

# Analyse SQL : Résumé des Requêtes .

## I. Les KPI calculés

### 1. Total Revenue:

The screenshot displays the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'SQLQuery1.sql' connected to 'DESKTOP-GNH63SI\SQLEXPRESS.Pizza Database (DESKTOP-GNH63SI\Lenovo (57))'. The 'Object Explorer' on the left shows the database structure, with 'dbo.pizza\_sales' selected under 'Tables'. The 'Query Editor' in the center contains the following SQL query:

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

The 'Results' pane at the bottom shows the output of the query, which is a single row with the value 817860.0499999993 under the column 'Total\_Revenue'. A status bar at the bottom indicates 'Query executed successfully.' and shows the cursor is at line 2, column 42.

Total_Revenue
817860.0499999993

## 2. Average Order Value

SQLQuery1.sql - DE...H63SI\Lenovo (57))*	
<pre>SELECT SUM(total_price) / COUNT(DISTINCT order_id) AS Average_Order_Value FROM pizza_sales</pre>	
00 %	
Results	Messages
Average_Order_Value	
1	38.3072622950816

## 3. Total Pizzas Sold

SQLQuery1.sql - DE...H63SI\Lenovo (57))*	
<pre>SELECT SUM(quantity) AS Total_Pizza_Solde FROM pizza_sales</pre>	
100 %	
Results	Messages
Total_Pizza_Solde	
1	49574

## 4. Total Orders

SQLQuery1.sql - DE...H63SI\Lenovo (57))*	
<pre>SELECT COUNT(DISTINCT order_id) AS Total_orders FROM pizza_sales</pre>	
100 %	
Results	Messages
Total_orders	
1	21350

## 5. Average Pizzas Per Order

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

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

100 %

Results Messages

	AVG_Pizzas_Per_Order
1	2.3219672131147

## 6. Daily Trend for Total Orders

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders  
FROM pizza_sales  
GROUP BY DATENAME(DW, order_date)
```

100 %

Results Messages

	order_day	total_orders
1	samedi	3158
2	dimanche	2624
3	mercredi	3024
4	lundi	2794
5	vendredi	3538
6	mardi	2973
7	jeudi	3239

## 7. Monthly Trend for Orders

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
select DATENAME(MONTH, order_date) as Month_Name, COUNT(DISTINCT order_id) as Total_Orders
from pizza_sales
GROUP BY DATENAME(MONTH, order_date)
```

100 %

Results Messages

	Month_Name	Total_Orders
1	août	1841
2	mars	1840
3	janvier	1845
4	février	1685
5	novembre	1792
6	décembre	1680
7	juillet	1935
8	avril	1799
9	mai	1853
10	septembre	1661
11	octobre	1646
12	juin	1773

## 8. % of Sales by Pizza Category

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
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
```

100 %

Results Messages

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

## 9. % of Sales by Pizza Size

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
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
```

100 %

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

## 10. Total Pizzas Sold by Pizza Category

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
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
```

100 %

Results Messages

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

## 11. Top 5 Pizzas by Revenue

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
SELECT Top 5 pizza_name, SUM(total_price) AS Total_Revenue
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue DESC
```

100 %

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

## 12. Bottom 5 Pizzas by Revenue

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

```
SELECT Top 5 pizza_name, SUM(total_price) AS Total_Revenue
FROM pizza_sales
GROUP BY pizza_name
ORDER BY Total_Revenue ASC
```

100 %

Results Messages

	pizza_name	Total_Revenue
1	The Brie Carre Pizza	11588,4999999999
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

## 13. Top 5 Pizzas by Quantity

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

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

100 %

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

#### 14. Bottom 5 Pizzas by Quantity

SQLQuery1.sql - DE...H63SI\Lenovo (57))\*

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

100 %

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