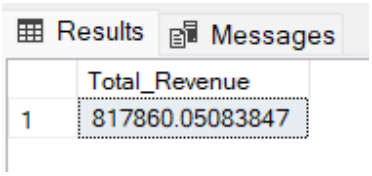
PIZZA SALES SQL QUERIES

A. KPI's

1. Total Revenue:

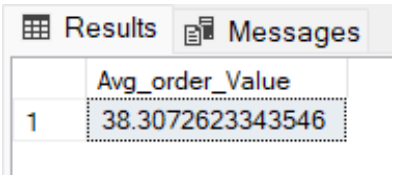
```
SELECT SUM(total_price) AS Total_Revenue
FROM pizza_sales;
```


Results Messages

	Total_Revenue
1	817860.05083847

2. Average Order Value

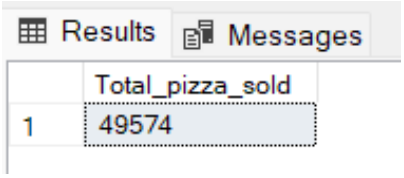
```
SELECT (SUM(total_price) / COUNT(DISTINCT order_id)) AS
Avg_order_Value FROM pizza_sales;
```


Results Messages

	Avg_order_Value
1	38.3072623343546

3. Total Pizzas Sold

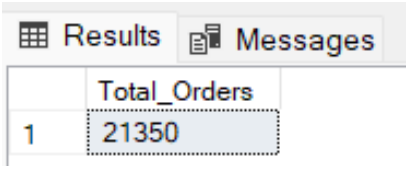
```
SELECT SUM(quantity) AS Total_pizza_sold
FROM pizza_sales;
```


Results Messages

	Total_pizza_sold
1	49574

4. Total Orders

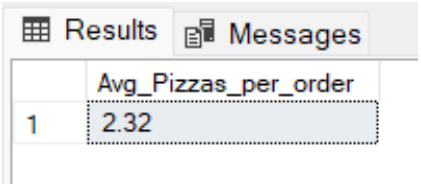
```
SELECT COUNT(DISTINCT order_id) AS Total_Orders
FROM pizza_sales;
```



	Total_Orders
1	21350

5. Average 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 Avg_Pizzas_per_order
FROM pizza_sales;
```

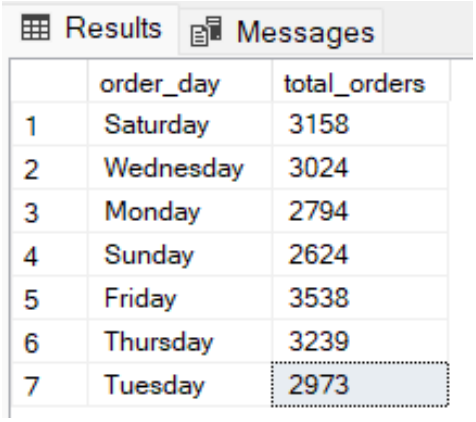


	Avg_Pizzas_per_order
1	2.32

B. 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);
```

Output:



	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

C. Hourly Trend for Orders

```
SELECT DATEPART(HOUR, order_time) as order_hours,
COUNT(DISTINCT order_id) as total_orders
FROM pizza_sales
GROUP BY DATEPART(HOUR, order_time)
ORDER BY DATEPART(HOUR, order_time);
```

Output

Results | Messages

	order_hours	total_orders
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
12	20	1642
13	21	1198
14	22	663
15	23	28

D. Percentage of Sales by Pizza Category

```
SELECT pizza_category, CAST(SUM(total_price) AS
DECIMAL(10,2)) 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_category;
```

Output

Results | Messages

	pizza_category	total_revenue	PCT
1	Classic	220053.10	26.91
2	Chicken	195919.50	23.96
3	Veggie	193690.45	23.68
4	Supreme	208197.00	25.46

E. Percentage of Sales by Pizza Size

```
SELECT pizza_size, CAST(SUM(total_price) AS DECIMAL(10,2))
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 pizza_size;
```

Output

Results Messages

	pizza_size	total_revenue	PCT
1	L	375318.70	45.89
2	M	249382.25	30.49
3	S	178076.50	21.77
4	XL	14076.00	1.72
5	XXL	1006.60	0.12

F. Total Pizzas Sold by Pizza Category

```
SELECT pizza_category, SUM(quantity) AS
Total_Quantity_Sold
FROM pizza_sales
WHERE MONTH(order_date) = 2
GROUP BY pizza_category
ORDER BY Total_Quantity_Sold DESC;
```

Output

	pizza_category	Total_Quantity_Sold
1	Classic	1178
2	Supreme	964
3	Veggie	944
4	Chicken	875

G. Top 5 Best Sellers by Total Pizzas Sold

```
SELECT Top 5 pizza_name, SUM(quantity) AS Total_Pizza_Sold
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Pizza_Sold DESC;
```

Output

	pizza_name	Total_Pizza_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

H. Bottom 5 Best Sellers by Total Pizzas Sold

```
SELECT TOP 5 pizza_name, SUM(quantity) AS Total_Pizza_Sold
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Pizza_Sold ASC;
```

Output

Results Messages

	pizza_name	Total_Pizza_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

NOTE

If you want to apply the Month, Quarter, Week filters to the above queries you can use WHERE clause. Follow some of below examples

```sql
SELECT DATENAME(DW, order_date) AS order_day,
COUNT(DISTINCT order_id) AS total_orders
FROM pizza_sales
WHERE MONTH(order_date) = 1
GROUP BY DATENAME(DW, order_date);
```

Results Messages

	order_day	total_orders
1	Friday	330
2	Monday	220
3	Saturday	303
4	Sunday	199
5	Thursday	329
6	Tuesday	242
7	Wednesday	222

Here MONTH(order_date) = 1 indicates that the output is for the month of January. MONTH(order_date) = 4 indicates output for Month of April.

```sql
SELECT DATENAME(DW, order_date) AS order_day,
COUNT(DISTINCT order_id) AS total_orders
FROM pizza_sales
WHERE DATEPART(QUARTER, order_date) = 1
GROUP BY DATENAME(DW, order_date);
```

Results Messages

	order_day	total_orders
1	Saturday	766
2	Wednesday	711
3	Monday	743
4	Sunday	682
5	Friday	885
6	Thursday	796
7	Tuesday	787

Here DATEPART(QUARTER, order_date) = 1 indicates that the output is for the Quarter 1. MONTH(order_date) = 3 indicates output for Quarter 3.