**ASSIGNMENT-4**

**Create Stored procedure in Northwind database to insert or update a record in a table**

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.

ANS.

CREATE PROCEDURE que1

AS

SELECT CustomerID, AVG(Freight) as AvgFreight

FROM Orders

GROUP BY CustomerID

GO

insert into orders values ( 'VAFFE', 8,'1997-08-25 00:00:00.000', '1997-08-01 00:00:00.000','1997-01-01 00:00:00.000', 1,80, 'Wolski Zajazd', 'ul. Filtrowa 68',

'Warszawa', 'Tachira', 24100, 'brazil')

exec que1

UPDATE Orders SET Freight=30 WHERE OrderID = 10248

SELECT \* FROM Orders WHERE OrderID = 10248

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 que1

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

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 que1

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

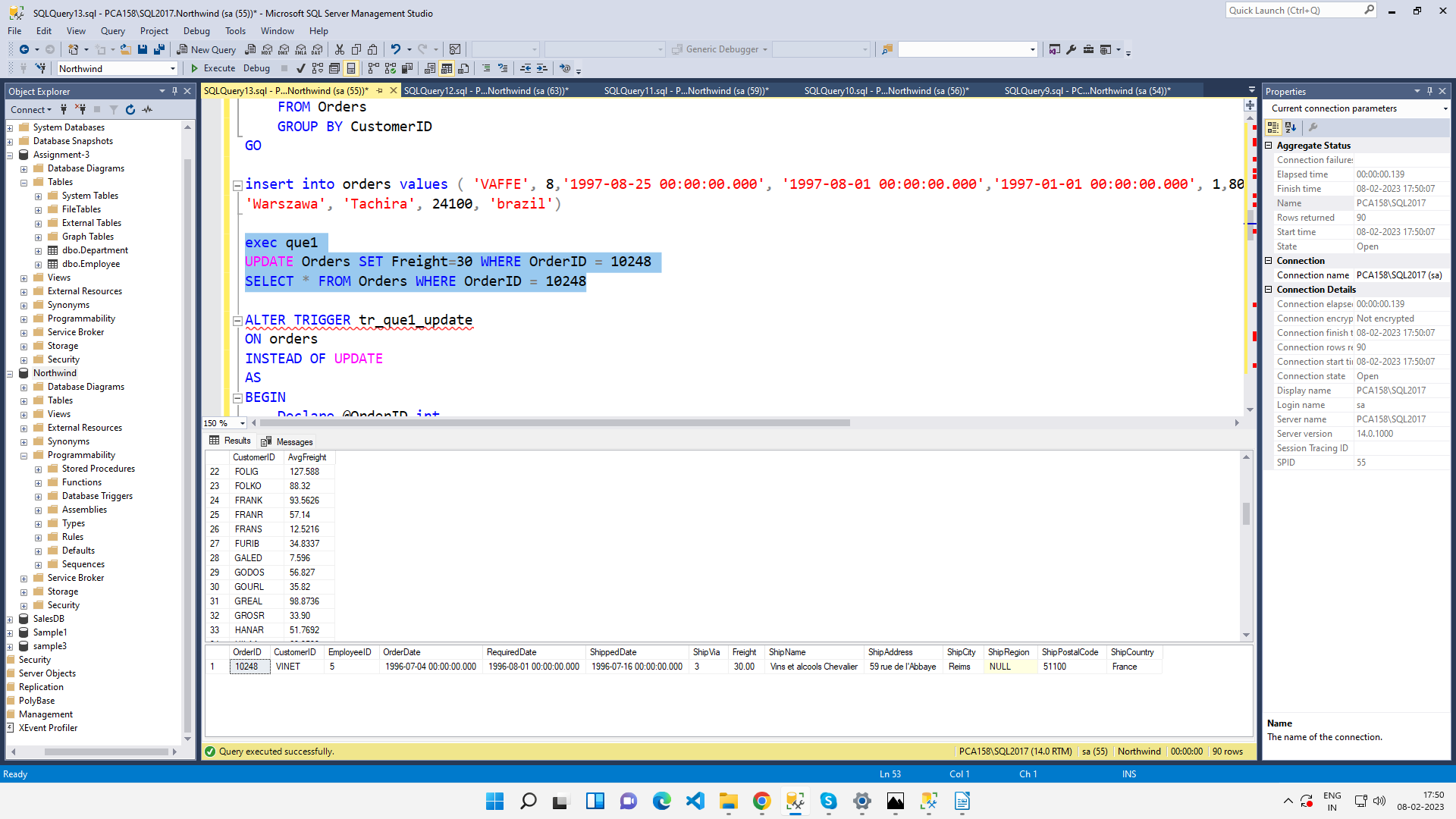
From Inserted

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

ANS.

create procedure [dbo].[Employee Sales by Country]

@Beginning\_Date DateTime, @Ending\_Date DateTime AS

SELECT Employees.Country, Employees.LastName, Employees.FirstName, Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal AS SaleAmount

FROM Employees INNER JOIN

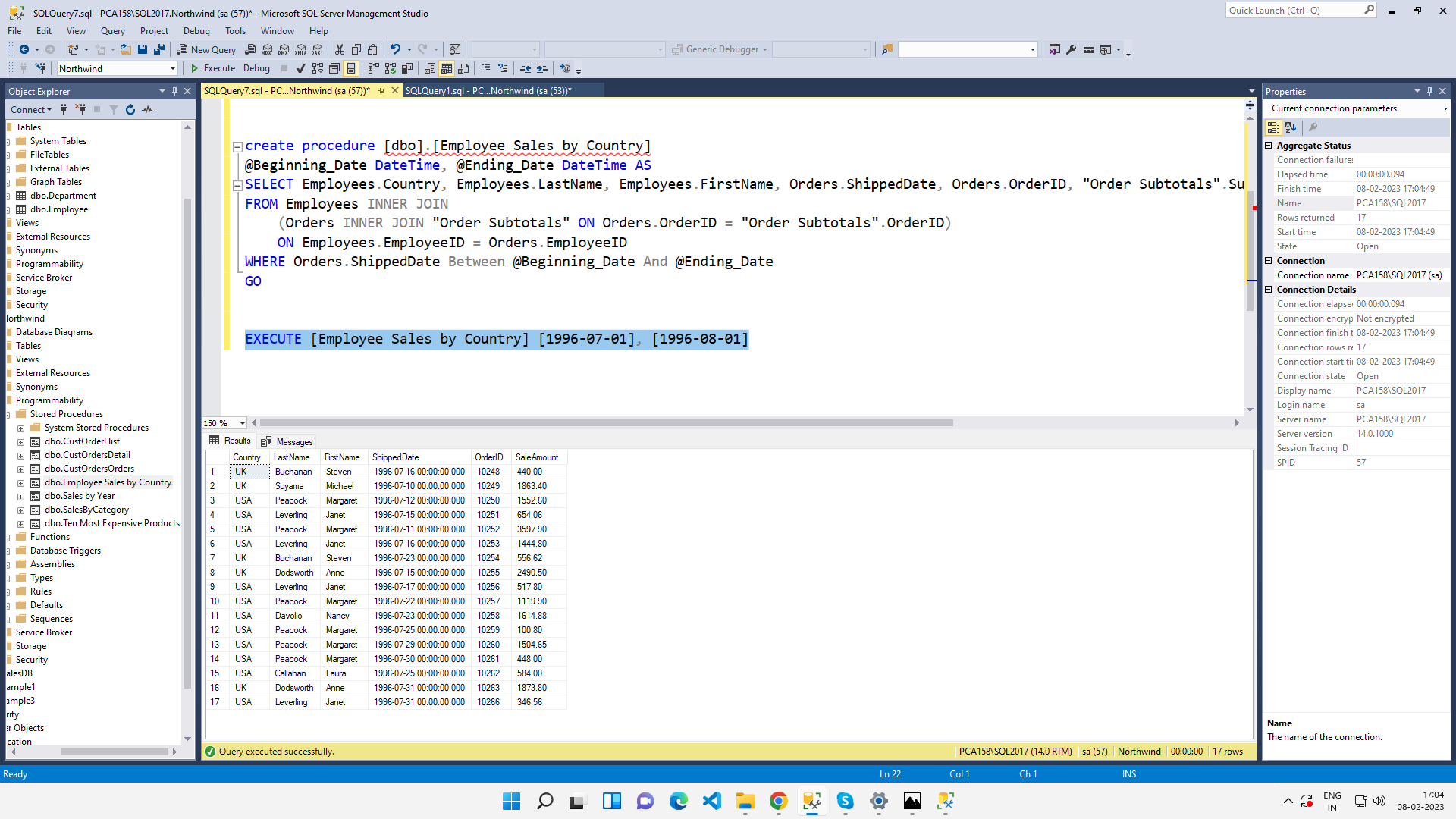
(Orders INNER JOIN "Order Subtotals" ON Orders.OrderID = "Order Subtotals".OrderID)

ON Employees.EmployeeID = Orders.EmployeeID

WHERE Orders.ShippedDate Between @Beginning\_Date And @Ending\_Date

GO

EXECUTE [Employee Sales by Country] [1996-07-01], [1996-08-01]



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

Sales by Year

ANS.

create procedure [dbo].[Sales by Year]

@Beginning\_Date DateTime, @Ending\_Date DateTime AS

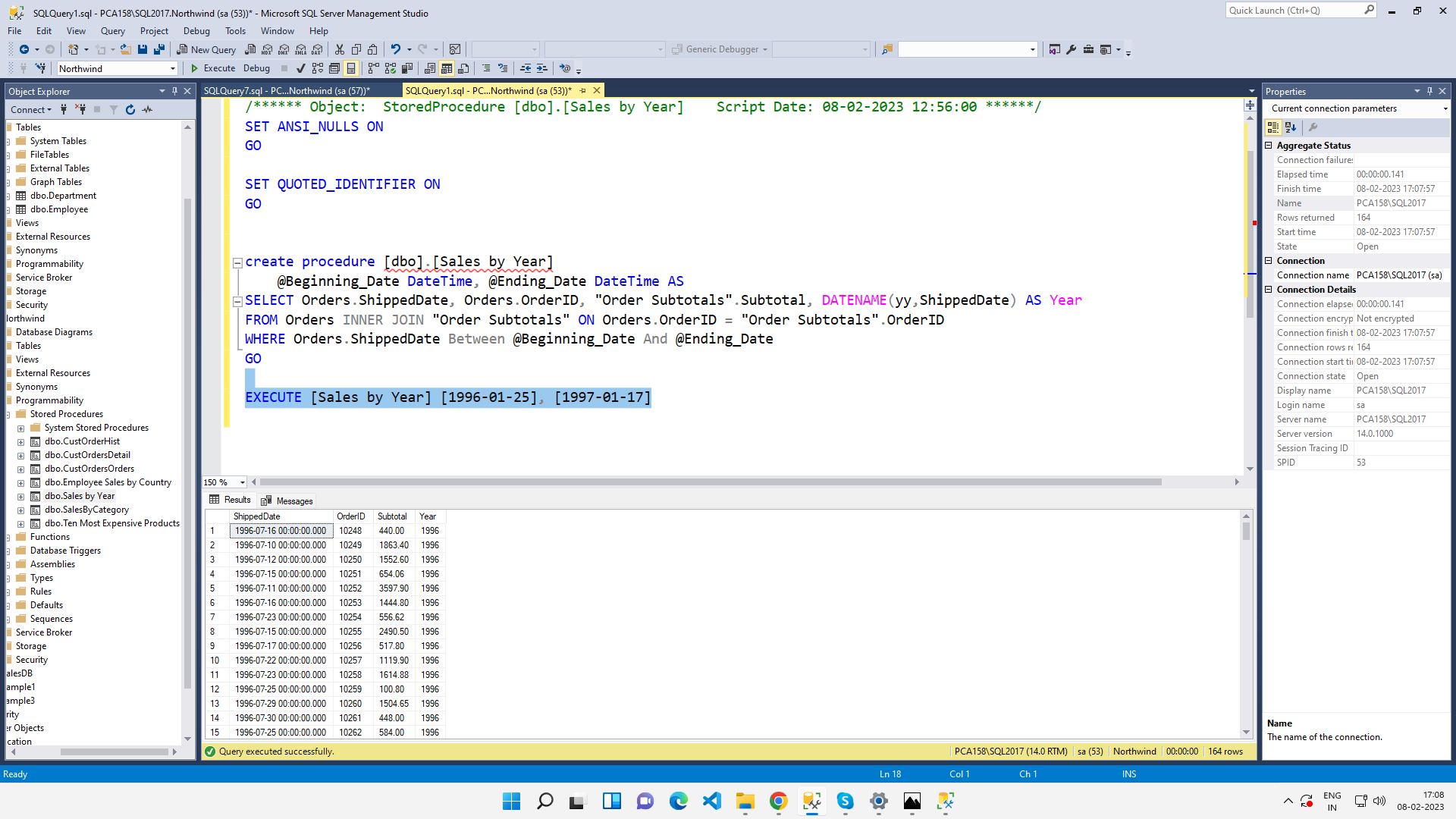
SELECT Orders.ShippedDate, Orders.OrderID, "Order Subtotals".Subtotal, DATENAME(yy,ShippedDate) AS Year

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

WHERE Orders.ShippedDate Between @Beginning\_Date And @Ending\_Date

GO

EXECUTE [Sales by Year] [1996-1-25], [1997-01-17]



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

Sales By Category

ANS.

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

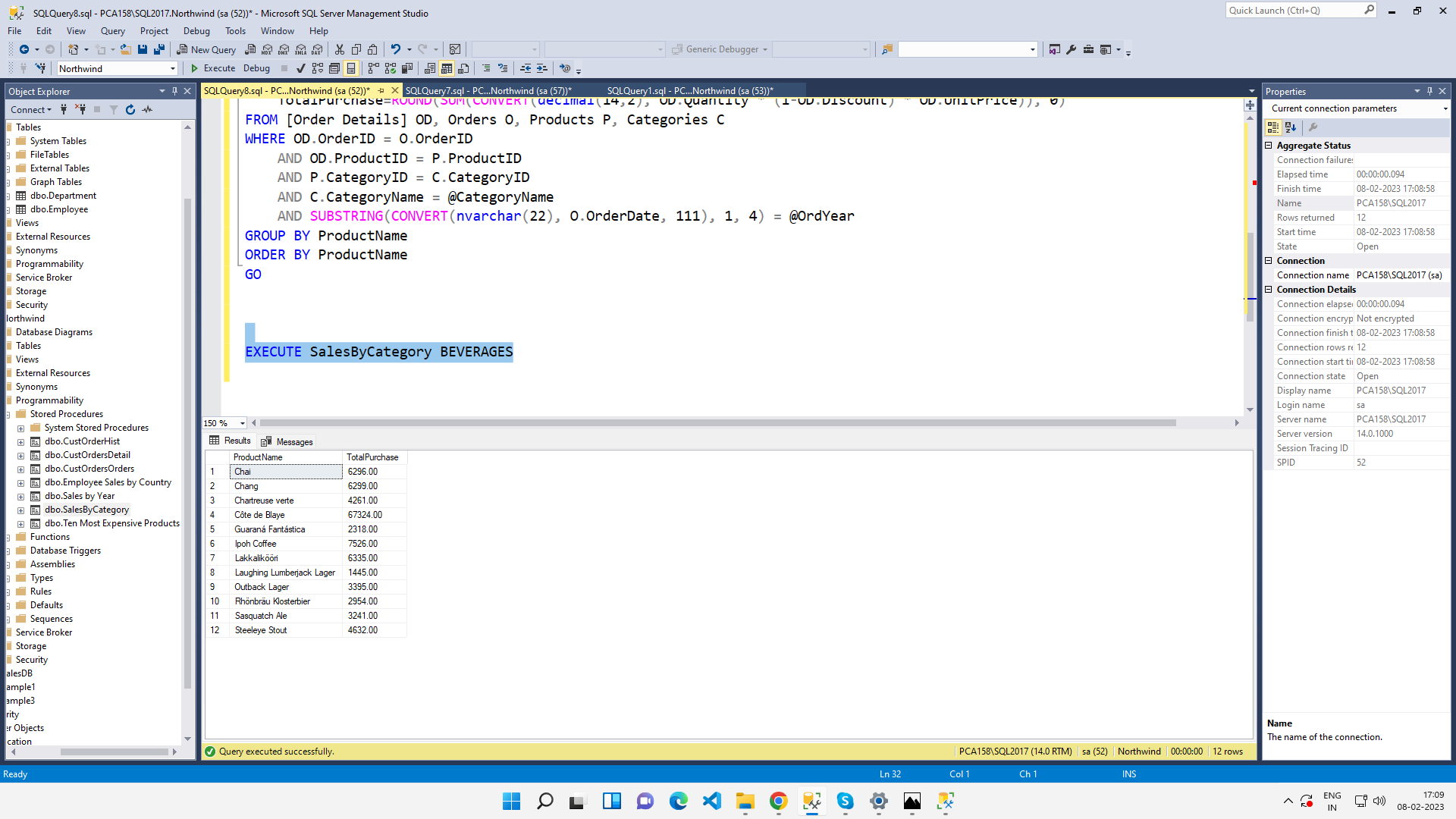
AND SUBSTRING(CONVERT(nvarchar(22), O.OrderDate, 111), 1, 4) = @OrdYear

GROUP BY ProductName

ORDER BY ProductName

GO

EXECUTE SalesByCategory BEVERAGES



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

Ten Most Expensive Products

ANS.

create procedure [dbo].[Ten Most Expensive Products] AS

SET ROWCOUNT 10

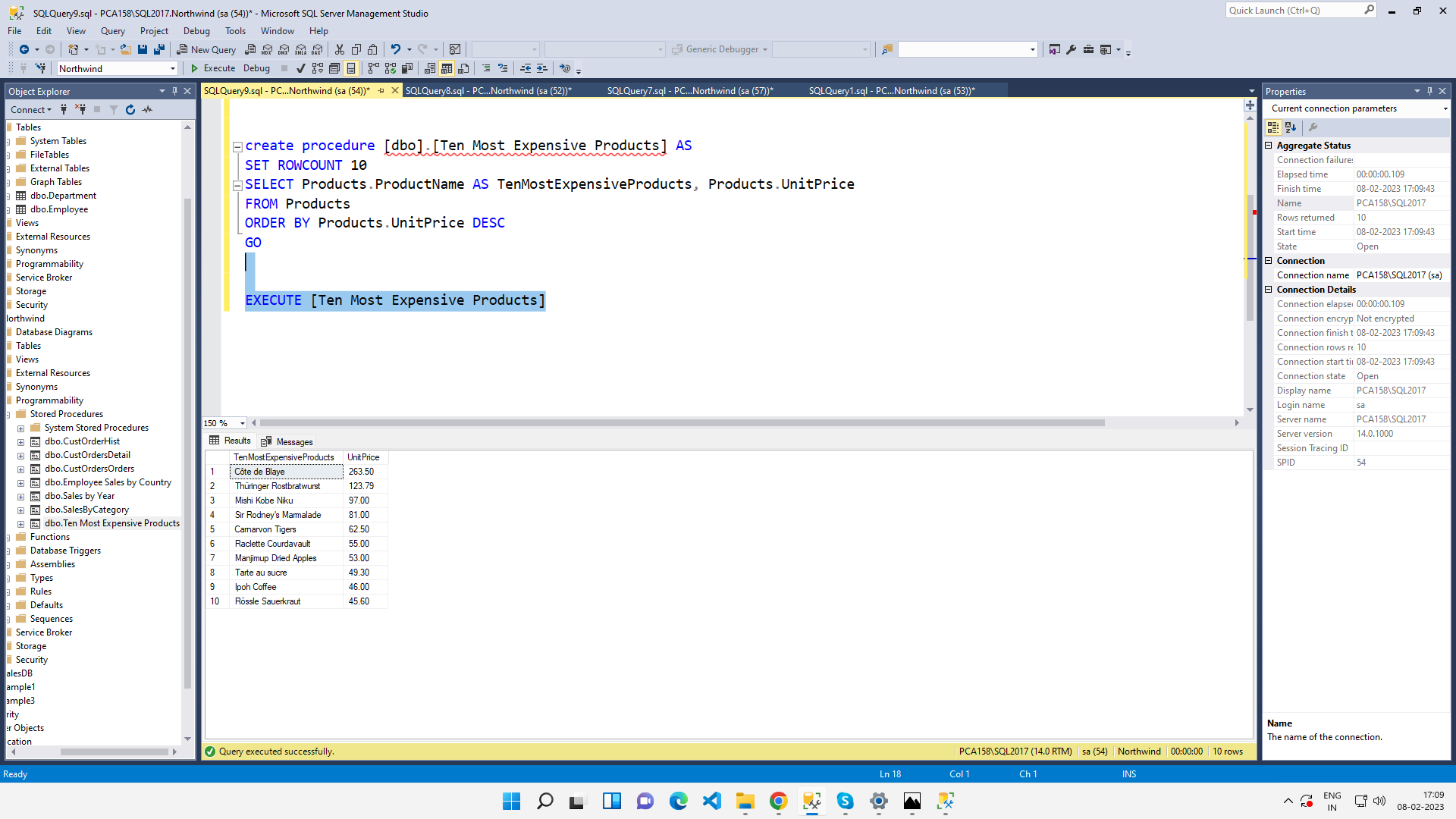
SELECT Products.ProductName AS TenMostExpensiveProducts, Products.UnitPrice

FROM Products

ORDER BY Products.UnitPrice DESC

GO

EXECUTE [Ten Most Expensive Products]



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

Customer Order Details

ANS.

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

AS

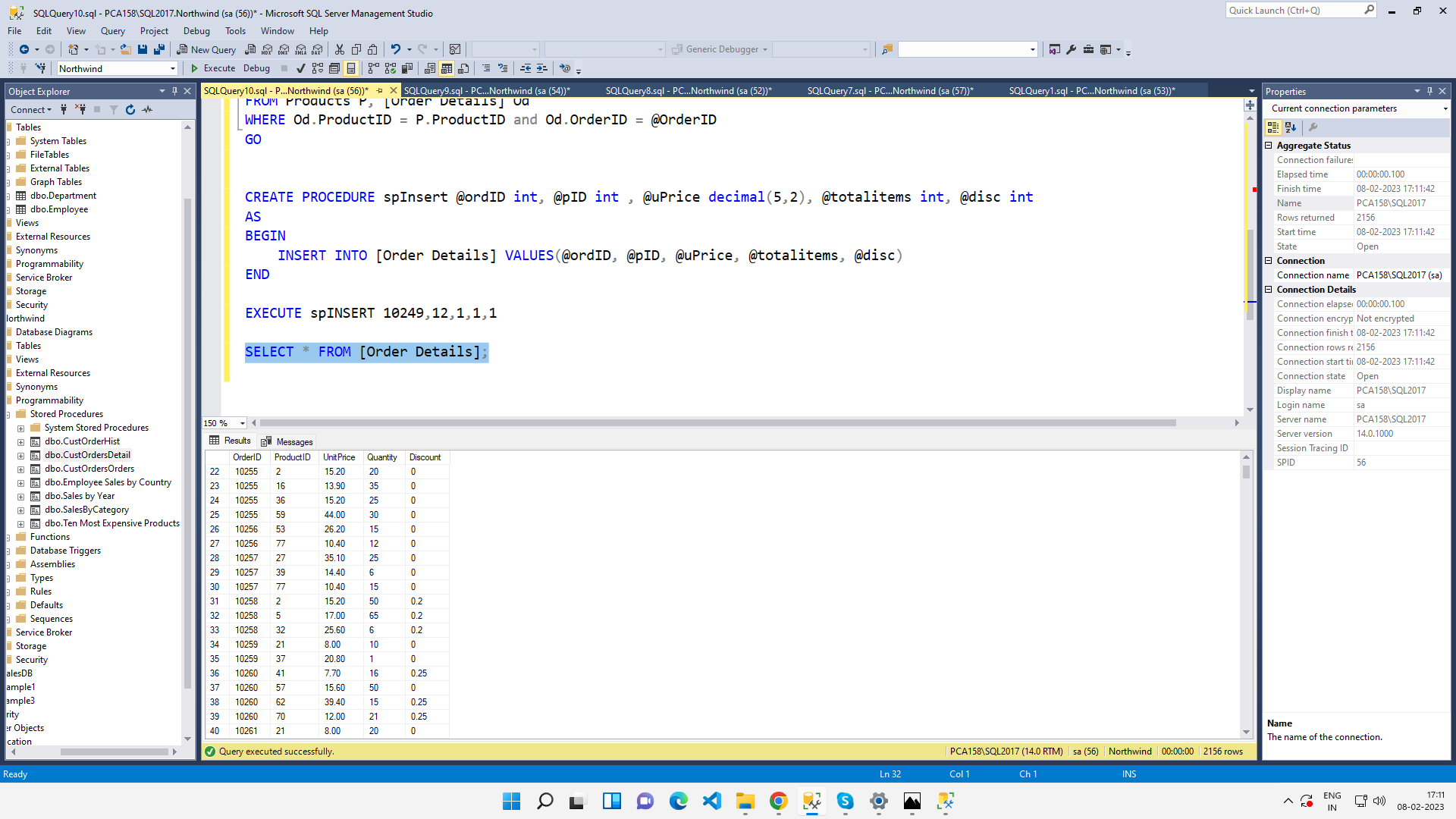
BEGIN

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

END

EXECUTE spINSERT 10249,12,1,1,1

SELECT \* FROM [Order Details];



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

Customer Order Details

ANS.

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

EXECUTE spUpdate 10248,1,10,10,0.85

SELECT \* FROM [Order Details];

