**Name:** Patel Aayushi Pravinbhai

**SQL Assignment-4**

**1. Create a stored procedure in the Northwind database that will calculate the average**

**value of Freight for a specified customer.Then, a business rule will be added that will**

**be triggered before every Update and Insert command in the Orders controller,and**

**will use the stored procedure to verify that the Freight does not exceed the average**

**freight. If it does, a message will be displayed and the command will be cancelled.**

CREATE PROCEDURE AvgFreight

AS

BEGIN

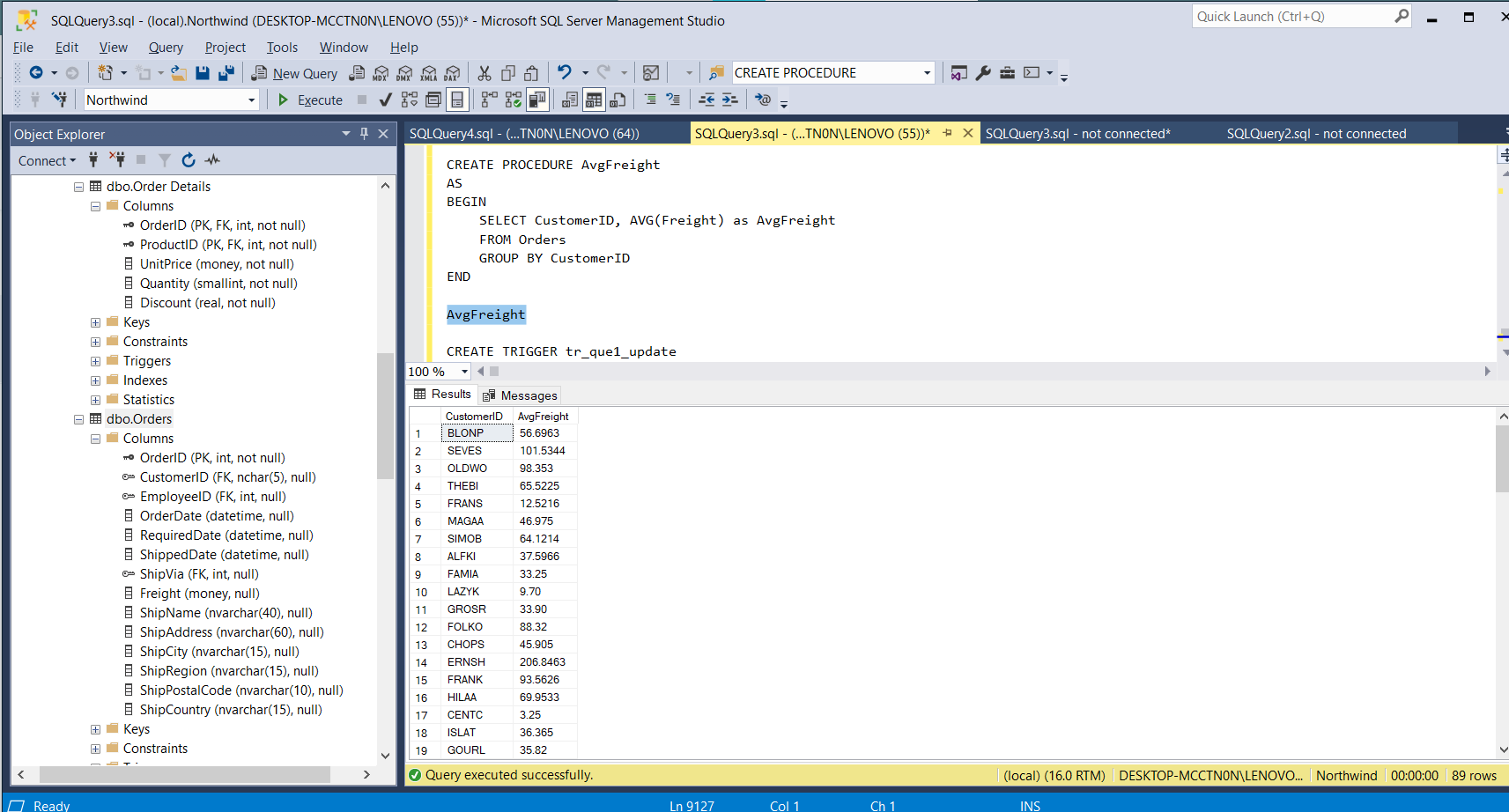
SELECT CustomerID, AVG(Freight) as AvgFreight

FROM Orders

GROUP BY CustomerID

END

AvgFreight

****

CREATE TRIGGER tr\_que1\_update

ON orders

INSTEAD OF UPDATE

AS

BEGIN

Declare @OrderID int

Declare @CustomerID varchar(50)

Declare @Freight money

Declare @AvgFreight money

Declare @t\_ave TABLE(CustomerID nchar(5), AvgFreight money)

INSERT @t\_ave

exec AvgFreight

Select \* Into #Temptable FROM Inserted

While(Exists(Select OrderID from #TempTable))

Begin

Select TOP 1 @OrderID = OrderID, @CustomerID = CustomerID, @Freight=Freight

FROM #Temptable

SET @AvgFreight = (SELECT AvgFreight FROM @t\_ave WHERE CustomerID = @CustomerID)

Print @Freight

Print @AvgFreight

IF @Freight > @AvgFreight

BEGIN

RAISERROR ('ABOVE AVERAGE',16,1)

END

ELSE

BEGIN

UPDATE Orders SET Freight = @Freight WHERE OrderID=@OrderID

END

Delete from #TempTable where OrderID = @OrderID

End

END

exec AvgFreight

UPDATE Orders SET Freight=5 WHERE OrderID = 10248

SELECT \* FROM Orders WHERE OrderID = 10248

CREATE TRIGGER tr\_que1\_insert

ON orders

INSTEAD OF INSERT

AS

BEGIN

Declare @OrderID int

Declare @CustomerID varchar(50)

Declare @Freight money

Declare @AvgFreight money

Declare @t\_ave TABLE(CustomerID nchar(5), AvgFreight money)

INSERT @t\_ave

exec AvgFreight

Select \* Into #Temptable FROM Inserted

While(Exists(Select OrderID from #TempTable))

Begin

Select TOP 1 @OrderID = OrderID, @CustomerID = CustomerID, @Freight=Freight

FROM #Temptable

SET @AvgFreight = (SELECT AvgFreight FROM @t\_ave WHERE CustomerID = @CustomerID)

IF @Freight > @AvgFreight

BEGIN

RAISERROR ('ABOVE AVERAGE',16,1)

END

ELSE

BEGIN

INSERT INTO Orders (CustomerID,EmployeeID,OrderDate,RequiredDate,ShippedDate,ShipVia,Freight,ShipName,ShipAddress,ShipCity,ShipRegion,ShipPostalCode,ShipCountry)

SELECT CustomerID,EmployeeID,OrderDate,RequiredDate,ShippedDate,ShipVia,Freight,ShipName,ShipAddress,ShipCity,ShipRegion,ShipPostalCode,ShipCountry

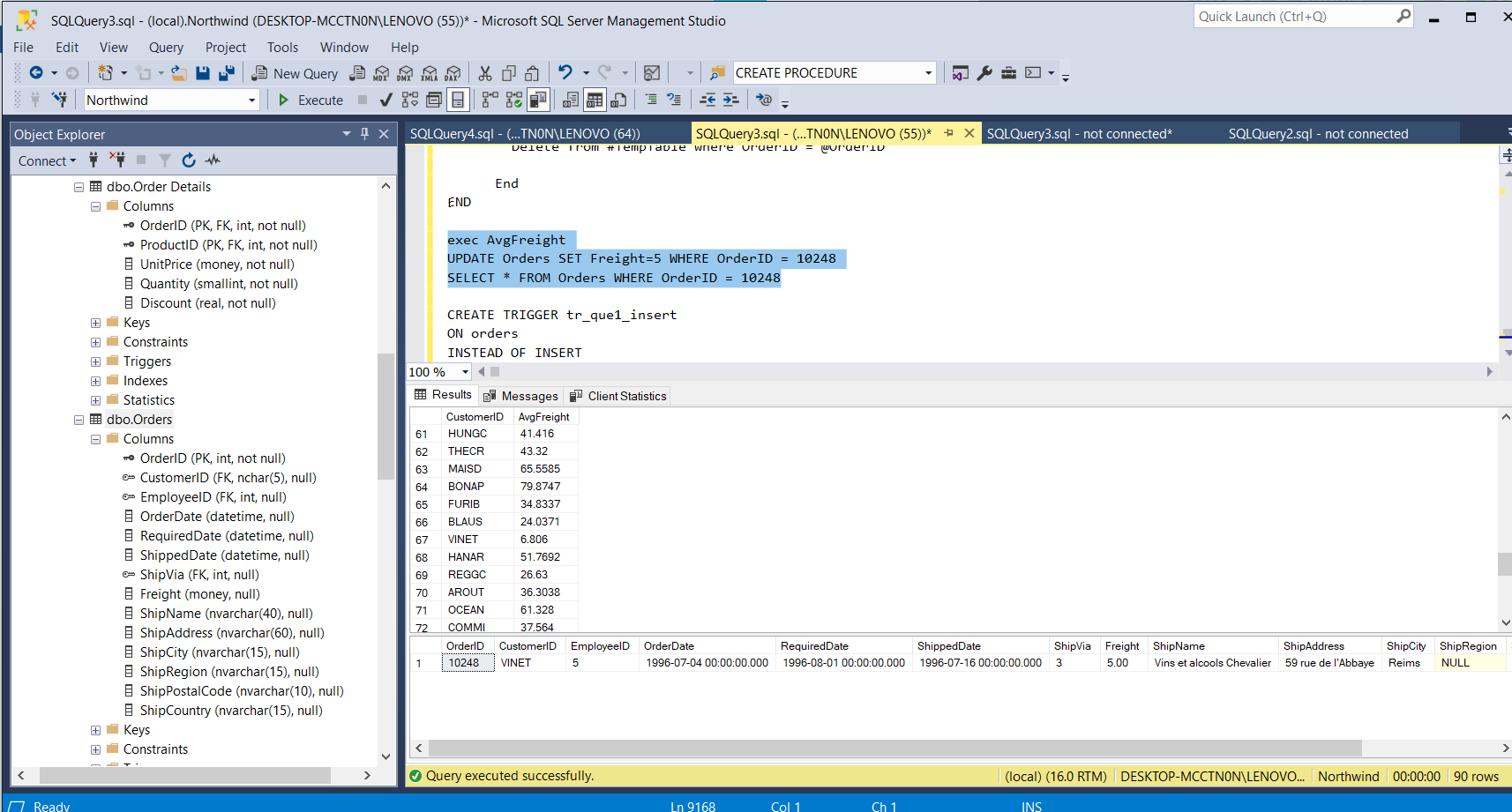
From Inserted

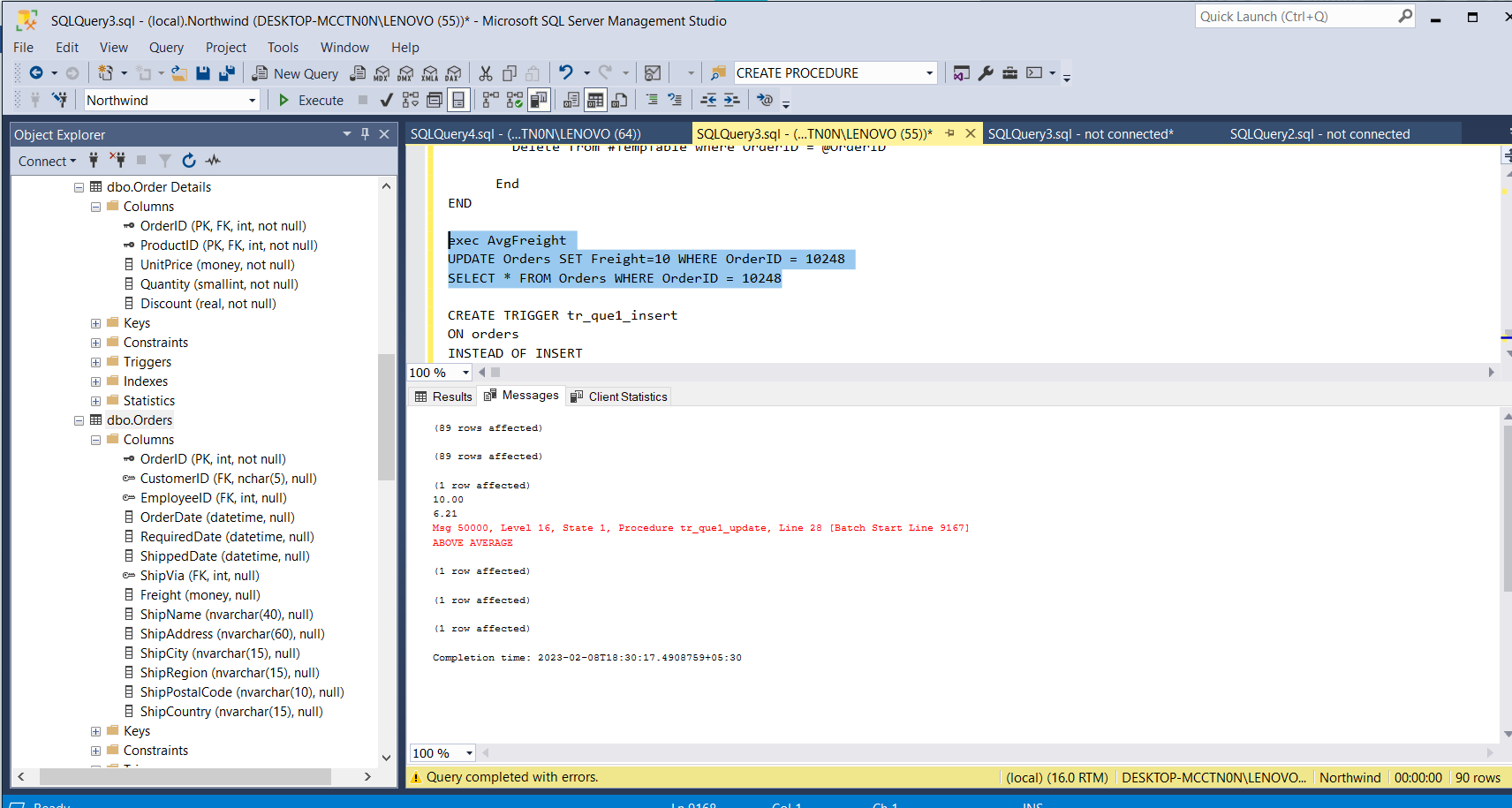
END

Delete from #TempTable where OrderID = @OrderID

End

END

****

****

**2. write a SQL query to Create Stored procedure in the Northwind database to retrieve**

**Employee Sales by Country.**

CREATE Procedure "Employee Sales by Country"

@Country nvarchar(15)

AS

BEGIN

Select Employees.Country, Employees.LastName, Employees.FirstName, Orders.ShippedDate, Orders.OrderID, "Order Details".ProductID,"Order Details".UnitPrice,"Order Details".Quantity, "Order Details".Discount

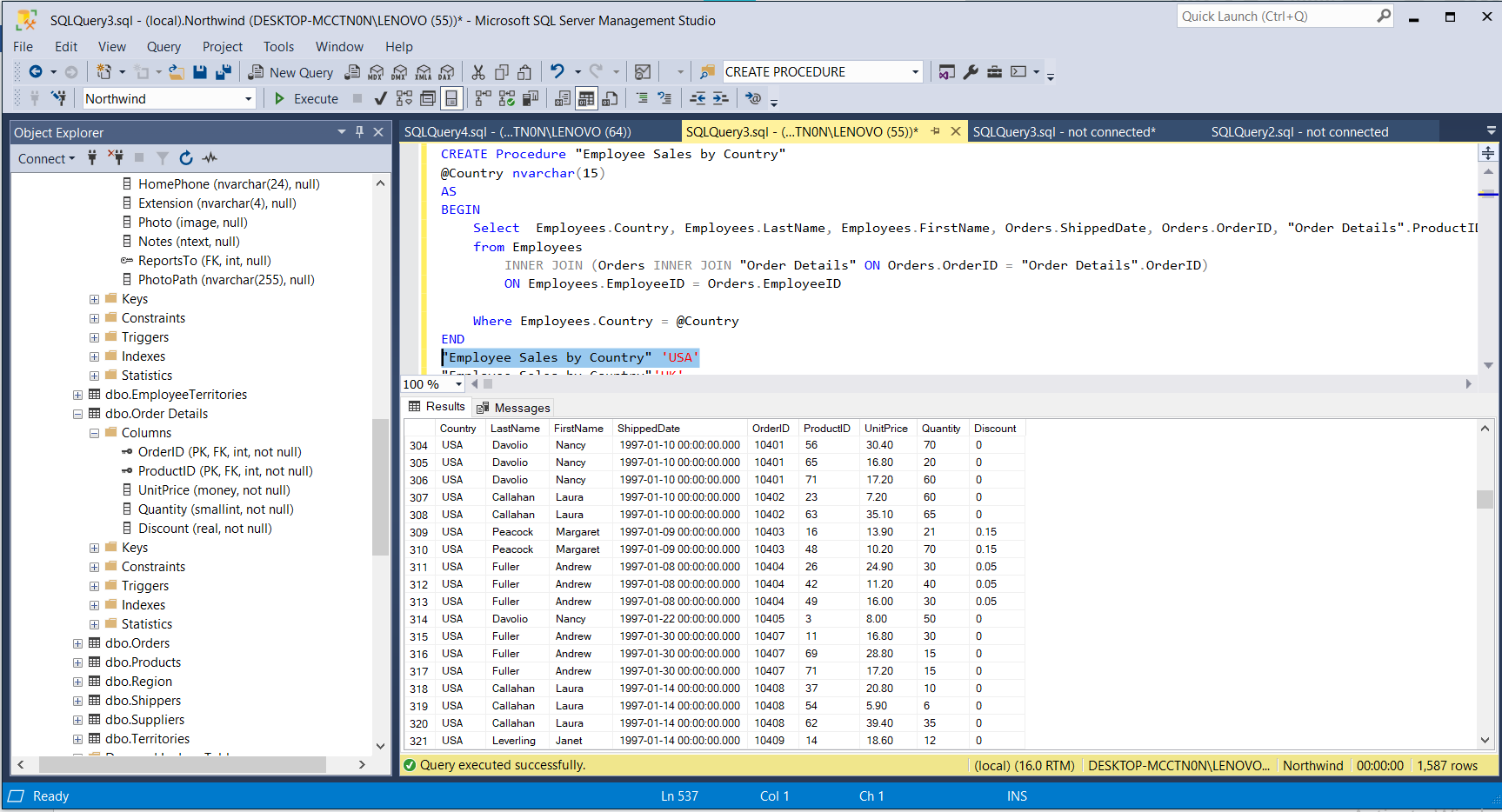
from Employees

INNER JOIN (Orders INNER JOIN "Order Details" ON Orders.OrderID = "Order Details".OrderID)

ON Employees.EmployeeID = Orders.EmployeeID

Where Employees.Country = @Country

END

****

**3. write a SQL query to Create Stored procedure in the Northwind database to retrieve**

**Sales by Year**

create view "Order Subtotals"

AS

BEGIN

SELECT "Order Details".OrderID, Sum(CONVERT(money,("Order Details".UnitPrice\*Quantity\*(1-Discount)/100))\*100) AS Subtotal

FROM "Order Details"

GROUP BY "Order Details".OrderID

END

CREATE procedure "Sales by Year"

@year int

AS

BEGIN

SELECT DATENAME(yy,ShippedDate) AS Year , Orders.OrderID, Orders.ShippedDate, Orders.OrderDate, "Order Subtotals".Subtotal

FROM Orders INNER JOIN "Order Subtotals" ON Orders.OrderID = "Order Subtotals".OrderID

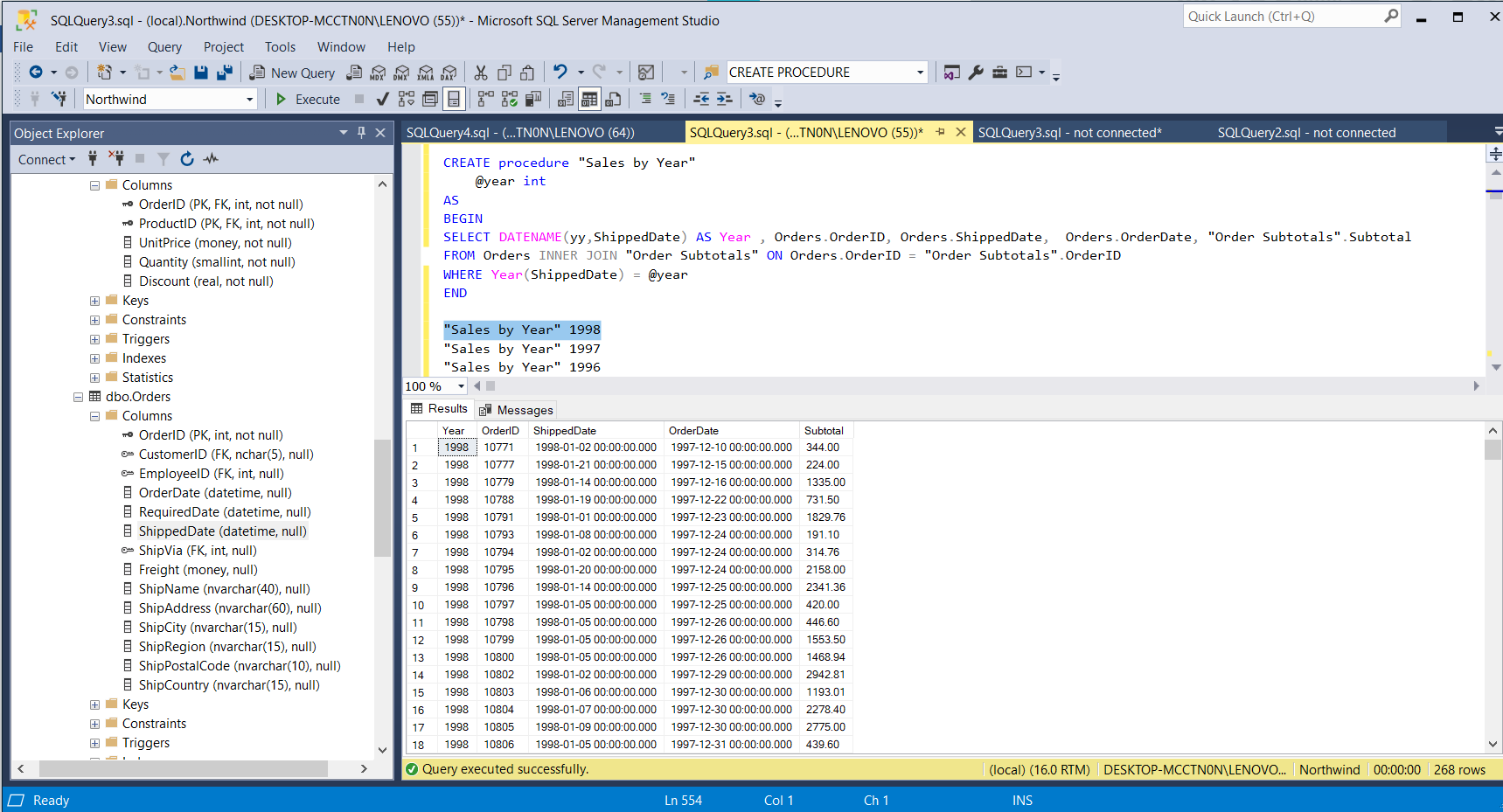
WHERE Year(ShippedDate) = @year

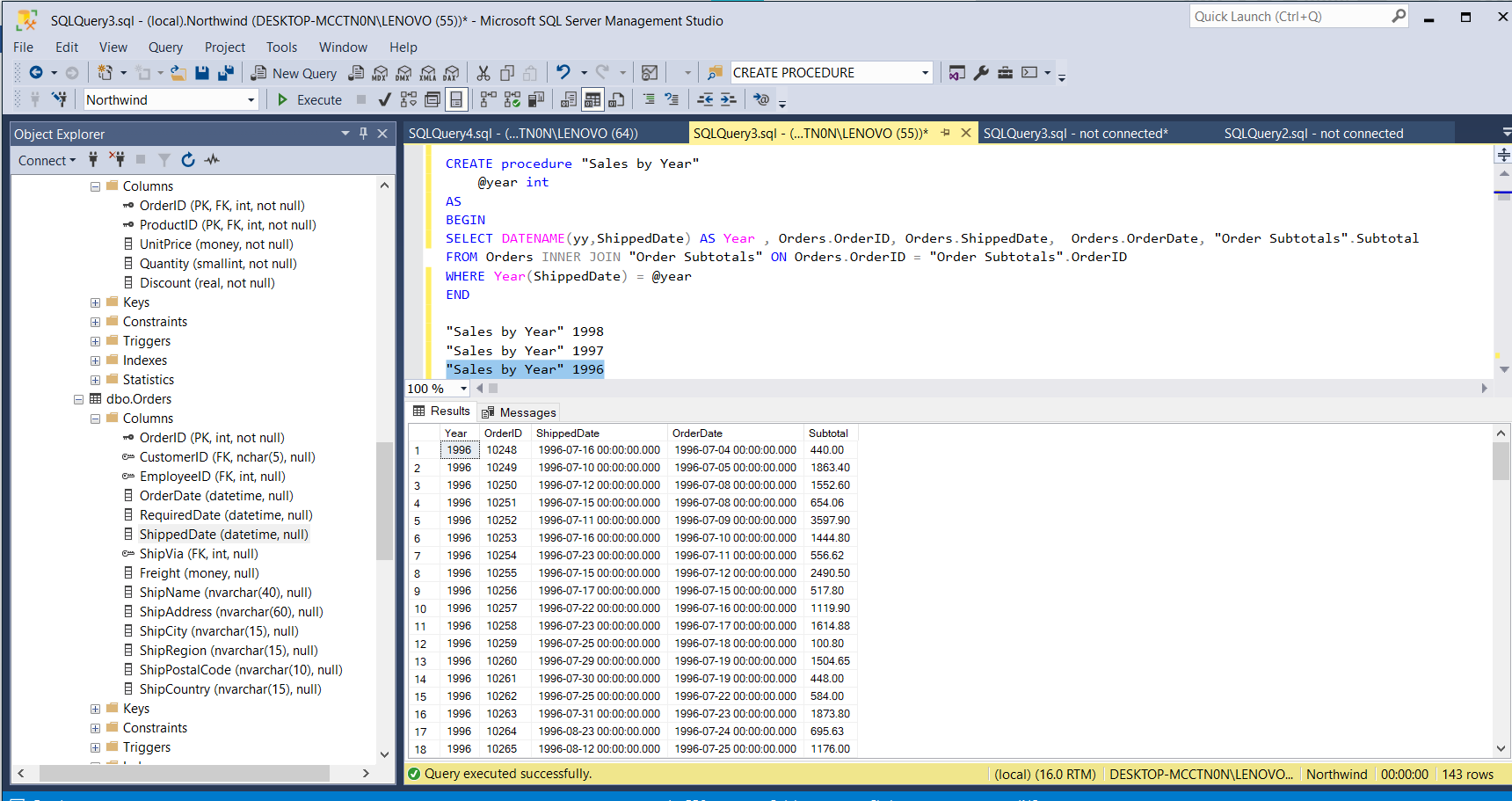
END

"Sales by Year" 1998

"Sales by Year" 1997

"Sales by Year" 1996

****

****

**4. write a SQL query to Create Stored procedure in the Northwind database to retrieve**

**Sales By Category**

CREATE PROCEDURE SalesByCategory

@CategoryName nvarchar(15)

AS

BEGIN

SELECT ProductName,

TotalPurchase=ROUND(SUM(CONVERT(decimal(14,2), OD.Quantity \* (1-OD.Discount) \* OD.UnitPrice)), 0)

FROM [Order Details] OD, Orders O, Products P, Categories C

WHERE OD.OrderID = O.OrderID

AND OD.ProductID = P.ProductID

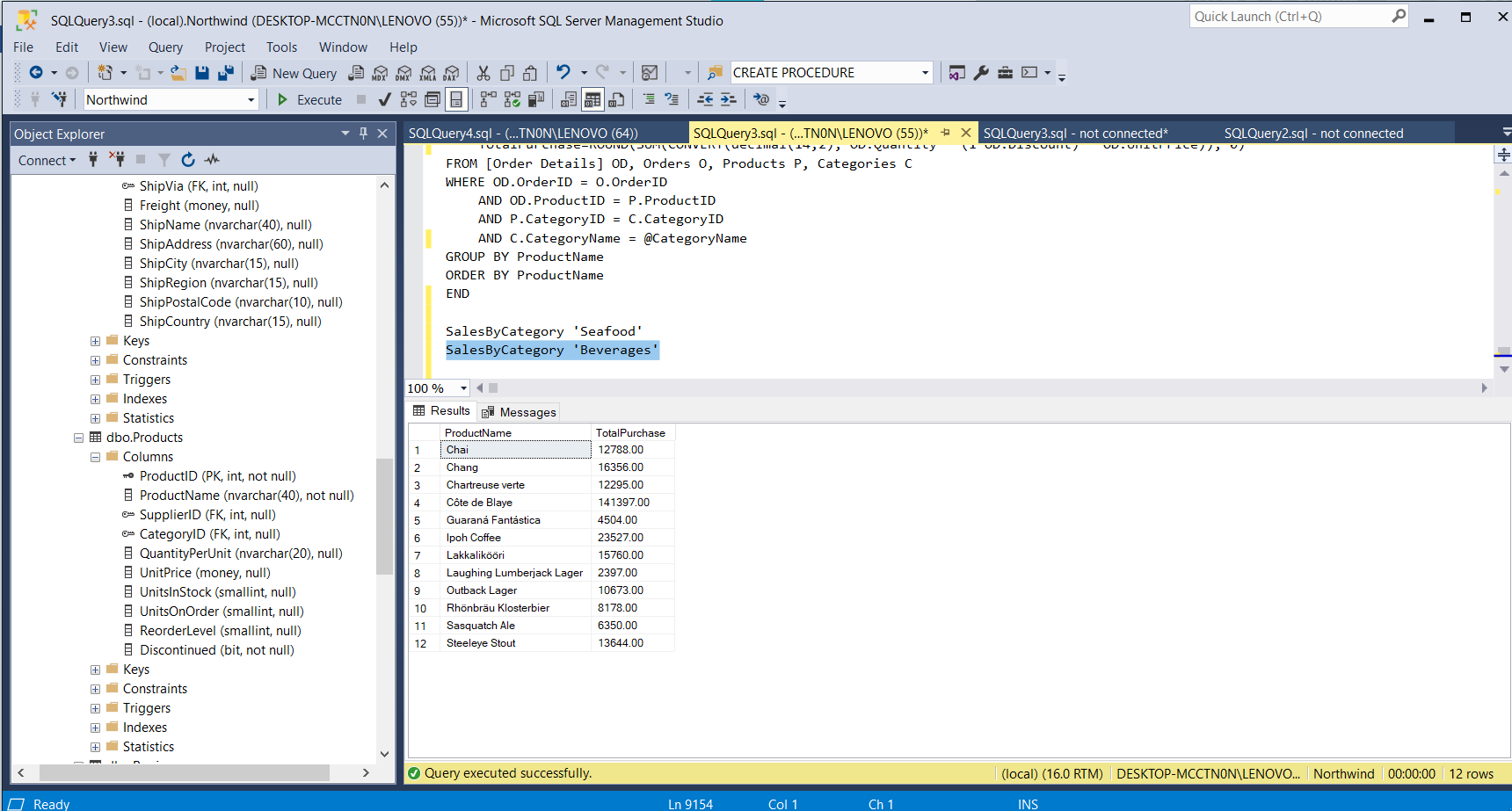
AND P.CategoryID = C.CategoryID

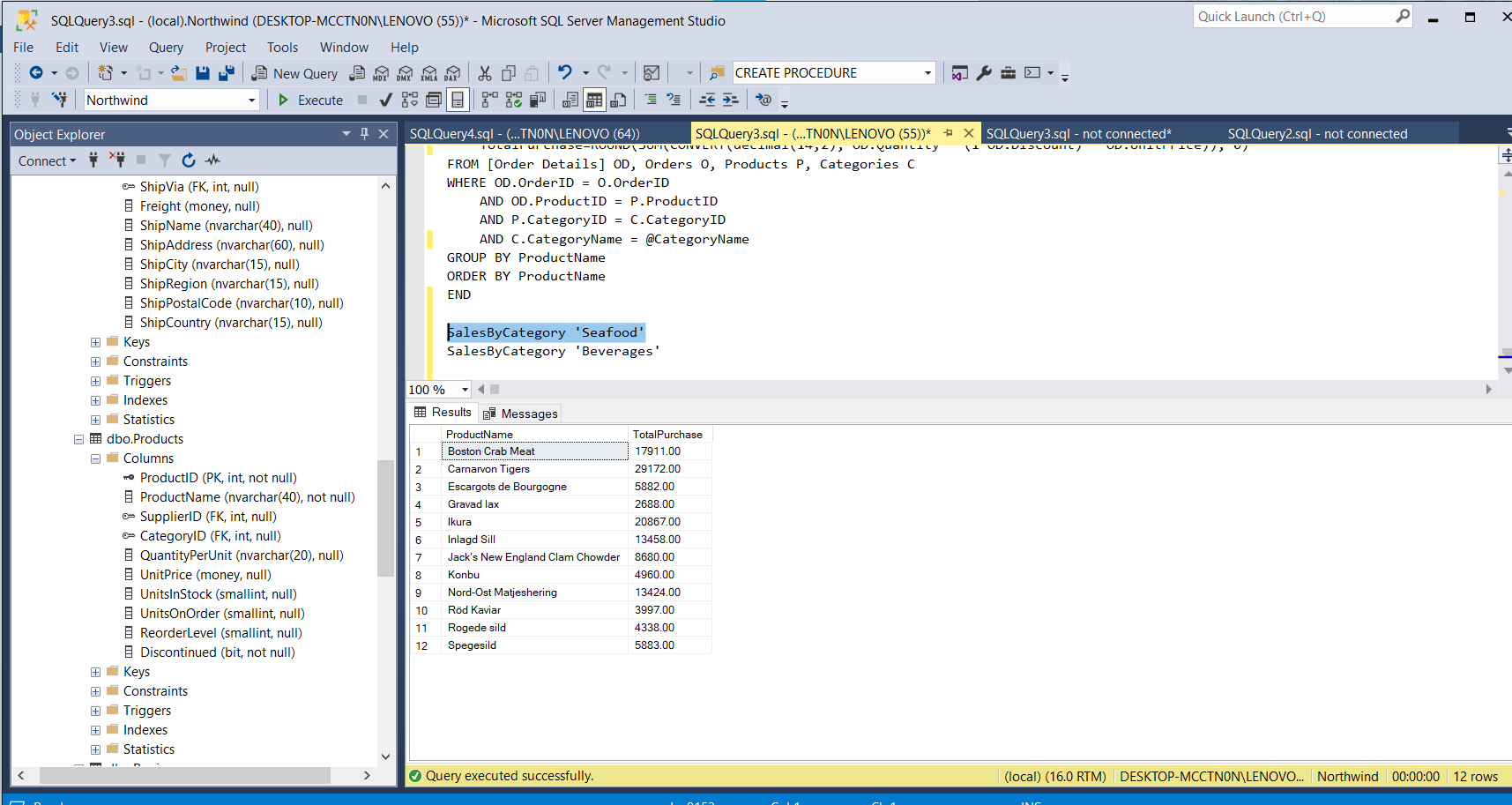
AND C.CategoryName = @CategoryName

GROUP BY ProductName

ORDER BY ProductName

END

****

****

**5. write a SQL query to Create Stored procedure in the Northwind database to retrieve**

**Ten Most Expensive Products**

Create procedure "Ten Most Expensive Products"

AS

BEGIN

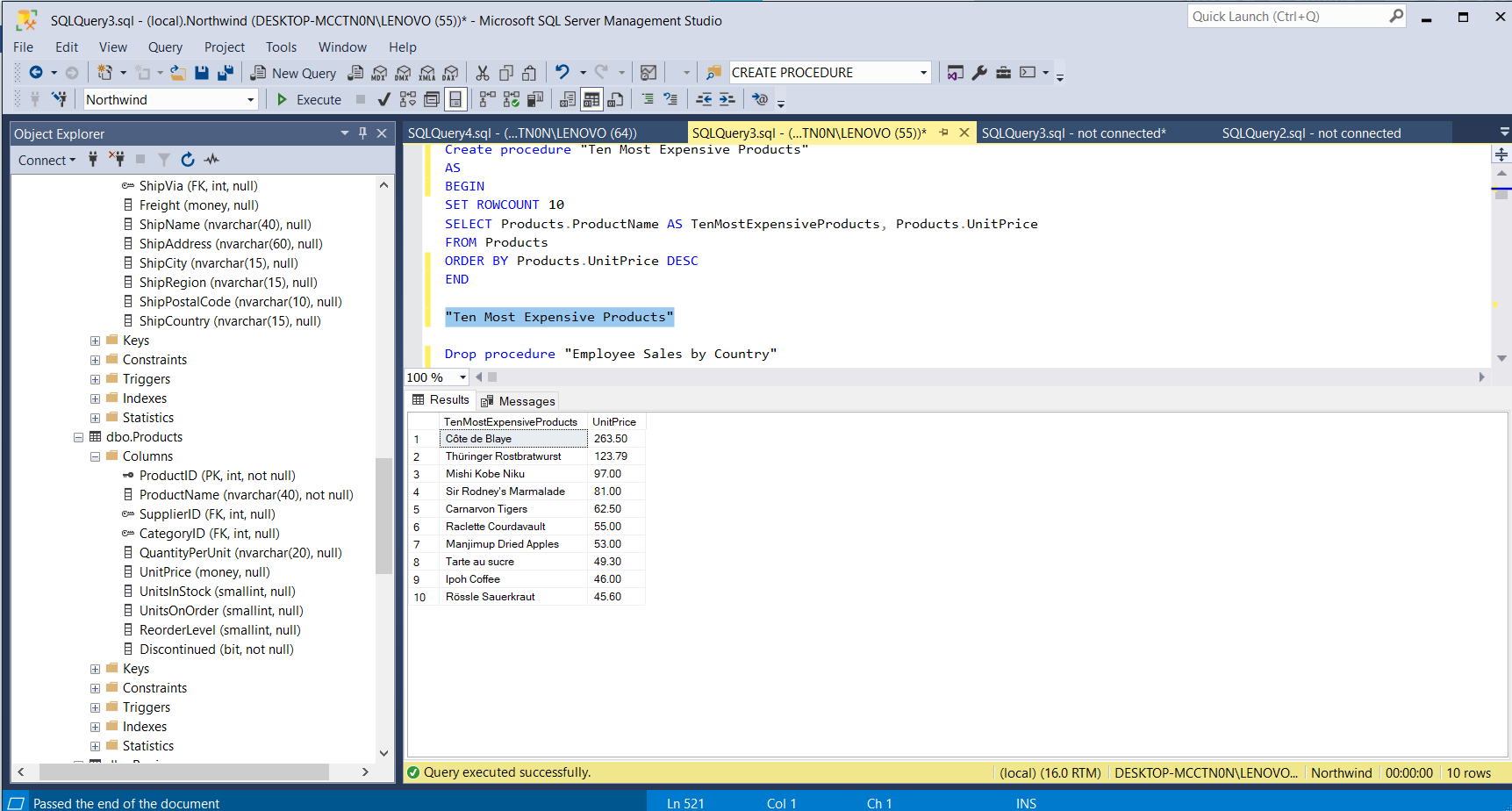
SET ROWCOUNT 10

SELECT Products.ProductName AS TenMostExpensiveProducts, Products.UnitPrice

FROM Products

ORDER BY Products.UnitPrice DESC

END

****

**6. write a SQL query to Create Stored procedure in the Northwind database to insert**

**Customer Order Details**

CREATE PROCEDURE spInsert @ordID int, @pID int , @uPrice decimal(5,2), @totalitems int, @disc int

AS

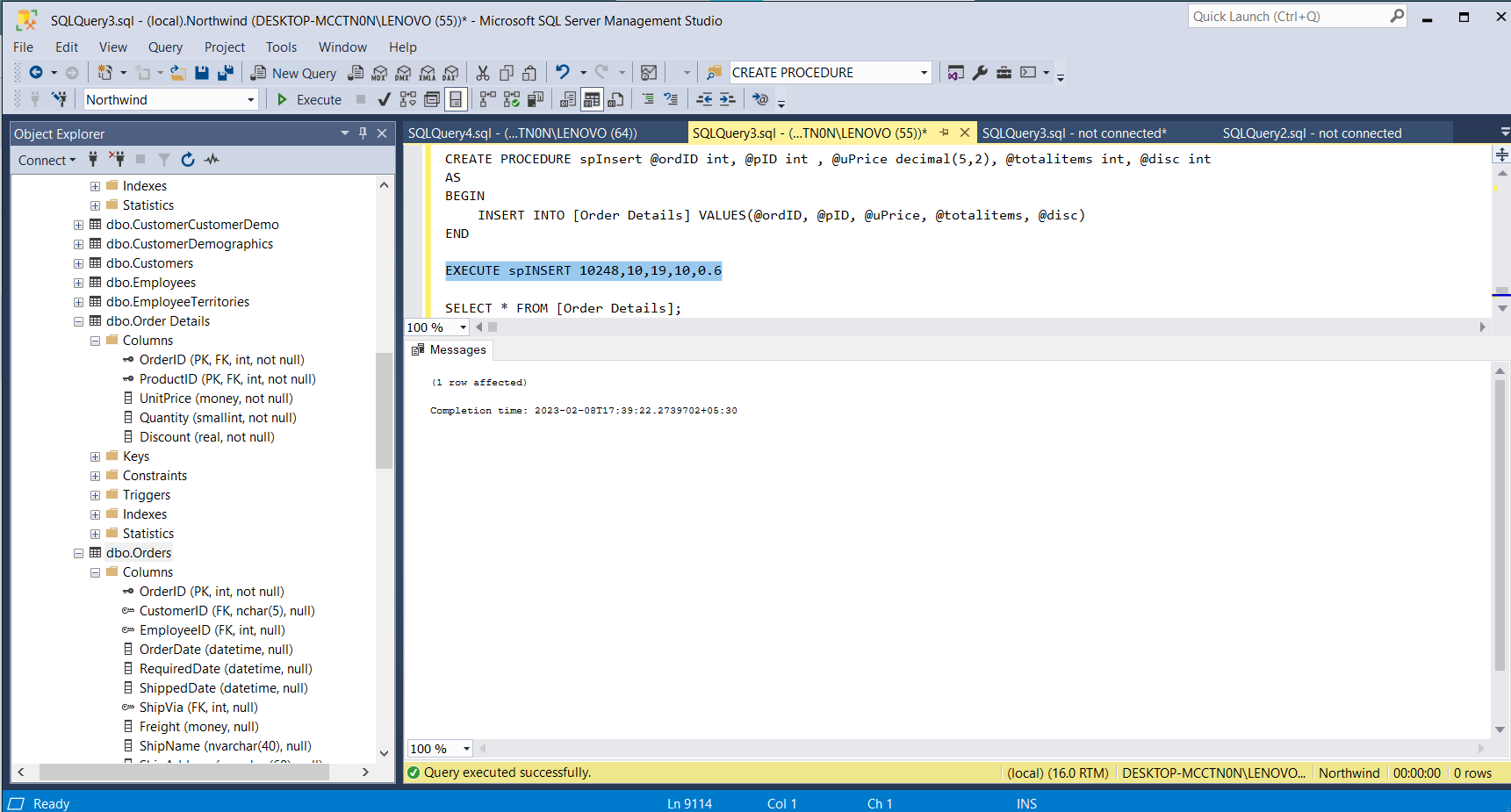
BEGIN

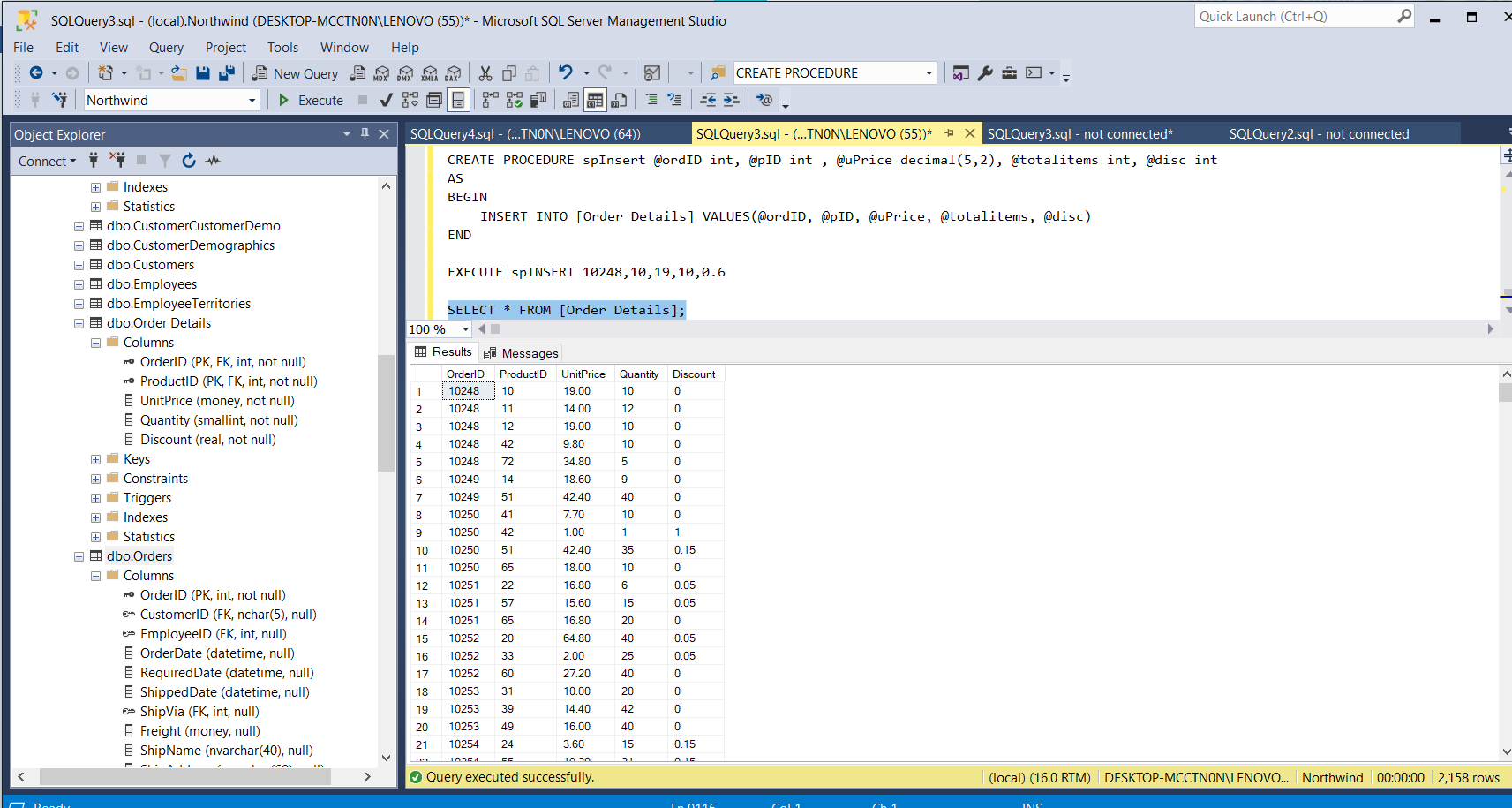
INSERT INTO [Order Details] VALUES(@ordID, @pID, @uPrice, @totalitems, @disc)

END

EXECUTE spINSERT 10248,10,19,10,0.6

SELECT \* FROM [Order Details];

****

****

**7. write a SQL query to Create Stored procedure in the Northwind database to update**

**Customer Order Details**

Select \* from "Order Details"

CREATE PROCEDURE spUpdate

@ordID int, @pID int , @uPrice decimal(5,2), @totalitems int, @disc int

AS

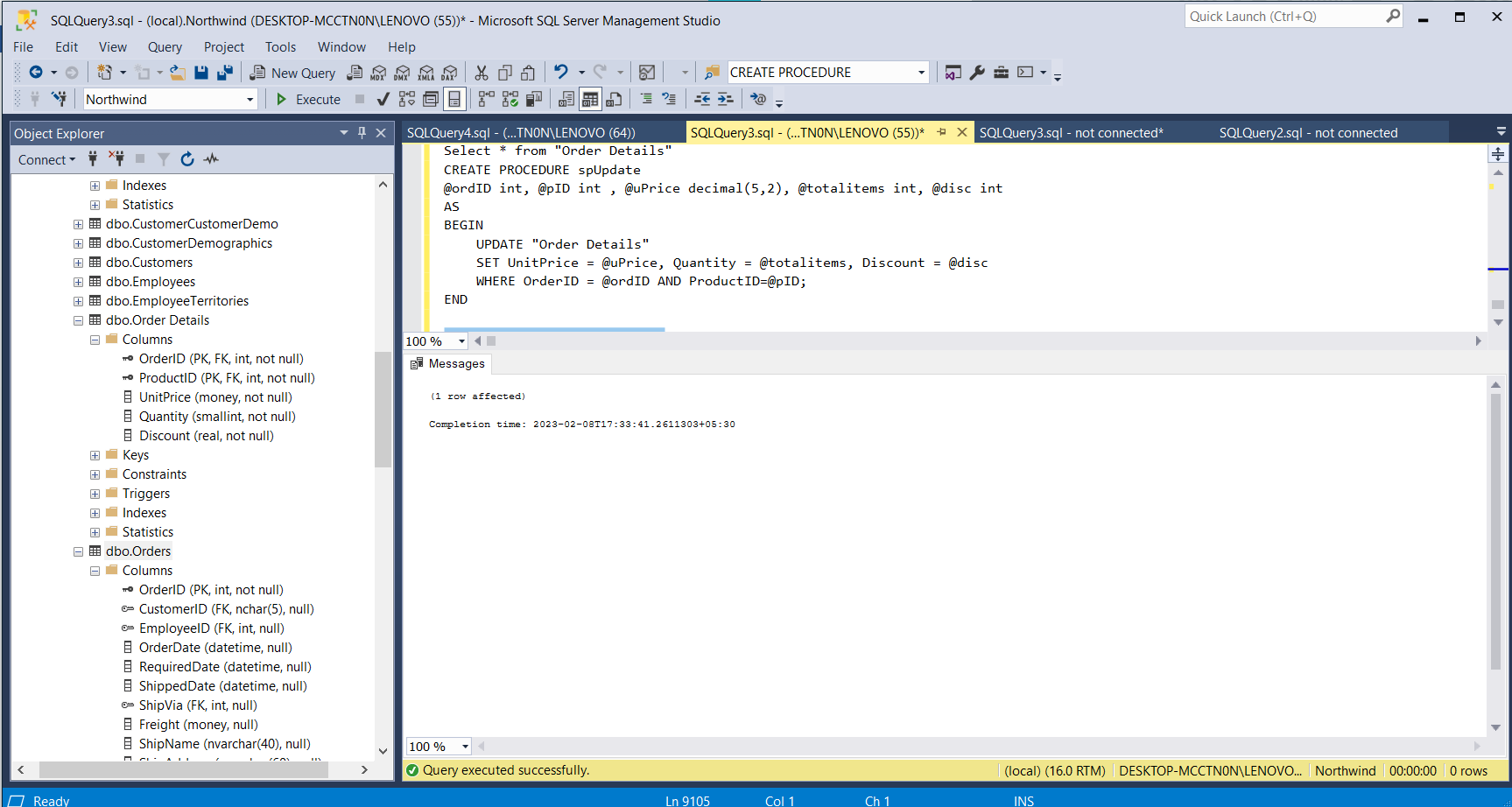
BEGIN

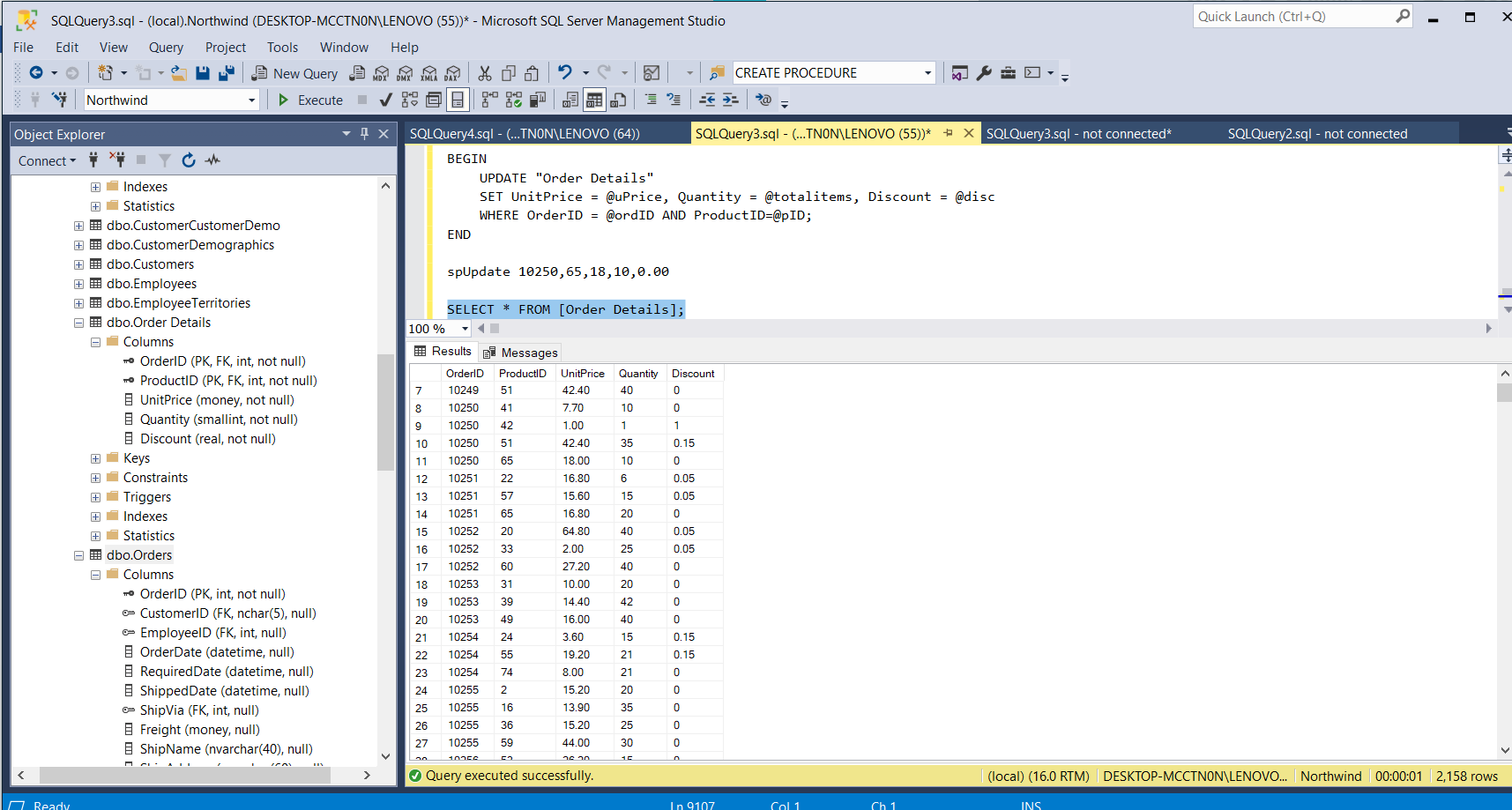
UPDATE "Order Details"

SET UnitPrice = @uPrice, Quantity = @totalitems, Discount = @disc

WHERE OrderID = @ordID AND ProductID=@pID;

END

****

****