

DataBase System LAB5



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 Lab-5 Tasks:	3
1.1 Return customers and their orders, including customers who placed no orders (CustomerID, OrderID, OrderDate)	3
1.2 Report only those customer IDs who never placed any order. (CustomerID, OrderID, OrderDate)	4
1.3 Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)	5
1.4 Report the total orders of each customer. (customerID, totalorders)	6
1.5 Write a query to generate a five copies of each employee. (EmployeeID, FirstName, LastName)	7
1.6 Write a query that returns a row for each employee and day in the range 04-07-1996 through 04-08-1997. (EmployeeID, Date)	8
1.7 Return US customers, and for each customer return the total number of orders and total quantities. (CustomerID, Totalorders, totalquantity)	9
1.8 Write a query that returns all customers in the output, but matches them with their respective orders only if they were placed on July 04,1997. (CustomerID, CompanyName, OrderID, Orderdate)	10
1.9 Are there any employees who are older than their managers?	11
1.10 List that names of those employees and their ages. (EmployeeName, Age, Manager Age)	12
1.11 List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)	13
1.12 List the addresses, cities, countries of all orders which were serviced by Anne and were shipped late. (Address, City, Country)	14
1.13 List all countries to which beverages have been shipped. (Country)	15

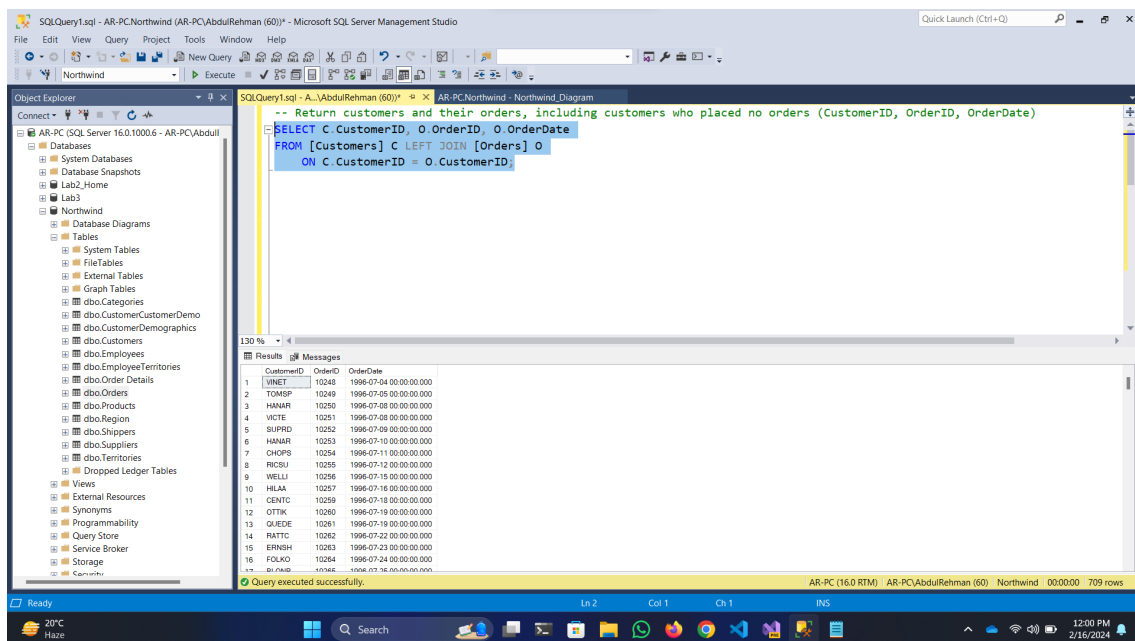
1 Lab-5 Tasks:

1.1 Return customers and their orders, including customers who placed no orders (CustomerID, OrderID, OrderDate)

SQL Query:

```
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
ON C.CustomerID = O.CustomerID;
```

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 'SQLQuery1.sql' file with the following query:

```
-- Return customers and their orders, including customers who placed no orders (CustomerID, OrderID, OrderDate)
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
ON C.CustomerID = O.CustomerID;
```

The 'Results' pane at the bottom shows the output of the query, which is a table with three columns: CustomerID, OrderID, and OrderDate. The table contains 17 rows of data, including customers with no orders (OrderID is null).

CustomerID	OrderID	OrderDate
VINET	10248	1996-07-04 00:00:00.000
TOMSP	10249	1996-07-05 00:00:00.000
HANAR	10250	1996-07-08 00:00:00.000
VICTE	10251	1996-07-08 00:00:00.000
SUPRD	10252	1996-07-09 00:00:00.000
HANAR	10253	1996-07-10 00:00:00.000
CHOPS	10254	1996-07-11 00:00:00.000
RICSU	10255	1996-07-12 00:00:00.000
WELLI	10256	1996-07-15 00:00:00.000
HILAA	10257	1996-07-18 00:00:00.000
CENTC	10259	1996-07-18 00:00:00.000
OTTIK	10260	1996-07-19 00:00:00.000
QUEDE	10261	1996-07-19 00:00:00.000
RIATC	10262	1996-07-22 00:00:00.000
ERNBH	10263	1996-07-23 00:00:00.000
FKLKD	10264	1996-07-24 00:00:00.000
BLVDW	10265	1996-07-26 00:00:00.000

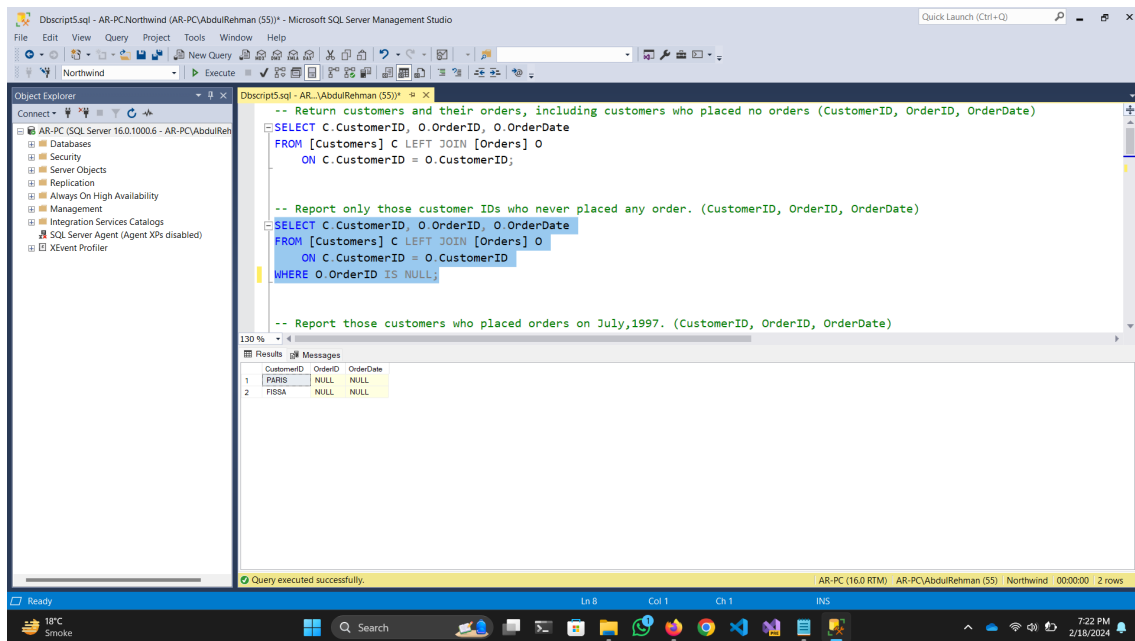
The status bar at the bottom indicates that the query was executed successfully, returning 709 rows.

1.2 Report only those customer IDs who never placed any order. (CustomerID, OrderID, OrderDate)

SQL Query:

```
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
WHERE O.OrderID IS NULL;
```

Result:



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

```
-- Return customers and their orders, including customers who placed no orders (CustomerID, OrderID, OrderDate)
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
    ON C.CustomerID = O.CustomerID;

-- Report only those customer IDs who never placed any order. (CustomerID, OrderID, OrderDate)
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
WHERE O.OrderID IS NULL;

-- Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)
```

The Results pane shows the output of the query, displaying two rows of data:

CustomerID	OrderID	OrderDate
1	PAIRS	NULL
2	FISSA	NULL

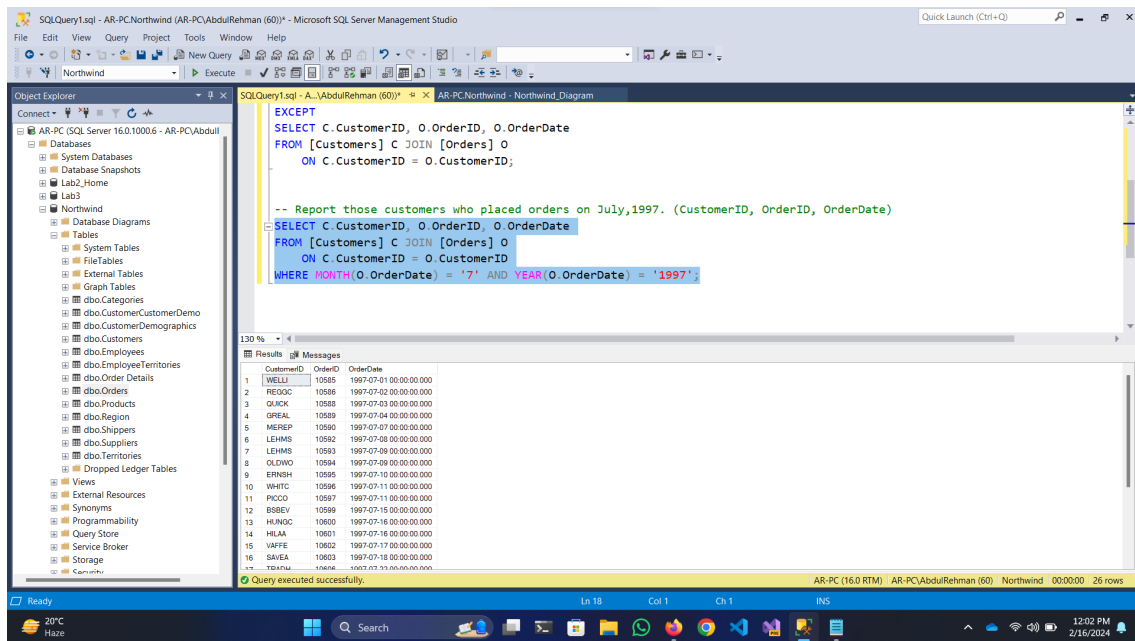
The status bar at the bottom indicates "Query executed successfully."

1.3 Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)

SQL Query:

```
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C JOIN [Orders] O
ON C.CustomerID = O.CustomerID
WHERE MONTH(O.OrderDate) = '7' AND YEAR(O.OrderDate) = '1997';
```

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:

```
EXCEPT
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C JOIN [Orders] O
ON C.CustomerID = O.CustomerID;

-- Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)
SELECT C.CustomerID, O.OrderID, O.OrderDate
FROM [Customers] C JOIN [Orders] O
ON C.CustomerID = O.CustomerID
WHERE MONTH(O.OrderDate) = '7' AND YEAR(O.OrderDate) = '1997';
```

The 'Results' pane at the bottom shows the output of the query, displaying 16 rows of data with columns CustomerID, OrderID, and OrderDate. The status bar at the bottom indicates 'Query executed successfully.' and '26 rows'.

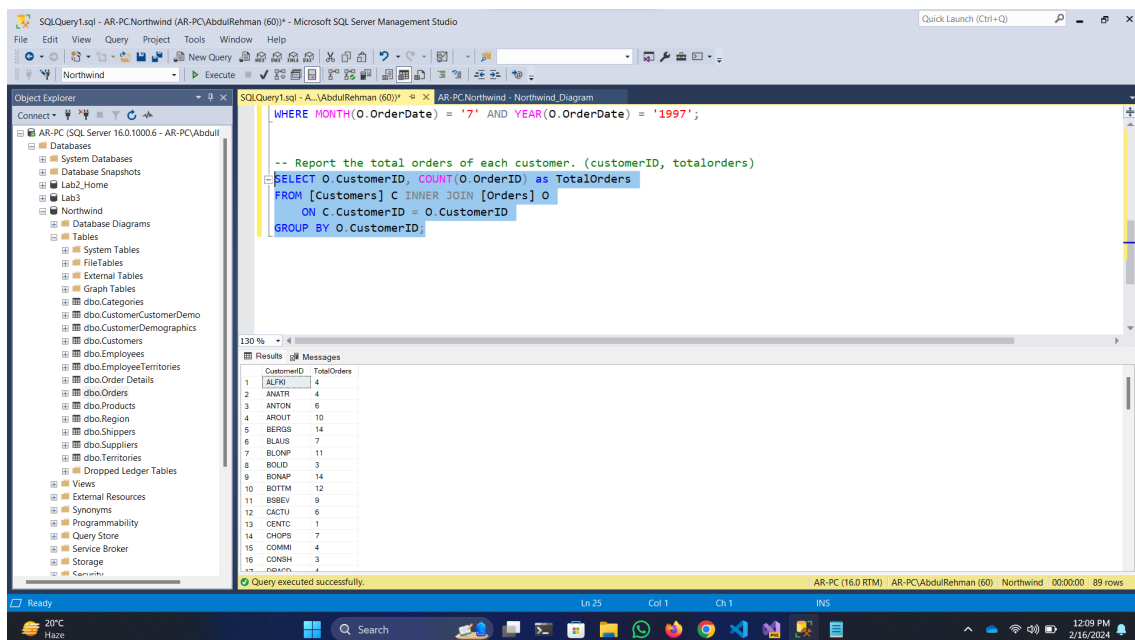
CustomerID	OrderID	OrderDate
WELLI	10505	1997-07-01 00:00:00.000
FRIDC	10506	1997-07-02 00:00:00.000
QUICK	10508	1997-07-03 00:00:00.000
GREAL	10509	1997-07-04 00:00:00.000
MEPSE	10590	1997-07-07 00:00:00.000
LEHMS	10592	1997-07-08 00:00:00.000
LEHMS	10593	1997-07-09 00:00:00.000
OLDWO	10594	1997-07-09 00:00:00.000
ERNST	10595	1997-07-10 00:00:00.000
WHITC	10596	1997-07-11 00:00:00.000
PICCO	10597	1997-07-11 00:00:00.000
BBSBY	10598	1997-07-15 00:00:00.000
HUNGC	10600	1997-07-16 00:00:00.000
HILAA	10601	1997-07-16 00:00:00.000
VAFFE	10602	1997-07-17 00:00:00.000
SAVEA	10603	1997-07-18 00:00:00.000
TSUTSU	10604	1997-07-19 00:00:00.000

1.4 Report the total orders of each customer. (customerID, totalorders)

SQL Query:

```
SELECT O.CustomerID, COUNT(O.OrderID) as TotalOrders
FROM [Customers] C INNER JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
GROUP BY O.CustomerID;
```

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:

```
WHERE MONTH(O.OrderDate) = '7' AND YEAR(O.OrderDate) = '1997';

-- Report the total orders of each customer. (customerID, totalorders)
SELECT O.CustomerID, COUNT(O.OrderID) as TotalOrders
FROM [Customers] C INNER JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
GROUP BY O.CustomerID;
```

The 'Results' pane at the bottom shows the output of the query, which is a table with two columns: 'CustomerID' and 'TotalOrders'. The data is as follows:

CustomerID	TotalOrders
ALFKI	4
ANATR	4
ANTON	6
AROUT	10
BERRG	14
BLAUS	7
BLOMP	11
BOLID	3
BONAP	14
BOTTM	12
BBSBEV	9
CACTU	6
CEHTC	1
CHOPB	7
COMM4	4
CONSH	3
PERCH	4

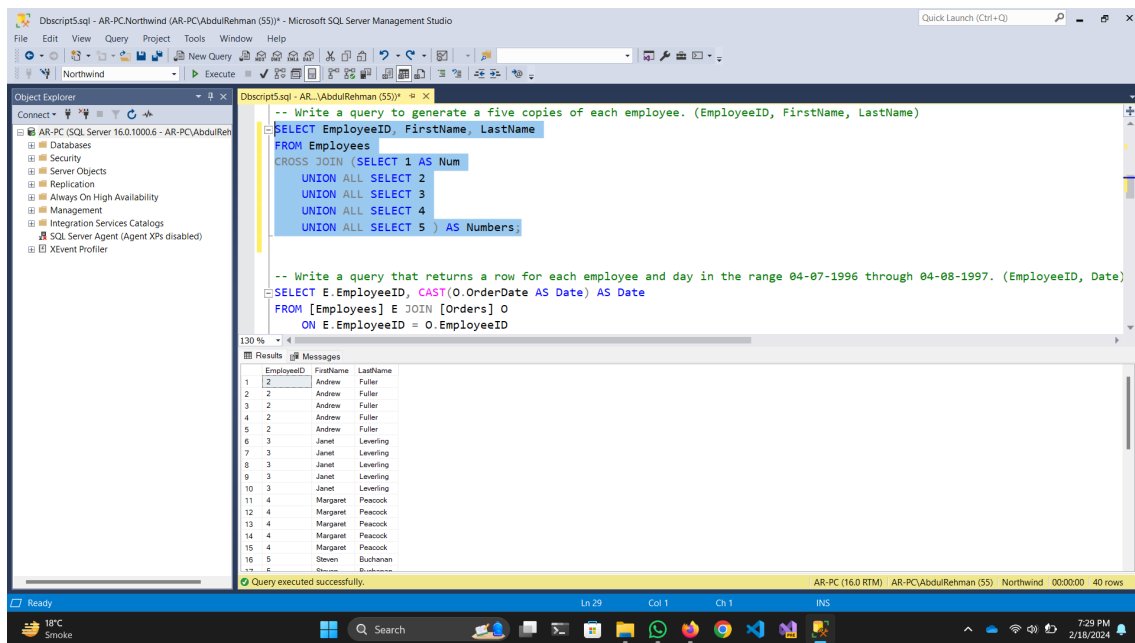
The status bar at the bottom indicates that the query was executed successfully, returning 89 rows.

1.5 Write a query to generate a five copies of each employee. (EmployeeID, FirstName, LastName)

SQL Query:

```
SELECT EmployeeID, FirstName, LastName
FROM Employees
CROSS JOIN (SELECT 1 AS Num
            UNION ALL SELECT 2
            UNION ALL SELECT 3
            UNION ALL SELECT 4
            UNION ALL SELECT 5) AS Numbers;
```

Result:



The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The query editor shows the following SQL query:

```
-- Write a query to generate a five copies of each employee. (EmployeeID, FirstName, LastName)
SELECT EmployeeID, FirstName, LastName
FROM Employees
CROSS JOIN (SELECT 1 AS Num
            UNION ALL SELECT 2
            UNION ALL SELECT 3
            UNION ALL SELECT 4
            UNION ALL SELECT 5) AS Numbers;
```

The query results are displayed in a table with the following data:

EmployeeID	FirstName	LastName
1	Andrew	Fuller
2	Andrew	Fuller
3	Andrew	Fuller
4	Andrew	Fuller
5	Andrew	Fuller
6	Janet	Leverling
7	Janet	Leverling
8	Janet	Leverling
9	Janet	Leverling
10	Janet	Leverling
11	Margaret	Peacock
12	Margaret	Peacock
13	Margaret	Peacock
14	Margaret	Peacock
15	Margaret	Peacock
16	Steven	Buchanan
17	Steven	Buchanan

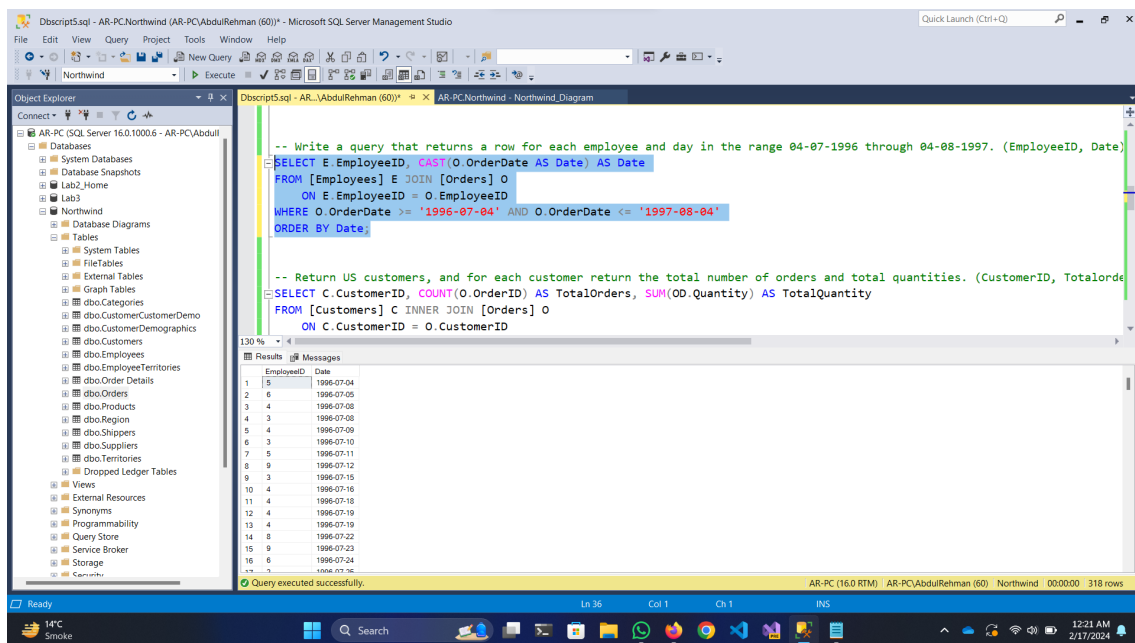
The status bar at the bottom indicates that the query was executed successfully, returning 40 rows.

1.6 Write a query that returns a row for each employee and day in the range 04-07-1996 through 04-08-1997. (EmployeeID, Date)

SQL Query:

```
SELECT E.EmployeeID, CAST(O.OrderDate AS Date) AS Date
FROM [Employees] E JOIN [Orders] O
ON E.EmployeeID = O.EmployeeID
WHERE O.OrderDate >= '1996-07-04' AND O.OrderDate <= '1997-08-04'
ORDER BY Date;
```

Result:



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

```
-- Write a query that returns a row for each employee and day in the range 04-07-1996 through 04-08-1997. (EmployeeID, Date)
SELECT E.EmployeeID, CAST(O.OrderDate AS Date) AS Date
FROM [Employees] E JOIN [Orders] O
ON E.EmployeeID = O.EmployeeID
WHERE O.OrderDate >= '1996-07-04' AND O.OrderDate <= '1997-08-04'
ORDER BY Date;
```

The query results are displayed in the Results pane, showing a table with two columns: EmployeeID and Date. The results are as follows:

EmployeeID	Date
1	1996-07-04
2	1996-07-05
3	1996-07-06
4	1996-07-07
5	1996-07-08
6	1996-07-09
7	1996-07-10
8	1996-07-11
9	1996-07-12
10	1996-07-13
11	1996-07-14
12	1996-07-15
13	1996-07-16
14	1996-07-17
15	1996-07-18
16	1996-07-19
17	1996-07-20
18	1996-07-21
19	1996-07-22
20	1996-07-23
21	1996-07-24
22	1996-07-25
23	1996-07-26
24	1996-07-27
25	1996-07-28
26	1996-07-29
27	1996-07-30
28	1996-07-31
29	1996-08-01
30	1996-08-02
31	1996-08-03
32	1996-08-04
33	1996-08-05
34	1996-08-06
35	1996-08-07
36	1996-08-08
37	1996-08-09
38	1996-08-10
39	1996-08-11
40	1996-08-12
41	1996-08-13
42	1996-08-14
43	1996-08-15
44	1996-08-16
45	1996-08-17
46	1996-08-18
47	1996-08-19
48	1996-08-20
49	1996-08-21
50	1996-08-22
51	1996-08-23
52	1996-08-24
53	1996-08-25
54	1996-08-26
55	1996-08-27
56	1996-08-28
57	1996-08-29
58	1996-08-30
59	1996-08-31
60	1996-09-01
61	1996-09-02
62	1996-09-03
63	1996-09-04
64	1996-09-05
65	1996-09-06
66	1996-09-07
67	1996-09-08
68	1996-09-09
69	1996-09-10
70	1996-09-11
71	1996-09-12
72	1996-09-13
73	1996-09-14
74	1996-09-15
75	1996-09-16
76	1996-09-17
77	1996-09-18
78	1996-09-19
79	1996-09-20
80	1996-09-21
81	1996-09-22
82	1996-09-23
83	1996-09-24
84	1996-09-25
85	1996-09-26
86	1996-09-27
87	1996-09-28
88	1996-09-29
89	1996-09-30
90	1996-10-01
91	1996-10-02
92	1996-10-03
93	1996-10-04
94	1996-10-05
95	1996-10-06
96	1996-10-07
97	1996-10-08
98	1996-10-09
99	1996-10-10
100	1996-10-11
101	1996-10-12
102	1996-10-13
103	1996-10-14
104	1996-10-15
105	1996-10-16
106	1996-10-17
107	1996-10-18
108	1996-10-19
109	1996-10-20
110	1996-10-21
111	1996-10-22
112	1996-10-23
113	1996-10-24
114	1996-10-25
115	1996-10-26
116	1996-10-27
117	1996-10-28
118	1996-10-29
119	1996-10-30
120	1996-10-31
121	1996-11-01
122	1996-11-02
123	1996-11-03
124	1996-11-04
125	1996-11-05
126	1996-11-06
127	1996-11-07
128	1996-11-08
129	1996-11-09
130	1996-11-10
131	1996-11-11
132	1996-11-12
133	1996-11-13
134	1996-11-14
135	1996-11-15
136	1996-11-16
137	1996-11-17
138	1996-11-18
139	1996-11-19
140	1996-11-20
141	1996-11-21
142	1996-11-22
143	1996-11-23
144	1996-11-24
145	1996-11-25
146	1996-11-26
147	1996-11-27
148	1996-11-28
149	1996-11-29
150	1996-11-30
151	1996-12-01
152	1996-12-02
153	1996-12-03
154	1996-12-04
155	1996-12-05
156	1996-12-06
157	1996-12-07
158	1996-12-08
159	1996-12-09
160	1996-12-10
161	1996-12-11
162	1996-12-12
163	1996-12-13
164	1996-12-14
165	1996-12-15
166	1996-12-16
167	1996-12-17
168	1996-12-18
169	1996-12-19
170	1996-12-20
171	1996-12-21
172	1996-12-22
173	1996-12-23
174	1996-12-24
175	1996-12-25
176	1996-12-26
177	1996-12-27
178	1996-12-28
179	1996-12-29
180	1996-12-30
181	1996-12-31
182	1997-01-01
183	1997-01-02
184	1997-01-03
185	1997-01-04
186	1997-01-05
187	1997-01-06
188	1997-01-07
189	1997-01-08
190	1997-01-09
191	1997-01-10
192	1997-01-11
193	1997-01-12
194	1997-01-13
195	1997-01-14
196	1997-01-15
197	1997-01-16
198	1997-01-17
199	1997-01-18
200	1997-01-19
201	1997-01-20
202	1997-01-21
203	1997-01-22
204	1997-01-23
205	1997-01-24
206	1997-01-25
207	1997-01-26
208	1997-01-27
209	1997-01-28
210	1997-01-29
211	1997-01-30
212	1997-01-31
213	1997-02-01
214	1997-02-02
215	1997-02-03
216	1997-02-04
217	1997-02-05
218	1997-02-06
219	1997-02-07
220	1997-02-08
221	1997-02-09
222	1997-02-10
223	1997-02-11
224	1997-02-12
225	1997-02-13
226	1997-02-14
227	1997-02-15
228	1997-02-16
229	1997-02-17
230	1997-02-18
231	1997-02-19
232	1997-02-20
233	1997-02-21
234	1997-02-22
235	1997-02-23
236	1997-02-24
237	1997-02-25
238	1997-02-26
239	1997-02-27
240	1997-02-28
241	1997-03-01
242	1997-03-02
243	1997-03-03
244	1997-03-04
245	1997-03-05
246	1997-03-06
247	1997-03-07
248	1997-03-08
249	1997-03-09
250	1997-03-10
251	1997-03-11
252	1997-03-12
253	1997-03-13
254	1997-03-14
255	1997-03-15
256	1997-03-16
257	1997-03-17
258	1997-03-18
259	1997-03-19
260	1997-03-20
261	1997-03-21
262	1997-03-22
263	1997-03-23
264	1997-03-24
265	1997-03-25
266	1997-03-26
267	1997-03-27
268	1997-03-28
269	1997-03-29
270	1997-03-30
271	1997-03-31
272	1997-04-01
273	1997-04-02
274	1997-04-03
275	1997-04-04
276	1997-04-05
277	1997-04-06
278	1997-04-07
279	1997-04-08
280	1997-04-09
281	1997-04-10
282	1997-04-11
283	1997-04-12
284	1997-04-13
285	1997-04-14
286	1997-04-15
287	1997-04-16
288	1997-04-17
289	1997-04-18
290	1997-04-19
291	1997-04-20
292	1997-04-21
293	1997-04-22
294	1997-04-23
295	1997-04-24
296	1997-04-25
297	1997-04-26
298	1997-04-27
299	1997-04-28
300	1997-04-29
301	1997-04-30
302	1997-05-01
303	1997-05-02
304	1997-05-03
305	1997-05-04
306	1997-05-05
307	1997-05-06
308	1997-05-07
309	1997-05-08
310	1997-05-09
311	1997-05-10
312	1997-05-11
313	1997-05-12
314	1997-05-13
315	1997-05-14
316	1997-05-15
317	1997-05-16
318	1997-05-17

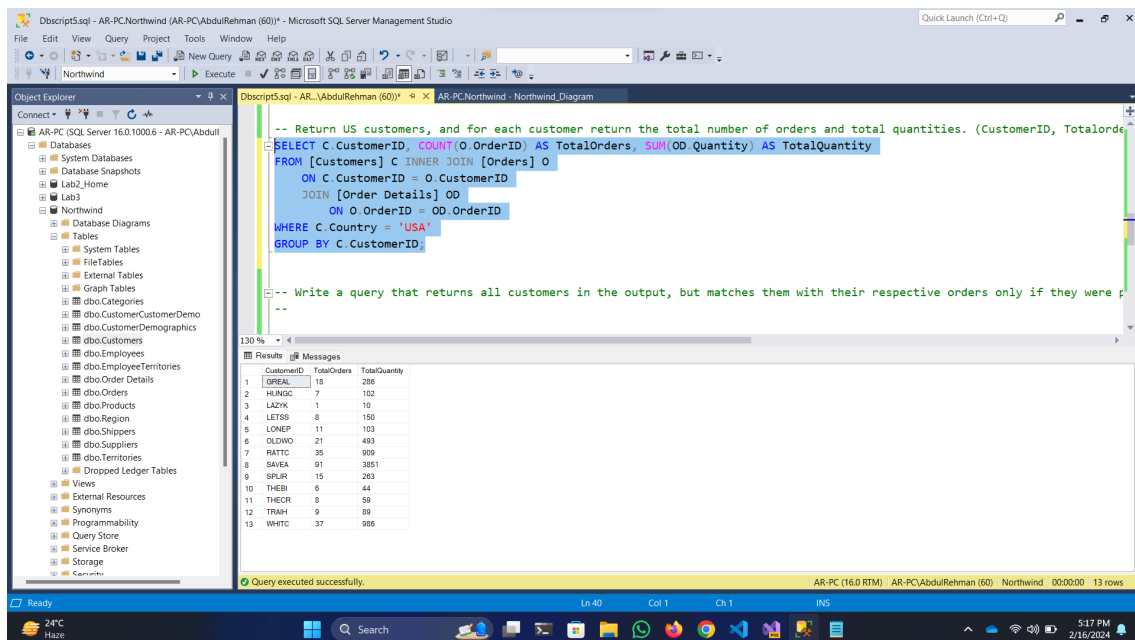
The status bar at the bottom indicates that the query was executed successfully, returning 318 rows.

1.7 Return US customers, and for each customer return the total number of orders and total quantities. (CustomerID, Totalorders, totalquantity)

SQL Query:

```
SELECT C.CustomerID, COUNT(O.OrderID) AS TotalOrders, SUM(OD.Quantity)
FROM [Customers] C INNER JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
    JOIN [Order Details] OD
    ON O.OrderID = OD.OrderID
WHERE C.Country = 'USA'
GROUP BY C.CustomerID;
```

Result:



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

```
-- Return US customers, and for each customer return the total number of orders and total quantities. (CustomerID, TotalOrders, TotalQuantity)
SELECT C.CustomerID, COUNT(O.OrderID) AS TotalOrders, SUM(OD.Quantity) AS TotalQuantity
FROM [Customers] C INNER JOIN [Orders] O
    ON C.CustomerID = O.CustomerID
    JOIN [Order Details] OD
    ON O.OrderID = OD.OrderID
WHERE C.Country = 'USA'
GROUP BY C.CustomerID;
```

The query results are displayed in the Results pane, showing 13 rows of data:

CustomerID	TotalOrders	TotalQuantity
1 GREAL	18	286
2 HUNGC	7	102
3 LAZYK	1	10
4 LETSR	8	150
5 LONEP	11	103
6 OLDWO	21	493
7 RATTC	35	909
8 SAVIA	91	3851
9 SPILA	15	263
10 THEBI	6	44
11 THECR	8	59
12 TRAPH	9	88
13 WHITC	37	986

The status bar at the bottom indicates that the query was executed successfully, returning 13 rows of data.

1.8 Write a query that returns all customers in the output, but matches them with their respective orders only if they were placed on July 04,1997. (CustomerID, Company-Name, OrderID, Orderdate)

SQL Query:

```
SELECT C.CustomerID, C.CompanyName, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
ON C.CustomerID = O.CustomerID AND O.OrderDate = '1997-07-04';
```

Result:

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

```
-- Write a query that returns all customers in the output, but matches them with their respective orders only if they were placed on July 04,1997. (CustomerID, Company-Name, OrderID, Orderdate)
SELECT C.CustomerID, C.CompanyName, O.OrderID, O.OrderDate
FROM [Customers] C LEFT JOIN [Orders] O
ON C.CustomerID = O.CustomerID AND O.OrderDate = '1997-07-04';

-- Are there any employees who are older than their managers?

-- List that names of those employees and their ages. (EmployeeName, Age, Manager Age)
```

The bottom pane shows the 'Results' tab with the following data:

CustomerID	CompanyName	OrderID	OrderDate
1	ALFKI	Alfreds Futterkiste	NULL
2	ANATR	Ana Trujillo Emparedados y helados	NULL
3	ANTON	Antonio Moreno Taqueria	NULL
4	AROUT	Around the Horn	NULL
5	BERGS	Berglunds snabbköp	NULL
6	BLAUS	Blauer See Delikatessen	NULL
7	BONAP	Blondel père et fils	NULL
8	BOLID	Bolidi Commodities	NULL
9	BONAP	Bon app'	NULL
10	BOTTM	Bottom-Dollar Markets	NULL
11	BIBBEV	B's Beverages	NULL
12	CACTU	Cactus Comidas para llevar	NULL
13	CENTC	Centro comercial Modestissimo	NULL
14	CHOPB	Chop-suey Chinese	NULL
15	CORMA	Comércio Mineiro	NULL
16	CONSH	Consolidated Holdings	NULL
17	DEMAT	Demazet's Produce	NULL

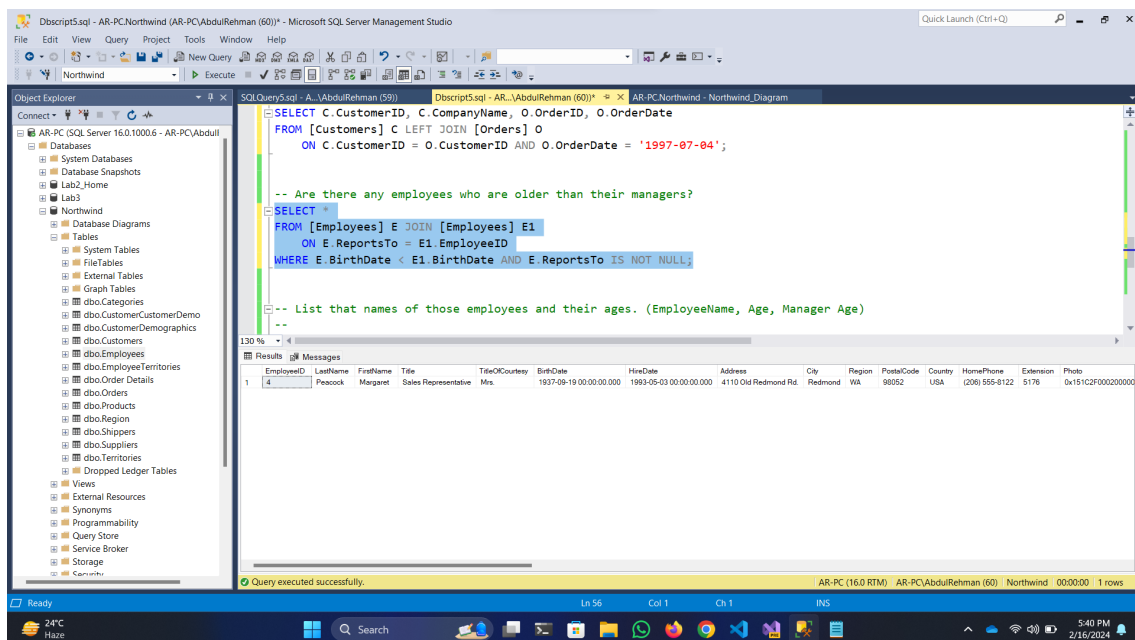
The status bar at the bottom indicates 'Query executed successfully.' and '91 rows'.

1.9 Are there any employees who are older than their managers?

SQL Query:

```
SELECT *
FROM [Employees] E JOIN [Employees] E1
    ON E.ReportsTo = E1.EmployeeID
WHERE E.BirthDate < E1.BirthDate AND E.ReportsTo IS NOT NULL;
```

Result:



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

```
-- Are there any employees who are older than their managers?
SELECT *
FROM [Employees] E JOIN [Employees] E1
    ON E.ReportsTo = E1.EmployeeID
WHERE E.BirthDate < E1.BirthDate AND E.ReportsTo IS NOT NULL;

-- List that names of those employees and their ages. (EmployeeName, Age, Manager Age)
--
```

The query was executed successfully, and the results are shown in the Results pane. The results table has the following columns: EmployeeID, LastName, FirstName, Title, TitleOfCourtesy, BirthDate, HireDate, Address, City, Region, PostalCode, Country, HomePhone, Extension, and Photo. The results pane shows one row of data:

EmployeeID	LastName	FirstName	Title	TitleOfCourtesy	BirthDate	HireDate	Address	City	Region	PostalCode	Country	HomePhone	Extension	Photo
4	Peacock	Margaret	Sales Representative	Mrs.	1937-09-19 00:00:00.0000	1993-05-03 00:00:00.0000	4110 Old Redmond Rd.	Redmond	WA	98052	USA	(206) 555-8122	5176	dx151c2f000200000

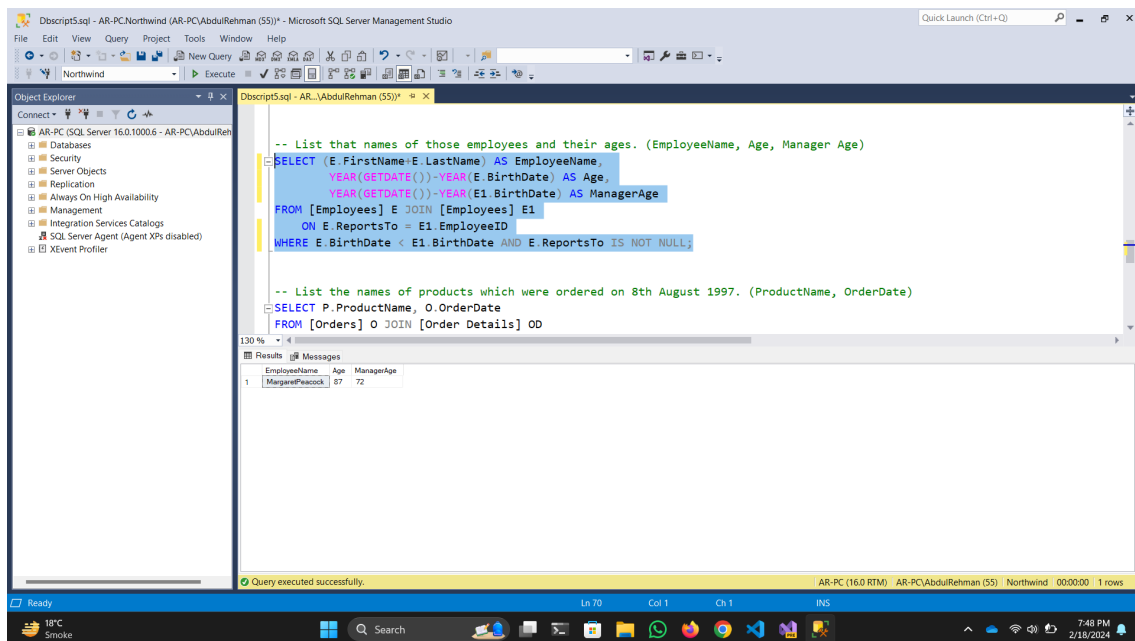
The status bar at the bottom indicates that the query was executed successfully, returning 1 row.

1.10 List that names of those employees and their ages. (EmployeeName, Age, Manager Age)

SQL Query:

```
SELECT (E.FirstName+E.LastName) AS EmployeeName,  
       YEAR(GETDATE())-YEAR(E.BirthDate) AS Age,  
       YEAR(GETDATE())-YEAR(E1.BirthDate) AS ManagerAge  
FROM [Employees] E JOIN [Employees] E1  
     ON E.ReportsTo = E1.EmployeeID  
WHERE E.BirthDate < E1.BirthDate AND E.ReportsTo IS NOT NULL;
```

Result:



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

```
-- List that names of those employees and their ages. (EmployeeName, Age, Manager Age)  
SELECT (E.FirstName+E.LastName) AS EmployeeName,  
       YEAR(GETDATE())-YEAR(E.BirthDate) AS Age,  
       YEAR(GETDATE())-YEAR(E1.BirthDate) AS ManagerAge  
FROM [Employees] E JOIN [Employees] E1  
     ON E.ReportsTo = E1.EmployeeID  
WHERE E.BirthDate < E1.BirthDate AND E.ReportsTo IS NOT NULL;  
  
-- List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)  
SELECT P.ProductName, O.OrderDate  
FROM [Orders] O JOIN [Order Details] OD
```

The query results are displayed in the Results pane, showing a single row:

EmployeeName	Age	ManagerAge
MargaretPaiscoe	67	72

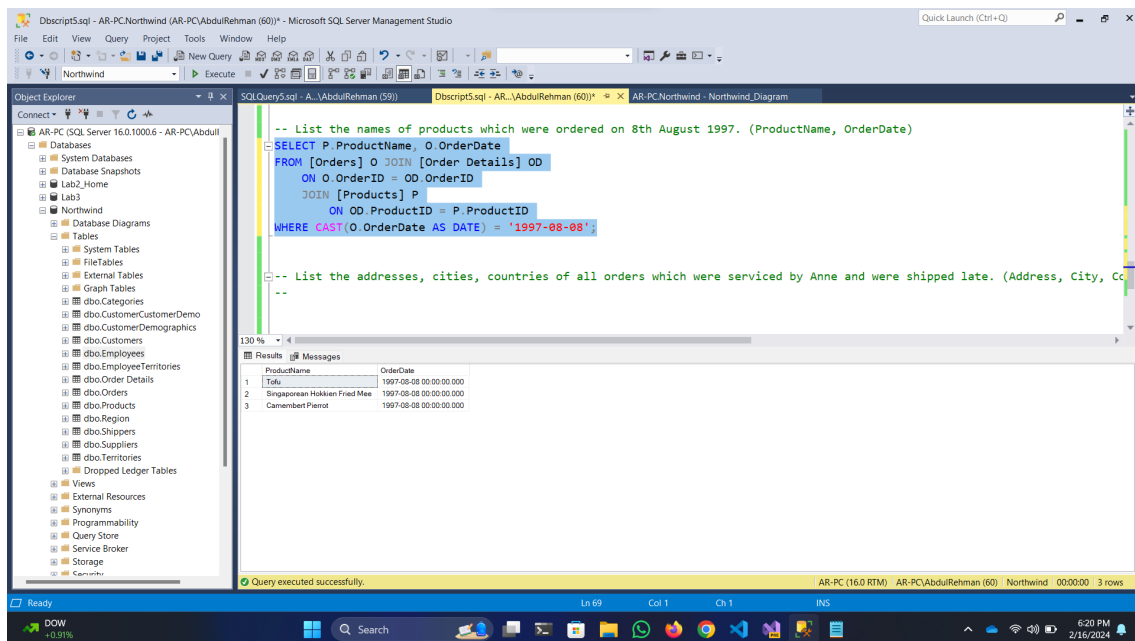
The status bar at the bottom indicates "Query executed successfully." and "1 rows".

1.11 List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)

SQL Query:

```
SELECT P.ProductName, O.OrderDate
FROM [Orders] O JOIN [Order Details] OD
    ON O.OrderID = OD.OrderID
JOIN [Products] P
    ON OD.ProductID = P.ProductID
WHERE CAST(O.OrderDate AS DATE) = '1997-08-08';
```

Result:



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

```
-- List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)
SELECT P.ProductName, O.OrderDate
FROM [Orders] O JOIN [Order Details] OD
    ON O.OrderID = OD.OrderID
JOIN [Products] P
    ON OD.ProductID = P.ProductID
WHERE CAST(O.OrderDate AS DATE) = '1997-08-08';
```

Below the query window, the 'Results' pane displays the output of the query:

	ProductName	OrderDate
1	Tofu	1997-08-08 00:00:00.000
2	Singaporean Hokkien Fried Mee	1997-08-08 00:00:00.000
3	Camembert Pierrot	1997-08-08 00:00:00.000

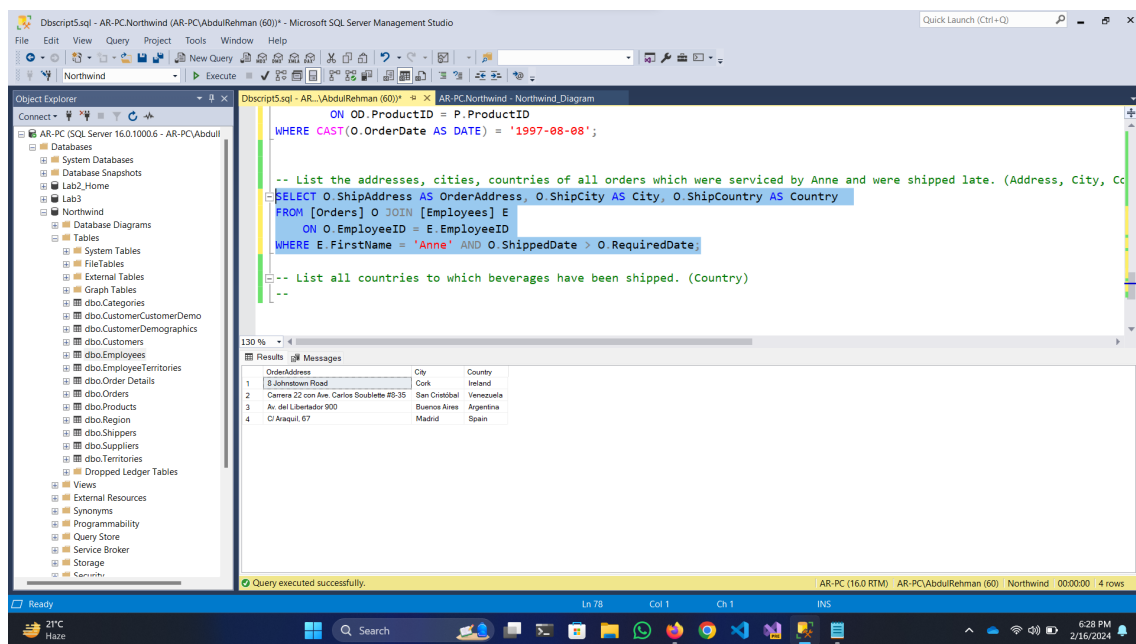
The status bar at the bottom indicates 'Query executed successfully.' and 'AR-PC (16.0 RTM) AR-PC\AbdulRehman (60) Northwind 00:00:00 3 rows'.

1.12 List the addresses, cities, countries of all orders which were serviced by Anne and were shipped late. (Address, City, Country)

SQL Query:

```
SELECT O.ShipAddress AS OrderAddress, O.ShipCity AS City, O.ShipCountry AS Country
FROM [Orders] O JOIN [Employees] E
ON O.EmployeeID = E.EmployeeID
WHERE E.FirstName = 'Anne' AND O.ShippedDate > O.RequiredDate;
```

Result:



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

```
-- List the addresses, cities, countries of all orders which were serviced by Anne and were shipped late. (Address, City, Country)
SELECT O.ShipAddress AS OrderAddress, O.ShipCity AS City, O.ShipCountry AS Country
FROM [Orders] O JOIN [Employees] E
ON O.EmployeeID = E.EmployeeID
WHERE E.FirstName = 'Anne' AND O.ShippedDate > O.RequiredDate;
```

The 'Results' pane shows the output of the query, which is a table with three columns: OrderAddress, City, and Country. The results are as follows:

OrderAddress	City	Country
8 Johnstown Road	Cork	Ireland
Carretera 22 con Ave. Carlos Soublette #8-35	San Cristóbal	Venezuela
Av. del Libertador 900	Buenos Aires	Argentina
C/ Araquil, 67	Madrid	Spain

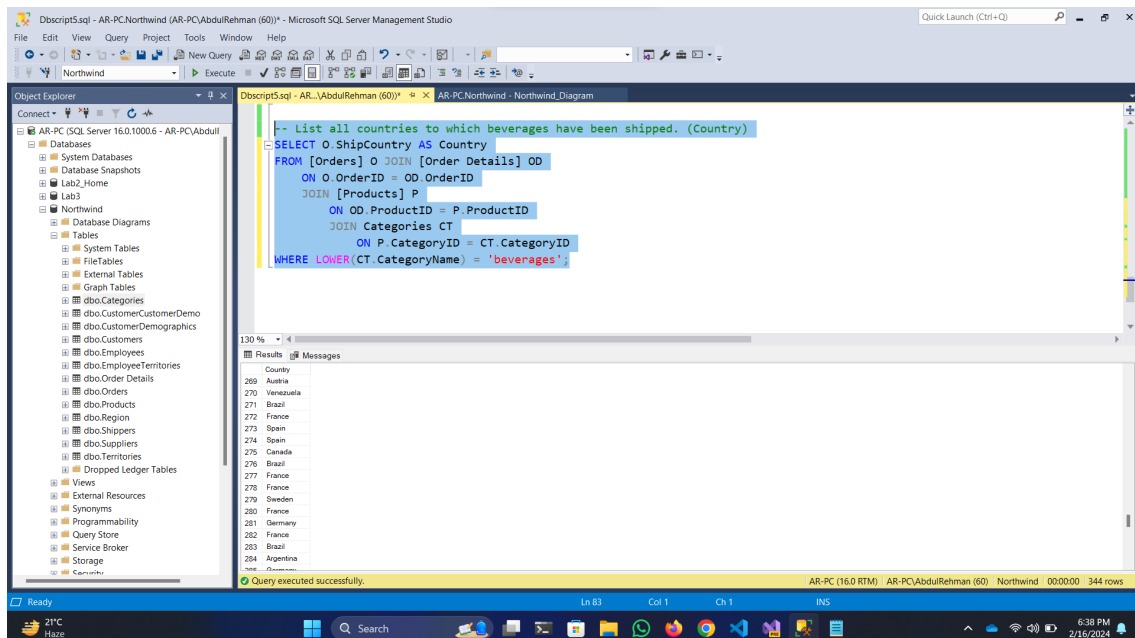
The status bar at the bottom indicates that the query was executed successfully and returned 4 rows.

1.13 List all countries to which beverages have been shipped. (Country)

SQL Query:

```
SELECT O.ShipCountry AS Country
FROM [Orders] O JOIN [Order Details] OD
ON O.OrderID = OD.OrderID
JOIN [Products] P
ON OD.ProductID = P.ProductID
JOIN Categories CT
ON P.CategoryID = CT.CategoryID
WHERE LOWER(CT.CategoryName) = 'beverages';
```

Result:



The screenshot displays the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including tables like Orders, Order Details, Products, and Categories. The Query Editor in the center contains the SQL query: `SELECT O.ShipCountry AS Country FROM [Orders] O JOIN [Order Details] OD ON O.OrderID = OD.OrderID JOIN [Products] P ON OD.ProductID = P.ProductID JOIN Categories CT ON P.CategoryID = CT.CategoryID WHERE LOWER(CT.CategoryName) = 'beverages';`. The Results pane at the bottom shows the output of the query, which is a list of countries: Austria, Venezuela, Brazil, France, Spain, Spain, Canada, Brazil, France, France, Sweden, Germany, France, Brazil, Argentina, and Mexico. The status bar at the bottom indicates that the query was executed successfully and returned 344 rows.

Country
269 Austria
270 Venezuela
271 Brazil
272 France
273 Spain
274 Spain
275 Canada
276 Brazil
277 France
278 France
279 Sweden
280 Germany
281 France
282 Brazil
283 Argentina
284 Mexico