

# Pizza Sales Analysis with Microsoft SQL

## Creating Database:

```
create database Pizza_DB;
```

## Use the Database:

```
use Pizza_DB;
```

## Show the table:

```
select * from pizza_sales;
```

-- Total Revenue: the sum of total price of all pizza orders.

```
select ROUND(SUM(total_price), 2) as Total_revenue from pizza_sales;
```

Results		Messages	
		Total_revenue	
1		817860.05	

-- Total Orders: The total number of orders placed.

```
select Count(distinct order_id) as Total_Orders from pizza_sales;
```

		Total_Orders	
1		21350	

-- Total Pizzas Sold: The Sum of the Quantities of all pizza sold.

```
select sum(quantity) as Total_Pizzas_Sold from pizza_sales;
```

Results		Messages	
		Total_Pizzas_Sold	
1		49574	

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

```
select ROUND(SUM(total_price)/ (select count(distinct order_id)),3) as Average_Order_Value from pizza_sales;
```

Results		Messages	
		Average_Order_Value	
1		38.307	

-- Average Pizzas Per Order: The Average number of pizza sold per order, calculated by dividing the total number of pizza sold by the total number of order.

```
select cast(cast(sum(quantity)as decimal(10,2))/ COUNT(distinct order_id) as decimal(10,2))
as Average_Pizza_per_order

from pizza_sales;
```

Results Messages	
	Average_Pizza_per_order
1	2.32

### -- Daily Trend For Total Orders:

create a bar chart that displays the daily trend of total orders over a specific time period. This chart will help us to identify any patterns or fluctuation in order volume on a daily bases.

```
select DATENAME(DW, order_date) as Order_day,
count(distinct order_id) as Total_orders

from 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

### -- Monthly Trend For Total Orders:

create a line chart that illustrates the monthly trend of total orders throughout t. This chart will allow us to identify peak month or period of high order activity.

```
select DATENAME(MM, order_date) as Month_Name,
COUNT(distinct order_id) as Total_order

from pizza_sales

group by DATENAME(MM, order_date)

order by Total_order DESC;
```

	Month_Name	Total_order
1	July	1935
2	May	1853
3	January	1845
4	August	1841
5	March	1840
6	April	1799
7	November	1792
8	June	1773
9	February	1685
10	December	1680
11	September	1661
12	October	1646

### -- Percentage Of Sales By Pizza Category:

create a pie chart that shows the distribution of sales across different pizza categories. This chart will provide insight into the popularity of various pizza categories and their contribution to overall sales

```
select pizza_category, Round(SUM(total_price),2) as Total_revenue,
cast((SUM(total_price)/ (select sum(total_price) from pizza_sales))* 100 as decimal(10,2)) as PCT
from pizza_sales
group by pizza_category
order by PCT DESC;
```

	pizza_category	Total_revenue	PCT
1	Classic	220053.1	26.91
2	Supreme	208197	25.46
3	Chicken	195919.5	23.96
4	Veggie	193690.45	23.68

### --Percentage Of Sales By Pizza Size:

Generate a pie chart that represents the percentage of sales attributed to different pizza sizes. this chart will help us to understand customer preferences for pizza sizes and their impact on sales.

```
select pizza_size, round(sum(total_price),2) as Total_revenue,
cast(SUM(total_price)*100 / (select sum(total_price) from pizza_sales) as decimal(10,2)) as PCT
from pizza_sales
group by pizza_size
order by PCT DESC;
```

Results Messages			
	pizza_size	Total_revenue	PCT
1	L	375318.7	45.89
2	M	249382.25	30.49
3	S	178076.5	21.77
4	XL	14076	1.72
5	XXL	1006.6	0.12

### -- Total Pizzas Sold By Pizza Category:

Create a funnel chart that presents the total number of pizzas sold for each pizza category. This chart will allow us to compare the sales performance of different pizza categories.

```
select pizza_category, SUM(quantity) as Total_quantity_sold
```

```
from pizza_sales
```

```
group by pizza_category;
```

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

### -- Top 5 Best Sellers By Revenue, Total Quantity and Total Orders:

Create a bar chart highlighting the top 5 best-selling pizzas based on the total number of pizzas sold. This chart will help us identify the most popular pizza options.

```
select top 5 pizza_name, sum(quantity) as Total_quantity, count(distinct order_id) as Total_orders,
```

```
round(sum(total_price),2) as Total_revenue
```

```
from pizza_sales
```

```
group by pizza_name
```



```
order by Total_revenue DESC;
```

Results Messages				
	pizza_name	Total_quantity	Total_orders	Total_revenue
1	The Thai Chicken Pizza	2371	2225	43434.25
2	The Barbecue Chicken Pizza	2432	2273	42768
3	The California Chicken Pizza	2370	2197	41409.5
4	The Classic Deluxe Pizza	2453	2329	38180.5
5	The Spicy Italian Pizza	1924	1822	34831.25

### -- Bottom 5 Worst Sellers By Revenue, Total Quantity and Total Orders:

Create a bar chart showcasing the bottom 5 worst-selling pizzas based on the total number of pizza sold. This chart will enable us to identify the underperforming or less popular pizza options

```
select top 5 pizza_name, sum(quantity) as Total_quantity, count(distinct order_id) as Total_orders,
round(sum(total_price),2) as Total_revenue
from pizza_sales
group by pizza_name
order by Total_revenue;
```

<div><div> Results</div><div> Messages</div></div>				
	pizza_name	Total_quantity	Total_orders	Total_revenue
1	The Brie Carre Pizza	490	480	11588.5
2	The Green Garden Pizza	997	976	13955.75
3	The Spinach Supreme Pizza	950	918	15277.75
4	The Mediterranean Pizza	934	912	15360.5
5	The Spinach Pesto Pizza	970	945	15596