sales by different pizza sizes.  
6) Top & Bottom 5 Pizzas (Bar Charts): Highlights the most and least performing pizzas by revenue, quantity, and order count.  
7) Funnel Chart: Illustrates total pizzas sold by category.

# Analytical Findings and Insights:

Based on SQL computations and Power BI visualizations, the following key insights were derived:

- The Thai Chicken Pizza generates the maximum revenue among all pizza types.  
- The Classic Deluxe Pizza records both the highest quantity sold and the highest number of total orders.  
- Orders peak on weekends, particularly during Friday and Saturday evenings.  
- The months of July and January witness the highest number of total orders.  
- The Classic category contributes the most to total sales and orders.  
- Large-sized pizzas dominate sales, generating the highest revenue share.

# Key Business Insights and Recommendations:

1) \*\*Menu Optimization:\*\* The popularity of Thai Chicken and Classic Deluxe pizzas indicates potential for premium pricing and promotional bundling.  
2) \*\*Inventory Planning:\*\* High weekend and monthly peaks suggest increased stock requirements for popular ingredients during those periods.  
3) \*\*Category Focus:\*\* The dominance of the Classic category implies consistent customer preference; introducing new variants under this category may enhance sales.  
4) \*\*Size Strategy:\*\* Large-sized pizzas are the most profitable, and combo offers could further boost order volume.  
5) \*\*Marketing Timing:\*\* Marketing campaigns should target weekend evenings and high-performing months to maximize engagement.

# Visual Dashboard & SQL Output Screenshots:

Below are the placeholders for inserting SQL query outputs and Power BI dashboards.  
You can add screenshots in these sections using Word's 'Insert → Pictures' feature.

## a) SQL Queries and Results

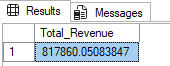
**KPIs**

We need to analyze 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:** The sum of the total price of all pizza orders.

SELECT SUM(total\_price) as Total\_Revenue

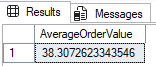
FROM pizza\_sales



1. **Average Order Value:** The average amount spent per order, calculated by dividing the total revenue by the total number of orders.

SELECT SUM(total\_price)/COUNT(DISTINCT order\_id) as AverageOrderValue

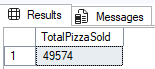
FROM pizza\_sales

****

1. **Total Pizzas Sold:** The sum of the quantities of all pizzas sold.

SELECT SUM(quantity) as TotalPizzaSold

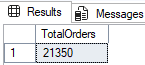
FROM pizza\_sales

****

1. **Total Orders:** The total number of orders placed.

SELECT COUNT(DISTINCT order\_id) as TotalOrders

FROM pizza\_sales

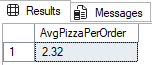
****

1. **Average Pizzas Per Order:** The average number of pizzas sold per order, calculated by dividing the total number of pizzas sold by the total number of orders.

SELECT CAST(CAST(SUM(quantity) AS DECIMAL(10,2))/CAST(COUNT(DISTINCT order\_id) AS DECIMAL(10,2)) AS DECIMAL(10,2))

AS AvgPizzaPerOrder

FROM pizza\_sales

****

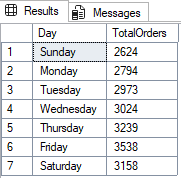
**Daily Trend For Total Orders**

SELECT DATENAME(dw,order\_date) Day, COUNT(DISTINCT order\_id) TotalOrders

FROM pizza\_sales

GROUP BY DATENAME(dw,order\_date),DATEPART(dw,order\_date)

ORDER BY DATEPART(dw,order\_date)

****

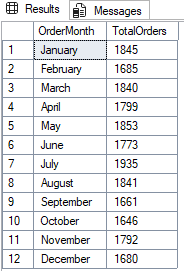
**Monthly Trend for Total Orders**

SELECT DATENAME(MONTH,order\_date),COUNT(DISTINCT order\_id)

FROM pizza\_sales

GROUP BY DATENAME(MONTH,order\_date), DATEPART(MONTH,order\_date)

ORDER BY DATEPART(MONTH,order\_date)

****

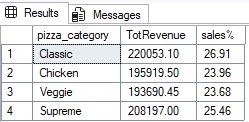
**Sales% By Pizza Category**

SELECT pizza\_category,

CAST(SUM(total\_price) AS DECIMAL(10,2)) TotRevenue,

CAST(SUM(total\_price) \*100/(SELECT SUM(total\_price) FROM pizza\_sales) AS DECIMAL(10,2)) AS [sales%] FROM pizza\_sales

GROUP BY pizza\_category

****

**Sales % BY Pizza Size**

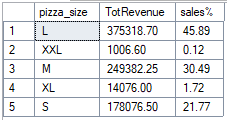
SELECT pizza\_size,

CAST(SUM(total\_price) AS DECIMAL(10,2)) TotRevenue,

CAST(SUM(total\_price) \*100/(SELECT SUM(total\_price) FROM pizza\_sales) AS DECIMAL(10,2)) AS [sales%]

FROM pizza\_sales

GROUP BY pizza\_size

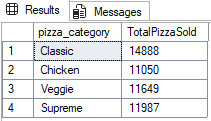
****

**Total Pizzas Sold By Pizza Category**

SELECT pizza\_category,SUM(quantity) as TotalPizzaSold

FROM pizza\_sales

GROUP BY pizza\_category

****

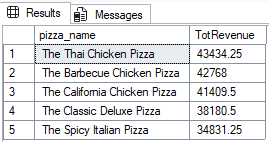
**Top 5 Pizzas By Revenue**

SELECT TOP 5 pizza\_name, SUM(total\_price) AS TotRevenue

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY SUM(total\_price) DESC

****

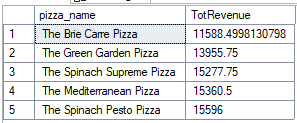
**Bottom 5 Pizzas By Revenue**

SELECT TOP 5 pizza\_name, SUM(total\_price) AS TotRevenue

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY SUM(total\_price)

****

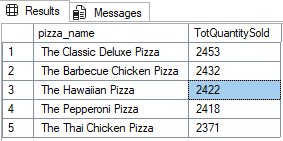
**Top 5 Pizzas By Quantity Sold :**

SELECT TOP 5 pizza\_name, SUM(quantity) AS TotQuantitySold

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY SUM(quantity) DESC

****

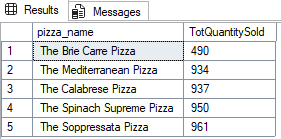
**Bottom 5 Pizzas By Quantity Sold**

SELECT TOP 5 pizza\_name, SUM(quantity) AS TotQuantitySold

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY SUM(quantity)

****

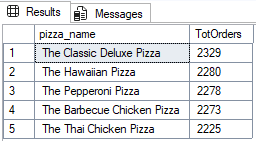
**Top 5 Pizzas By Total Orders**

SELECT TOP 5 pizza\_name, COUNT(DISTINCT order\_id) AS TotOrders

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY COUNT(DISTINCT order\_id) DESC

****

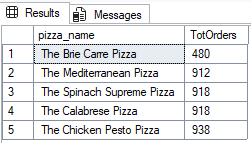
**Bottom 5 Pizzas By Total Orders**

SELECT TOP 5 pizza\_name, COUNT(DISTINCT order\_id) AS TotOrders

FROM pizza\_sales

GROUP BY pizza\_name

ORDER BY COUNT(DISTINCT order\_id)

****

**Identify Peak Sales Hours**

SELECT DATEPART(HOUR,order\_time) OrderHour,

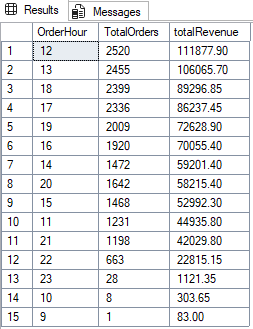
COUNT(DISTINCT order\_id) TotalOrders,

CAST(SUM(total\_price) AS DECIMAL(10,2)) totalRevenue

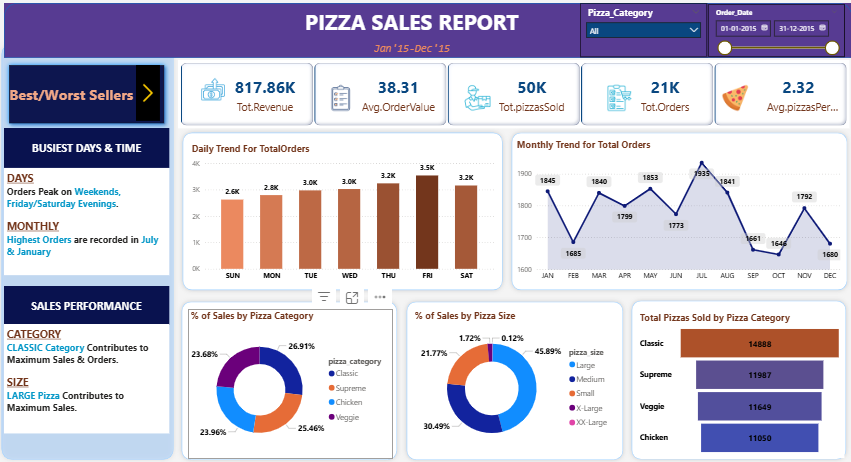
FROM pizza\_sales

GROUP BY DATEPART(HOUR,order\_time)

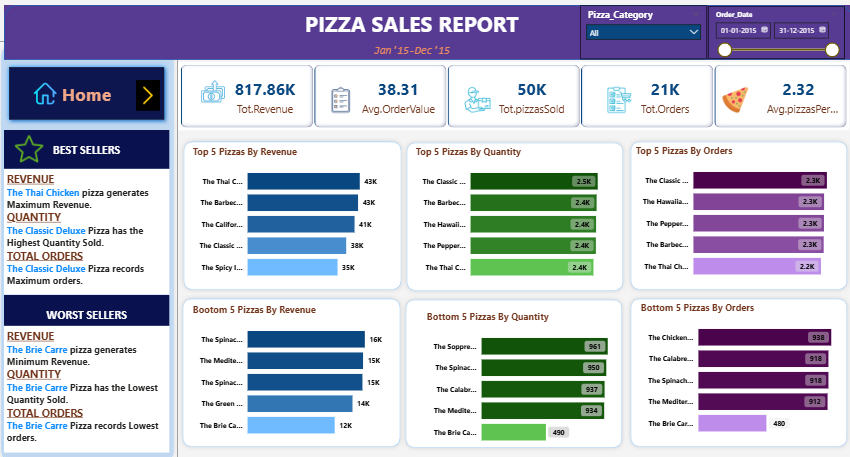
ORDER BY totalRevenue DESC

****

## b) Power BI Dashboard – Overview



**c) Power BI Dashboard – Top & Bottom Performers**



# Significance of the Project:

The Pizza Sales Analysis Project demonstrates the practical application of SQL and Power BI in real-world business analytics. By translating raw transactional data into actionable insights, the project highlights how data visualization and performance metrics can support operational decision-making. The project underscores the value of integrating analytical tools to optimize menu planning, streamline operations, and improve profitability.

From a business standpoint, this project helps stakeholders understand sales distribution, identify high-performing products, and strategize future promotions effectively. The findings can be leveraged by restaurant managers, supply chain teams, and marketing departments to make evidence-based decisions.

# Conclusion:

The Pizza Sales Analysis Project successfully utilizes SQL and Power BI to extract, analyze, and visualize key insights from sales data. Through this analysis, meaningful patterns related to revenue, category performance, and consumer behavior were identified. The comprehensive dashboard provides a clear understanding of business performance, enabling data-driven strategies for future growth. This project highlights the significance of combining database querying with visualization tools to transform data into powerful business intelligence.