**SQL ASSIGNMENT-4**

**SQL-Queries**

**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.**

Answer: -

CREATE PROCEDURE spCalAvgFreight

@CustomerID NVARCHAR(5),

@AverageFreight MONEY OUTPUT

AS

BEGIN

SELECT @AverageFreight = AVG(Freight)

FROM Orders

WHERE CustomerID = @CustomerID

END

GO

[dbo].[spEmployeeSalesByCategory]

-- Create Trigger for Verifing Freight before Insert

CREATE TRIGGER tr\_VerifyFreightForInsert

ON Orders

INSTEAD OF INSERT

AS

BEGIN

DECLARE @AvgFreightOfOrders MONEY

DECLARE @CustID NVARCHAR(5)

DECLARE @Freight MONEY

SELECT @CustId=CustomerID FROM inserted

SELECT @Freight=Freight FROM inserted

-- execute stored procedure

EXECUTE spCalAvgFreight @CustID,

@AverageFreight = @AvgFreightOfOrders OUTPUT

-- check the freight

IF @AvgFreightOfOrders IS NOT NULL

AND @AvgFreightOfOrders < @Freight

BEGIN

Raiserror('Invalid data as Freight value exceeds the average freight value',16,1)

RETURN

END

END

INSERT INTO Orders VALUES('VINET',null,null,null,null,null,23,null,null,null,null,null,null)

-- Create Trigger for Verifing Freight before Update

CREATE TRIGGER tr\_VerifyFreightForUpdate

ON Orders

INSTEAD OF UPDATE

AS

BEGIN

DECLARE @AvgFreightOfOrders MONEY

DECLARE @CustID NVARCHAR(5)

DECLARE @Freight MONEY

SELECT @CustId=CustomerID FROM inserted

SELECT @Freight=Freight FROM inserted

-- execute stored procedure

EXECUTE spCalAvgFreight @CustID,

@AverageFreight = @AvgFreightOfOrders OUTPUT

-- check the freight

IF @AvgFreightOfOrders IS NOT NULL

AND @AvgFreightOfOrders < @Freight

BEGIN

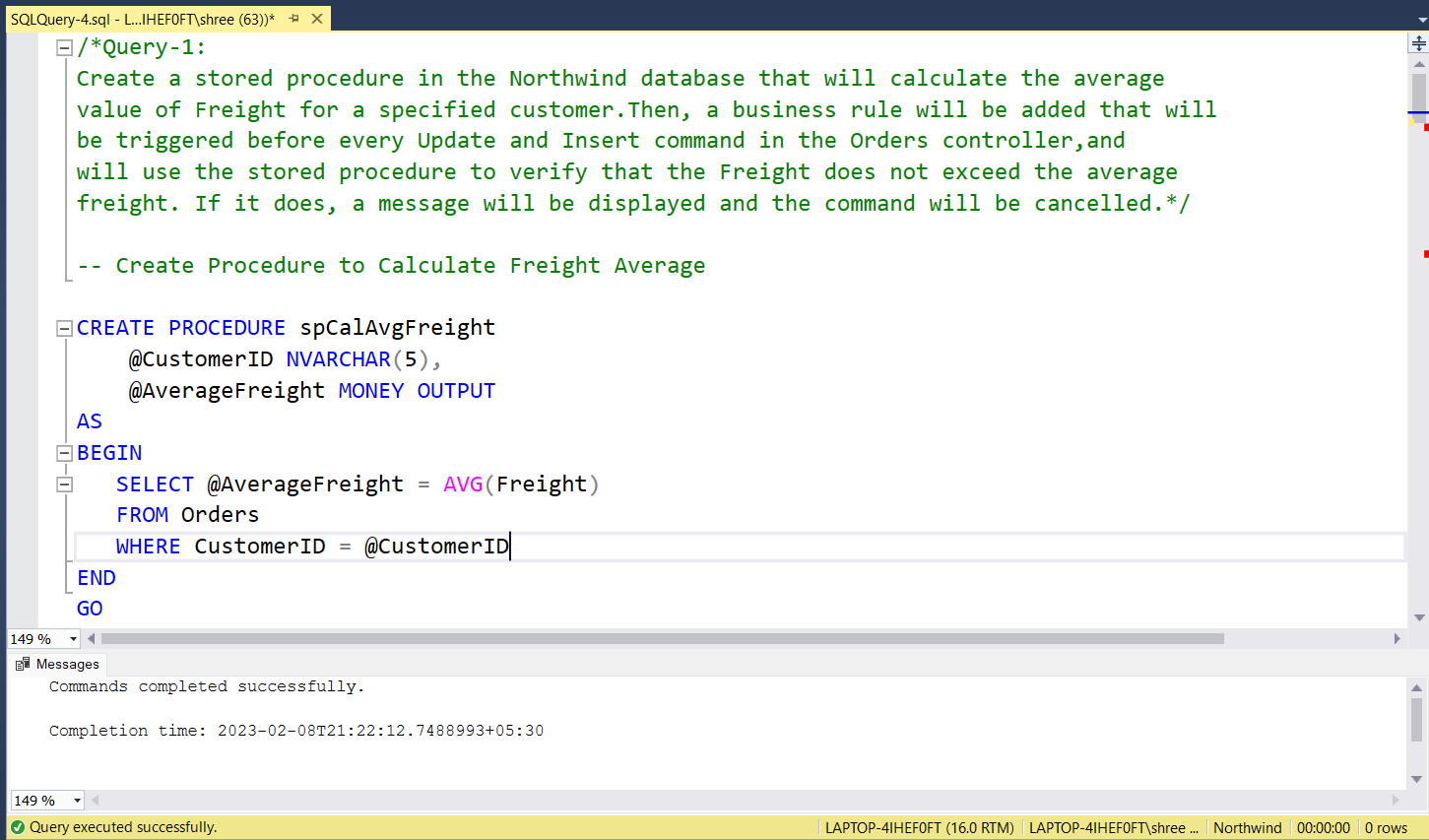
Raiserror('Invalid data as Freight value exceeds the average freight value',16,1)

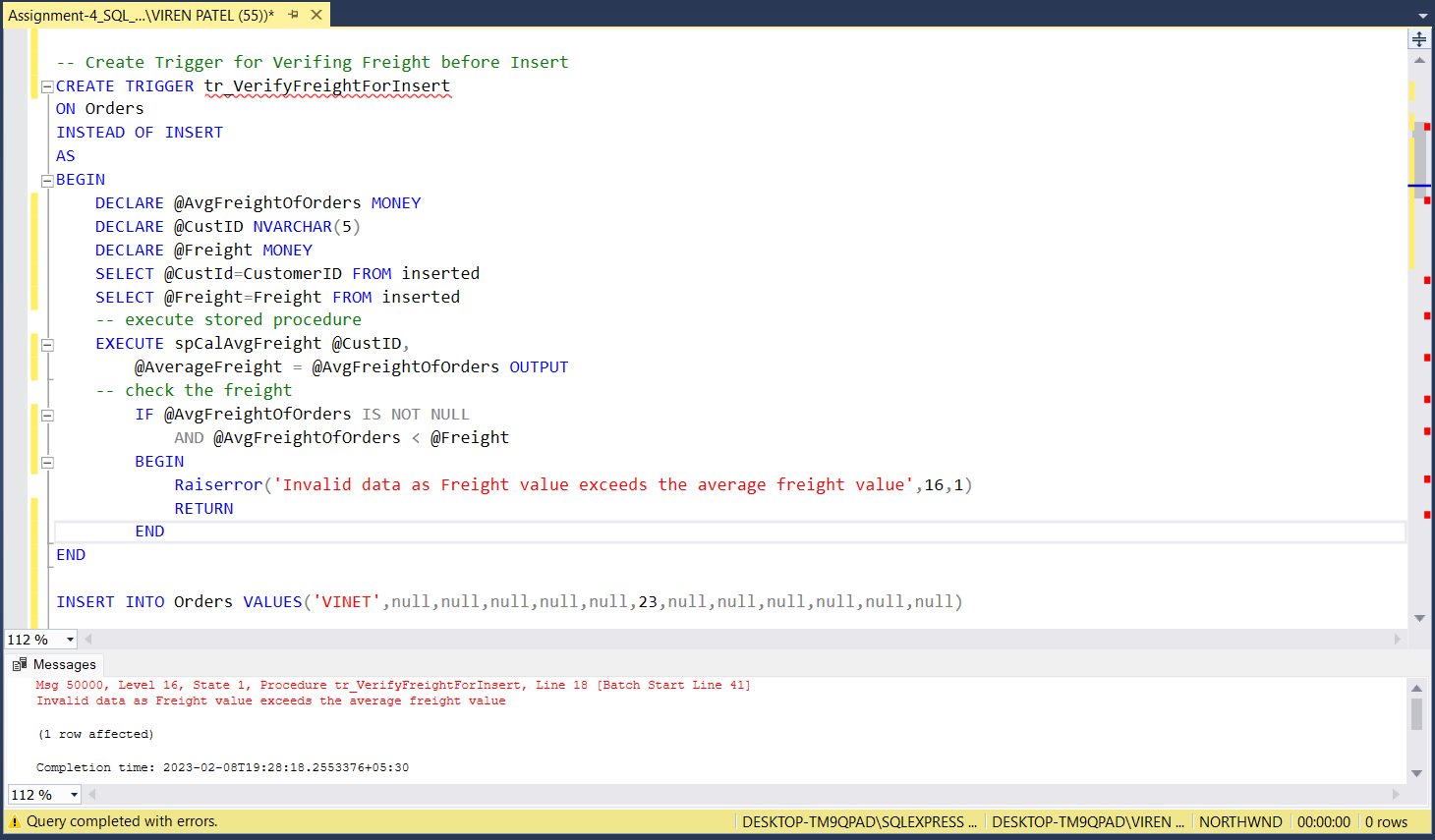
RETURN

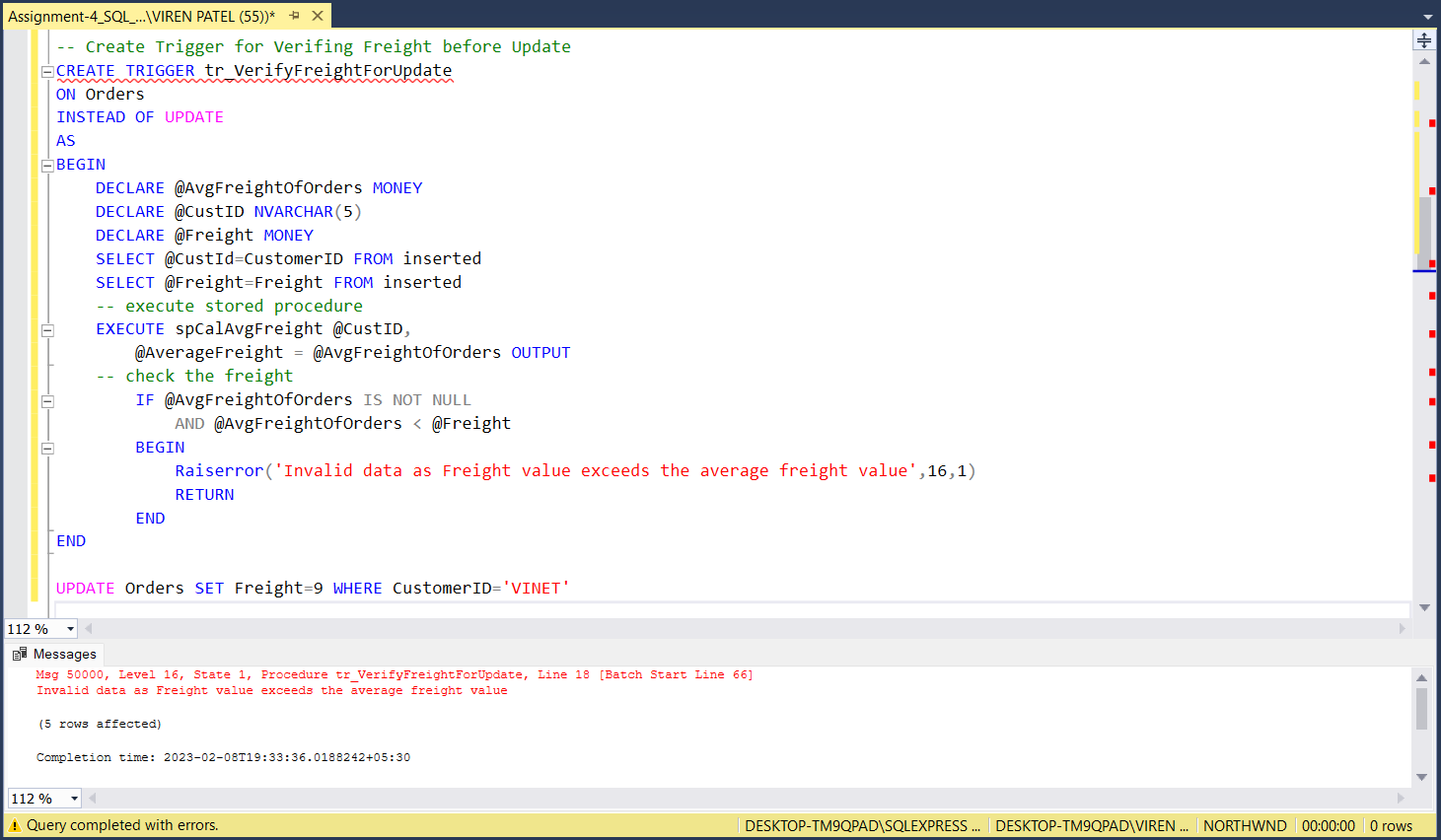
END

END

UPDATE Orders SET Freight=9 WHERE CustomerID='VINET'







**2) Write a SQL query to Create Stored procedure in the Northwind database to retrieve Employee Sales by Country.**

Answer: -

CREATE PROCEDURE spEmployeeSalesByCountry

@Country NVARCHAR(30)

AS

BEGIN

SELECT

Orders.EmployeeID,

Employees.FirstName AS 'Employee',

Orders.ShipCountry AS 'Country',

COUNT(Orders.EmployeeID) AS [Total Sales]

FROM Orders

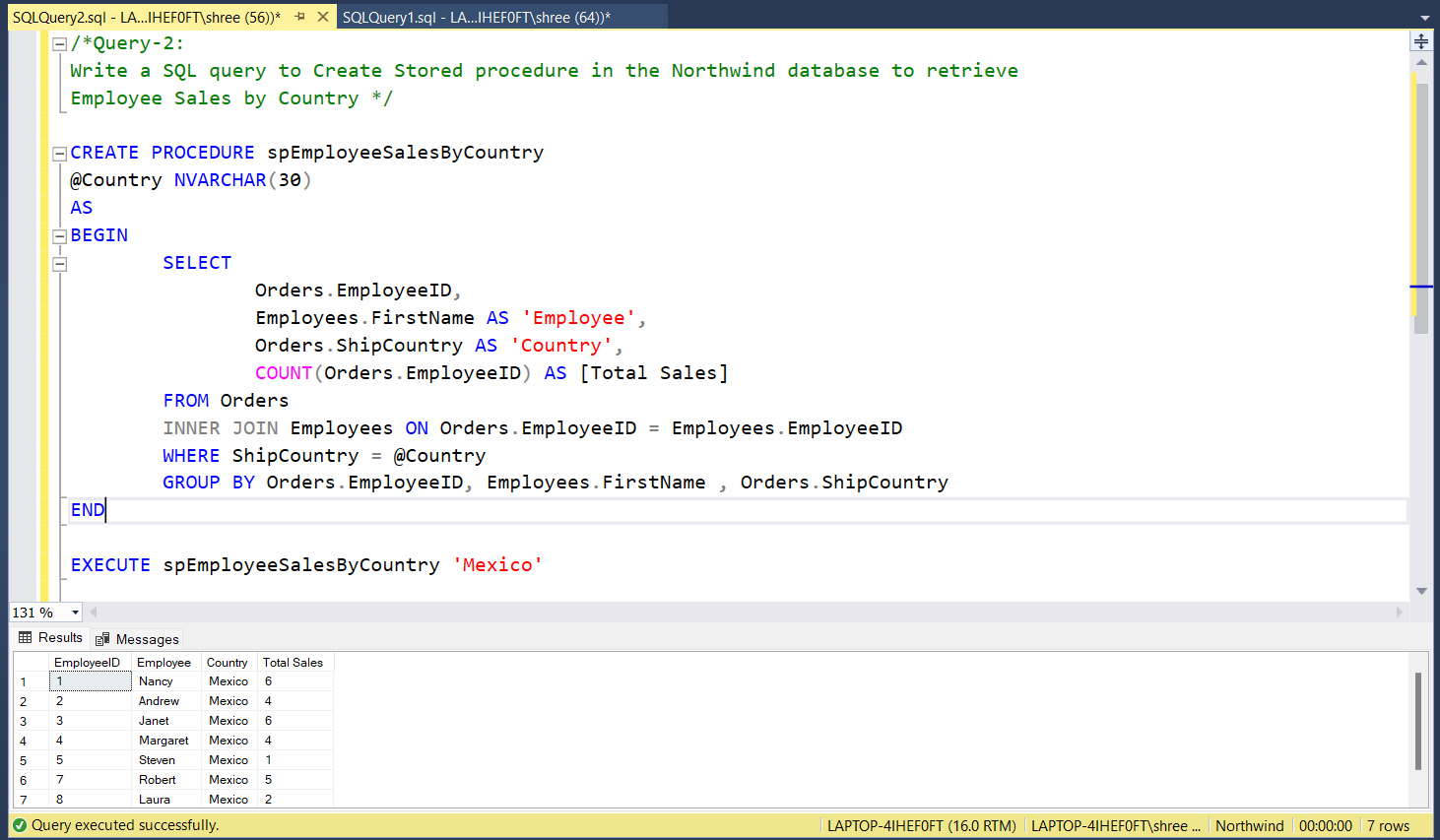
INNER JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID

WHERE ShipCountry = @Country

GROUP BY Orders.EmployeeID, Employees.FirstName , Orders.ShipCountry

END

EXECUTE spEmployeeSalesByCountry 'Mexico'



**3) Write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales by Year.**

Answer: -

CREATE PROCEDURE spEmployeeSalesByYear

@Year INT

AS

BEGIN

SELECT

YEAR(Orders.OrderDate) AS [Sales Year],

COUNT(YEAR(Orders.OrderDate)) AS [Total Sales]

FROM Orders

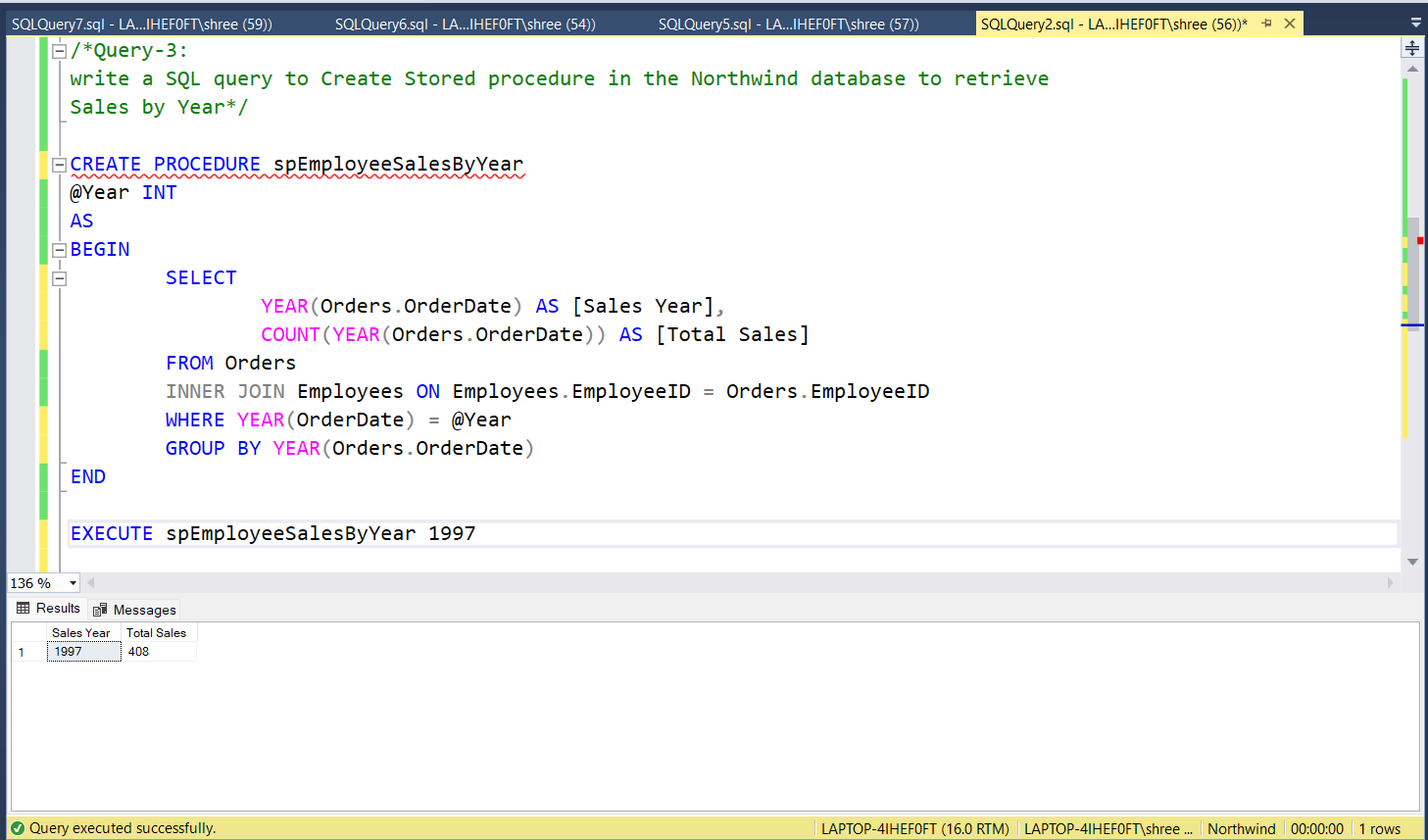
INNER JOIN Employees ON Employees.EmployeeID = Orders.EmployeeID

WHERE YEAR(OrderDate) = @Year

GROUP BY YEAR(Orders.OrderDate)

END

EXECUTE spEmployeeSalesByYear 1997



**4) Write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales By Category.**

Answer: -

CREATE PROCEDURE spEmployeeSalesByCategory

@Category VARCHAR(30)

AS

BEGIN

SELECT

Categories.CategoryName,

COUNT(Categories.CategoryName) AS [Total Sales]

FROM Products

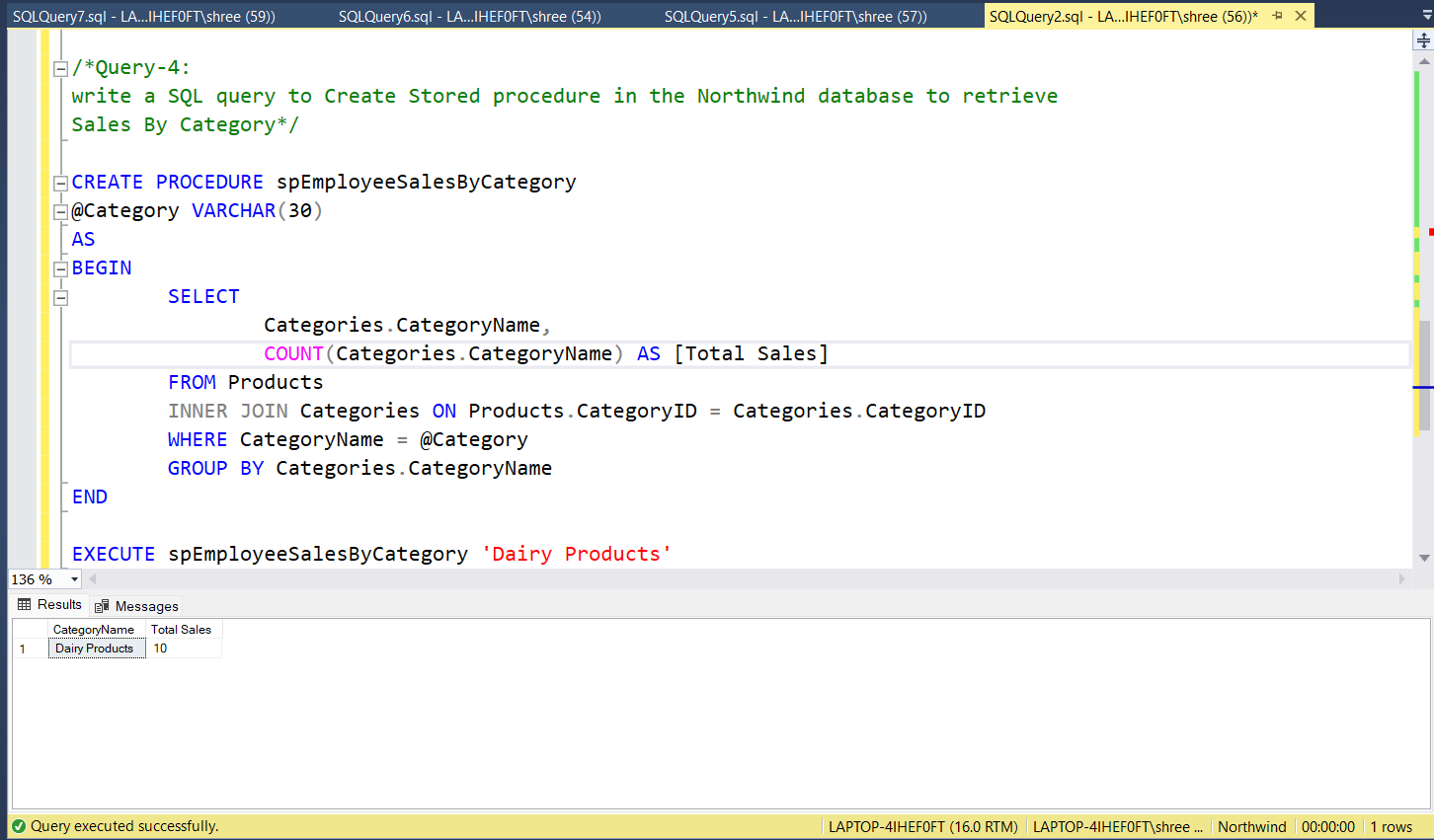
INNER JOIN Categories ON Products.CategoryID = Categories.CategoryID

WHERE CategoryName = @Category

GROUP BY Categories.CategoryName

END

EXECUTE spEmployeeSalesByCategory 'Dairy Products'



**5) write a SQL query to Create Stored procedure in the Northwind database to retrieve Ten Most Expensive Products.**

Answer: -

CREATE PROCEDURE spTenMostExpensiveProducts

AS

BEGIN

SELECT

TOP 10

ProductID,

ProductName,

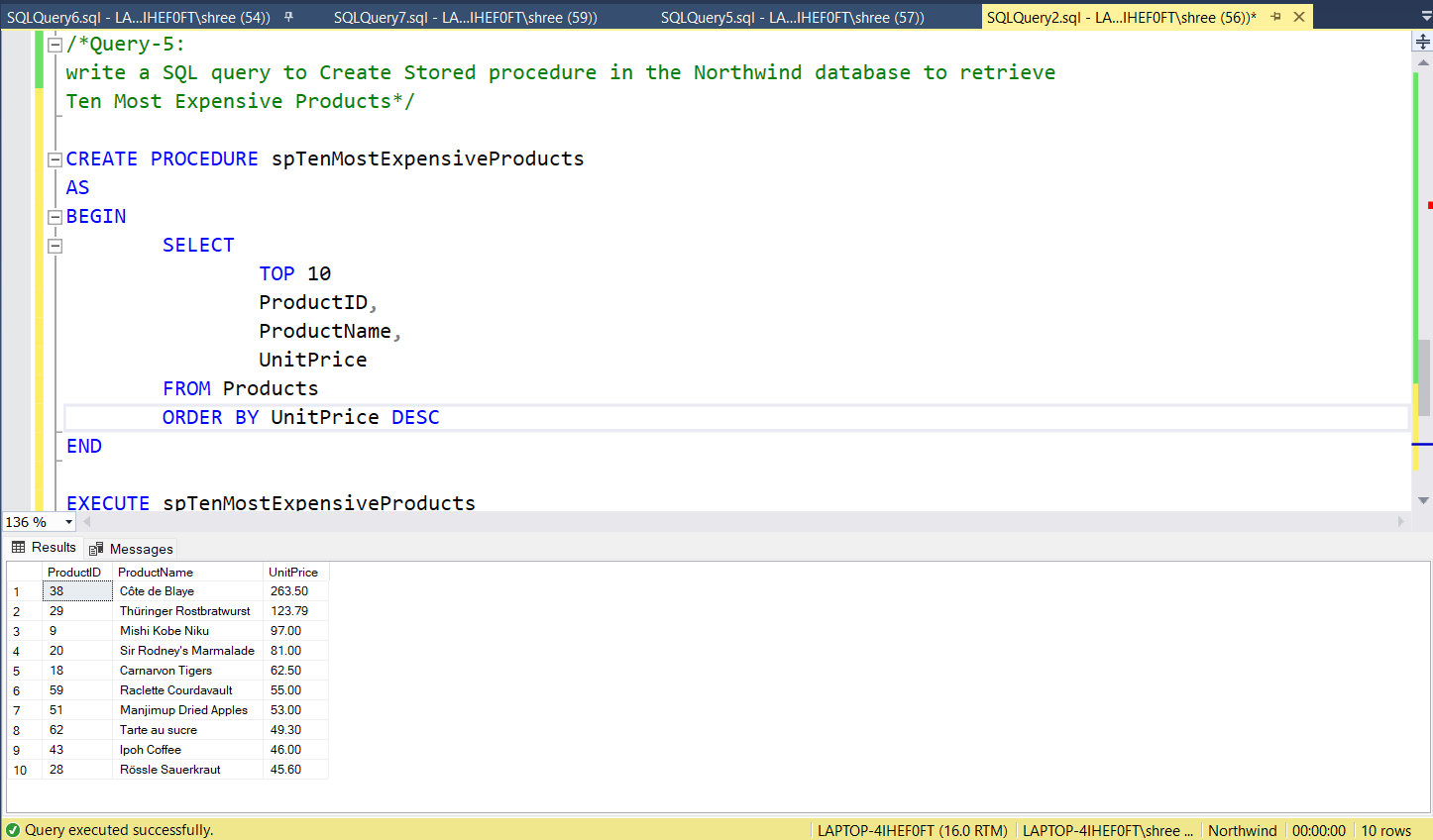
UnitPrice

FROM Products

ORDER BY UnitPrice DESC

END

EXECUTE spTenMostExpensiveProducts



**6)** **Write a SQL query to Create Stored procedure in the Northwind database to insert Customer Order Details.**

Answer: -

CREATE PROCEDURE spInsertCustOrderDetail

@OrderID INT,

@ProductID INT,

@UnitPrice Decimal(4,2),

@Quantity INT,

@Discount FLOAT

AS

BEGIN

INSERT INTO [Order Details]

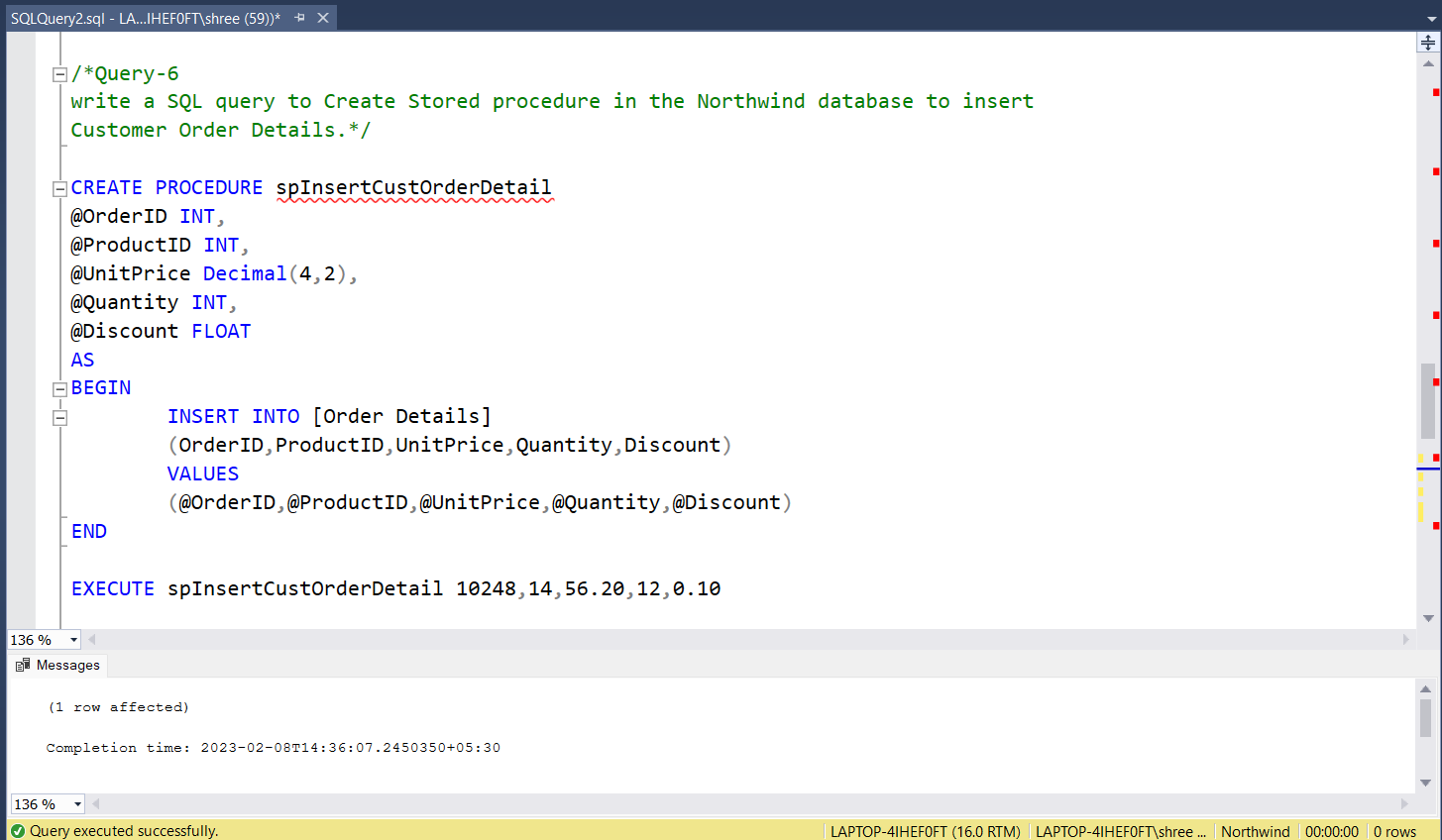
(OrderID,ProductID,UnitPrice,Quantity,Discount)

VALUES

(@OrderID,@ProductID,@UnitPrice,@Quantity,@Discount)

END

EXECUTE spInsertCustOrderDetail 10248,14,56.20,12,0.10



**7)** **Write a SQL query to Create Stored procedure in the Northwind database to update Customer Order Details.**

Answer: -

CREATE PROCEDURE spUpdateCustOrderDetail

@OrderID INT,

@ProductID INT,

@UnitPrice Decimal(4,2),

@Quantity INT,

@Discount FLOAT

AS

BEGIN

UPDATE [Order Details]

SET

UnitPrice = @UnitPrice,

Quantity = @Quantity,

Discount = @Discount

WHERE OrderID = @OrderID and ProductID = @ProductID

END

EXECUTE spUpdateCustOrderDetail 10248,14,56.20,12,0.20

