



Pirra

SALES ANALYSIS

REPORT



BY

Alipriya

Ghosh

alipriyaghosh17@gmail.com

Software Utilized

- Microsoft Excel
 - 1. Imported raw dataset into Power Query.
 - 2. Cleaned data (removed errors, duplicates, and unnecessary columns).
 - 3. Transformed Order Date column into YYYY-MM-DD format (1996-07-21) for SQL compatibility.
 - 4. Exported the cleaned dataset as a CSV file for use in SQL queries.

• MySQL

1. Imported the cleaned CSV file into SQL.
2. Calculated Key Performance Indicators (KPIs) such as:
 - Total Sales
 - Number of Orders
 - Average Order Value
 - Customer Count / Retention
3. Generated Monthly Trends (grouped by MONTH(Order_Date)).
4. Generated Daily Trends (grouped by DAY(Order_Date)).
5. Identified Top 5 Best-Selling Pizzas (ranked by sales quantity/revenue).
6. Identified Bottom 5 Least-Selling Pizzas (ranked by sales quantity/revenue).
7. Stored queries for further visualization/reporting.

• Microsoft Power BI

1. Imported SQL data into Power BI.
2. Calculated KPIs using DAX formulas (e.g., Total Sales, Avg Order Value, Customer Count).
3. Formatted the Date column into MM-DD-YYYY and also extracted:
 - Month Number (for chronological order in graphs)
 - Month Name (for readability in visuals)
 - Day Number & Day Name (for daily trend charts).
4. Applied these fields in charts to build:
 - Monthly Sales Trend Graph
 - Daily Sales Trend Graph
 - Top 5 & Bottom 5 Pizza Charts
5. Designed an interactive dashboard for business insights.

KPI's

MySQL

1. Calculate the Total Revenue.

```
Select sum(total_price) as Total_revenue  
from pizza_sales;
```

Total_revenue
817860.049999993

2. Average Order Value.

```
With total_orders as (  
Select count( distinct order_id) as  
tcount from pizza_sales),  
Total_revenue as(  
Select sum(total_price) as Total_rev  
from pizza_sales),  
Average_order as  
(Select round(Total_rev/tcount,2 )as  
avg_order from  
total_orders,Total_revenue )  
SELECT *  
FROM average_order;
```

avg_order
38.31



3. Total Pizzas Sold.

```
Select Sum(quantity) as total_pizza_sold  
from pizza_sales;
```

total_pizza_sold
49574

4. Total Orders.

```
Select count(distinct order_id) as  
total_no_of_order  
from pizza_sales;
```

total_no_of_order
21350

5. Average Pizzas Per Order.

```
With total_count as( select sum(quantity) as  
total_quantity from pizza_sales),  
Total_count2 as ( Select count(distinct  
order_id) as total_c from pizza_sales) ,  
AvgO as ( select total_quantity/total_c as  
p_sold from total_count ,total_count2)  
select p_sold  
from avgO ;
```

p_sold
2.3220



CHARTS

6. Daily Trend for Total Orders

```
SELECT  
DAYOFWEEK(order_date) AS day_number,  
DAYNAME(order_date) AS day_name,  
COUNT(DISTINCT order_id) AS total_orders  
FROM pizza_sales  
GROUP BY day_number, day_name  
ORDER BY day_number;
```

	day_number	day_name	total_orders
▶	1	Sunday	2624
	2	Monday	2794
	3	Tuesday	2973
	4	Wednesday	3024
	5	Thursday	3239
	6	Friday	3538
	7	Saturday	3158



7. Monthly Trend for Orders.

```
SELECT  
MONTH(order_date) AS month_number,  
MONTHNAME(order_date) AS month_name,  
COUNT(DISTINCT order_id) AS total_orders  
FROM pizza_sales  
GROUP BY month_number, month_name  
ORDER BY month_number;
```

month_number	month_name	total_orders
3	March	1840
4	April	1799
5	May	1853
6	June	1773
7	July	1935
8	August	1841
9	September	1661
10	October	1646
11	November	1792
12	December	1680



8. % of Sales by Pizza Category

```
With Pizza_category as (
Select pizza_category, sum(total_price) as
Tp from pizza_sales
group by pizza_category ) ,
total_sales as ( Select sum(total_price) as
Ts from pizza_sales) ,
percentage_sales as ( select pizza_category,
(Tp/Ts)*100 as ps from Pizza_category
,total_sales)
Select pizza_category , ps as sales
from percentage_sales ;
```

pizza_category	sales
Classic	26.905960255669903
Veggie	23.682590927384783
Supreme	25.45631126009884
Chicken	23.955137556847493



9.% of Sales by Pizza Size

```
With pizza_size as ( Select pizza_size,
sum(total_price) as Pz
from pizza_sales
group by pizza_size) ,
total_sales as ( Select sum(total_price) as
Ts from pizza_sales) ,
percentage_sales as ( select pizza_size ,
(Pz/Ts)*100 as ps
from pizza_size ,total_sales)
Select pizza_size , ps as percentage_sales
from percentage_sales ;
```

pizza_size	percentage_sales
Large	45.890332948774294
Medium	30.492044451859723
Small	21.773468455880682
Extra large	1.7210768517181052
Extra extra large	0.12307729176892906



10 Total Pizza Sold by Category

```
SELECT pizza_category, SUM(quantity) as  
Total_Quantity_Sold  
FROM pizza_sales  
GROUP BY pizza_category  
ORDER BY Total_Quantity_Sold DESC;
```

pizza_category	Total_Quantity_Sold
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

11. Top 5 Pizzas by Revenue

```
SELECT pizza_category, SUM(quantity) as  
Total_Quantity_Sold  
FROM pizza_sales  
GROUP BY pizza_category  
ORDER BY Total_Quantity_Sold DESC;
```

pizza_name	total_price
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Spicy Italian Pizza	34831.25



12. Bottom 5 Pizzas by Revenue

```
Select pizza_name , sum(total_price) as  
total_price  
from pizza_sales group by pizza_name  
order by total_price limit 5;
```

pizza_name	total_price
The Brie Carre Pizza	11588.4999999999
The Green Garden Pizza	13955.75
The Spinach Supreme Pizza	15277.75
The Mediterranean Pizza	15360.5
The Spinach Pesto Pizza	15596

13. Top 5 Pizzas by Quantity.

```
Select pizza_name , sum(quantity) as  
qnt_ordered  
from pizza_sales  
group by pizza_name  
order by qnt_ordered desc  
limit 5;
```

pizza_name	qnt_ordered
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371



14. Bottom 5 Pizzas by Quantity

```
Select pizza_name ,sum(quantity)as  
qnt_ordered  
from pizza_sales group by pizza_name  
order by qnt_ordered limit 5;
```

pizza_name	qnt_ordered
The Brie Carre Pizza	490
The Mediterranean Pizza	934
The Calabrese Pizza	937
The Spinach Supreme Pizza	950
The Soppressata Pizza	961

15. Top 5 by Total Orders

```
Select pizza_name , count(distinct order_id)  
as total_ordered  
from pizza_sales group by pizza_name  
order by total_ordered desc limit 5;
```

pizza_name	total_ordered
The Classic Deluxe Pizza	2329
The Hawaiian Pizza	2280
The Pepperoni Pizza	2278
The Barbecue Chicken Pizza	2273
The Thai Chicken Pizza	2225



16. Bottom 5 by Total Orders.

```
Select pizza_name , count(distinct order_id)  
as total_ordered  
from pizza_sales  
group by pizza_name  
order by total_ordered asc  
limit 5;
```

pizza_name	total_ordered
The Brie Carre Pizza	480
The Mediterranean Pizza	912
The Calabrese Pizza	918
The Spinach Supreme Pizza	918
The Chicken Pesto Pizza	938



DAX

PowerBI

Total Order =

```
DISTINCTCOUNT(pizza_sales_a[order_id])
```

Total Revenue =

```
sum(pizza_sales_a[total_price])
```

Total Pizza Sold =

```
Sum(pizza_sales_a[quantity])
```

AVG Order Value = [Total Revenue]/[Total Order]

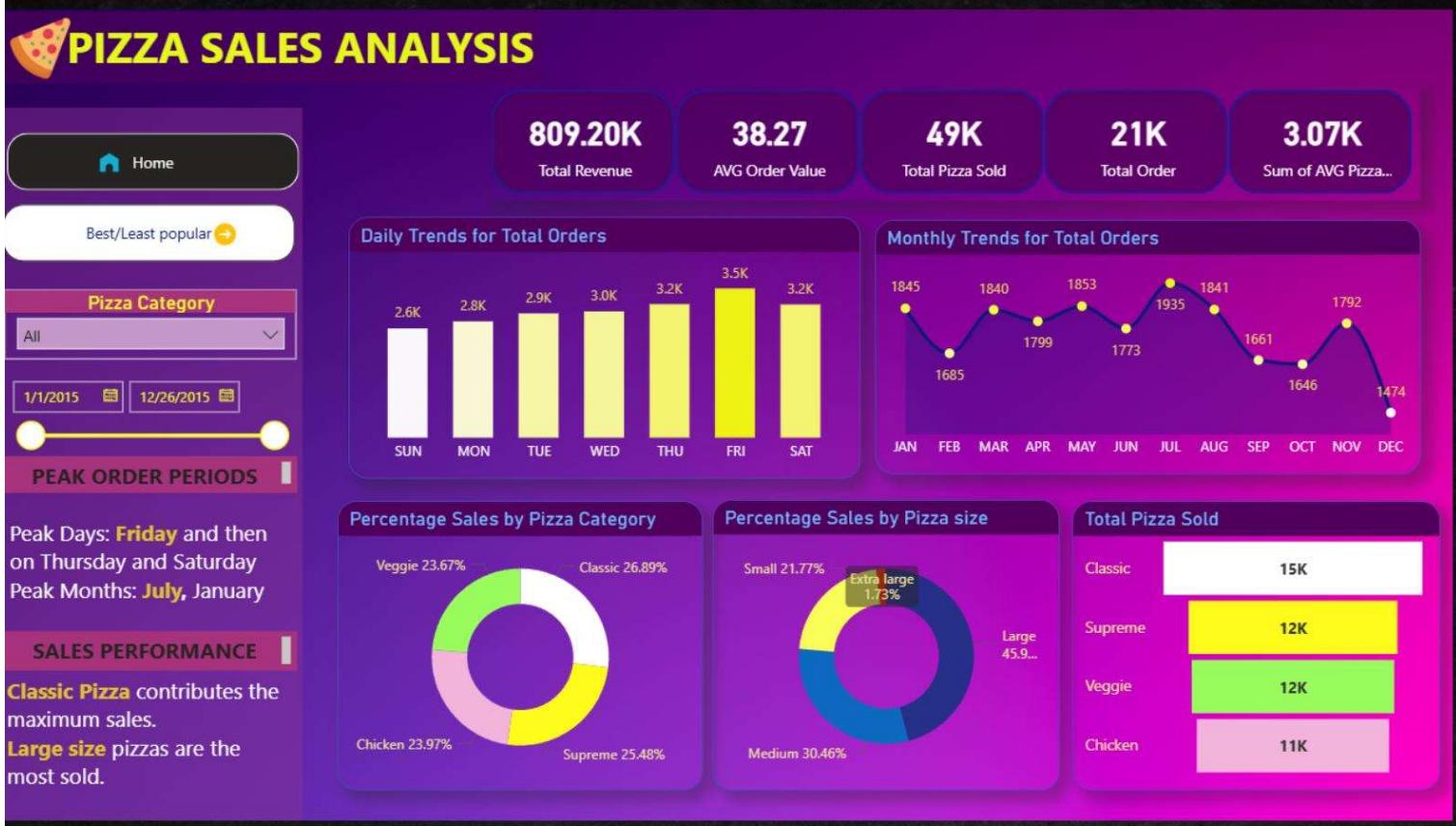
AVG Pizza Per Order = [Total Pizza Sold]/[Total Revenue]

Order Month =

```
UPPER(left(pizza_sales_a[Month Name],3))
```

Order Day =

```
UPPER(left(pizza_sales_a[Day Name],3))
```



Conclusion

- Peak Order Periods (Time-Series Insights)
- Temporal Trend: Highest demand observed on Fridays, followed by Thursdays and Saturdays.
- Seasonality: Sales spike in July and January.
- Sales Performance (Contribution Analysis)
- Category Contribution: Classic Pizza shows the maximum contribution to overall sales.
- Size Segmentation: Large-size pizzas dominate in terms of sales volume.
- Best Sellers (Revenue & Quantity Metrics)
- Revenue Driver: Thai Chicken Pizza delivers the highest revenue.
- Order Frequency: Classic Deluxe Pizza records the maximum order count.
- Least Popular (Low-Performance Outliers)
- Revenue Outlier: Brie Carre Pizza ranks lowest in revenue generation.
- Demand Outlier: Brie Carre Pizza also shows the minimum order frequency.

THE END