

--PROBLEM STATEMENT 1: KPI

USE [PizzaSales]

1. --Total Revenue calculation
SELECT
ROUND (SUM (total_price),2) AS Total_revenue
FROM pizza_sales

Results Messages	
Total_revenue	
1	817860.05

2. --Average order value
SELECT
ROUND (SUM (total_price) / COUNT (DISTINCT order_id),2) AS Average_order_value
FROM pizza_sales

Average_order_value
38.31

3. --Total pizzas sold
SELECT
COUNT (quantity) AS Total_pizzas_sold
FROM pizza_sales

Total_pizzas_sold
49574

4. --Total orders
SELECT
COUNT (order_id) AS Total_orders
FROM pizza_sales

Total_orders
21350

5. --Average number of pizzas per order
SELECT CAST (
CAST (SUM(quantity) AS decimal(10,2))
/ CAST (COUNT(DISTINCT order_id) AS decimal(10,2))
AS decimal (10,2))
AS average_number_of_pizzas_per_order
FROM pizza_sales

average_number_of_pizzas_per_order
2.32

--PROBLEM STATEMENT 2:CHART REQUIREMENTS

USE PizzaSales

1.--Daily trend for total orders

```
SELECT DATENAME(DW, order_date) as Order_day , COUNT(DISTINCT
order_id) as Total_orders
FROM pizza_sales
GROUP BY DATENAME(DW, order_date)
```

	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.--Hourly trend for total orders

```
SELECT DATEPART(HH, order_time) AS Order_hour , COUNT(DISTINCT
order_id) as Total_orders,
CASE
WHEN COUNT(DISTINCT order_id) >= 1500 THEN 'Peak times'
WHEN COUNT(DISTINCT order_id) >= 500 AND COUNT(DISTINCT order_id) <
1500 THEN 'Normal'
ELSE 'Slow'
END AS Business_status
FROM pizza_sales
GROUP BY DATEPART(HH, order_time)
ORDER BY DATEPART(HH, order_time)
```

	Order_hour	Total_orders	Business_status
1	9	1	Slow
2	10	8	Slow
3	11	1231	Normal
4	12	2520	Peak times
5	13	2455	Peak times
6	14	1472	Normal
7	15	1468	Normal
8	16	1920	Peak times
9	17	2336	Peak times
10	18	2399	Peak times
11	19	2009	Peak times
12	20	1642	Peak times
13	21	1198	Normal
14	22	663	Normal
15	23	28	Slow

3--Total Sales per month

```

SELECT
DATENAME(MONTH, order_date) AS order_month, ROUND(SUM(total_price),2)
AS Total_revenue
FROM pizza_sales
GROUP BY DATENAME(MONTH, order_date)
ORDER BY DATENAME(MONTH, order_date) DESC

```

	order_month	Total_revenue
1	September	64180.05
2	October	64027.6
3	November	70395.35
4	May	71402.75
5	March	70397.1
6	June	68230.2
7	July	72557.9
8	January	69793.3
9	February	65159.6
10	December	64701.15
11	August	68278.25
12	April	68736.8

4--Percentage of sales per category

```
SELECT pizza_category,ROUND(SUM(total_price),2) AS Total_revenue,
ROUND(SUM(total_price)*100
/ (SELECT SUM(total_price) FROM pizza_sales),2) AS pct_sales
FROM pizza_sales
GROUP BY pizza_category
```

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

5--Percentage of sales by pizza size

```
SELECT pizza_size,
ROUND(SUM(total_price),2) AS Total_revenue, ROUND(SUM(total_price)*100 /
(SELECT SUM(total_price) FROM pizza_sales),2) AS pct_sales
FROM pizza_sales
GROUP BY pizza_size
ORDER BY pct_sales DESC
```

	pizza_size	Total_revenue	pct_sales
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

6--Total pizzas sold by category

```
SELECT pizza_category, ROUND(SUM(total_price),2) AS Total_sales
FROM pizza_sales
GROUP BY pizza_category
```

	pizza_category	Total_sales
1	Classic	220053.1
2	Chicken	195919.5
3	Veggie	193690.45
4	Supreme	208197

7--Top 5 Best sellers by total Pizzas sold

```
SELECT TOP(5)
pizza_name , COUNT(quantity) AS 'total_pizzas_sold'
FROM pizza_sales
GROUP BY pizza_name
```

	pizza_name	total_pizzas_sold
1	The Sicilian Pizza	1887
2	The Spicy Italian Pizza	1887
3	The Thai Chicken Piz...	2315
4	The Big Meat Pizza	1811
5	The Four Cheese Piz...	1850

8--Bottom 5 worst sellers by total pizzas sold

```
SELECT TOP(5)
pizza_name , COUNT(quantity) AS 'total_pizzas_sold'
FROM pizza_sales
GROUP BY pizza_name
ORDER BY total_pizzas_sold DESC
```

	pizza_name	total_pizzas_sold
1	The Classic Deluxe Pizza	2416
2	The Barbecue Chicken Pizza	2372
3	The Hawaiian Pizza	2370
4	The Pepperoni Pizza	2369
5	The Thai Chicken Pizza	2315