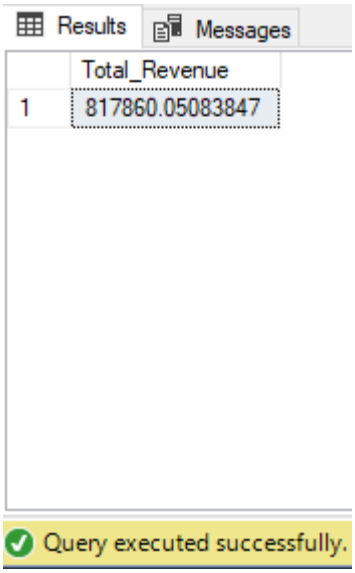
PIZZA SALES SQL QUERIES

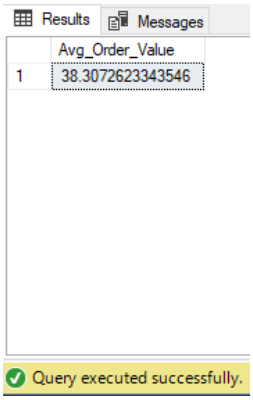
1. Total Revenue:

```
/* Q1: Total Revenue: The sum of the total price of all pizza
orders.*/
SELECT SUM (total_price)AS Total_Revenue from pizza_sales
```



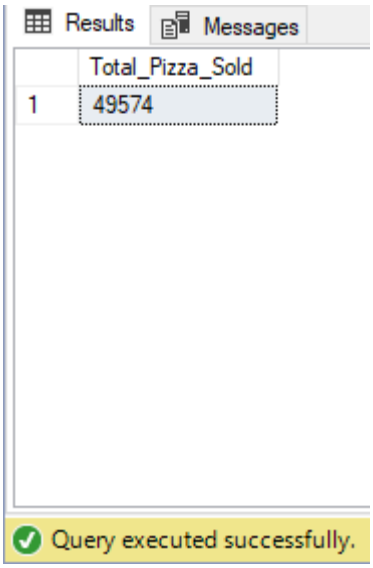
2. Average Order Value

```
/* Q2: 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
Avg_Order_Value from pizza_sales
```



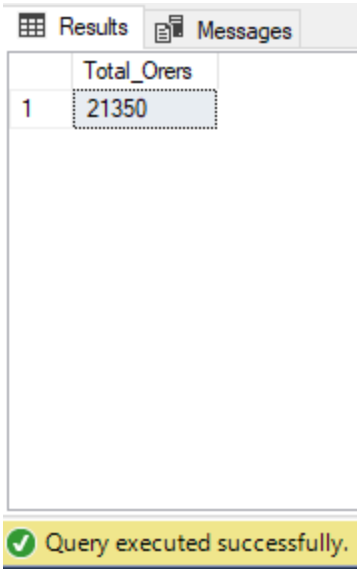
3. Total Pizzas Sold

```
/* Q3: Total Pizzas Sold: The sum of the quantities of all pizzas
sold..*/

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



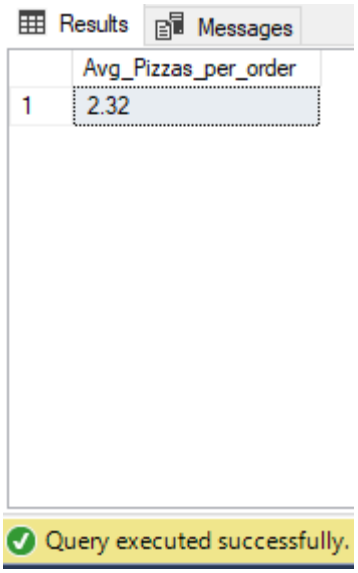
4. Total Orders

```
/* Q4: Total Orders: The total number of orders placed.*/
SELECT COUNT (DISTINCT order_id ) AS Total_Orers from pizza_sales
```



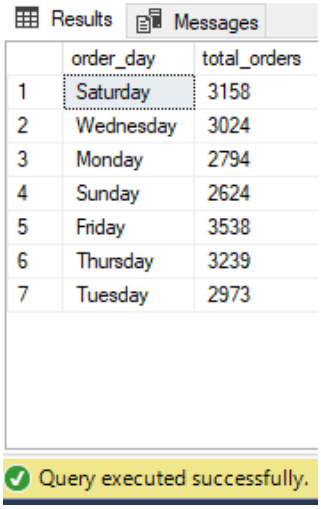
5. Average Pizzas Per Order

```
/* Q5: 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 * from pizza_sales
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

/* So in the upper query we use CAST it is used for converting the
data type in decimal as average values are mostly answer in decimals
for that purpose we use CAST and we use (10,2) it is use for as we
getting decimals like 2.987647755 and from that 10 decimals we want
only 2 decimals thats why we used this  */
```



Queries for Chart Requirements:

```
/* 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)

/* In upper query we use  DATENAME it is an arguement it is used to
derive the date of the week and DW retrives the day of the week as a
chacracter string like sunday, monday etc.  .*/
```



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

Query executed successfully.

```sql
/* 2: Mothly Trends for Total Orders .*/

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
```

	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

Query executed successfully.

```sql
/* 3: Monthly Trend for Orders.*/
SELECT * FROM pizza_sales
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 Percntge_of_sales
from pizza_sales
WHERE MOnTH (order_date) =1
GROUP BY pizza_category
```

/* 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. */

Results Messages

	pizza_category	Total_Sales	Percntge_of_sales
1	Classic	18619.4000015259	26.6779189176038
2	Chicken	16188.75	23.1952780348435
3	Veggie	17055.4000778198	24.4370162489706
4	Supreme	17929.7499866486	25.6897867985821

```sql
/* 4: % of Sales by Pizza Size.*/
SELECT * FROM pizza_sales
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) AS DECIMAL (10,2)) AS
Percntge_of_sales
FROM pizza_sales
GROUP BY pizza_size
ORDER BY Percntge_of_sales DESC
```

Results | Messages

	pizza_size	Total_Sales	Percntge_of_sales
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

```sql
/* 5: Total Pizzas Sold by Pizza Category.*/
SELECT * FROM pizza_sales
SELECT pizza_category, SUM (quantity) as Total_Quantity_Sold
FROM pizza_sales
GROUP BY pizza_category
ORDER BY Total_Quantity_Sold DESC
```

Results | Messages

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

```sql
6: Top 5 Pizzas by Revenue.Total Quantity and Total Orders. */
SELECT TOP 5 pizza_name, SUM (total_price) AS Total_Revenue  FROM
pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue ASC

/* To limit our data to TOP 5 or something like this we write TOP 5 as
we did in our query ABOVE we write TOP after SELECT */

/* Whenever you are using agregation in any of the statement with
respect to category or fieled like  " pizza_name " we use GROUP BY for
thta specific fieled. */
```

Results | Messages

	pizza_name	Total_Revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5
4	The Classic Deluxe Pizza	38180.5
5	The Spicy Italian Pizza	34831.25

```sql
/* 7: Bottom 5 Pizzas by Revenue. */
SELECT * FROM pizza_sales
SELECT TOP  5 pizza_name, SUM (total_price) AS Total_Revenue  FROM
pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue ASC
```

Results | Messages

	pizza_name	Total_Revenue
1	The Brie Carre Pizza	11588.4998130798
2	The Green Garden Pizza	13955.75
3	The Spinach Supreme Pizza	15277.75
4	The Mediterranean Pizza	15360.5
5	The Spinach Pesto Pizza	15596

```sql
/* 8: TOP 5 Pizzas by Quantity. */
SELECT Top 5 pizza_name, SUM (quantity) AS Total_Pizza_Sold
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Pizza_Sold DESC
```

Results | Messages

	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

```sql
/* 9: Bottom 5 Pizzas by Quantity. */
SELECT Top 5 pizza_name, SUM (quantity) AS Total_Pizza_Sold
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Pizza_Sold ASC
```

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

```sql
* 10: Top 5 Pizzas by Total Orders. */
SELECT Top 5 pizza_name, COUNT (DISTINCT order_id) AS Total_Orders
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Orders DESC
```

Results Messages

	pizza_name	Total_Orders
1	The Classic Deluxe Pizza	2329
2	The Hawaiian Pizza	2280
3	The Pepperoni Pizza	2278
4	The Barbecue Chicken Pizza	2273
5	The Thai Chicken Pizza	2225

```sql
/* 11: Bottom 5 Pizzas by Total Orders. */
SELECT Top 5 pizza_name, COUNT (DISTINCT order_id) AS Total_Orders
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Orders ASC
```

Results Messages

	pizza_name	Total_Orders
1	The Brie Carre Pizza	480
2	The Mediterranean Pizza	912
3	The Spinach Supreme Pizza	918
4	The Calabrese Pizza	918
5	The Chicken Pesto Pizza	938