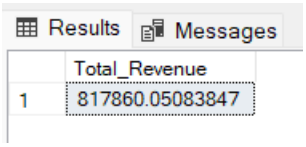
PIZZA SALES SQL QUERIES

A. KPI's

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

```
SELECT SUM(total_price) AS Total_Revenue FROM pizza_sales;
```

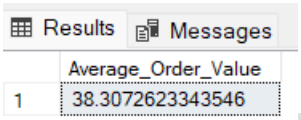


Results | Messages

	Total_Revenue
1	817860.05083847

2. 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 Average_Order_Value FROM
pizza_sales;
```

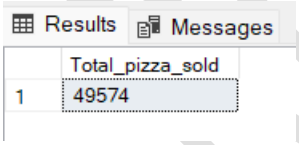


Results | Messages

	Average_Order_Value
1	38.3072623343546

3. Total Pizza Sold: The sum of the quantities of all pizza sold.

```
SELECT SUM(quantity) AS Total_pizza_sold FROM pizza_sales;
```

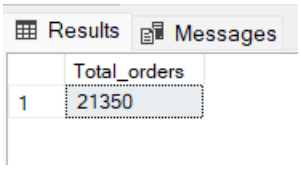


Results | Messages

	Total_pizza_sold
1	49574

4. Total Orders: The total numbers of orders placed.

```
SELECT COUNT(DISTINCT order_id) AS Total_orders FROM pizza_sales;
```



Results | Messages

	Total_orders
1	21350

5. 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  FROM pizza_sales;
```

Results | Messages

	Avg_pizzas_per_order
1	2.32

B. CHARTS REQUIREMENT

1. 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 identify any patterns or fluctuations in order volumes on a daily basis.

```
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

2. Monthly Trend for Total Orders: Create a line chart that illustrates the hourly trend of total orders throughout the day. This chart will allow us to identify peak hours or periods of high order activity.

```
SELECT DATENAME(MONTH, order_date) AS Month_Name, COUNT(DISTINCT order_id) AS
Total_Orders FROM pizza_sales
GROUP BY DATENAME(MONTH, order_date)
ORDER BY Total_Orders DESC;
```

Results | Messages

	Month_Name	Total_Orders
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

3. Percentage of Sales by Pizza Category: Create a pie chart that shows the distribution of sales across different pizza categories. This chart will provide insights into the popularity of various pizza categories and their contribution to overall sales.

```
SELECT pizza_category, SUM(total_price) AS Total_Sales,
(SUM(total_price)/ (SELECT SUM(total_price) FROM pizza_sales))* 100 AS
Percentage_sales FROM pizza_sales
GROUP BY pizza_category;
```

Results | Messages

	pizza_category	Total_Sales	Percentage_sales
1	Classic	220053.100021362	26.9059602306976
2	Chicken	195919.5	23.9551375322885
3	Veggie	193690.451004028	23.6825910258677
4	Supreme	208196.99981308	25.4563112111462

4. Percentage of Sales by Pizza Size: Generate a pie chart that represents the percentage of sales attribution to different pizza sizes. This chart will help us understand customer preferences for pizza sizes and their impact on sales.

```
SELECT pizza_size, SUM(total_price) AS Total_Sales,(SUM(total_price)/ (SELECT
SUM(total_price) FROM pizza_sales))* 100 AS Percentage_sales FROM pizza_sales
GROUP BY pizza_size
ORDER BY Percentage_sales DESC;
```

Results | Messages

	pizza_size	Total_Sales	Percentage_sales
1	L	375318.701004028	45.8903330244889
2	M	249382.25	30.492044420599
3	S	178076.49981308	21.7734684107037
4	XL	14076	1.72107684995364
5	XXL	1006.6000213623	0.123077294254725

5. Top 5 best sellers by Revenue and Total Quantity: Create a bar chart highlighting the top 5 best-selling pizzas based on the revenue and Total Quantity. This chart will help us identify the most popular pizza options.

```
SELECT TOP 5 pizza_name, SUM(total_price) AS total_revenue, SUM(quantity)
AS Total_quantity FROM pizza_sales_excel_file
Group by pizza_name
order by total_revenue DESC;
```

Results | Messages

	pizza_name	total_revenue	Total_quantity
1	The Thai Chicken Pizza	43434.25	2371
2	The Barbecue Chicken Pizza	42768	2432
3	The California Chicken Pizza	41409.5	2370
4	The Classic Deluxe Pizza	38180.5	2453
5	The Spicy Italian Pizza	34831.25	1924

6. Bottom 5 sellers by Total Revenue and Total Quantity: Create a bar chart highlighting the Bottom 5 selling pizzas based on the Revenue and Total Quantity. This chart will help us identify the least popular pizza options.

```
SELECT TOP 5 pizza_name, SUM(total_price) AS total_revenue, SUM(quantity)
AS Total_quantity FROM pizza_sales_excel_file
Group by pizza_name
order by total_revenue ASC;
```

	pizza_name	total_revenue	Total_quantity
1	The Brie Carre Pizza	11588.4998130798	490
2	The Green Garden Pizza	13955.75	997
3	The Spinach Supreme Pizza	15277.75	950
4	The Mediterranean Pizza	15360.5	934
5	The Spinach Pesto Pizza	15596	970