

DataBase System LAB-4



Session 2022 - 2026

Abdul Rehman 2022-CS-79

Submitted To:

Mr. Nazeef Ul Haq

Department of Computer Science
University of Engineering and Technology Lahore Pakistan

Contents

| | | |
|----------|--|-----------|
| 1 | Question 1: | 4 |
| 1.1 | List name of all the products whose price is above average. (Product Name) . . . | 4 |
| 1.2 | Result | 4 |
| 2 | Question 2: | 5 |
| 2.1 | Write a query to generate report showing date wise orders shipped. (Shipped- Date, numberoforders) | 5 |
| 2.2 | Result | 5 |
| 3 | Question 3: | 6 |
| 3.1 | List name of all countries from where two or more suppliers belong to. (Country) | 6 |
| 3.2 | Result | 6 |
| 4 | Question 4: | 7 |
| 4.1 | Write a query to generate report showing month wise orders delayed shipped. Your output should look like this (Month Number, Orders Delayed) | 7 |
| 4.2 | Result | 7 |
| 5 | Question 5: | 8 |
| 5.1 | Report all the orders which have been discounted. Your result should show the total discount against each order. Output should look like this (Order ID, Discount) | 8 |
| 5.2 | Result | 8 |
| 6 | Question 6: | 9 |
| 6.1 | Write a query to list the number of orders which were shipped in the cities of USA in 1997. Show the number of order against each city. (Ship City, Number of orders) | 9 |
| 6.2 | Result | 9 |
| 7 | Question 7: | 10 |
| 7.1 | Write a query to generate report showing country wise orders delayed shipped. Your output should look like this: (Country, Orders Delayes) | 10 |
| 7.2 | Result | 10 |
| 8 | Question 8: | 11 |

| | | |
|----------|---|-----------|
| 8.1 | <i>Report all the orders which have been discounted with total price of order. Your result should show the total discount against each order. Output should look like this: (Order ID, Discount, Total Price)</i> | <i>11</i> |
| 8.2 | <i>Result</i> | <i>11</i> |
| 9 | Question 9: | 12 |
| 9.1 | <i>Write a query to list the number of orders which were shipped in the cities of each region in 1997. Show the number of order against each city. Your results should look like this: (ShipRegion, ShipCity, Numberoforders)</i> | <i>12</i> |
| 9.2 | <i>Result</i> | <i>12</i> |

1 Question 1:

1.1 List name of all the products whose price is above average. (Product Name)

SQL Query:

```
SELECT [ProductName]
FROM [Northwind].[dbo].[Products]
WHERE UnitPrice > (SELECT AVG(Products.UnitPrice) FROM [Northwind].[dbo].[P
```

1.2 Result

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the 'Object Explorer' with the 'Northwind' database selected. The right pane shows the 'SQL Query Editor' with the following query:

```
-- List name of all the products whose price is above average. (Product Name)
SELECT [ProductName]
FROM [Northwind].[dbo].[Products]
WHERE UnitPrice > (SELECT AVG(Products.UnitPrice) FROM [Northwind].[dbo].[Products]);

-- Write a query to generate report showing date wise orders shipped. (ShippedDate, numberoforders)
SELECT [ShippedDate], COUNT(*) as [numberoforders]
FROM [Northwind].[dbo].[Orders]
GROUP BY ShippedDate;

-- List name of all countries from where two or more suppliers belong to. (Country)
SELECT [Country]
```

The 'Results' pane shows the output of the first query, listing 16 products with their names:

| ProductID | ProductName |
|-----------|---------------------------------|
| 1 | Uncle Bob's Organic Dried Pears |
| 2 | Northwoods Cranberry Sauce |
| 3 | Maisie Kobe Niku |
| 4 | Buns |
| 5 | Queso Manchego La Pastora |
| 6 | Alice Mutton |
| 7 | Carnarvon Tigers |
| 8 | Sir Rodney's Marmalade |
| 9 | Gumbär Gummbirichen |
| 10 | Schoggi Schokolade |
| 11 | Rössle Sauerkraut |
| 12 | Thüringer Rostbratwurst |
| 13 | Mascarpone Fabiol |
| 14 | Côte de Blaye |
| 15 | Irish Coffee |
| 16 | Marzipan Dried Apples |

The status bar at the bottom indicates 'Query executed successfully' and '25 rows'.

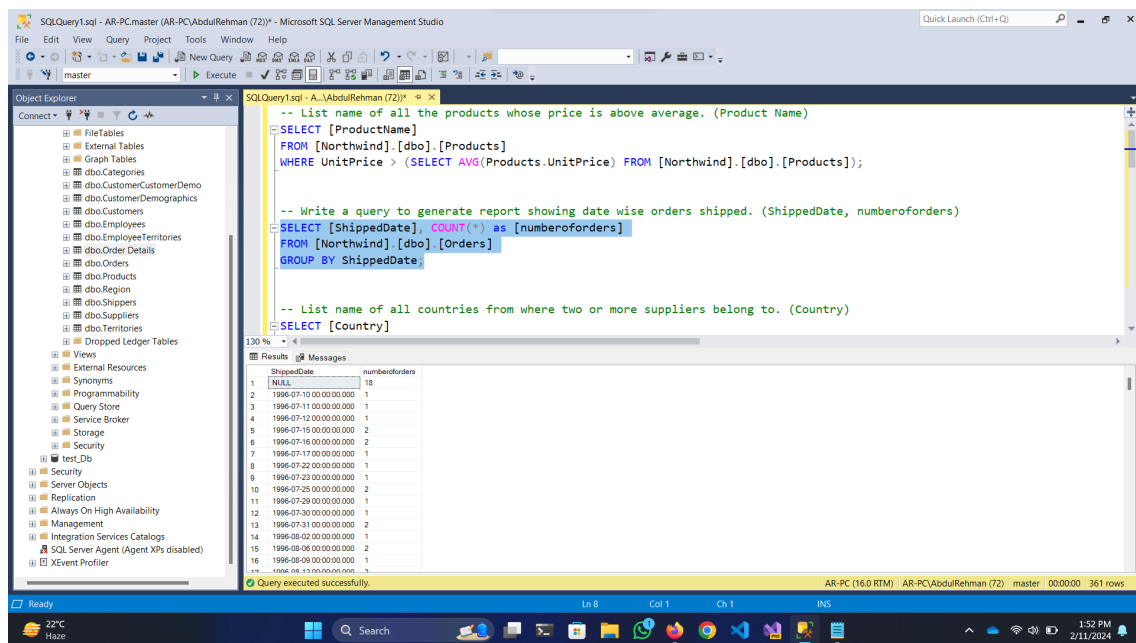
2 Question 2:

2.1 Write a query to generate report showing date wise orders shipped. (ShippedDate, numberoforders)

SQL Query:

```
SELECT [ShippedDate], COUNT(*) as [numberoforders]
FROM [Northwind].[dbo].[Orders]
GROUP BY ShippedDate;
```

2.2 Result



The screenshot displays the Microsoft SQL Server Management Studio interface. The query editor shows a SQL query to list products with prices above the average, followed by the query for the report, and then a query to list countries with two or more suppliers. The results pane shows the output of the third query, listing countries and the number of suppliers.

```
-- List name of all the products whose price is above average. (Product Name)
SELECT [ProductName]
FROM [Northwind].[dbo].[Products]
WHERE UnitPrice > (SELECT AVG(Products.UnitPrice) FROM [Northwind].[dbo].[Products]);

-- Write a query to generate report showing date wise orders shipped. (ShippedDate, numberoforders)
SELECT [ShippedDate], COUNT(*) as [numberoforders]
FROM [Northwind].[dbo].[Orders]
GROUP BY ShippedDate;

-- List name of all countries from where two or more suppliers belong to. (Country)
SELECT [Country]
```

| ShippedDate | numberoforders |
|-------------------------|----------------|
| 1996-07-10 00:00:00.000 | 1 |
| 1996-07-11 00:00:00.000 | 1 |
| 1996-07-12 00:00:00.000 | 1 |
| 1996-07-15 00:00:00.000 | 2 |
| 1996-07-16 00:00:00.000 | 2 |
| 1996-07-17 00:00:00.000 | 1 |
| 1996-07-22 00:00:00.000 | 1 |
| 1996-07-23 00:00:00.000 | 1 |
| 1996-07-25 00:00:00.000 | 2 |
| 1996-07-29 00:00:00.000 | 1 |
| 1996-07-30 00:00:00.000 | 1 |
| 1996-07-31 00:00:00.000 | 2 |
| 1996-08-02 00:00:00.000 | 1 |
| 1996-08-06 00:00:00.000 | 2 |
| 1996-08-08 00:00:00.000 | 1 |
| 1996-08-13 00:00:00.000 | 1 |

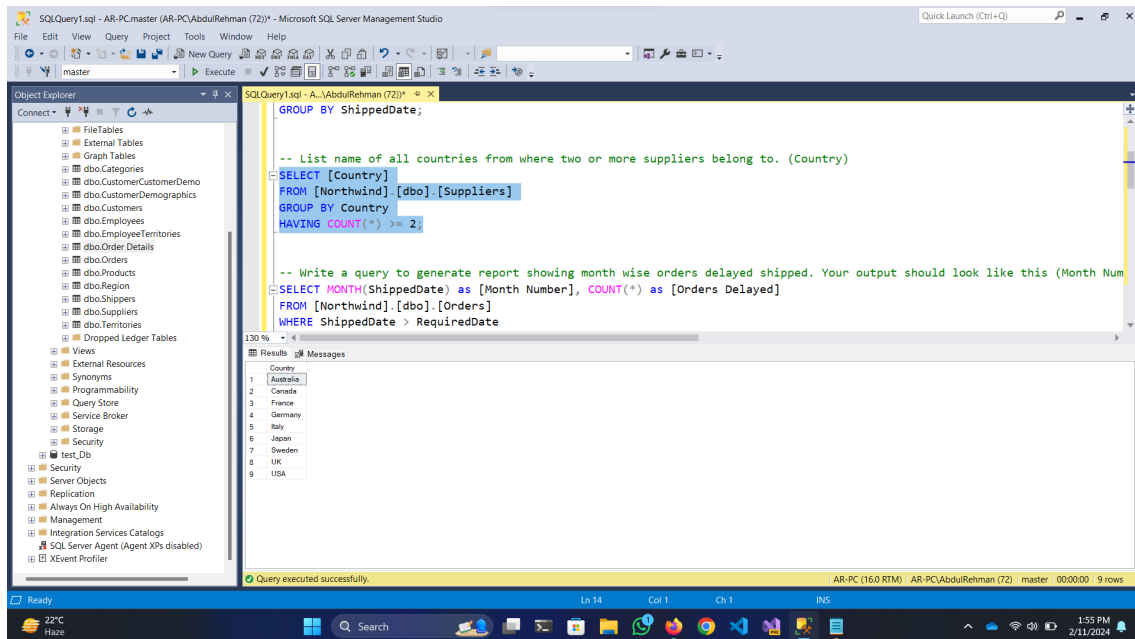
3 Question 3:

3.1 List name of all countries from where two or more suppliers belong to. (Country)

SQL Query:

```
SELECT [Country]
FROM [Northwind].[dbo].[Suppliers]
GROUP BY Country
HAVING COUNT(*) >= 2;
```

3.2 Result



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor displays the following SQL query:

```
GROUP BY ShippedDate;

-- List name of all countries from where two or more suppliers belong to. (Country)
SELECT [Country]
FROM [Northwind].[dbo].[Suppliers]
GROUP BY Country
HAVING COUNT(*) >= 2;

-- Write a query to generate report showing month wise orders delayed shipped. Your output should look like this (Month Num
SELECT MONTH(ShippedDate) as [Month Number], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
```

The Results pane shows the output of the first query, which is a list of countries:

| Country |
|-------------|
| 1 Australia |
| 2 Canada |
| 3 France |
| 4 Germany |
| 5 Italy |
| 6 Japan |
| 7 Sweden |
| 8 UK |
| 9 USA |

The status bar at the bottom indicates that the query was executed successfully, returning 9 rows.

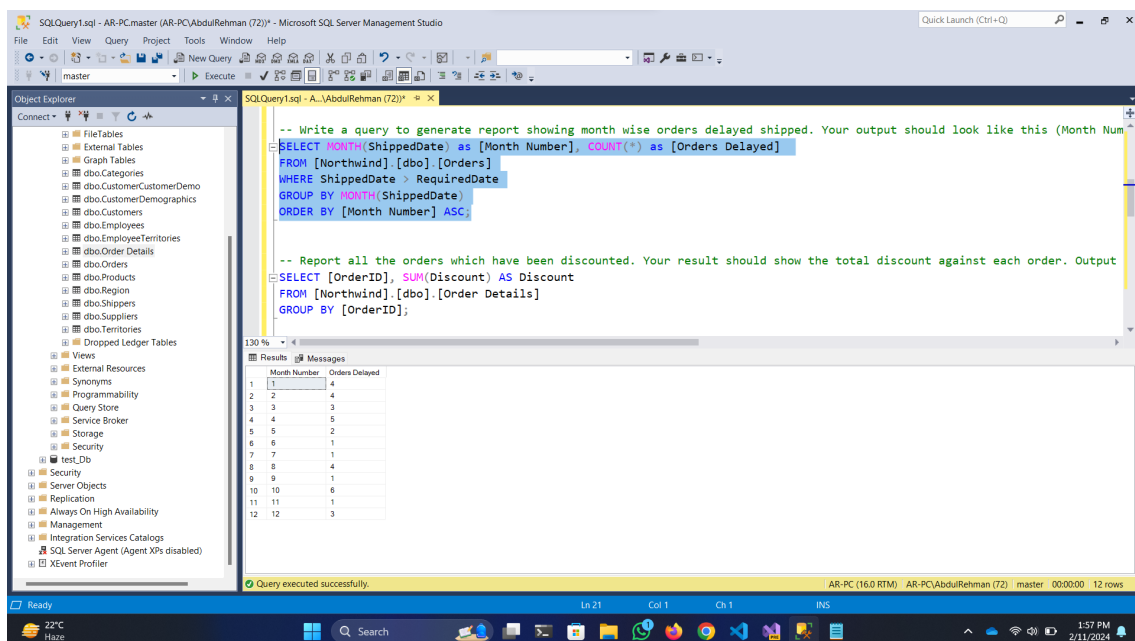
4 Question 4:

4.1 Write a query to generate report showing month wise orders delayed shipped. Your output should look like this (Month Number, Orders Delayed)

SQL Query:

```
SELECT MONTH(ShippedDate) as [Month Number], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
GROUP BY MONTH(ShippedDate)
ORDER BY [Month Number] ASC;
```

4.2 Result



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL query:

```
-- Write a query to generate report showing month wise orders delayed shipped. Your output should look like this (Month Number, Orders Delayed)
SELECT MONTH(ShippedDate) as [Month Number], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
GROUP BY MONTH(ShippedDate)
ORDER BY [Month Number] ASC;

-- Report all the orders which have been discounted. Your result should show the total discount against each order. Output
SELECT [OrderID], SUM(Discount) AS Discount
FROM [Northwind].[dbo].[Order Details]
GROUP BY [OrderID];
```

The query results are displayed in the Results pane, showing a table with two columns: Month Number and Orders Delayed. The data is as follows:

| Month Number | Orders Delayed |
|--------------|----------------|
| 1 | 4 |
| 2 | 4 |
| 3 | 3 |
| 4 | 5 |
| 5 | 2 |
| 6 | 1 |
| 7 | 1 |
| 8 | 4 |
| 9 | 1 |
| 10 | 6 |
| 11 | 1 |
| 12 | 3 |

The status bar at the bottom indicates that the query was executed successfully, returning 12 rows.

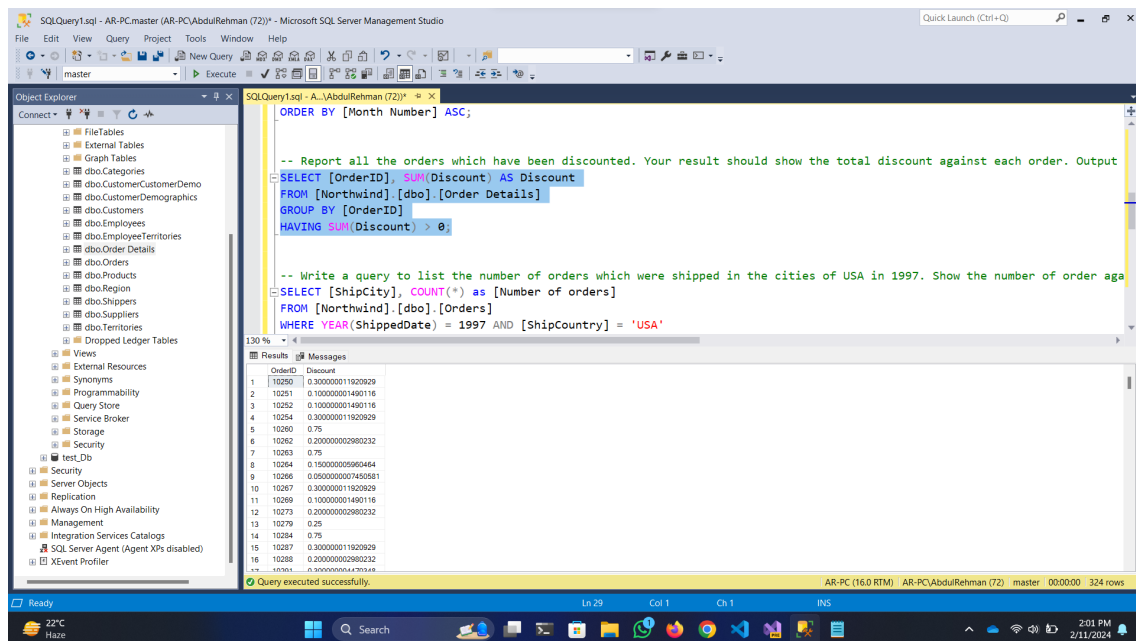
5 Question 5:

5.1 Report all the orders which have been discounted. Your result should show the total discount against each order. Output should look like this (Order ID, Discount)

SQL Query:

```
SELECT [OrderID], SUM(Discount) AS Discount
FROM [Northwind].[dbo].[Order Details]
GROUP BY [OrderID]
HAVING SUM(Discount) > 0;
```

5.2 Result



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
ORDER BY [Month Number] ASC;

-- Report all the orders which have been discounted. Your result should show the total discount against each order. Output
SELECT [OrderID], SUM(Discount) AS Discount
FROM [Northwind].[dbo].[Order Details]
GROUP BY [OrderID]
HAVING SUM(Discount) > 0;

-- Write a query to list the number of orders which were shipped in the cities of USA in 1997. Show the number of order aga
SELECT [ShipCity], COUNT(*) as [Number of orders]
FROM [Northwind].[dbo].[Orders]
WHERE YEAR([ShippedDate]) = 1997 AND [ShipCountry] = 'USA'
```

The Results pane shows the output of the first query, displaying a table with two columns: OrderID and Discount. The data is as follows:

| OrderID | Discount |
|---------|--------------------|
| 10250 | 0.3000000011920929 |
| 10251 | 0.1000000001490116 |
| 10252 | 0.1000000001490116 |
| 10254 | 0.3000000011920929 |
| 10260 | 0.75 |
| 10262 | 0.200000002980232 |
| 10263 | 0.75 |
| 10264 | 0.1500000005960464 |
| 10266 | 0.0500000007450581 |
| 10267 | 0.3000000011920929 |
| 10269 | 0.1000000001490116 |
| 10273 | 0.200000002980232 |
| 10279 | 0.25 |
| 10284 | 0.75 |
| 10287 | 0.3000000011920929 |
| 10288 | 0.200000002980232 |
| 10301 | 0.3000000011920929 |

The status bar at the bottom indicates that the query was executed successfully, returning 324 rows.

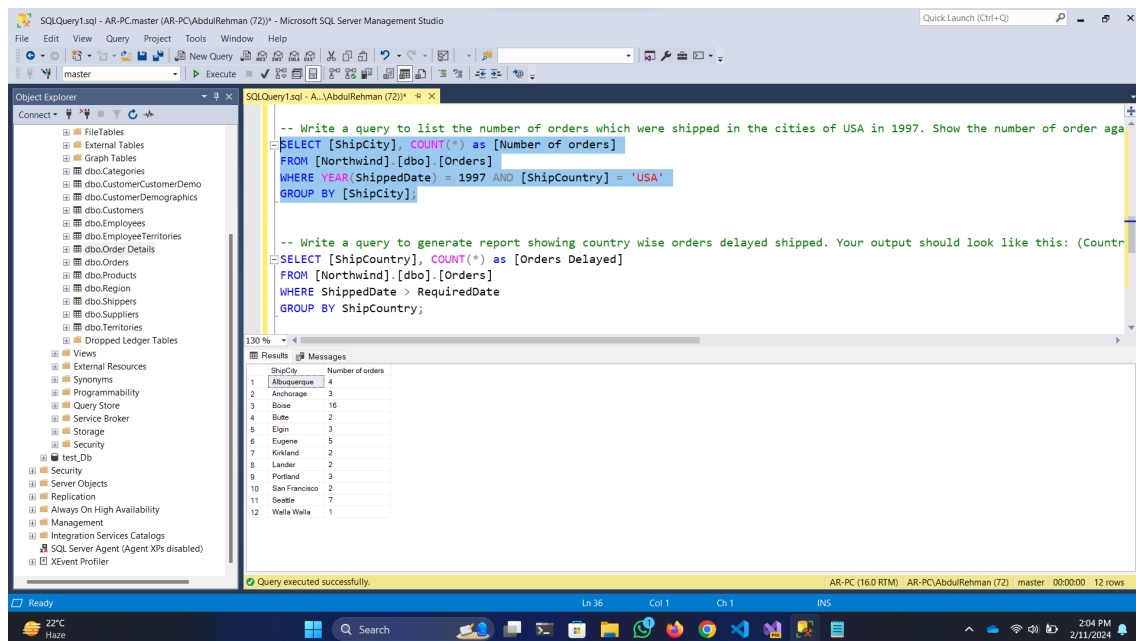
6 Question 6:

6.1 Write a query to list the number of orders which were shipped in the cities of USA in 1997. Show the number of order against each city. (Ship City, Number of orders)

SQL Query:

```
SELECT [ShipCity], COUNT(*) as [Number of orders]
FROM [Northwind].[dbo].[Orders]
WHERE YEAR(ShippedDate) = 1997 AND [ShipCountry] = 'USA'
GROUP BY [ShipCity];
```

6.2 Result



The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The left pane shows the Object Explorer with the 'test_Db' database selected. The right pane shows the SQL Query window with the following query:

```
-- Write a query to list the number of orders which were shipped in the cities of USA in 1997. Show the number of order aga
SELECT [ShipCity], COUNT(*) as [Number of orders]
FROM [Northwind].[dbo].[Orders]
WHERE YEAR(ShippedDate) = 1997 AND [ShipCountry] = 'USA'
GROUP BY [ShipCity];

-- Write a query to generate report showing country wise orders delayed shipped. Your output should look like this: (Countr
SELECT [ShipCountry], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
GROUP BY ShipCountry;
```

The Results pane shows the output of the first query, displaying a table with two columns: ShipCity and Number of orders. The data is as follows:

| ShipCity | Number of orders |
|---------------|------------------|
| Anchorage | 4 |
| Anchorage | 3 |
| Boise | 16 |
| Butte | 2 |
| Elgin | 3 |
| Eugene | 5 |
| Kirkland | 2 |
| Lander | 2 |
| Portland | 3 |
| San Francisco | 2 |
| Seattle | 7 |
| Walla Walla | 1 |

The status bar at the bottom indicates that the query was executed successfully, returning 12 rows.

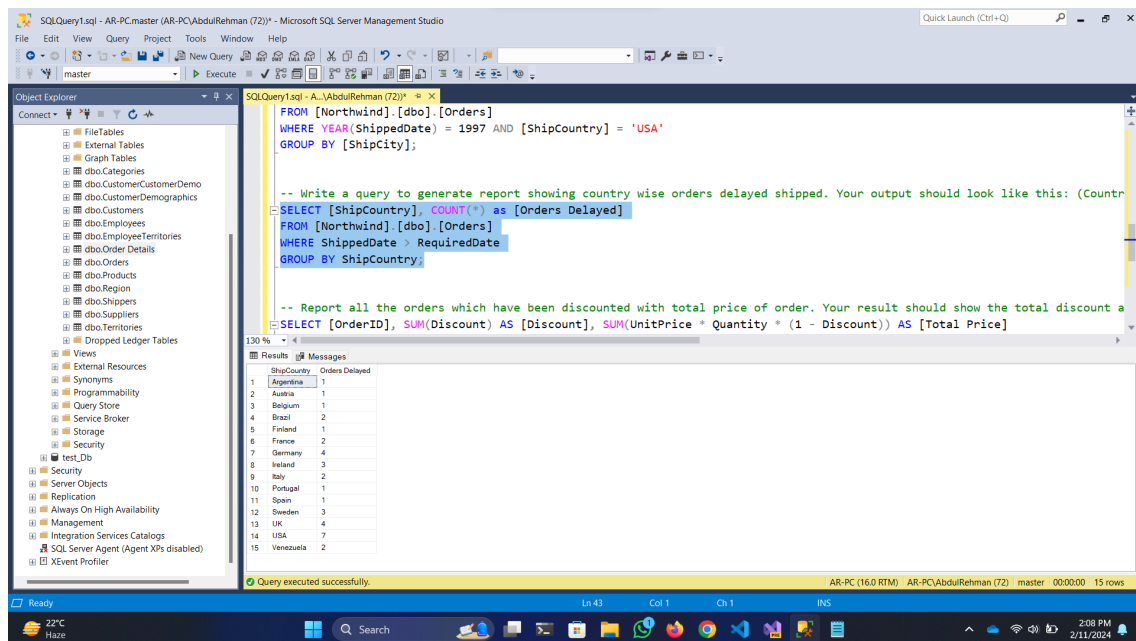
7 Question 7:

7.1 Write a query to generate report showing country wise orders delayed shipped. Your output should look like this: (Country, Orders Delays)

SQL Query:

```
SELECT [ShipCountry], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
GROUP BY ShipCountry;
```

7.2 Result



The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the 'test_Db' database selected. The right pane shows the SQL Query Editor with the following query:

```
-- Write a query to generate report showing country wise orders delayed shipped. Your output should look like this: (Country, Orders Delays)
SELECT [ShipCountry], COUNT(*) as [Orders Delayed]
FROM [Northwind].[dbo].[Orders]
WHERE ShippedDate > RequiredDate
GROUP BY ShipCountry;
```

Below the query editor, the Results pane displays the output of the query:

| ShipCountry | Orders Delayed |
|-------------|----------------|
| Argentina | 1 |
| Australia | 1 |
| Belgium | 1 |
| Brazil | 2 |
| Finland | 1 |
| France | 2 |
| Germany | 4 |
| Ireland | 3 |
| Italy | 2 |
| Portugal | 1 |
| Spain | 1 |
| Sweden | 3 |
| UK | 4 |
| USA | 7 |
| Venezuela | 2 |

The status bar at the bottom indicates that the query was executed successfully, returning 15 rows.

8 Question 8:

8.1 Report all the orders which have been discounted with total price of order. Your result should show the total discount against each order. Output should look like this: (Order ID, Discount, Total Price)

SQL Query:

```
SELECT [OrderID], SUM(Discount) AS [Discount], SUM(UnitPrice * Quantity * (1 - Discount)) AS [Total Price]
FROM [Northwind].[dbo].[Order Details]
WHERE Discount > 0
GROUP BY OrderID;
```

8.2 Result

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the 'Object Explorer' with the 'Server Objects' folder expanded, showing various database objects like Tables, Views, and Security. The right pane shows a SQL query window with the following query:

```
-- Report all the orders which have been discounted with total price of order. Your result should show the total discount a
SELECT [OrderID], SUM(Discount) AS [Discount], SUM(UnitPrice * Quantity * (1 - Discount)) AS [Total Price]
FROM [Northwind].[dbo].[Order Details]
WHERE Discount > 0
GROUP BY OrderID;
```

Below the query window, the 'Results' pane displays the output of the query as a table with 3 columns: OrderID, Discount, and Total Price. The table contains 16 rows of data.

| OrderID | Discount | Total Price |
|---------|-------------------|------------------|
| 10250 | 0.300000011920929 | 1475.60003662109 |
| 10251 | 0.100000001490116 | 318.60000187988 |
| 10252 | 0.100000001490116 | 2509.89990234375 |
| 10254 | 0.300000011920929 | 388.82003326416 |
| 10260 | 0.75 | 724.649993896484 |
| 10262 | 0.200000002980232 | 163.199999945242 |
| 10263 | 0.75 | 1773 |
| 10264 | 0.150000005990464 | 163.625 |
| 10266 | 0.050000007450581 | 346.99997958594 |
| 10267 | 0.300000011920929 | 2801.60000103052 |
| 10269 | 0.100000001490116 | 642.200012207031 |
| 10273 | 0.200000002980232 | 1997.27996826172 |
| 10279 | 0.25 | 351 |
| 10284 | 0.75 | 844.875 |
| 10287 | 0.300000011920929 | 595.000007629395 |
| 10288 | 0.200000002980232 | 80.999984741211 |

The status bar at the bottom indicates that the query was executed successfully, returning 324 rows.

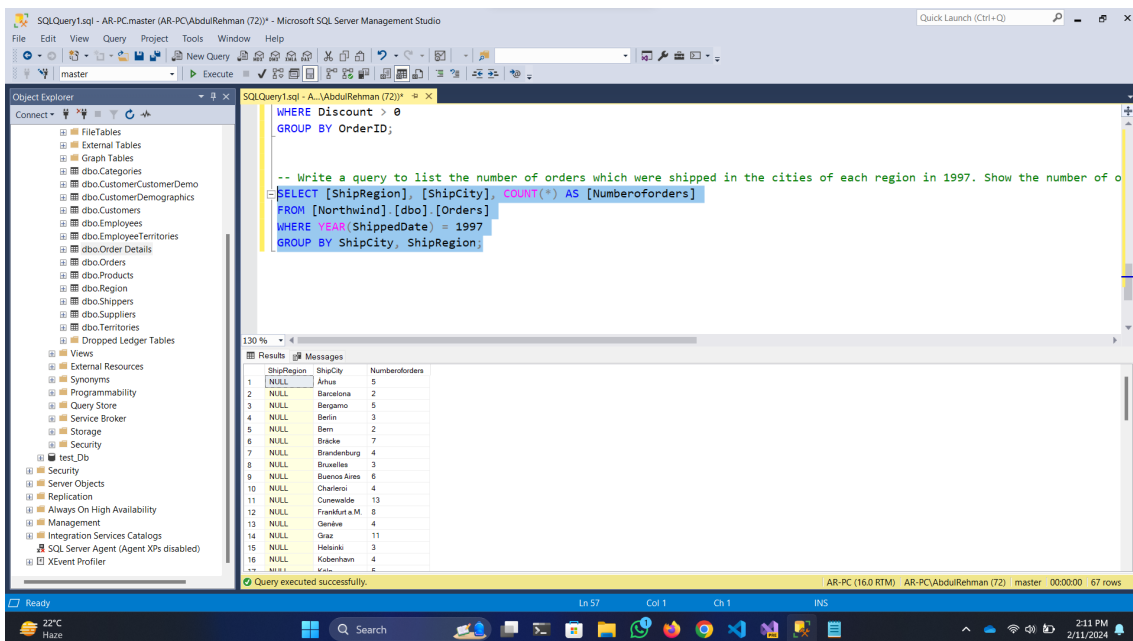
9 Question 9:

9.1 Write a query to list the number of orders which were shipped in the cities of each region in 1997. Show the number of order against each city. Your results should look like this: (ShipRegion, ShipCity, Numberoforders)

SQL Query:

```
SELECT [ShipRegion], [ShipCity], COUNT(*) AS [Numberoforders]
FROM [Northwind].[dbo].[Orders]
WHERE YEAR(ShippedDate) = 1997
GROUP BY ShipCity, ShipRegion;
```

9.2 Result



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor displays the following SQL query:

```
WHERE Discount > 0
GROUP BY OrderID;

-- Write a query to list the number of orders which were shipped in the cities of each region in 1997. Show the number of o
SELECT [ShipRegion], [ShipCity], COUNT(*) AS [Numberoforders]
FROM [Northwind].[dbo].[Orders]
WHERE YEAR(ShippedDate) = 1997
GROUP BY ShipCity, ShipRegion;
```

The Results pane shows the output of the query, displaying a table with three columns: ShipRegion, ShipCity, and Numberoforders. The data is as follows:

| ShipRegion | ShipCity | Numberoforders |
|------------|----------------|----------------|
| NULL | Anhui | 5 |
| NULL | Banarasi | 2 |
| NULL | Bergamo | 5 |
| NULL | Berlin | 3 |
| NULL | Bern | 2 |
| NULL | Brake | 7 |
| NULL | Brandenburg | 4 |
| NULL | Bruxelles | 3 |
| NULL | Buenos Aires | 6 |
| NULL | Chattanooga | 4 |
| NULL | Cunewalde | 13 |
| NULL | Frankfurt a.M. | 8 |
| NULL | Geneve | 4 |
| NULL | Grax | 11 |
| NULL | Helsinki | 3 |
| NULL | Kobenhavn | 4 |

The status bar at the bottom indicates that the query was executed successfully, returning 67 rows.