DataBase System
Lab-6

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

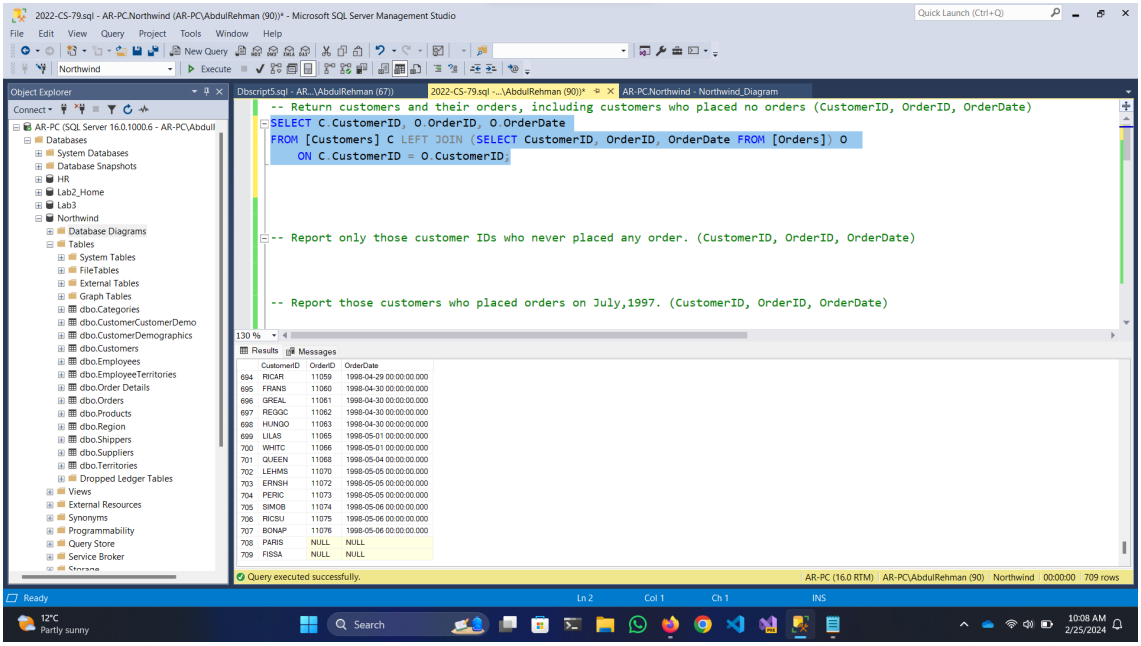
1 Lab-6 Tasks:

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 (SELECT CustomerID, OrderID, OrderDate FROM [Orders]) O
 ON C.CustomerID = O.CustomerID;
```

Result:

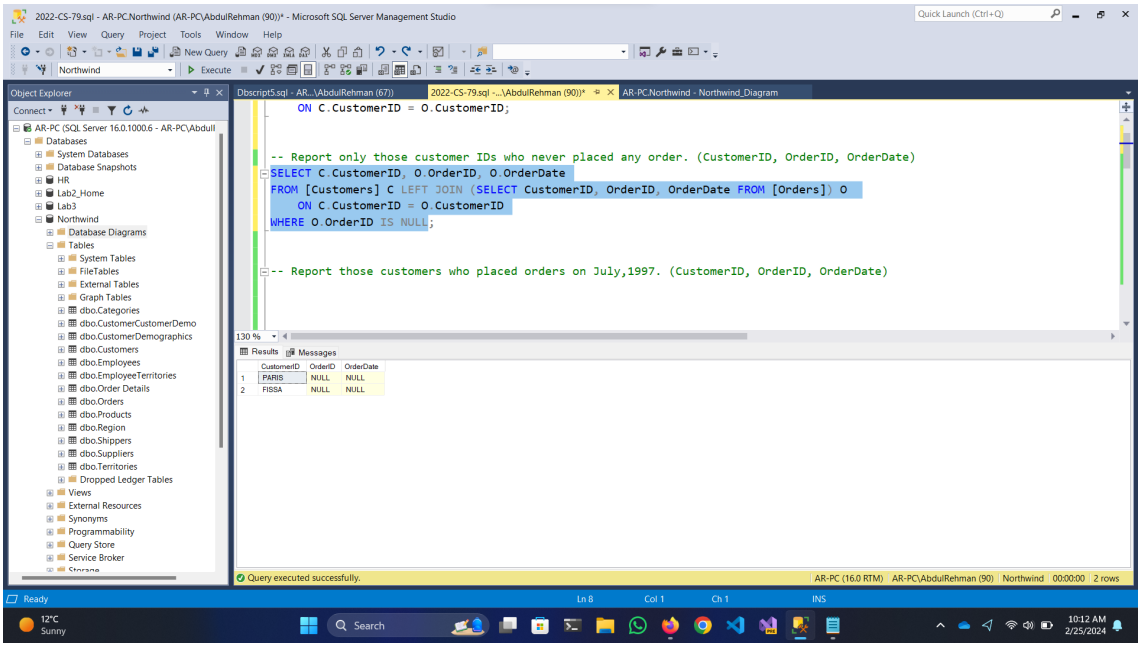


1.2 Report only those customer IDs who never placed any order. (CustomerID, OrderID, OrderDate)

SQL Query:

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

Result:

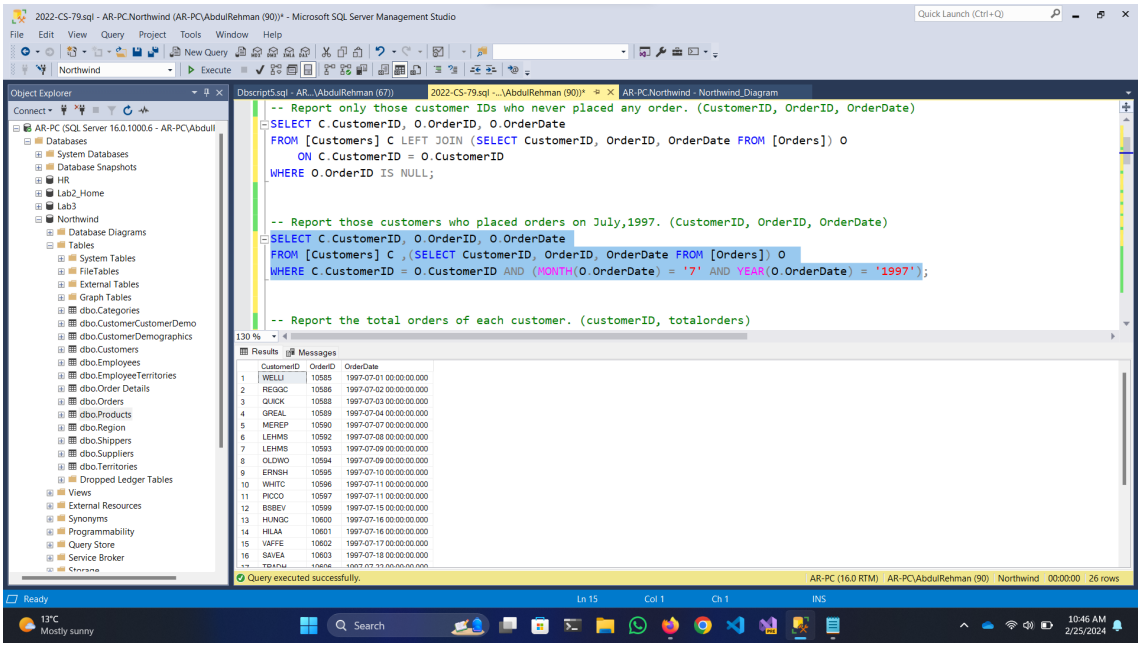


1.3 Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)

SQL Query:

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

Result:

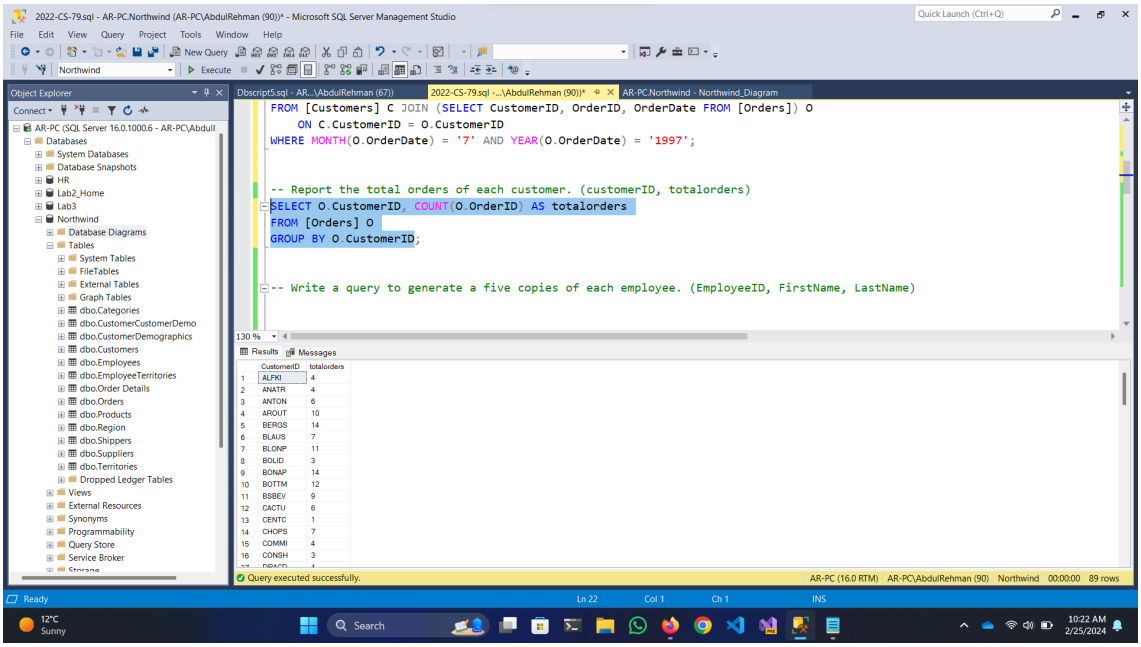


1.4 Report the total orders of each customer. (customerID, totalorders)

SQL Query:

```
SELECT O.CustomerID, COUNT(O.OrderID) AS totalorders
FROM [Orders] O
GROUP BY O.CustomerID;
```

Result:

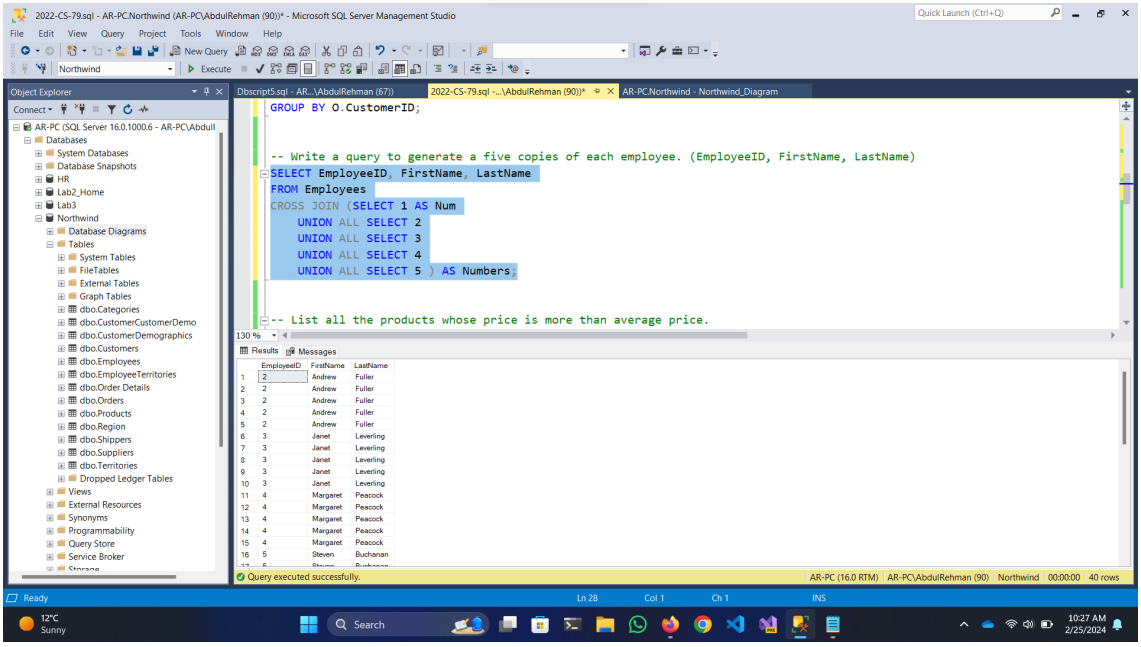


1.5 Write a query to generate a five copies of each employee. (EmployeeID, FirstName, LastName)

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:

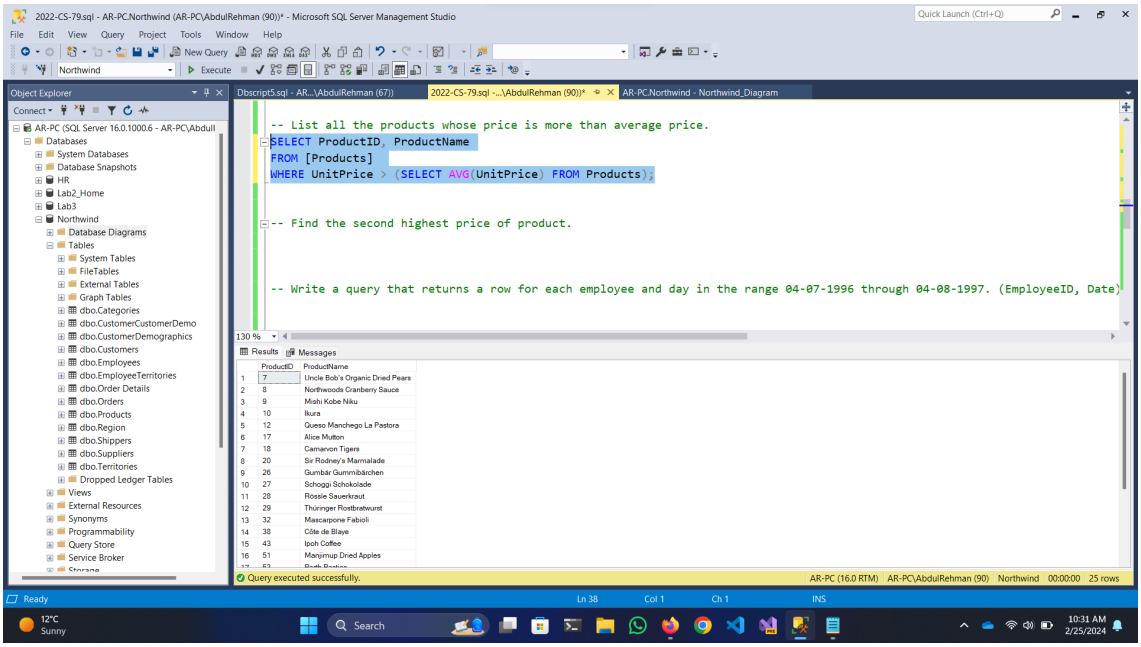


1.6 List all the products whose price is more than average price.

SQL Query:

```
SELECT ProductID, ProductName
FROM [Products]
WHERE UnitPrice > (SELECT AVG(UnitPrice) FROM Products);
```

Result:

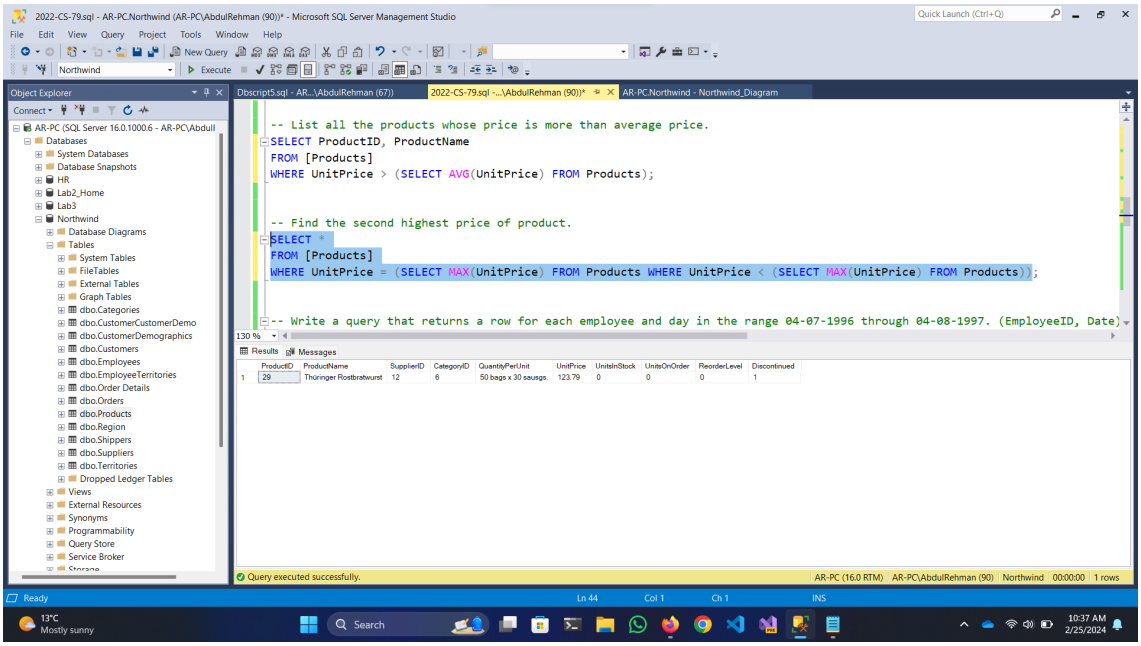


1.7 Find the second highest price of product.

SQL Query:

```
SELECT *
FROM [Products]
WHERE UnitPrice = (
    SELECT MAX(UnitPrice) FROM Products WHERE UnitPrice < (
        SELECT MAX(UnitPrice) FROM Products));
```

Result:

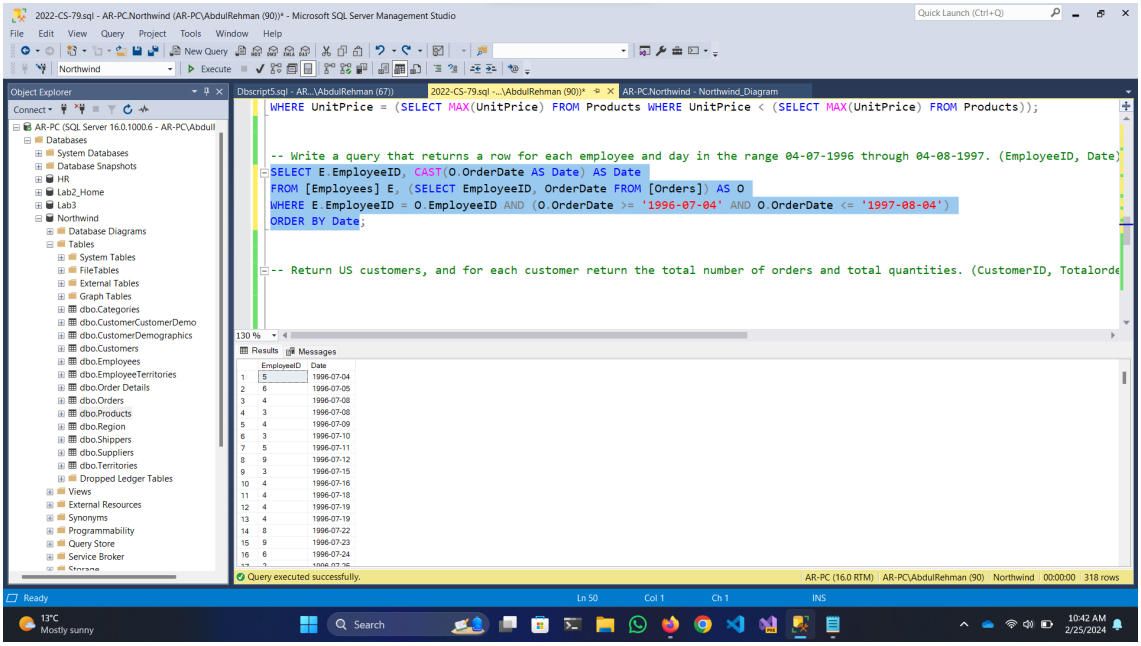


1.8 Write a query that returns a row for each employee and day in the range 04-07-1996 through 04-08-1997. (EmployeeID, Date)

SQL Query:

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

Result:

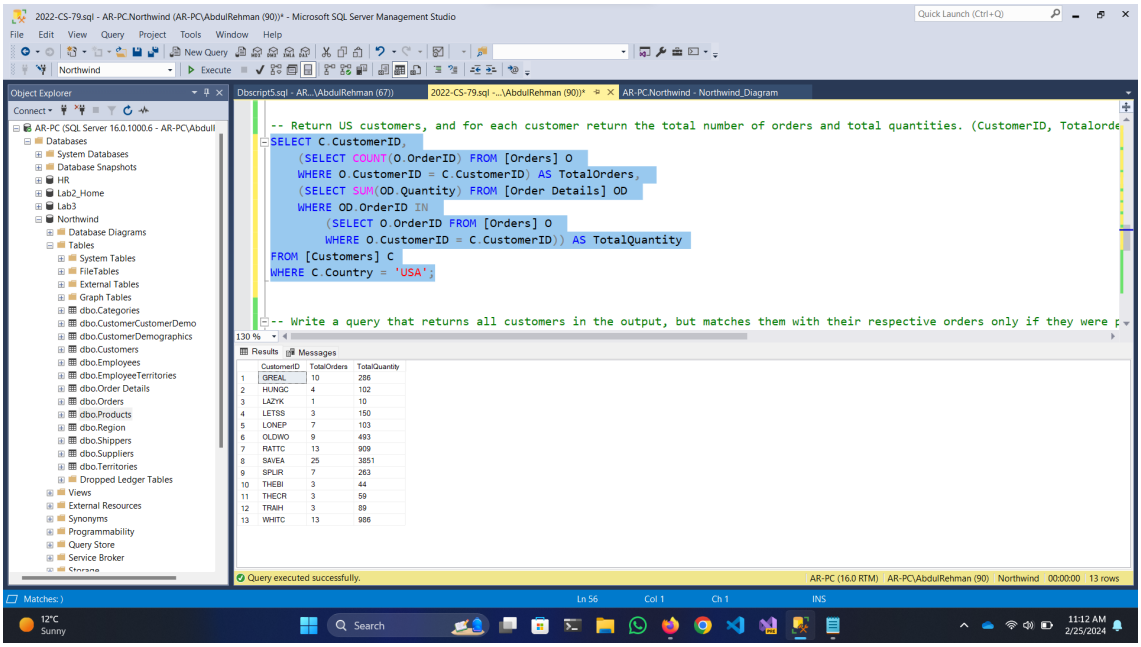


1.9 Return US customers, and for each customer return the total number of orders and total quantities. (CustomerID, Totalorders, totalquantity)

SQL Query:

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

Result:

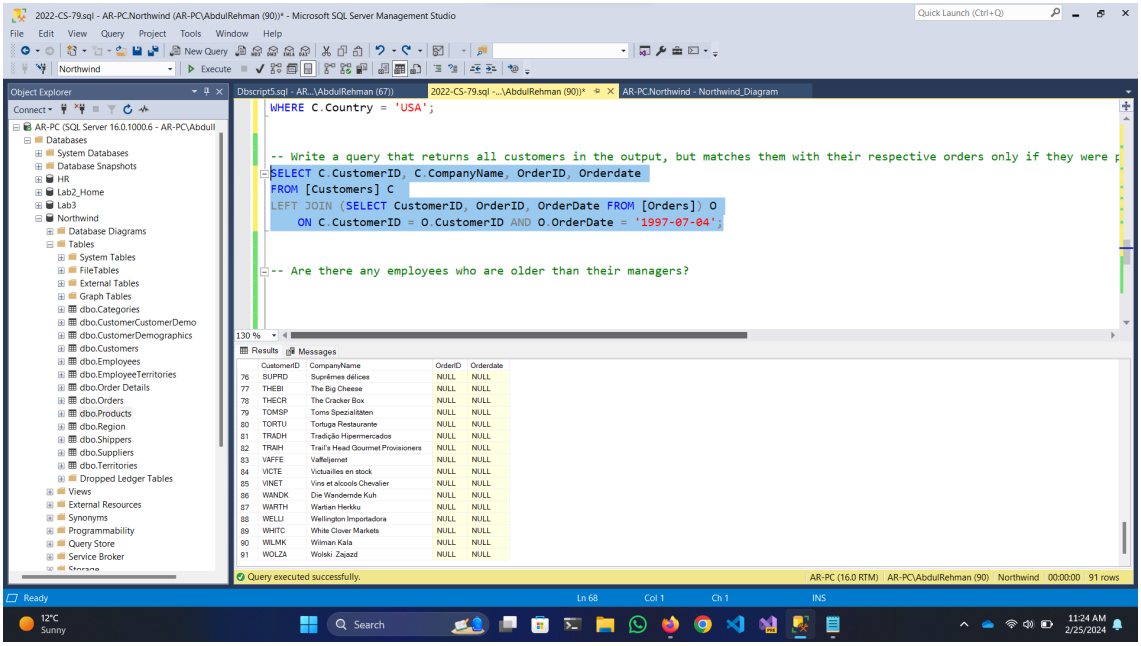


1.10 Write a query that returns all customers in the output, but matches them with their respective orders only if they were placed on July 04,1997. (CustomerID, CompanyName, OrderID, Orderdate)

SQL Query:

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

Result:

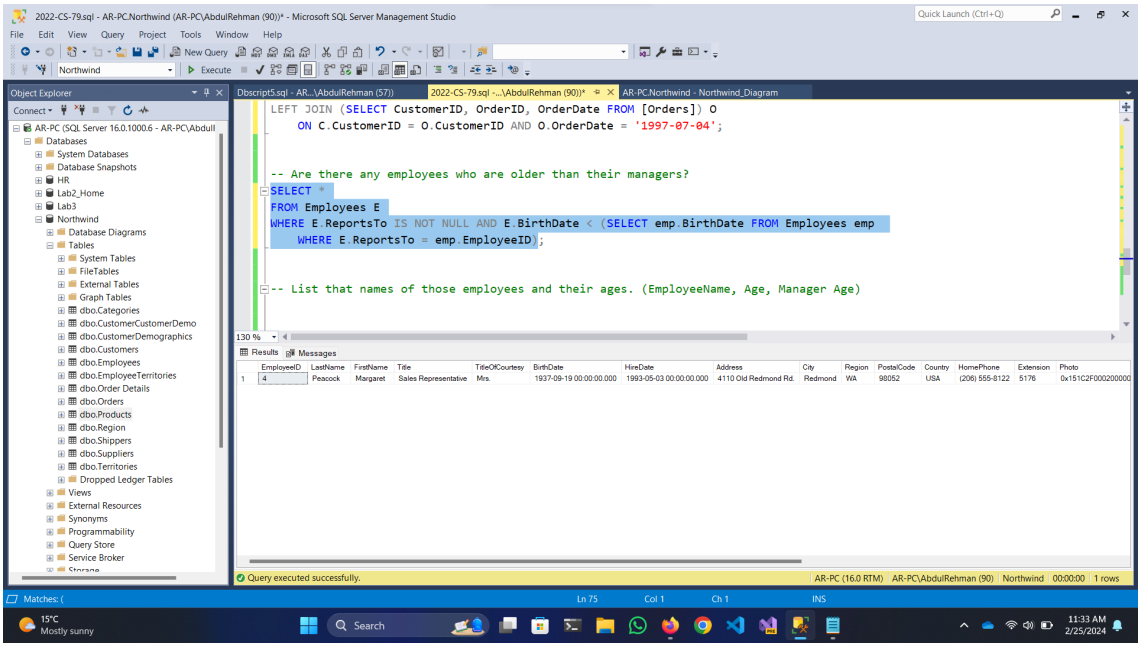


1.11 Are there any employees who are older than their managers?

SQL Query:

```
SELECT *
FROM Employees E
WHERE E.ReportsTo IS NOT NULL AND E.BirthDate <
    (SELECT emp.BirthDate FROM Employees emp
 WHERE E.ReportsTo = emp.EmployeeID);
```

Result:

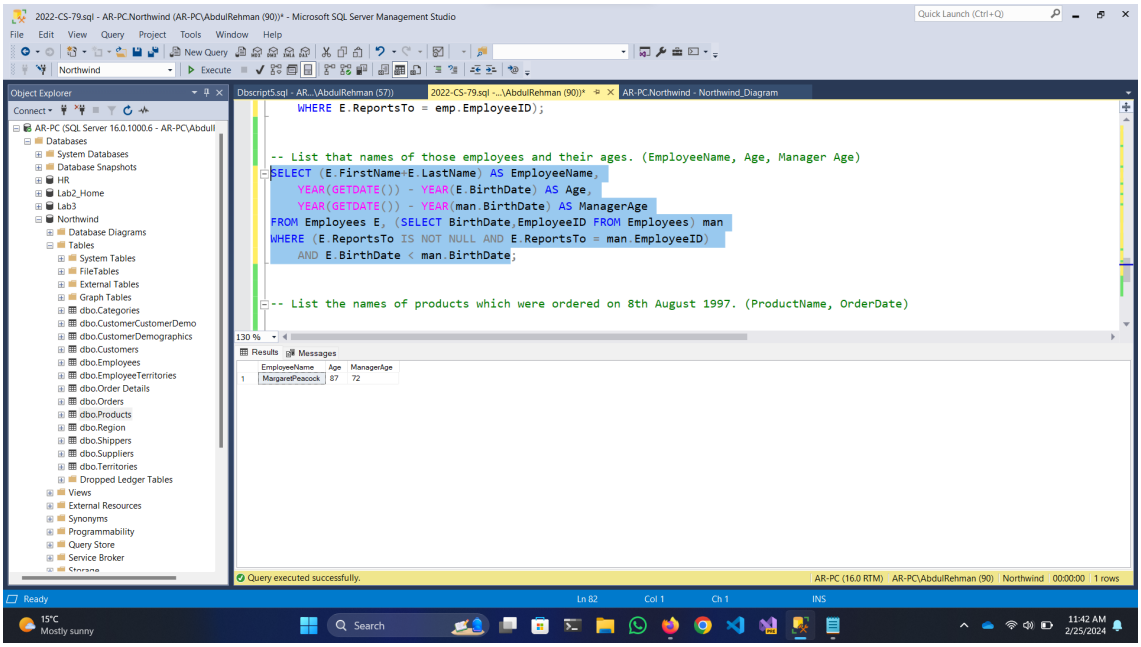


1.12 List that names of those employees and their ages. (EmployeeName, Age, Manager Age)

SQL Query:

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

Result:

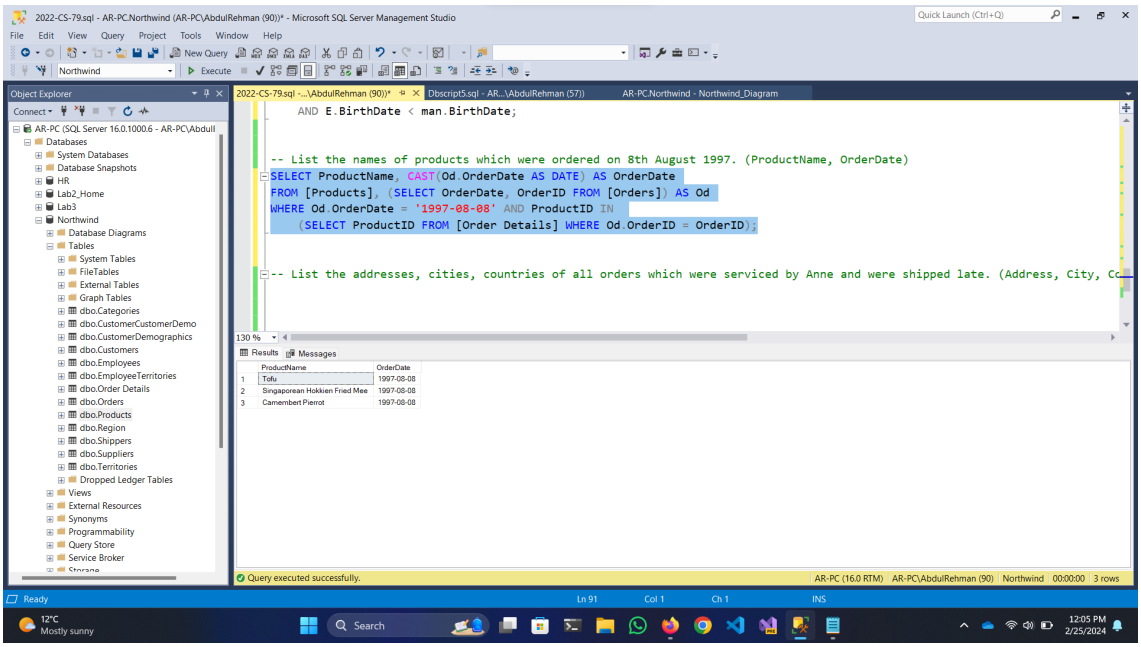


1.13 List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)

SQL Query:

```
SELECT ProductName, CAST(Od.OrderDate AS DATE) AS OrderDate
FROM [Products], (SELECT OrderDate, OrderID FROM [Orders]) AS Od
WHERE Od.OrderDate = '1997-08-08' AND ProductID IN
  (SELECT ProductID FROM [Order Details] WHERE Od.OrderID = OrderID);
```

Result:

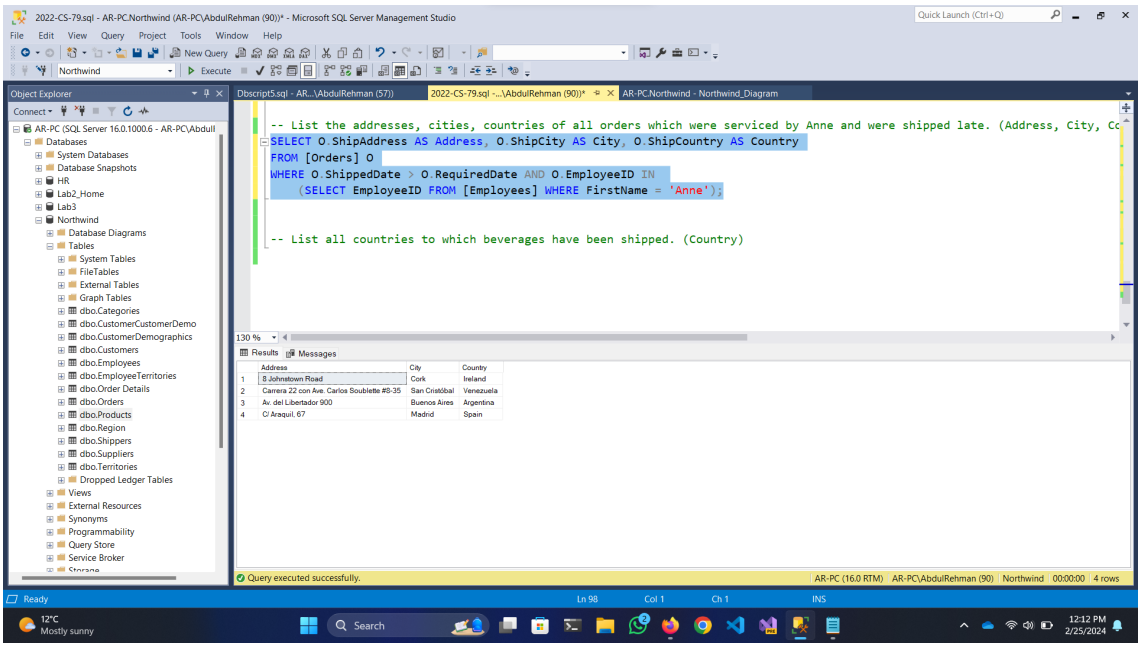


1.14 List the addresses, cities, countries of all orders which were serviced by Anne and were shipped late. (Address, City, Country)

SQL Query:

```
SELECT O.ShipAddress AS Address, O.ShipCity AS City, O.ShipCountry AS C
FROM [Orders] O
WHERE O.ShippedDate > O.RequiredDate AND O.EmployeeID IN
 (SELECT EmployeeID FROM [Employees] WHERE FirstName = 'Anne');
```

Result:



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

SQL Query:

```
SELECT O.ShipCountry AS Country
FROM [Orders] O
WHERE O.OrderID IN
  (SELECT OD.OrderID FROM [Order Details] OD WHERE OD.ProductID IN
  (SELECT P.ProductID FROM [Products] P WHERE P.CategoryID =
  (SELECT CategoryID FROM [Categories] WHERE LOWER(CategoryName) = 'beve
```

Result: