Problem Statement

KPIs

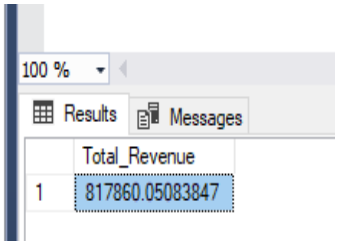
We need to analyse key indicators for our pizza sales data to gain insights into our business performance. Specifically, we want to calculate the following metrics.

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

SYNTAX:

```
SELECT SUM(total_price) AS Total_Revenue from pizza_sales
```

OUTPUT:

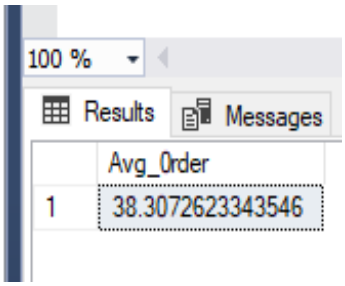


2) Average Order Value: The average amount spent per order, calculated by dividing the total revenue by the total number of orders.

SYNTAX:

```
SELECT * FROM pizza_sales
SELECT SUM(total_price)/ COUNT(DISTINCT order_id) AS Avg_Order from pizza_sales
```

OUTPUT:

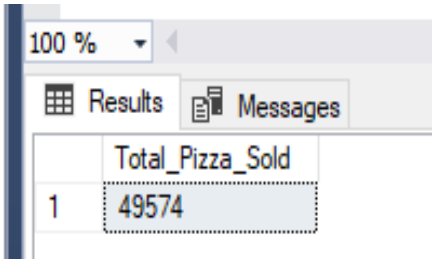


3) Total Pizzas Sold: The sum of the quantities of all pizzas sold.

SYNTAX:

```
SELECT SUM(quantity) as Total_Pizza_Sold from pizza_sales
```

OUTPUT:



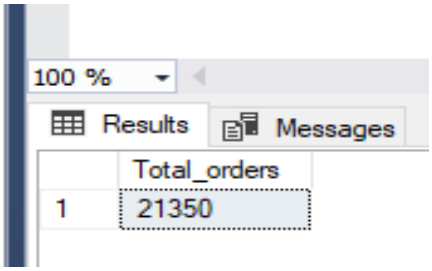
	Total_Pizza_Sold
1	49574

4) Total orders: The total number of orders placed.

SYNTAX:

```
SELECT COUNT(DISTINCT order_id) as Total_orders from pizza_sales
```

OUTPUT:



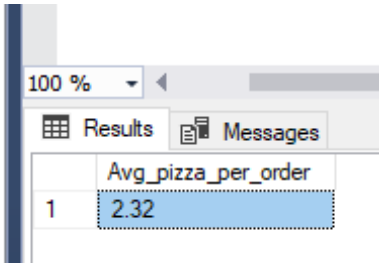
	Total_orders
1	21350

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.

SYNTAX:

```
SELECT CAST(CAST(SUM(quantity) AS DECIMAL(10,2))/CAST(COUNT(DISTINCT order_id) AS DECIMAL(10,2)) AS
DECIMAL(10,2)) AS Avg_pizza_per_order from pizza_sales
```

OUTPUT:



	Avg_pizza_per_order
1	2.32

CHARTS REQUIREMENT

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

6) Daily Trend for Total Orders: Create bar chart that displays daily trend of total orders over specific period. This chart help in identifying any pattern or fluctuations in order volumes on daily basis.

SYNTAX:

```
--Daily Trend
SELECT DATENAME(DW, order_date) as order_day, COUNT(DISTInCT order_id) as Total_order from pizza_sales
GROUP BY DATENAME(DW, order_date)
```

OUTPUT:

100 %

Results | Messages

	order_day	Total_order
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 Total Orders: Create line chart that illustrates hourly trend of total orders throughout the day. This chart identifies peak hours or periods of high order activity.

SYNTAX:

```
--Hourly trend
SELECT DATEPART(HOUR, order_time) AS order_hours, COUNT(DISTInCT order_id) as Total_order from pizza_sales
GROUP BY DATEPART(HOUR, order_time)
ORDER BY DATEPART(HOUR, order_time)
```

OUTPUT:

100 %

Results | Messages

	order_hours	Total_order
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

Query executed successfully.

8) Percentage of Sales by Pizza Category: Create a pie chart showing the distribution of sales across different pizza categories. This chart provides insights into the popularity of various pizza categories and their contribution to overall sales.

SYNTAX:

```
 SELECT pizza_category, SUM(total_price) as Total_Sales,SUM(total_price) * 100/
(SELECT SUM(total_price) from pizza_sales WHERE MONTH(order_date) =1)AS PCT
from pizza_sales
WHERE MONTH(order_date)=1
```

```
GROUP BY pizza_category
```

OUTPUT:

100 %

Results | Messages

	pizza_category	Total_Sales	PCT
1	Classic	18619.4000015259	2.27659976574687
2	Chicken	16188.75	1.97940344236196
3	Veggie	17055.4000778198	2.0853690139694
4	Supreme	17929.7499866486	2.19227604628285

9) Percentage of Sales by Pizza Size: Using pie chart to represent percentage of sales attributed to different pizza sizes. This chart helps us understand customer preferences for pizza sizes and their impact on sales.

SYNTAX:

```
 SELECT pizza_size, CAST(SUM(total_price)AS DECIMAL(10,2)) as Total_Sales, CAST(SUM(total_price) * 100/
(SELECT SUM(total_price) from pizza_sales WHERE DATEPART(QUARTER, order_date)=1) AS DECIMAL(10,2)) AS PCT
from pizza_sales
WHERE DATEPART(QUARTER, order_date)=1
GROUP BY pizza_size
ORDER BY PCT DESC
```

OUTPUT:

100 %

Results | Messages

	pizza_size	Total_Sales	PCT
1	L	95229.65	46.37
2	M	61159.00	29.78
3	S	45384.25	22.10
4	XL	3289.50	1.60
5	XXL	287.60	0.14

10) Total Pizzas Sold by Pizza category: Using funnel chart to present total number of pizzas sold for each category. This chart allow comparing the sales performance of different pizza categories.

SYNTAX:

```
 SELECT pizza_category, sum(quantity) as Total_Pizzas_Sold
from pizza_sales
GROUP BY pizza_category
```

OUTPUT:

100 %

Results | Messages

	pizza_category	Total_Pizzas_Sold
1	Classic	14888
2	Chicken	11050
3	Veggie	11649
4	Supreme	11987

11) Top five Best Sellers by Total Pizzas Sold: Using bar chart showing top five best-selling pizzas based on total number of pizzas sold.

SYNTAX:

```
SELECT TOP 5 pizza_name,sum(quantity) as Total_Pizzas_Sold
from pizza_sales
GROUP BY pizza_name
order by sum(quantity) DESC
```

OUTPUT:

100 %

Results | Messages

	pizza_name	Total_Pizzas_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: Using bar chart showing bottom five worst selling pizzas based on total number of pizzas sold.

SYNTAX:

```
SELECT TOP 5 pizza_name,sum(quantity) as Total_Pizzas_Sold
from pizza_sales
GROUP BY pizza_name
order by sum(quantity) asc
```

OUTPUT:

100 %

Results | Messages

	pizza_name	Total_Pizzas_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