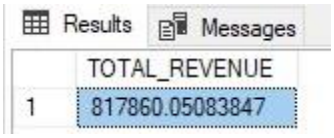
PIZZA SALES SQL QUERIES

A. KPI's

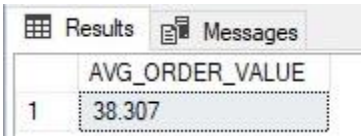
1. Total Revenue:

```
SELECT SUM( total_price ) AS TOTAL_REVENUE FROM pizza_sales
```



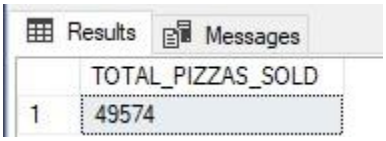
2. Average Order Value:

```
SELECT CAST( SUM ( total_price ) / COUNT( DISTINCT order_id ) AS DECIMAL ( 10, 3 ) ) AS
AVG_ORDER_VALUE FROM pizza_sales
```



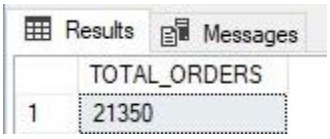
3. Total Pizzas Sold:

```
SELECT SUM( quantity ) AS TOTAL_PIZZAS_SOLD FROM pizza_sales
```



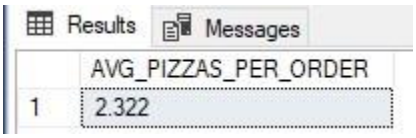
4. Total Orders:

```
SELECT COUNT( DISTINCT order_id ) AS TOTAL_ORDERS FROM pizza_sales
```



5. Average Pizzas Per Order:

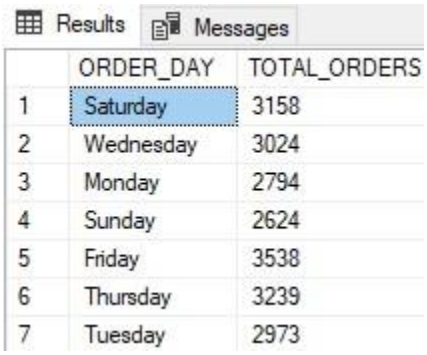
```
SELECT (CAST( CAST( SUM( quantity ) AS DECIMAL (10, 3)) /
CAST( COUNT( DISTINCT order_id) AS DECIMAL (10, 3)) AS DECIMAL (10, 3)))
AS AVG_PIZZAS_PER_ORDER
FROM pizza_sales
```



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:



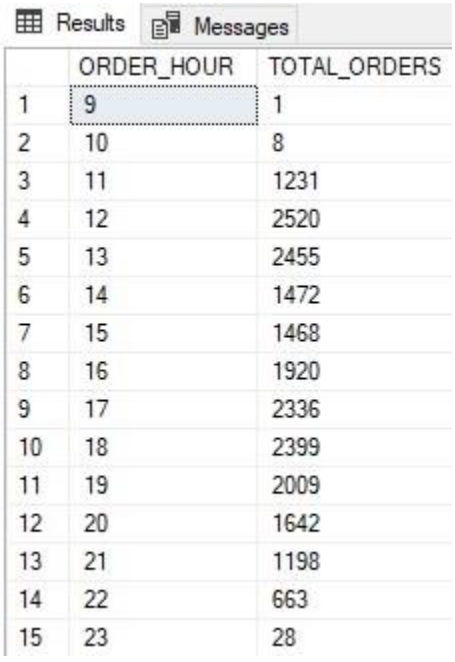
Results | Messages

	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

C. Hourly Trend for Orders

```
SELECT DATEPART(HOUR, order_time) AS ORDER_HOUR, 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_HOUR	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, SUM(quantity) AS TOTAL_SOLD,
CAST(SUM(quantity)*100/(SELECT CAST(SUM(quantity) AS DECIMAL (10, 2)) FROM pizza_sales) AS
DECIMAL (10, 2)) AS PCT
FROM pizza_sales
GROUP BY pizza_category
ORDER BY pizza_category
```

OUTPUT:

Results Messages

	pizza_category	TOTAL_SOLD	PCT
1	Chicken	11050	22.29
2	Classic	14888	30.03
3	Supreme	11987	24.18
4	Veggie	11649	23.50

E. Percentage of Sales by Pizza Size

```
SELECT pizza_size, SUM(quantity) AS TOTAL_SOLD,
CAST(SUM(quantity)*100/(SELECT CAST(SUM(quantity) AS DECIMAL (10, 2)) 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_SOLD	PCT
1	L	18956	38.24
2	M	15635	31.54
3	S	14403	29.05
4	XL	552	1.11
5	XXL	28	0.06

F. Total Pizzas Sold by Pizza Category

```
SELECT pizza_category, SUM(quantity) AS TOTAL_SOLD
FROM pizza_sales
GROUP BY pizza_category
ORDER BY TOTAL_SOLD DESC
```

OUTPUT:

Results Messages

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

G. Top 5 Best Selling Pizza's

```
SELECT TOP 5 pizza_name, SUM(quantity) AS TOTAL_SOLD
FROM pizza_sales
GROUP BY pizza_name
ORDER BY TOTAL_SOLD DESC
```

OUTPUT:

Results | Messages

	pizza_name	TOTAL_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 Selling Pizza's

```
SELECT TOP 5 pizza_name, SUM(quantity) AS TOTAL_SOLD
FROM pizza_sales
GROUP BY pizza_name
ORDER BY TOTAL_SOLD DESC
```

Results | Messages

	pizza_name	TOTAL_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

```
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)
```

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.

```
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)
```

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