SQL PROJECT

BY

IVOKE CHINWENDU MABEL

INTRODUCTION

- SQL is a structured query language for data creation and manipulation.
- As a programming language, SQL has commands which are used to define data tables.
- SQL explains works by breaking down the query into its operational components and displaying how the data base intends to execute them.
- It considers various factors such as table joins, indices and filtering criteria and presents this plan in a tabular format.

BACKGROUND PROJECT

- This SQL for data analysis helped me to practice key commands like **ORDER BY,GROUP BY, SELECT, WHERE, COUNT**, and some aggregate functions to answer some questions as can be seen in the visuals.

- By analyzing products, employee and operation’s data, employers can gain insights into their products and employees performance and identify areas for improvement and make data-driven decisions to increase revenue and profitability while discovering competent and incompetent employees.

PROJECT OBJECTIVES

The aim of this project is to to build visuals answering some questions like;

- Average lead time
- Average production time
- Downtime duration
- Production quantity
- Inventory quantity
- Employees rating etc.

PROJECT METHODOLOGY

The data I used for this project was given by my SQL instructor from Dahel consultant and Techies by name Modinat Ganiyu.

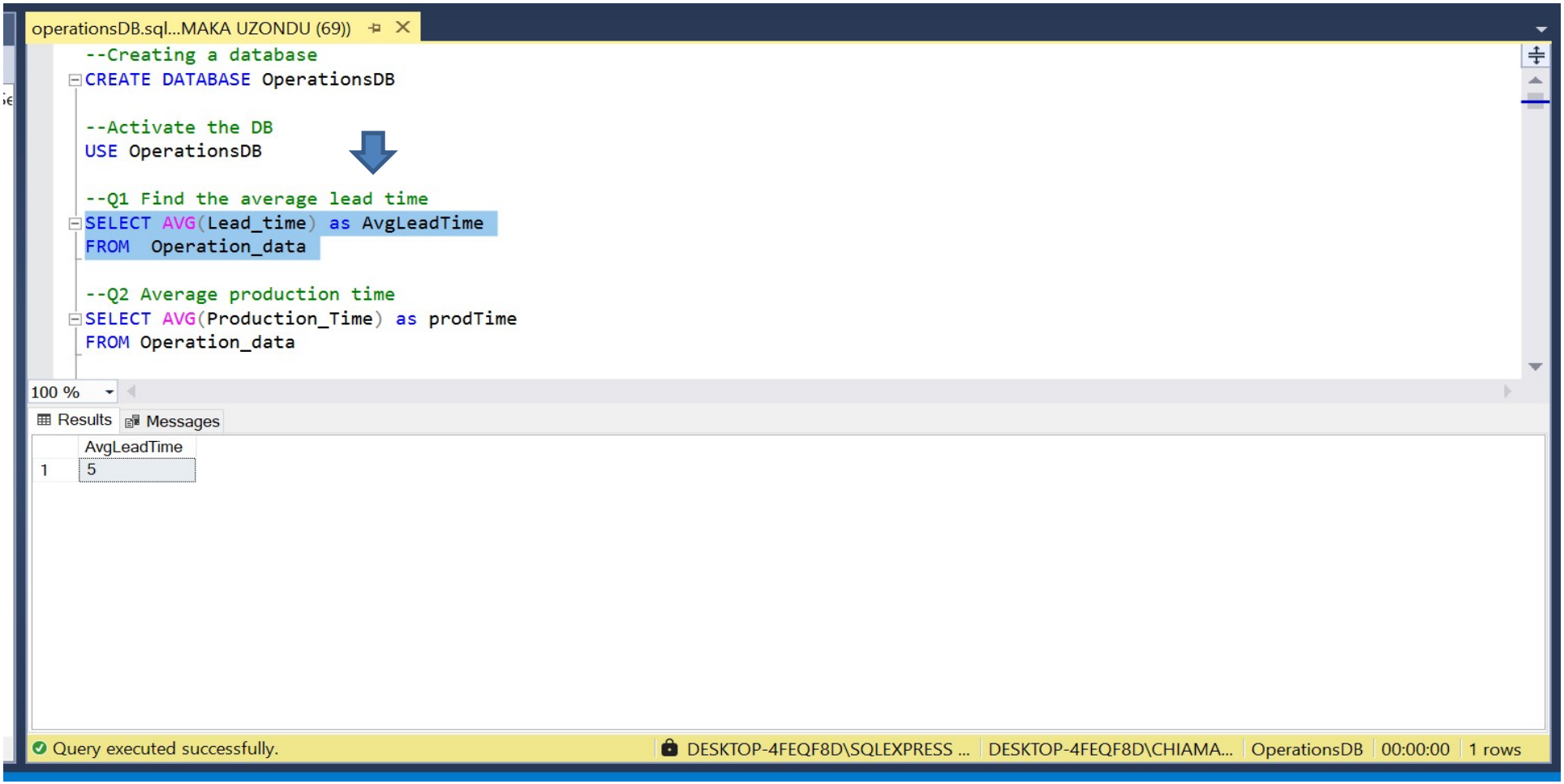
I used Microsoft SQL Server Management Studio for this project.

This project was conducted in steps below;

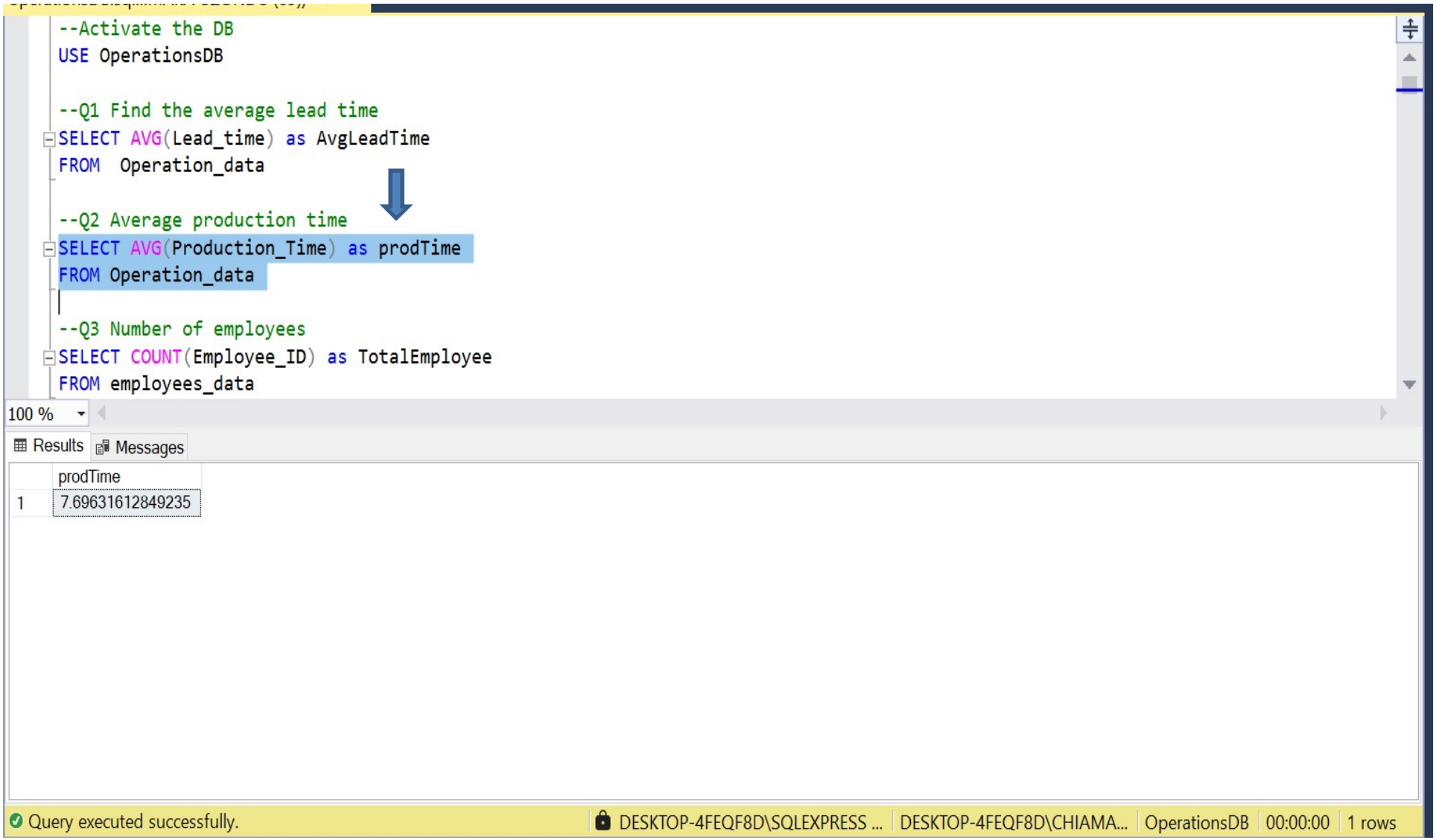
- **DATABASE CREATION**: I created a database called operationDB and executed it to change the name Master.
- **DATA CLEANING:** To get accurate result, I used DISTINCT function to clean the tables before execution to avoid including duplicate in my analysis.
- **DATA EXPLORATION**: I inserted the three tables used(employees, operations and product tables)into the database created. Then I explored the three tables to know its content and characteristics.
- **SELECT statement**: I used SELECT statement to answer the queries and sreenshot the views base on the number of queries as can be seen in this project.
- **NOTE**: Each of the questions answered were highlighted for easy comprehension of the analysis and the arrow shows the exact question answered.

DATA ANALYSIS

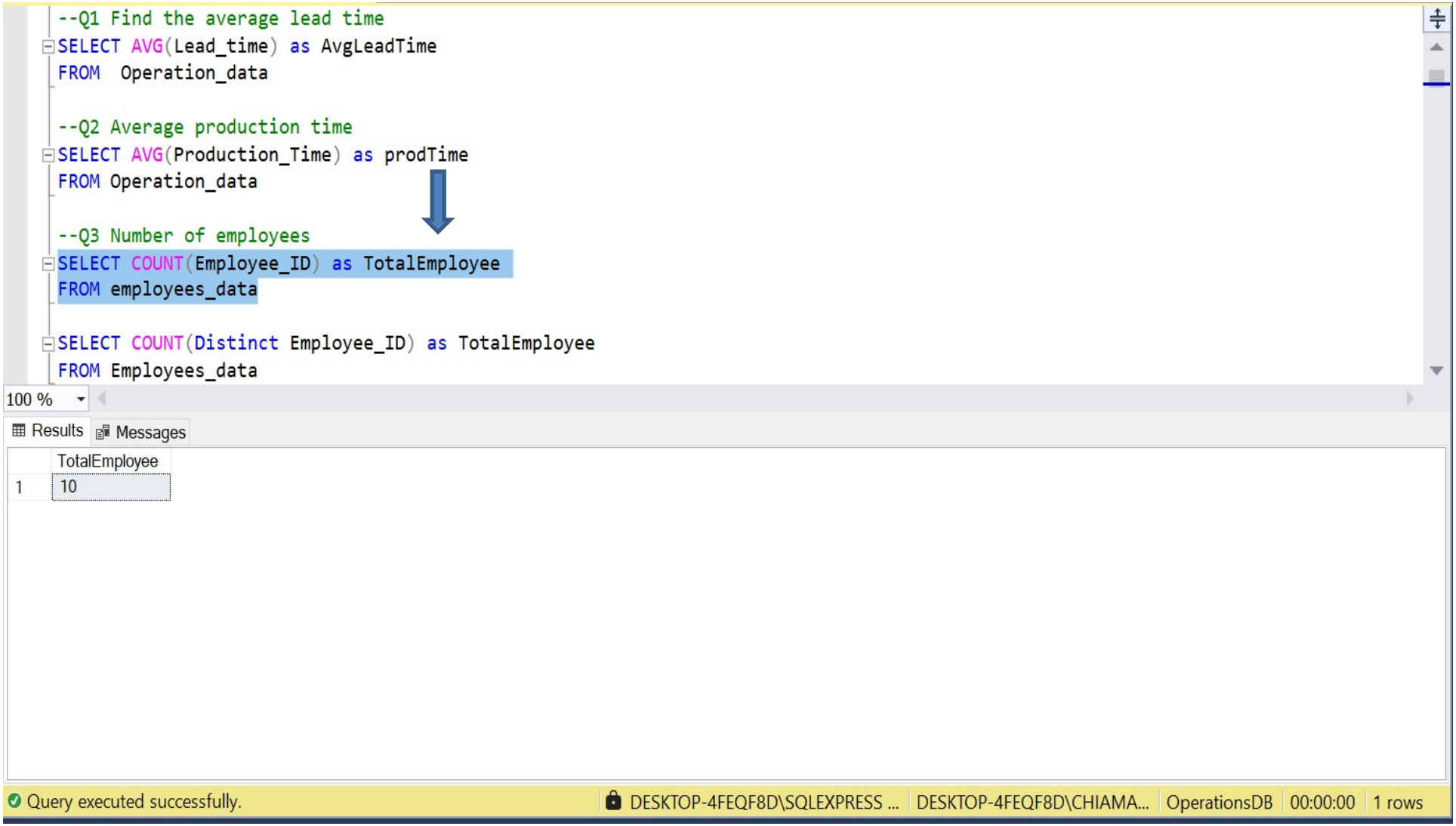
- The different queries were answered and the views recorded as follows:
- The average amount of time it takes to complete the sales was seen as **5** base on my analysis.



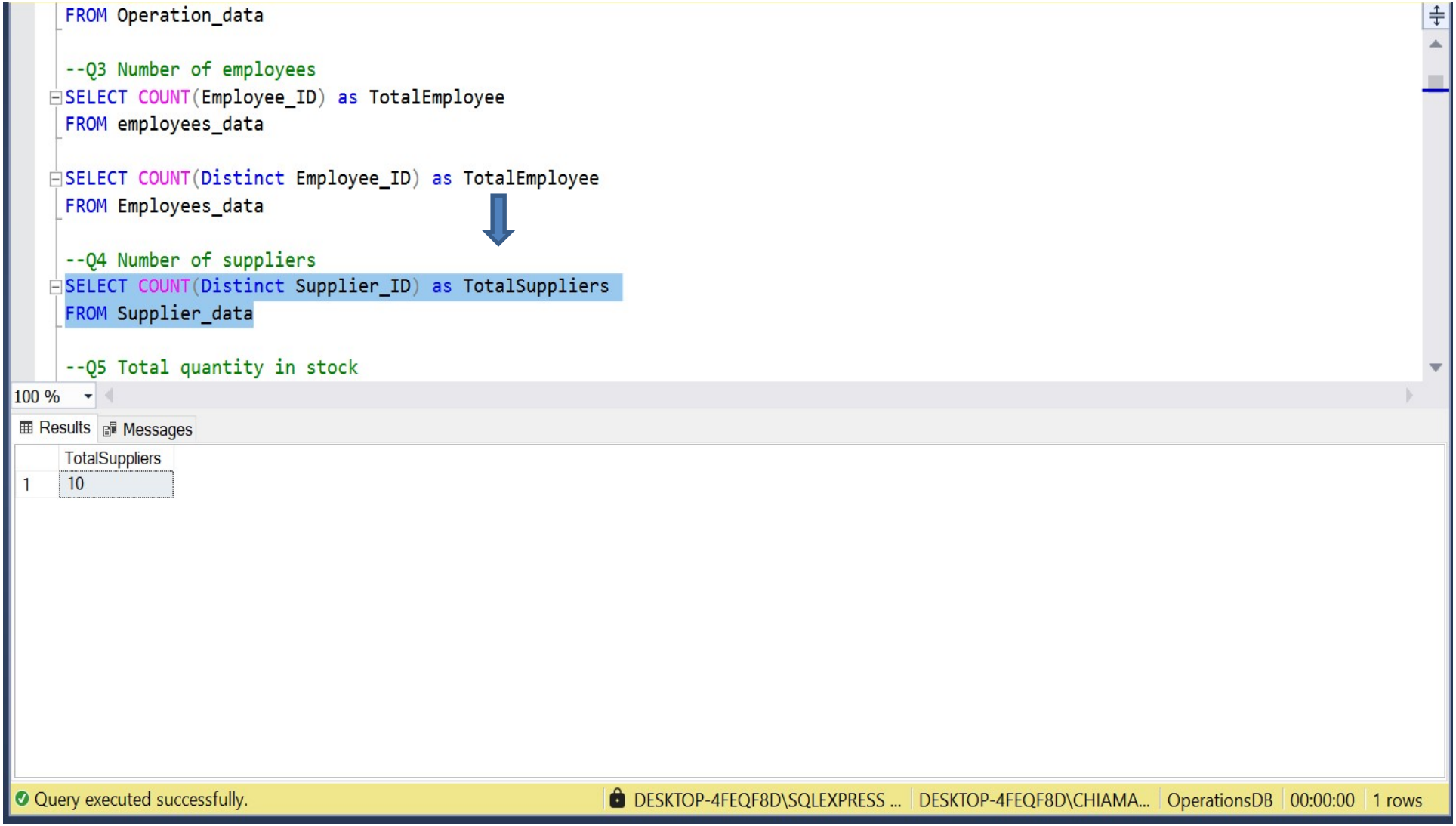
- The average production time were also gotten using the **SELECT AVG**.
- With the average time it took to sell, it can be seen that production is slow which is not an encouraging situation.



The number of employees were counted using **DISTINCT COUNT** function to erase any duplicate. The view showed **10** employees which may be low compared to the number of products they produce per day reason for slow production rate.



The optimal number of suppliers is not fixed, therefore the total suppliers were counted inorder to determine their ability to offer flexibility and risk mitigation. Although the total quantity in stock were seen as **279200**,and suppliers seen as **10**,it means that sale is low because suppliers are not many ,therefore more suppliers are needed to boost sales.



The **SUM** function was used to calculate the total quantity in stock and this was gotten from the operation table and the total quantity was analyzed using the **SUM** function and it was given as **279200** products.

```
SELECT COUNT(Distinct Employee_ID) as TotalEmployee
FROM Employees_data

--Q4 Number of suppliers
SELECT COUNT(Distinct Supplier_ID) as TotalSuppliers
FROM Supplier_data

--Q5 Total quantity in stock
SELECT SUM(Quantity_in_stock) as TotalQuantity
FROM Operation_data

--Q6 Total downtime duration
SELECT SUM(Downtime_Duration) as Total_Down_Time
FROM Operation_data
```

100 %

Results | Messages

	TotalQuantity
1	279200

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 1 rows

The downtime duration(the times the machines used for production, became faulty or runned down)thereby slowing the services was analyzed using the **SUM** function.

```
--Q4 Number of suppliers
SELECT COUNT(Distinct Supplier_ID) as TotalSuppliers
FROM Supplier_data

--Q5 Total
SELECT SUM(Quantity_in_stock) as TotalQuantity
FROM Operation_data

--Q6 Total downtime duration
SELECT SUM(Downtime_Duration) as Total_Down_Time
FROM Operation_data

--Q7 Total quantity in stock
SELECT SUM(Quantity_in_stock) as TotalQty
```

table OperationsDB.dbo.supplier_data

100 %

Results | Messages

	Total_Down_Time
1	2507.68790769577

Query executed successfully. DESKTOP-4FEOE8D\SQLEXPRESS | DESKTOP-4FEOE8D\CHIAMA | OperationsDB | 00:00:00 | 1 rows

The complete amount of available products for sale were analyzed inorder to determine if the available products will be enough for distributors.

```
SELECT SUM(Quantity_in_stock) as TotalQuantity
FROM Operation_data

--Q6 Total downtime duration
SELECT SUM(Downtime_Duration) as Total_Down_Time
FROM Operation_data

--Q7 Total quantity in stock
SELECT SUM(Quantity_in_stock) as TotalQty
FROM Operation_data

--Q8 Total production quantity
```

100 %

Results | Messages

	TotalQty
1	279200

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 1 rows

The production quantity of the products were analyzed and the amount of goods and services produced within a particular period was determined base on the orders placed.

```
SELECT SUM(Downtime_Duration) as Total_Down_Time
FROM Operation_data

--Q7 Total quantity in stock
SELECT SUM(Quantity_in_stock) as TotalQty
FROM Operation_data

--Q8 Total production quantity
SELECT(Production_Quantity) as TotalQty
 FROM Operation_Data

 --Q9 Number of entries in each downtime reason
 SELECT COUNT(Operations_ID) as TotalOperation,Maintenance_Type
 FROM Operation_data
```

100 %

Results | Messages

	TotalQty
1	497
2	437
3	246
4	591
5	470
6	448
7	275
8	938
9	488
10	610
11	422
12	675

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 1,000 rows

The number of entries in each downtime was counted inorder to capture the maintenance type and the total operations done in each downtime.

```sql
--Q7 Total quantity in stock
SELECT SUM(Quantity_in_stock) as TotalQty
FROM Operation_data

--Q8 Total production quantity
SELECT(Production_Quantity) as TotalQty
 FROM Operation_Data

 --Q9 Number of entries in each downtime reason
 SELECT COUNT(Operations_ID) as TotalOperation,Maintenance_Type
 FROM Operation_data
 Group by Maintenance_Type

 --Q10 Number of entries in each downtime reason
```

100 %

Results | Messages

	TotalOperation	Maintenance_Type
1	484	Repairs
2	516	Routine Maintenance

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 2 rows

The downtime entries were also analyzed to record the impact of the downtime and from my analysis, Material shortage was the top reason for downtime, followed by Human error and Technical issues

```
FROM Operation_Data

--Q9 Number of entries in each downtime reason
SELECT COUNT(Operations_ID) as TotalOperation,Maintenance_Type
FROM Operation_data
Group by Maintenance_Type

--Q10 Number of entries in each downtime reason
SELECT COUNT(Operations_ID) as Totaloperations, Downtime_Reason
FROM operation_data
Group by Downtime_Reason
Order by COUNT(Operations_ID)DESC

--Q11 Products by qty in stock(to get the inventory level of each products)
```

100 %

Results | Messages

	Totaloperations	Downtime_Reason
1	347	Material Shortage
2	346	Human Error
3	307	Technical Issues

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 3 rows

The inventory level of stock was calculated to record the number of goods in stock inorder to meet up with orders and to avoid overstock situations.

```
--Q11 Products by qty in stock(to get the inventory level of each products)
SELECT product_ID,SUM(Quantity_in_stock) as TotalQty
FROM Operation_data
Group by product_ID
ORDER BY SUM(Quantity_in_stock)DESC

--Q12 Average employees performance(employees rating) by month
```

100 %

Results | Messages

	product_ID	TotalQty
1	3	44475
2	4	41097
3	5	40194
4	7	39396
5	1	39353
6	2	38008
7	6	36677

Query executed successfully. | DESKTOP-4FEOF8D\SQLEXPRESS ... | DESKTOP-4FEOF8D\CHIAMA... | OperationsDB | 00:00:00 | 7 rows

The employees rating were done monthly to track progress and to provide regular feedback and to help in broader performance review cycle. Although it was not categorized but from my analyzed output, the poor, fair, above average, good or excellent can be easily determined

```
--Q11 Products by qty in stock(to get the inventory level of each products)
SELECT product_ID,SUM(Quantity_in_stock) as TotalQty
FROM Operation_data
Group by product_ID
ORDER BY SUM(Quantity_in_stock)DESC

--Q12 Average employees performance(employees rating) by month
SELECT AVG(employee_rating) as AvgempPerf,MONTH(production_Date) as promonth
FROM Operation_data
Group by MONTH(Production_Date)
ORDER BY AVG(Employee_Rating)DESC

--Q13 Average downtime(downtime duration)  by month
```

100 %

Results | Messages

	AvgempPerf	promonth
1	3	6
2	3	5
3	3	8
4	2	7
5	2	4
6	2	9

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 6 rows

The number of times a slack in service happened was recorded as the average downtime in months.

```
--Q12 Average employees performance(employees rating) by month
SELECT AVG(employee_rating) as AvgempPerf,MONTH(production_Date) as promonth
FROM Operation_data
Group by MONTH(Production_Date)
ORDER BY AVG(Employee_Rating)DESC

--Q13 Average downtime(downtime duration)  by month
SELECT AVG(Downtime_Duration) as avgDownTime,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
ORDER BY AVG(DownTime_Duration)DESC

--Q14 Average lead time by month
```

100 %

Results | Messages

	avgDownTime	(No column name)
1	2.75374515451623	7
2	2.62022131975786	6
3	2.49457954327265	4
4	2.45440268516541	5
5	2.41882705499255	8
6	2.30863244785285	9

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 6 rows

To determine the effect of the availability and unavailability of raw materials for production, the average lead-time was calculated and it was seen that there was no significant difference between the months.

```
--Q13 Average downtime(downtime duration)  by month
SELECT AVG(Downtime_Duration) as avgDownTime,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
ORDER BY AVG(DownTime_Duration)DESC

--Q14 Average lead time by month
SELECT AVG(CAST(Lead_Time as Decimal(10,2))) as AvgLeadTime,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
Order by AvgLeadTime DESC

--Q15 Total production quantity by month
```

100 %

Results | Messages

	AvgLeadTime	(No column name)
1	5.763313	5
2	5.740740	8
3	5.732919	9
4	5.444444	6
5	5.349112	7
6	4.920000	4

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 6 rows

The quantity of products produced at a particular month was analyzed to determine the production month, the quantity produced and the expiry date.

```
--Q14 Average lead time by month
SELECT AVG(CAST(Lead_Time as Decimal(10,2))) as AvgLeadTime,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
Order by AvgLeadTime DESC

--Q15 Total production quantity by month
SELECT SUM(production_quantity) as TotalProQty,MONTH(Production_date) as ProMonth
FROM Operation_data
Group by MONTH(Production_Date)
Order by SUM(Production_quantity)Desc

-- Q16 Quality metrics by each products
```

100 %

Results | Messages

	TotalProQty	ProMonth
1	102318	8
2	96790	5
3	96140	9
4	95495	7
5	90950	6
6	82806	4

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 6 rows

The quality metrics also known as the KPI's were counted inorder to measure the quality of the products, the processes involved and the services rendered. It tracked the effectiveness of the quality control processes during production of each products. From the view below, there was no significant difference in the quality of all the products as all of them are within the range of **132** to **156**

```
Group by MONTH(Production_Date)
Order by SUM(Production_quantity)Desc

-- Q16 Quality metrics by each products
SELECT product_ID,COUNT(Quantity_sold) as Quality_metrics
FROM Operation_data
Group by Product_ID
Order by COUNT(Quantity_Sold)DESC


-- Q17 Quantity in stock by product
select Product_ID,SUM(Production_Quantity) as Quantity_in_stock
FROM Operation_data
```

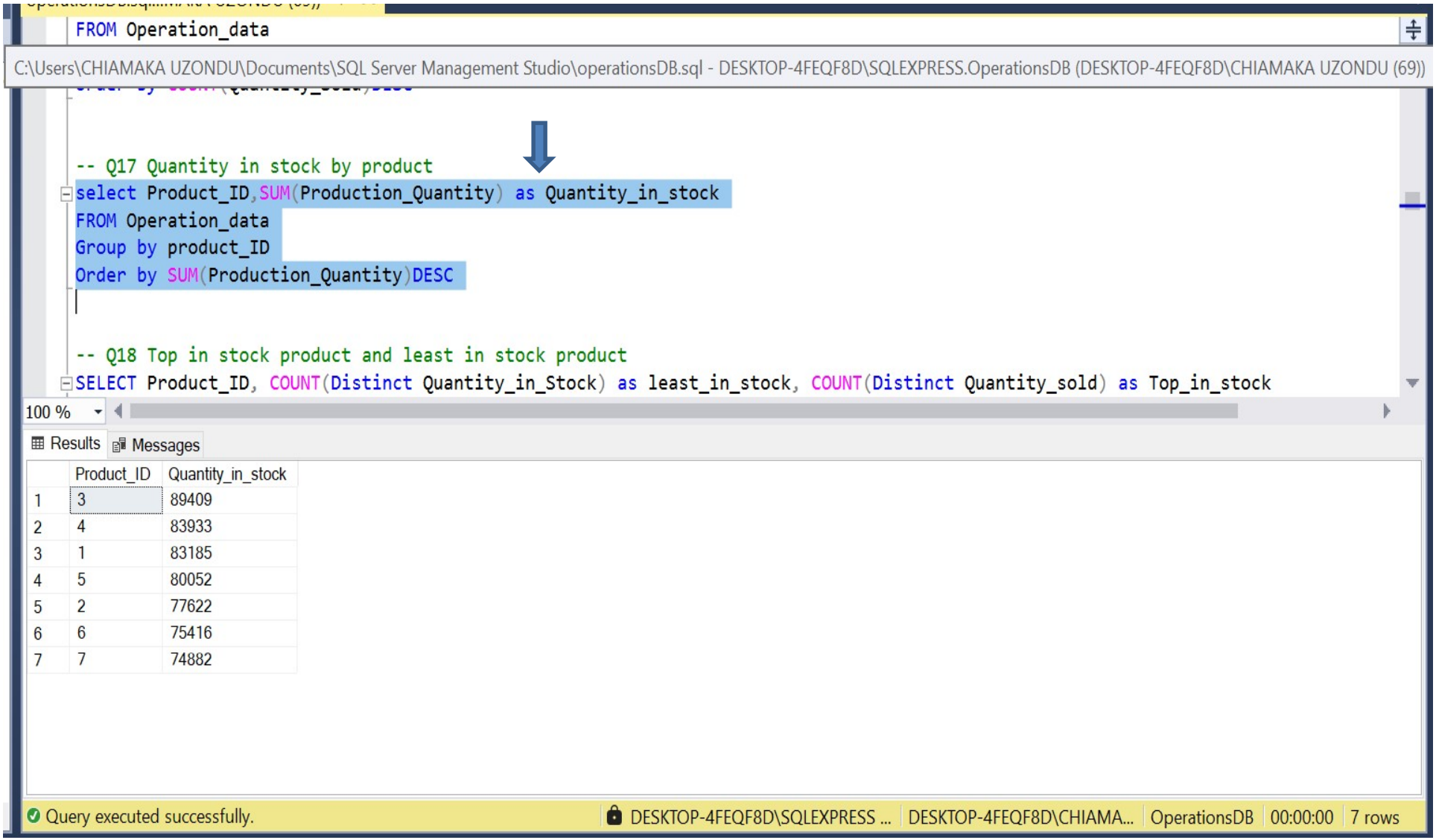
100 %

Results | Messages

	product_ID	Quality_metrics
1	3	156
2	5	148
3	1	147
4	4	145
5	6	137
6	7	135
7	2	132

Query executed successfully. DESKTOP-4FEQF8D\SQLEXPRESS ... DESKTOP-4FEQF8D\CHIAMA... OperationsDB 00:00:00 7 rows

The inventory quantity of stock was calculated base on products. This was done by counting each products in the store seperatly to determine the exact amount of each products in the inventory.



The top and least product in stock were counted inorder to categorize products base on importance and to track minimum and maximum stock levels.

```
select Product_ID,SUM(Production_Quantity) as Quantity_in_stock
FROM Operation_data
Group by product_ID
Order by SUM(Production_Quantity)DESC


-- Q18 Top in stock product and least in stock product
SELECT Product_ID, COUNT(Distinct Quantity_in_Stock) as least_in_stock, COUNT(Distinct Quantity_sold) as Top_in_stock
FROM Operation_data
Group by product_ID
Order by SUM(Production_Quantity)DESC


-- Q19 product with the top average production time
```

100 %

Results | Messages

	Product_ID	least_in_stock	Top_in_stock
1	3	131	111
2	4	127	112
3	1	120	113
4	5	125	108
5	2	114	96
6	6	112	101
7	7	115	95

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 7 rows

The products with the top average production time was calculated to determine the product that can be produced in bulk than the others especially those requiring craftmanship and manufacturing processes.

```
-- Q19 product with the top average production time
SELECT AVG(CAST(Production_Time as decimal)) as Top_Average_Pro_Time,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
Order by AVG(production_time)DESC

SELECT AVG(CAST(Lead_time as decimal)) as Top_Average_Pro_Time,MONTH(Production_Date)
FROM Operation_data
Group by MONTH(production_Date)
Order by AVG(Production_Time)DESC

-- Q20 product with the least average production time
SELECT AVG(Downtime_Duration) as least_Avg_pro_Time, MONTH(Production_Date)
```

100 %

Results | Messages

	Top_Average_Pro_Time	(No column name)
1	7.857988	5
2	7.840000	4
3	7.782608	9
4	7.586419	6
5	7.579881	7
6	7.513227	8

	Top_Average_Pro_Time	(No column name)
1	5.763313	5
2	4.920000	4
3	5.732919	9
4	5.444444	6

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 12 rows

Products with the least average production time was also calculated to record the products that rely on streamlined and automated machinery and standardized processes to achieve rapid production.

```
Group by MONTH(production_Date)
Order by AVG(Production_Time)DESC

-- Q20 product with the least average production time
SELECT AVG(Downtime_Duration) as least_Avg_pro_Time, MONTH(Production_Date)
FROM Operation_data
Group by MONTH(Production_Date)
Order by AVG(Production_Time)DESC

-- Q21 most produced product and the least produced product
SELECT product_ID, COUNT(Distinct Production_Quantity) as most_produced_product,COUNT(Distinct Quantity_in_Stock) as least_produce
FROM operation_data
Group by Product_ID
order by SUM(Production_Quantity)DESC
```

100 %

Results | Messages

	least_Avg_pro_Time	(No column name)
1	2.45440268516541	5
2	2.49457954327265	4
3	2.30863244785285	9
4	2.62022131975786	6
5	2.75374515451623	7
6	2.41882705499255	8

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 6 rows

DISTINCT COUNT was used to determine the exact most and least produced products inorder to avoid any duplicate that will bring error to the analysis.

```
Group by MONTH(Production_Date)
Order by AVG(Production_Time)DESC

-- Q21 most produced product and the least produced product
SELECT product_ID, COUNT(Distinct Production_Quantity) as most_produced_product,COUNT(Distinct Quantity_in_Stock) as least_produce
FROM operation_data
Group by Product_ID
order by SUM(Production_Quantity)DESC

-- Q22 show each quality metrics by month
SELECT Product_ID,COUNT(Quality_metrics) as Quality_by_months
FROM Operation_data
Group by product_ID
Order by COUNT(Quality_metrics)
```

100 %

Results | Messages

	product_ID	most_produced_product	least_produced_product
1	3	141	131
2	4	135	127
3	1	136	120
4	5	134	125
5	2	122	114
6	6	128	112
7	7	124	115

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 7 rows

The quality metrics by month was recorded to show the product quality, monthly yield, supplier quality metrics, overall equipment effectiveness. All these were calculated and recorded in numbers named quality by month

```
-- Q21 most produced product and the least produced product
SELECT product_ID, COUNT(Distinct Production_Quantity) as most_produced_product,COUNT(Distinct Quantity_in_Stock) as least_produce
FROM operation_data
Group by P
order by S

-- Q22 show each quality metrics by month
SELECT Product_ID,COUNT(Quality_metrics) as Quality_by_months
FROM Operation_data
Group by product_ID
Order by COUNT(Quality_metrics)

-- Q23 most produced and sold employee (pro qty and qty sold)
SELECT Employee_ID,COUNT(Distinct Production_Quantity) as most_produced,COUNT(Distinct Quantity_Sold) as sold_employee
```

table OperationsDB.dbo.Operation_data

100 %

Results | Messages

	Product_ID	Quality_by_months
1	2	132
2	7	135
3	6	137
4	4	145
5	1	147
6	5	148
7	3	156

Query executed successfully. DESKTOP-4FEQF8D\SQLEXPRESS ... DESKTOP-4FEQF8D\CHIAMA... OperationsDB 00:00:00 7 rows

The employees who produced the highest amount of products and those that sold the most products were counted and recorded accordingly.

```
-- Q22 show each quality metrics by month
SELECT Product_ID,COUNT(Quality_metrics) as Quality_by_months
FROM Operation_data
Group by product_ID
Order by COUNT(Quality_metrics)

-- Q23 most produced and sold employee (pro qty and qty sold)
SELECT Employee_ID,COUNT(Distinct Production_Quantity) as most_produced,COUNT(Distinct Quantity_Sold) as sold_employee
FROM Operation_data
Group by Employee_ID
Order by SUM(Production_Quantity)DESC

-- Q24 Number of employees by each rating
SELECT Employee_ID,COUNT(Distinct Employee_Rating) as number_of_employees
```

100 %

Results | Messages

	Employee_ID	most_produced	sold_employee
1	8	109	88
2	3	100	79
3	5	103	88
4	6	99	75
5	4	94	80
6	1	93	72
7	9	94	71
8	10	92	72
9	2	80	64
10	7	83	65

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 10 rows

On a scale of 1 -5,the employees were rated and after analysis they all stood in the range of 5.

```sql
-- Q23 most produced and sold employee (pro qty and qty sold)
SELECT Employee_ID,COUNT(Distinct Production_Quantity) as most_produced,COUNT(Distinct Quantity_Sold) as sold_employee
FROM Operation_data
Group by Employee_ID
Order by SUM(Production_Quantity)DESC

-- Q24 Number of employees by each rating
SELECT Employee_ID,COUNT(Distinct Employee_Rating) as number_of_employees
FROM Operation_data
Group by Employee_ID
Order by COUNT(Employee_Rating)DESC

-- Q25 Most performing employee (sum of rating)
SELECT Employee_ID,SUM(Employee_Rating) as most_performing_employee
```

100 %

Results | Messages

	Employee_ID	number_of_employees
1	8	5
2	5	5
3	6	5
4	3	5
5	4	5
6	1	5
7	9	5
8	10	5
9	7	5
10	2	5

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 10 rows

The employees who go over and beyond exceeding expectations and demonstrating initiatives were counted. This was done by the analysis of how they can manage time and raw materials for production.

```
-- Q24 Number of employees by each rating
SELECT Employee_ID,COUNT(Distinct Employee_Rating) as number_of_employees
FROM Operation_data
Group by Employee_ID
Order by COUNT(Employee_Rating)DESC

-- Q25 Most performing employee (sum of rating)
SELECT Employee_ID,SUM(Employee_Rating) as most_performing_employee
FROM Operation_data
Group by Employee_ID
Order by SUM(Employee_Rating)DESC

-- Q26 Least performing employee
SELECT Employee_ID,COUNT(Employee_Rating) as least_performing_employee
```

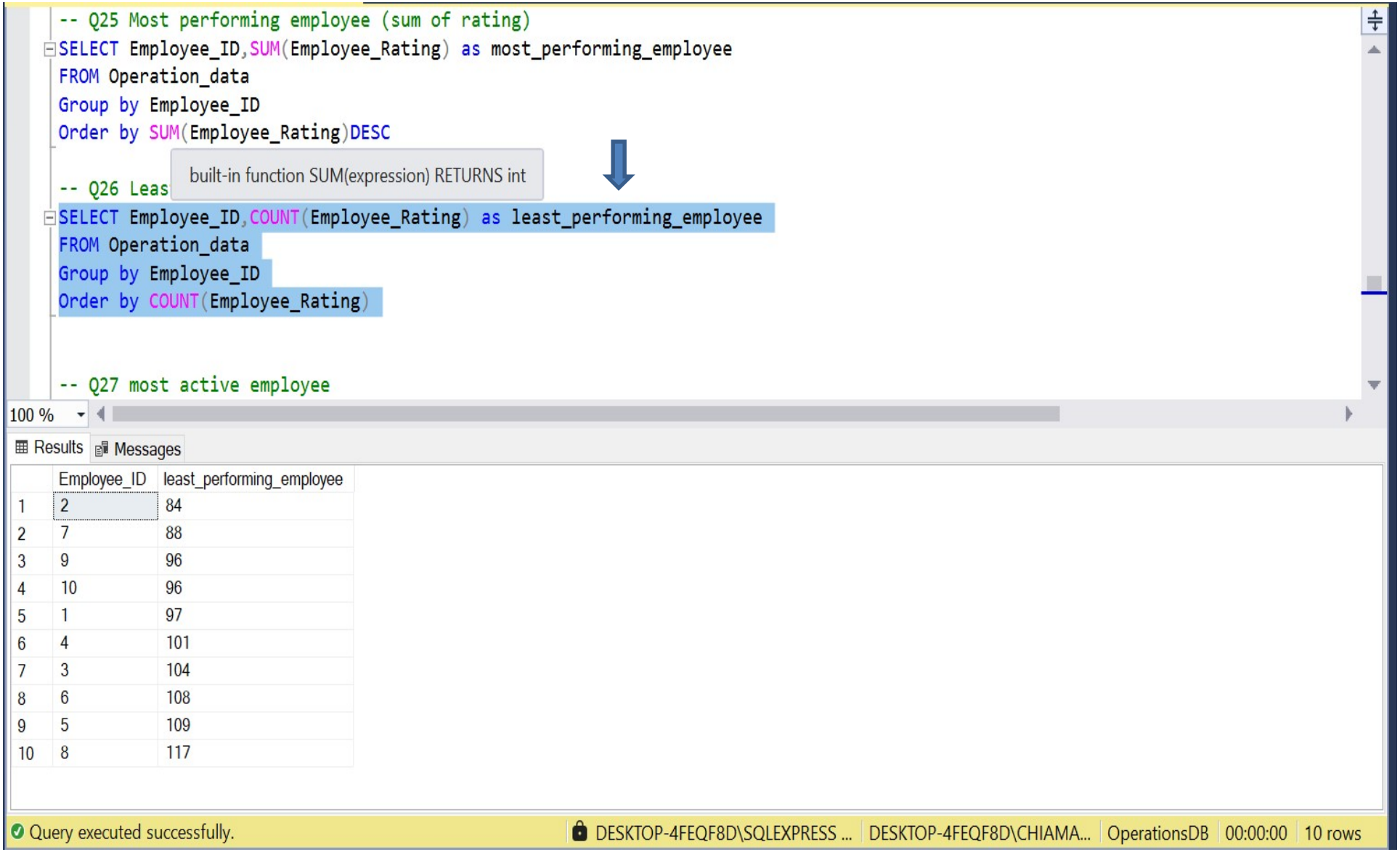
100 %

Results | Messages

	Employee_ID	most_performing_employee
1	8	344
2	6	320
3	5	320
4	1	314
5	9	293
6	3	293
7	4	281
8	10	273
9	7	266
10	2	264

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 10 rows

On the other hand, the least performing employees were also counted base on time management and the amount of products produced and sold at a particular time.



The employees who are highly engaged and involved in their work were counted and recorded. This was calculated by taking cognizance of reliability, punctuality, communication, positive attitude, adaptability etc.

```
Group by Employee_ID
Order by COUNT(Employee_Rating)


-- Q27 most active employee
SELECT Employee_ID,COUNT(Operations_ID) as most_active_employee
FROM Operation_data
Group by Employee_ID
Order by COUNT(Operations_ID)DESC

-- Q28 Avg lead time by suppliers
SELECT supplier_ID,AVG(Lead_Time) as Avg_LT_by_suppliers
FROM Operation_data
Group by supplier_ID
```

100 %

Results | Messages

	Employee_ID	most_active_employee
1	8	117
2	5	109
3	6	108
4	3	104
5	4	101
6	1	97
7	10	96
8	9	96
9	7	88
10	2	84

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 10 rows

The overall time it took the suppliers to complete the processes of supplies from from procurement to delivery were counted and recorded in the ratio of **1:5** and when analyzed, all the supplies were seen in the range of **5**.

```
-- Q27 most active employee
SELECT Employee_ID,COUNT(Operations_ID) as most_active_employee
FROM Operation_data
Group by Employee_ID
Order by COUNT(Operations_ID)DESC

-- Q28 Avg lead time by suppliers
SELECT supplier_ID,AVG(Lead_Time) as Avg_LT_by_suppliers
FROM Operation_data
Group by supplier_ID
Order by SUM(Lead_Time)DESC
```

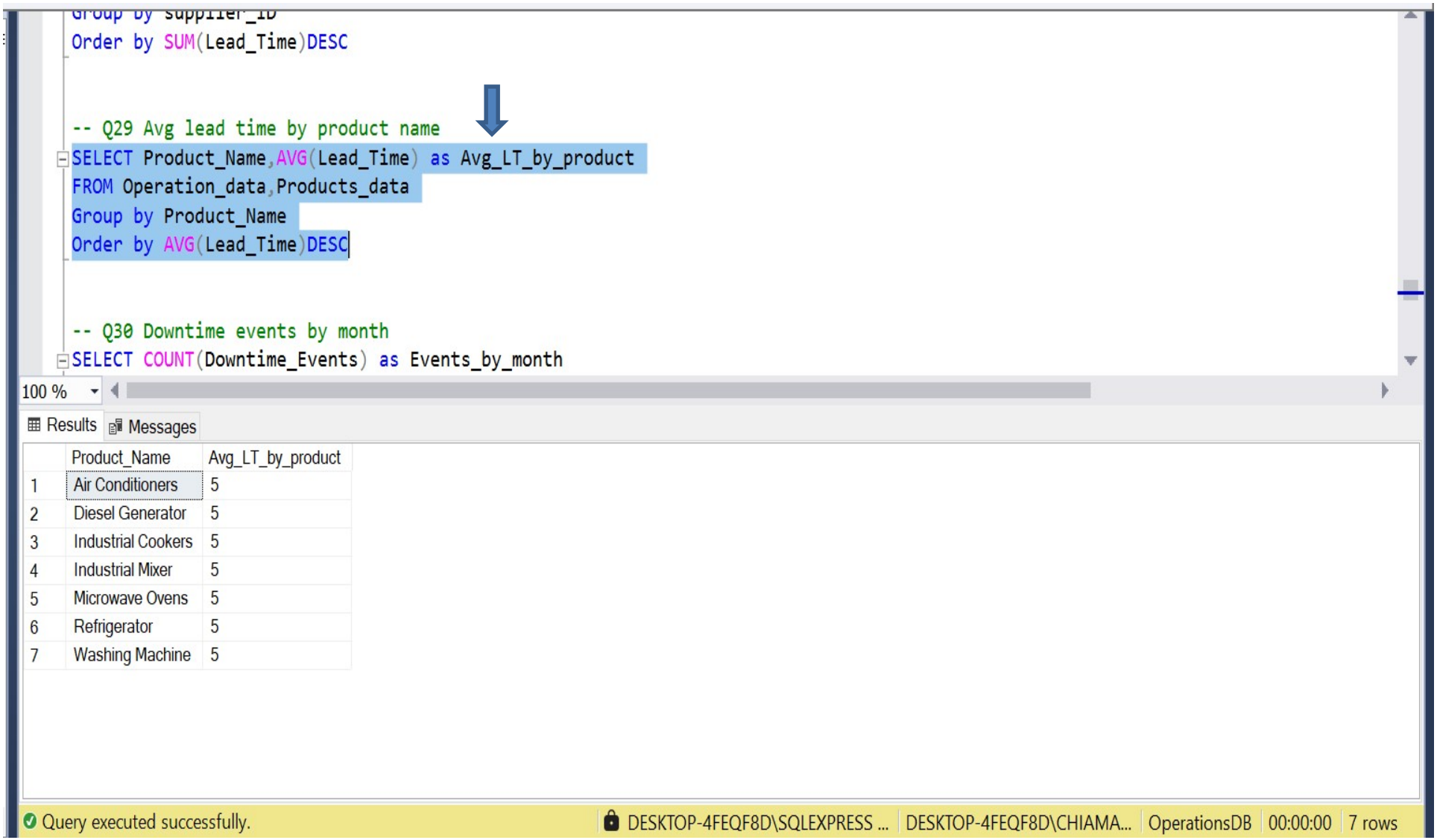
100 %

Results | Messages

	supplier_ID	Avg_LT_by_suppliers
1	10	5
2	5	5
3	4	5
4	3	5
5	9	5
6	1	5
7	2	5
8	7	5
9	8	5
10	6	5

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 10 rows

The analysis of recording the average lead-time of a product followed the process of calculating the time it takes from when the customer places an order for a particular product to when it was delivered. This overall analysis recorded **5** in all.



The downtime events by month was done to identify trends, root causes and to derive preventive measures to reduce future downtime and improve operational efficiency. After the analysis, the down-time was recorded as seen on a monthly bases.

```
SELECT Product_Name,AVG(Lead_Time) as Avg_LT_by_product
FROM Operation_data,Products_data
Group by Product_Name
Order by AVG(Lead_Time)DESC


-- Q30 Downtime events by month
SELECT COUNT(Downtime_Events) as Events_by_month
FROM Operation_data
Group by MONTH(Production_Date)
Order by COUNT(Downtime_Events)

-- Q31 Downtime events by product
```

100 %

Results | Messages

	Events_by_month
1	150
2	161
3	162
4	169
5	169
6	189

Query executed successfully. DESKTOP-4FEQF8D\SQLEXPRESS ... DESKTOP-4FEQF8D\CHIAMA... OperationsDB 00:00:00 6 rows

The downtime event by products were analyzed to track and identify recurring issues and to record areas of improvement. This was done to help in optimizing production processes and reducing cost associated with downtime. The analysis recorded different downtime by products and was duly recorded.

```
Group by MONTH(Production_Date)
Order by COUNT(Downtime_Events)


-- Q31 Downtime events by product
SELECT product_ID,COUNT(Downtime_Events) as Events_by_product
FROM Operation_data
Group by (Product_ID)
Order by COUNT(Downtime_Events)


SELECT * FROM employees_data
SELECT * FROM Operation_data
SELECT * FROM products_data
```

100 %

Results | Messages

	product_ID	Events_by_product
1	2	132
2	7	135
3	6	137
4	4	145
5	1	147
6	5	148
7	3	156

Query executed successfully. | DESKTOP-4FEQF8D\SQLEXPRESS ... | DESKTOP-4FEQF8D\CHIAMA... | OperationsDB | 00:00:00 | 7 rows

RECOMMENDATIONS

- I recommend smaller and more strategic supplier base because it allows for more effective, closer relationships with customers and producers and better quality control.
- I recommend that customers satisfaction should be tracked to know if the products will be produced in bulk or in minute quantity.
- Leverage reviews to build trust and credibility.

  Promote daily offers
- Highlight bestsellers by displaying the hottest items as popular products, customer favourites. This makes them more enticing to consumers who always want to get their hands on the best and most trending products.