**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.**

alter procedure spavgfreight

@customerID varchar(10), @avgval float output

AS

BEGIN

select @avgval=AVG(Freight)

FROM orders

WHERE CustomerID = @customerID

return @avgval

END

alter trigger [dbo].[AverageFreightValidation]

ON [dbo].[orders]

after INSERT, UPDATE

AS

BEGIN

set nocount on

DECLARE @avgval float,

@customerID varchar(10),

@Freight real

select @customerID = CustomerID,

@Freight= Freight

from inserted

EXEC spavgfreight @customerID , @avgval OUTPUT

IF (@Freight > @avgval)

BEGIN

RAISERROR('Freight exceeds the average freight', 16, 1)

ROLLBACK TRANSACTION

END

END

SET IDENTITY\_INSERT orders ON

Go

INSERT INTO orders(

OrderID,

CustomerID,

EmployeeID ,

OrderDate,

RequiredDate ,

"ShippedDate",

"ShipVia" ,

"Freight" ,

"ShipName" ,

"ShipAddress" ,

"ShipCity",

"ShipRegion" ,

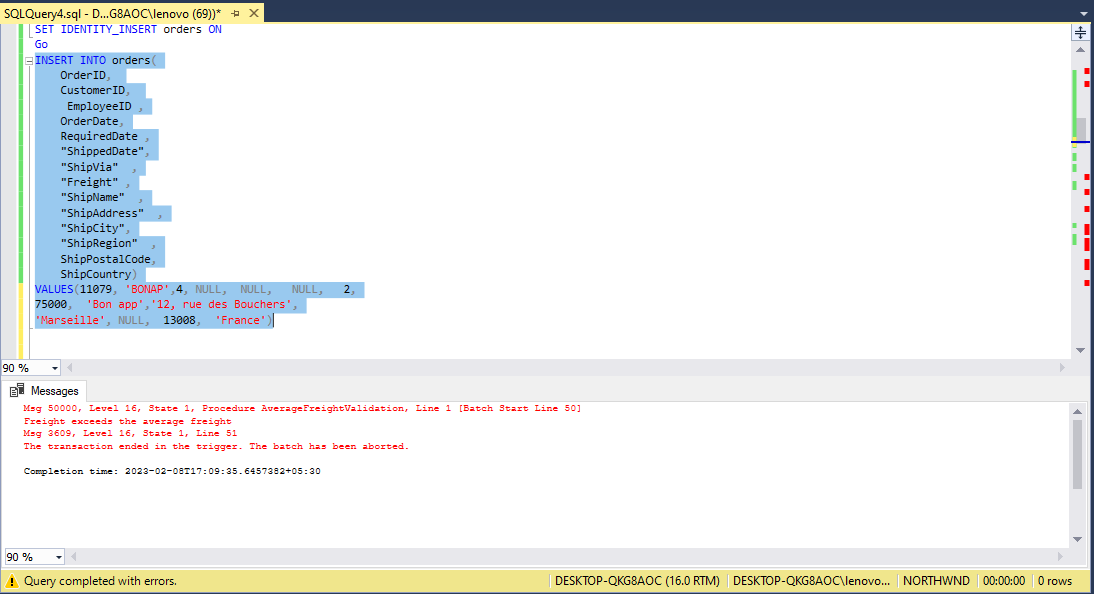
ShipPostalCode,

ShipCountry)

VALUES(11079, 'BONAP',4, NULL, NULL, NULL, 2,

75000, 'Bon app','12, rue des Bouchers',

'Marseille', NULL, 13008, 'France')



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

**Employee Sales by Country**

alter procedure spGetEmployeeSalesByCountry

@country nvarchar(50)

as

begin

select o.EmployeeID, sum(d.Quantity) as salesum

from Orders o

right join [Order Details] d

on o.OrderID = d.OrderID

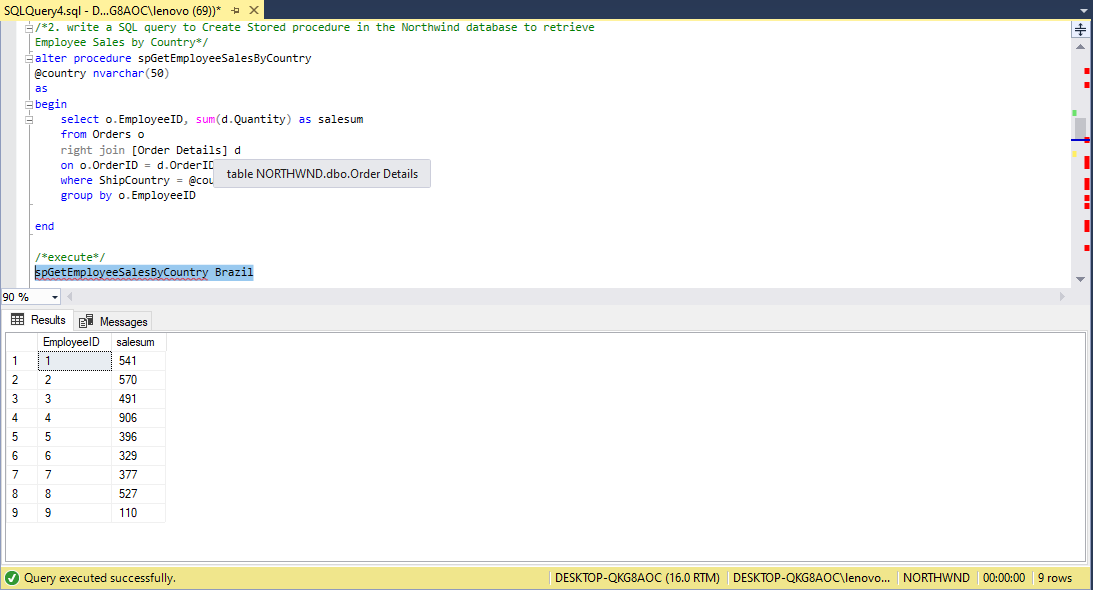
where ShipCountry = @country

group by o.EmployeeID

end

/\*execute\*/

spGetEmployeeSalesByCountry Brazil



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

**Sales by Year**

alter procedure spGetSalesByYear

@year int

as

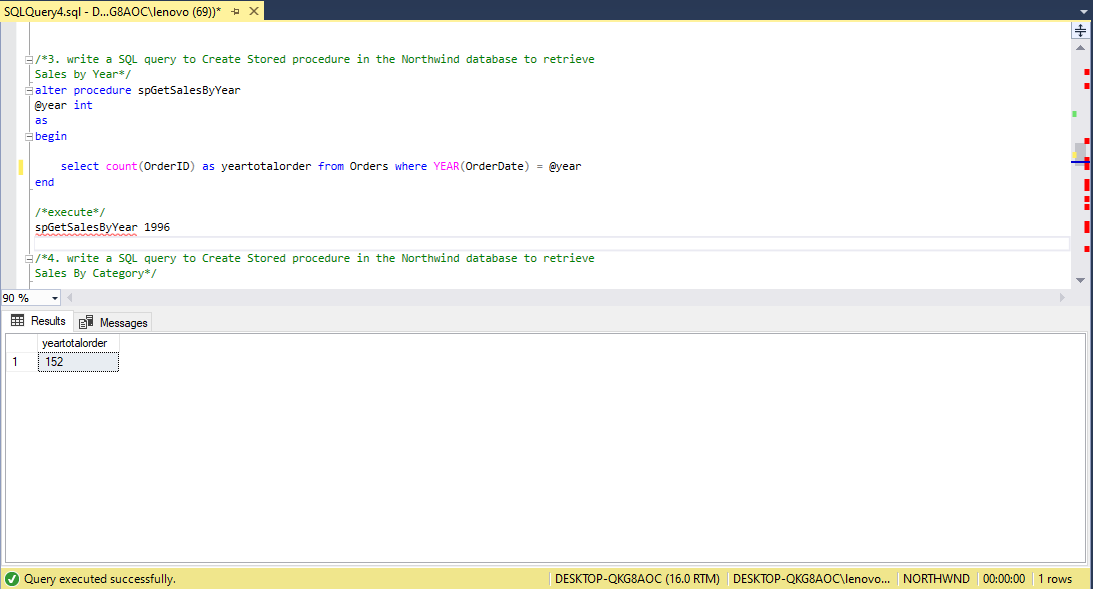
begin

select count(OrderID) as yeartotalorder from Orders where YEAR(OrderDate) = @year

end

/\*execute\*/

spGetSalesByYear 1996



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

**Sales By Category**

alter procedure spGetSalesByCategory

@category nvarchar(25)

as

begin

select count(OrderID) as sales from Orders where OrderID in

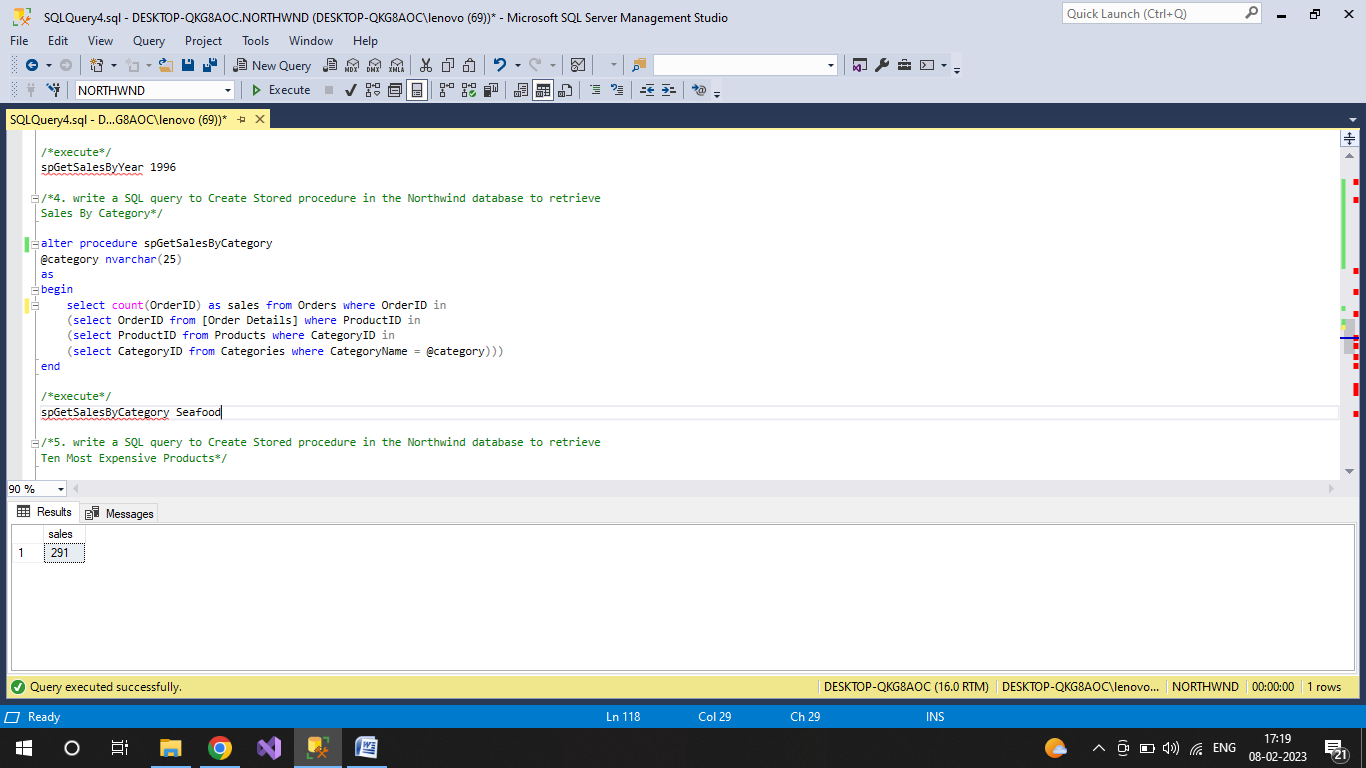
(select OrderID from [Order Details] where ProductID in

(select ProductID from Products where CategoryID in

(select CategoryID from Categories where CategoryName = @category)))

end

/\*execute\*/

spGetSalesByCategory Seafood

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

**Ten Most Expensive Products**

create procedure spGetTenMostExpensiveProducts

as

begin

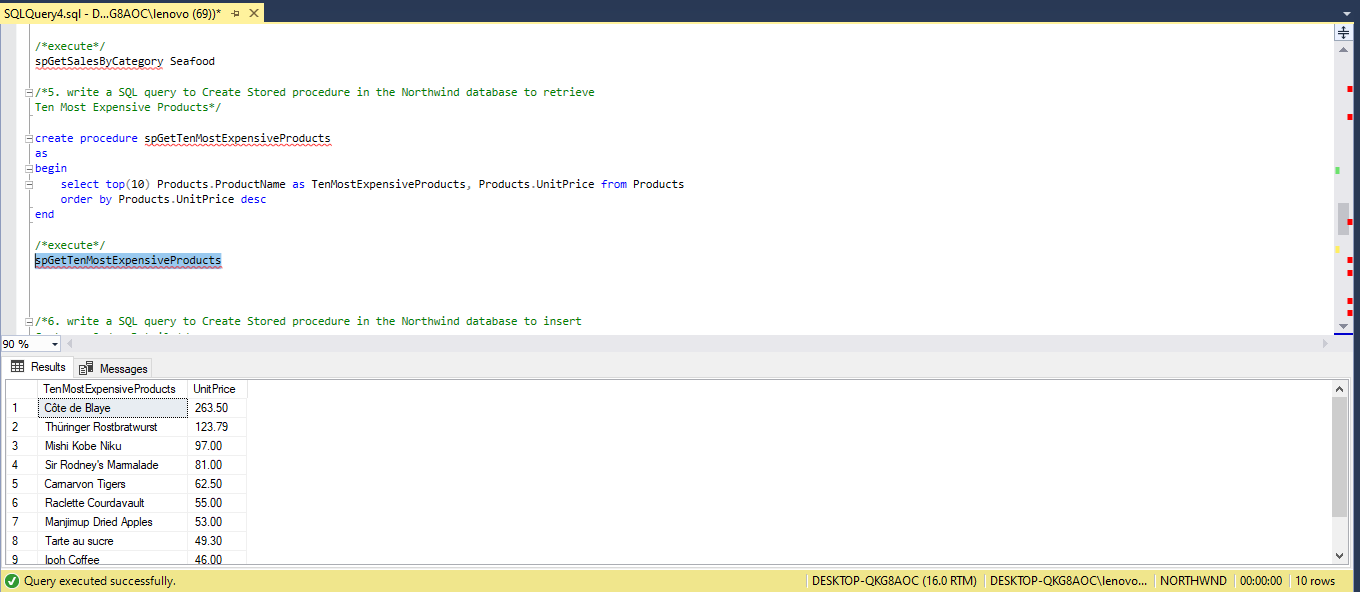
select top(10) Products.ProductName as TenMostExpensiveProducts, Products.UnitPrice from Products

order by Products.UnitPrice desc

end

/\*execute\*/

spGetTenMostExpensiveProducts



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

**Customer Order Details**

create procedure spGetInsertCustomerOrderDetails

@orderid int,

@productid int,

@unitprice money,

@quantity smallint,

@discount real

as

begin

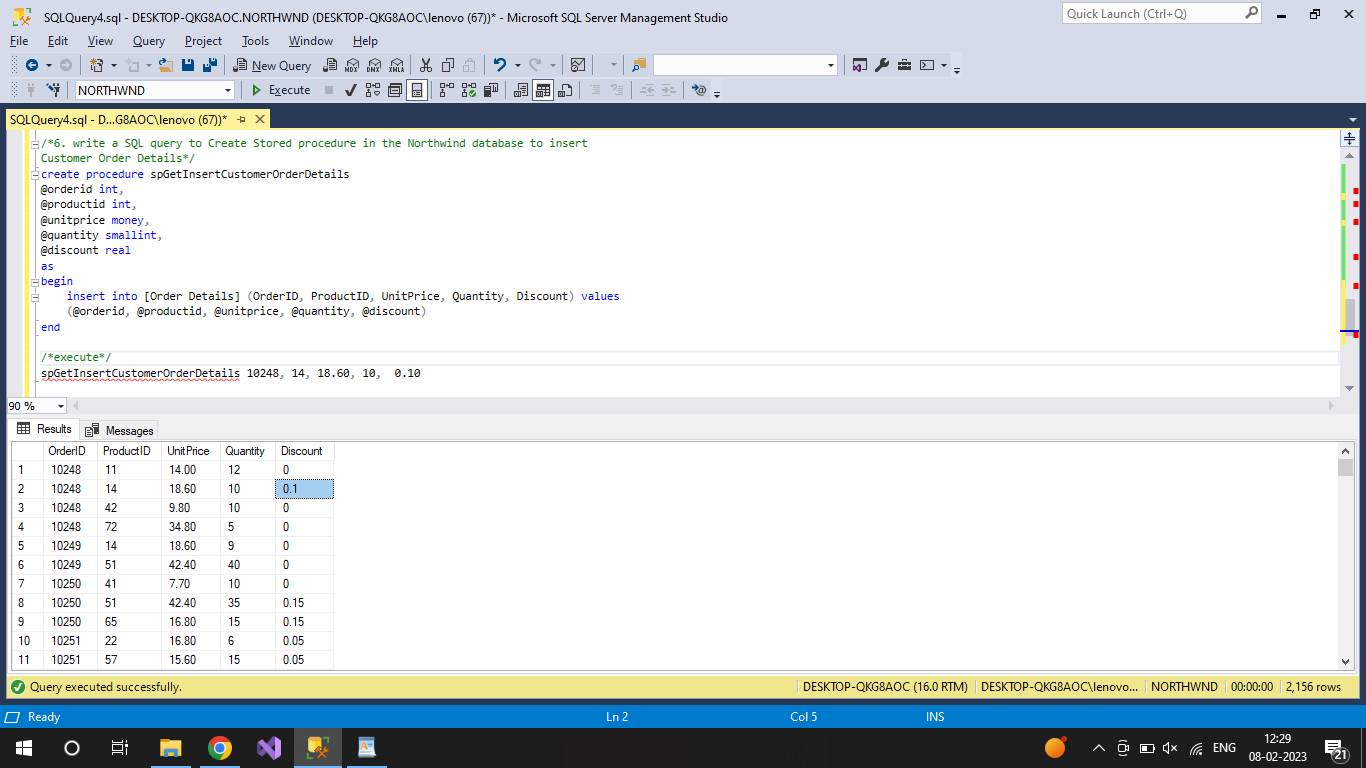
insert into [Order Details] (OrderID, ProductID, UnitPrice, Quantity, Discount) values

(@orderid, @productid, @unitprice, @quantity, @discount)

end

/\*execute\*/

spGetInsertCustomerOrderDetails 10248, 14, 18.60, 10, 0.10



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

**Customer Order Details**

create procedure spGetUpdateCustomerOrderDetails

@orderid int,

@productid int,

@unitprice money,

@quantity smallint,

@discount real

as

begin

update [Order Details]

set UnitPrice=@unitprice,Quantity=@quantity,Discount=@discount

WHERE OrderID=@orderid and ProductID = @productid

end

/\*execute\*/

spGetUpdateCustomerOrderDetails 10248, 14, 16.80, 8, 0.15

