**Name: Nandani Patel**

**Department: Dotnet**

**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.**
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]

@Beginning\_Date DateTime, @Ending\_Date DateTime, @Country varchar(50)AS

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

FROM Employees INNER JOIN

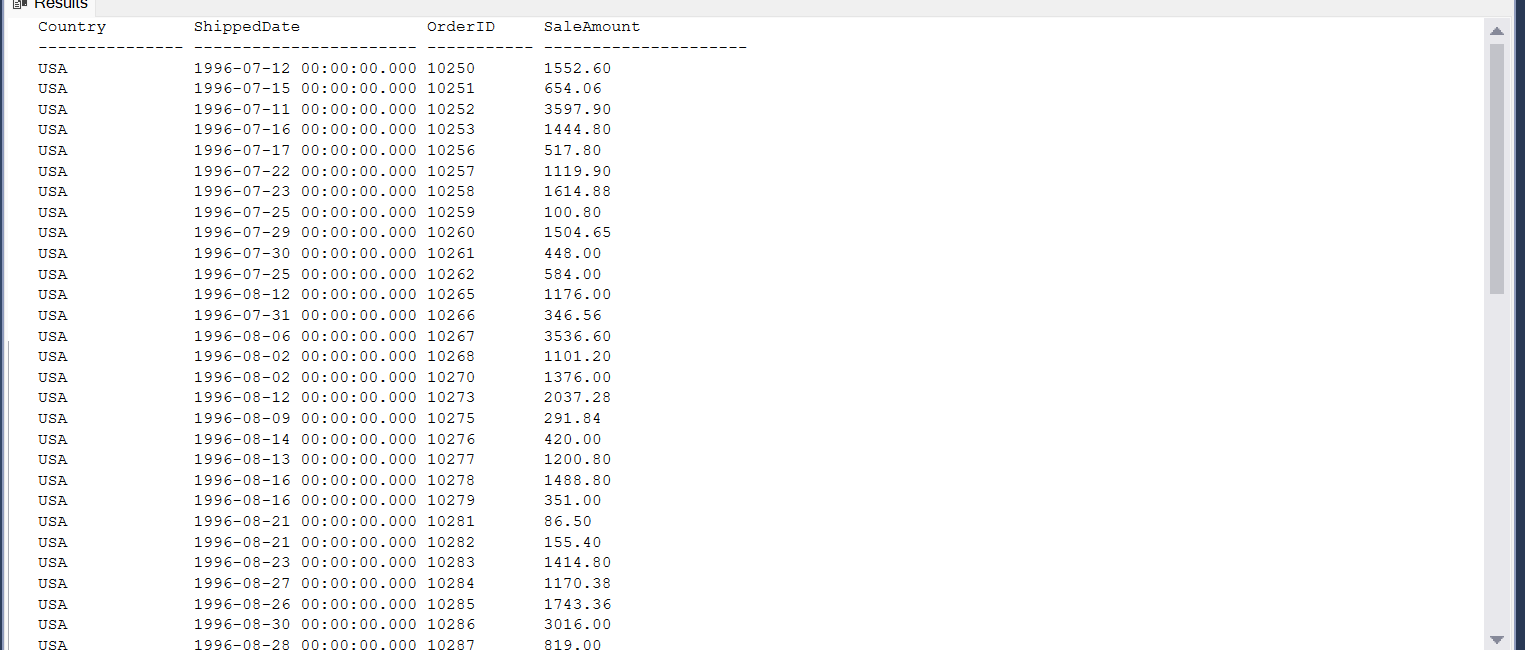
(Orders INNER JOIN "Order Subtotals" ON Orders.OrderID = "Order Subtotals".OrderID)

ON Employees.EmployeeID = Orders.EmployeeID

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

AND Employees.Country=@Country;

EXEC [Employee Sales by Country] @Beginning\_Date='1996-03-05 00:00:00', @Ending\_Date='1996-09-04 00:00:00' ,@Country='USA



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

CREATE PROCEDURE SalesByYear

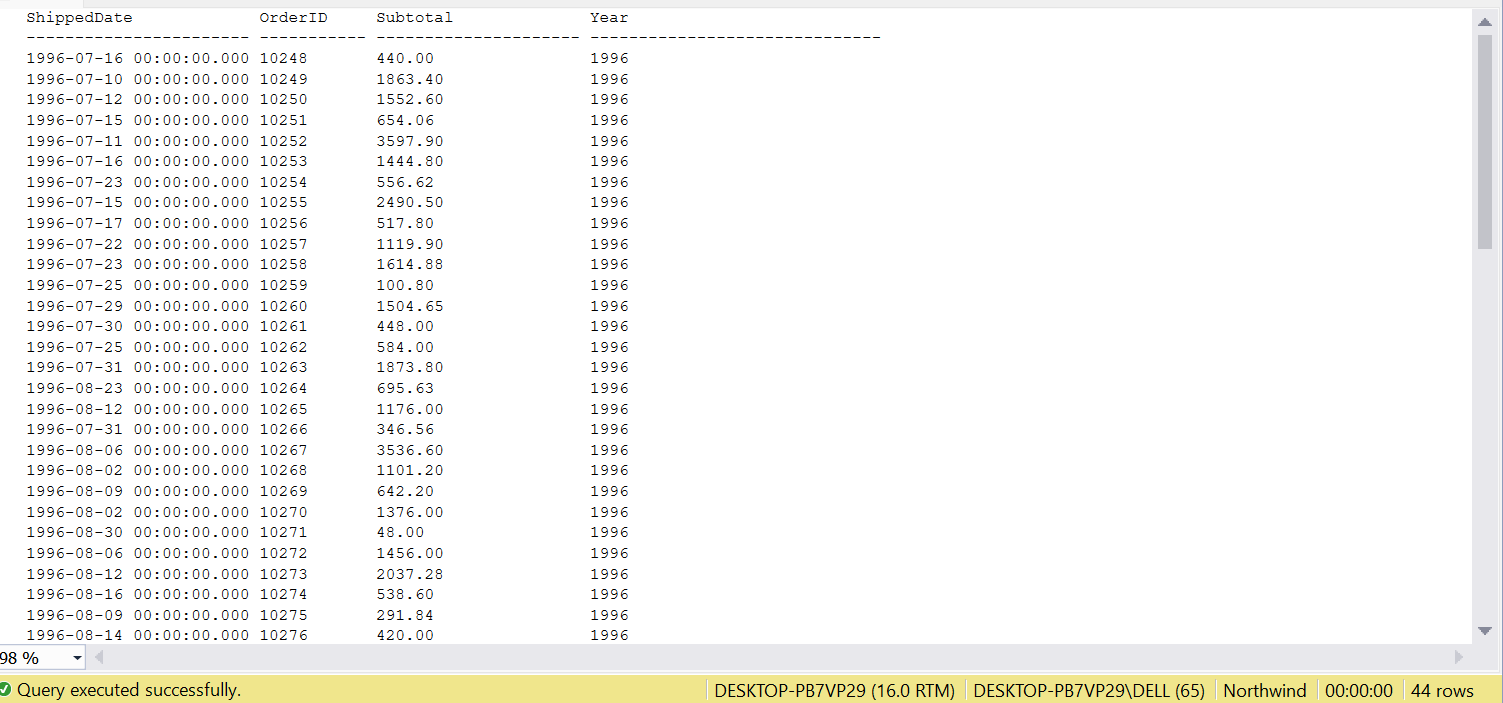
@Beginning\_Date DateTime, @Ending\_Date DateTime AS

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;

EXEC SalesByYear @Beginning\_Date='1995-03-05 00:00:00', @Ending\_Date='1996-09-04 00:00:00';

****

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

CREATE PROCEDURE SalesByCategory

@CategoryName nvarchar(15)

AS

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

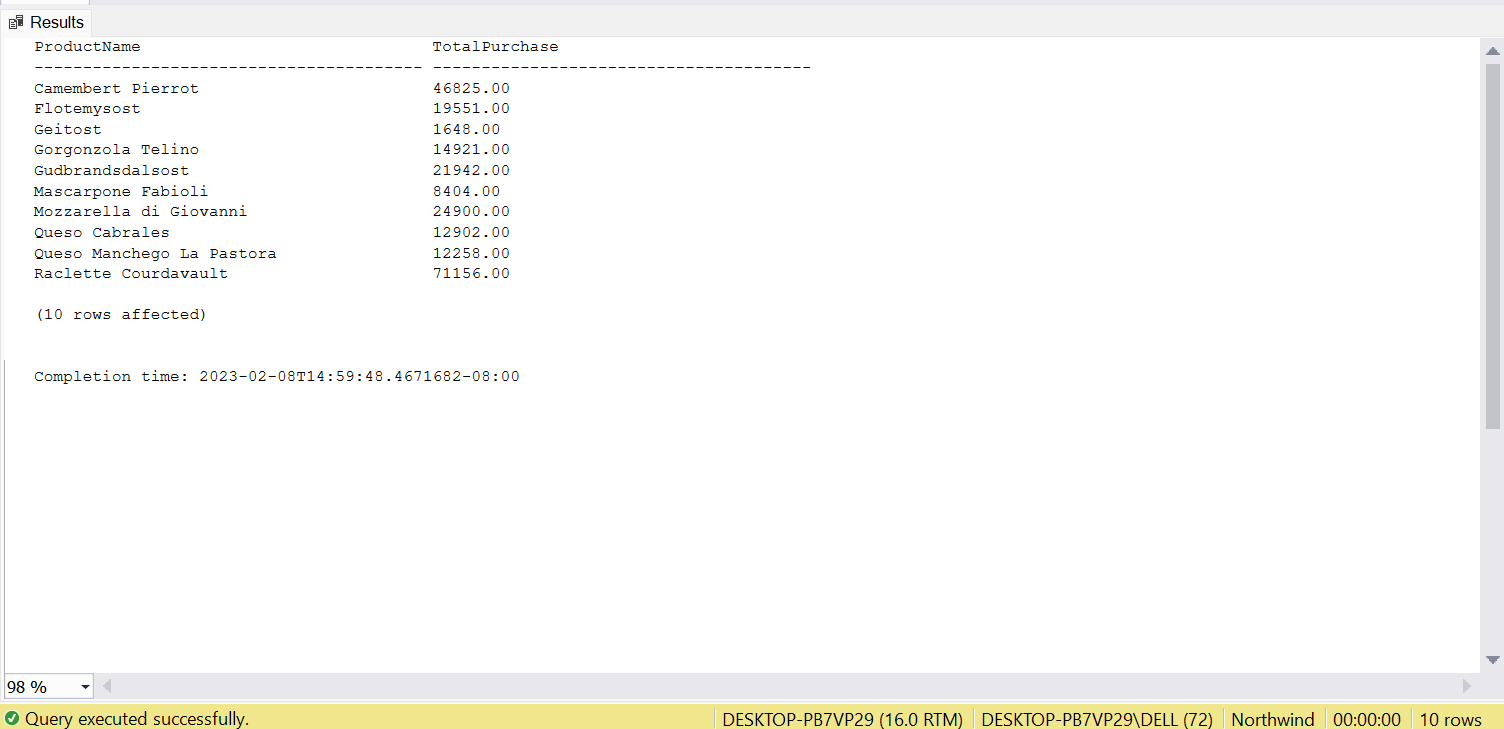
AND OD.ProductID = P.ProductID

AND P.CategoryID = C.CategoryID

AND C.CategoryName = @CategoryName

GROUP BY ProductName;

EXEC SalesByCategory @CategoryName='Dairy Products';

****

1. **Write a SQL query to Create Stored procedure in the Northwind database to retrieve Ten Most Expensive Products.**

CREATE PROCEDURE TenMostExpensiveProducts AS

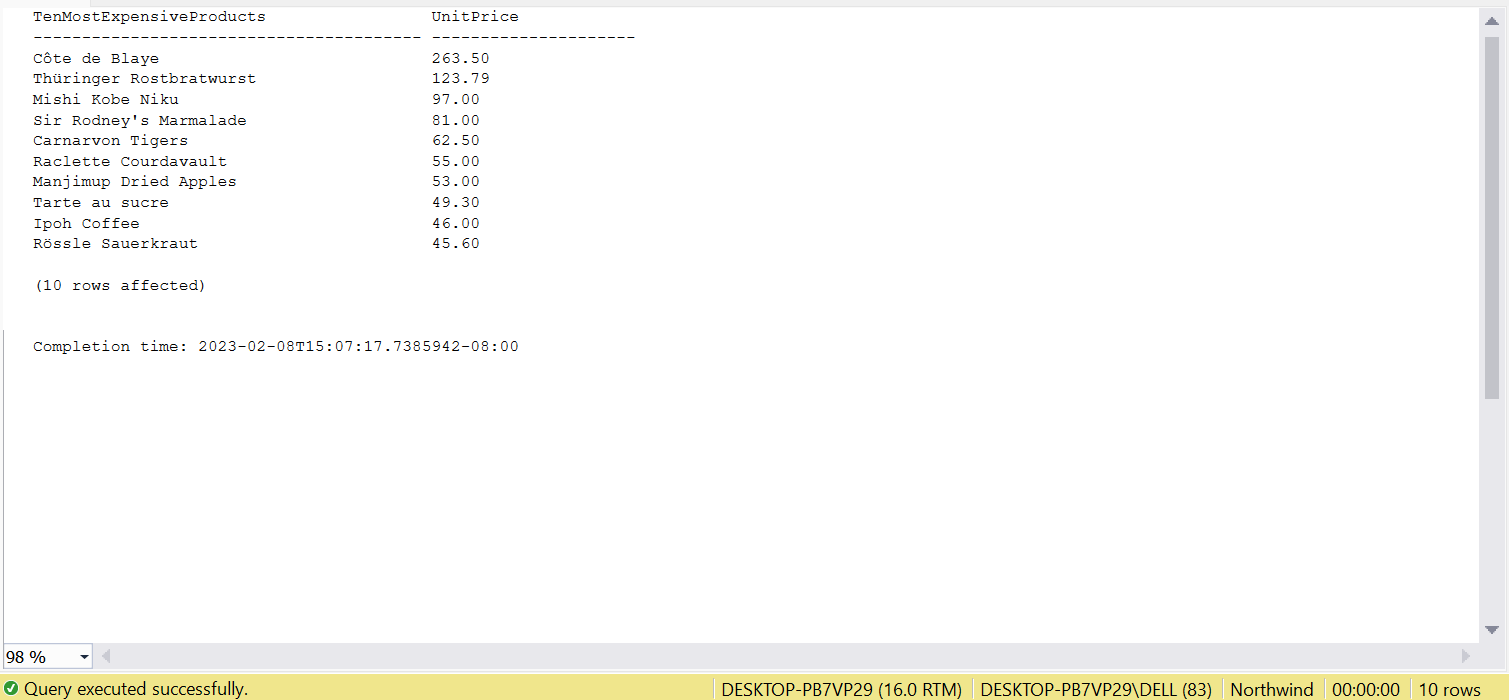
SET ROWCOUNT 10

SELECT Products.ProductName AS TenMostExpensiveProducts, Products.UnitPrice

FROM Products

ORDER BY Products.UnitPrice DESC;

EXEC TenMostExpensiveProducts;

****

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

ALTER PROCEDURE [dbo].[InsertData]

@OrderID int ,

@ProductID int,

@UnitPrice Money,

@Quantity int,

@Discount Real

AS

BEGIN

INSERT INTO [Order Details](OrderID,ProductID,UnitPrice,Quantity,Discount)

VALUES (@OrderID,@ProductID,ROUND(@UnitPrice,2),@Quantity,CONVERT(INT,@Discount\*100))

END

EXEC [dbo].[InsertData]

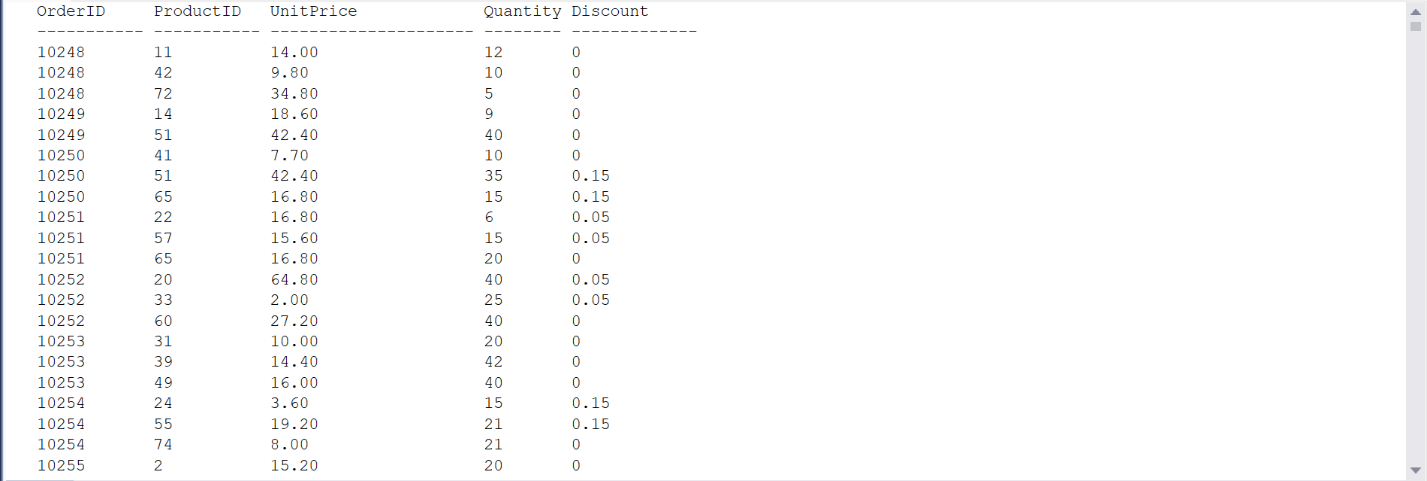
@OrderID=10448,

@ProductID=10,

@UnitPrice=25.00,

@Quantity=8,

@Discount=0

****

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

CREATE PROCEDURE [dbo].[UpdateData]

@OrderID int ,

@ProductID int,

@UnitPrice Money,

@Quantity int,

@Discount Real

AS

BEGIN

UPDATE[Order Details]

SET

UnitPrice=ROUND(@UnitPrice,2),

Quantity=@Quantity,

Discount=CONVERT(int,@Discount\*100)

WHERE OrderID=@OrderID AND ProductID=@ProductID

END

EXEC [dbo].[UpdateData]

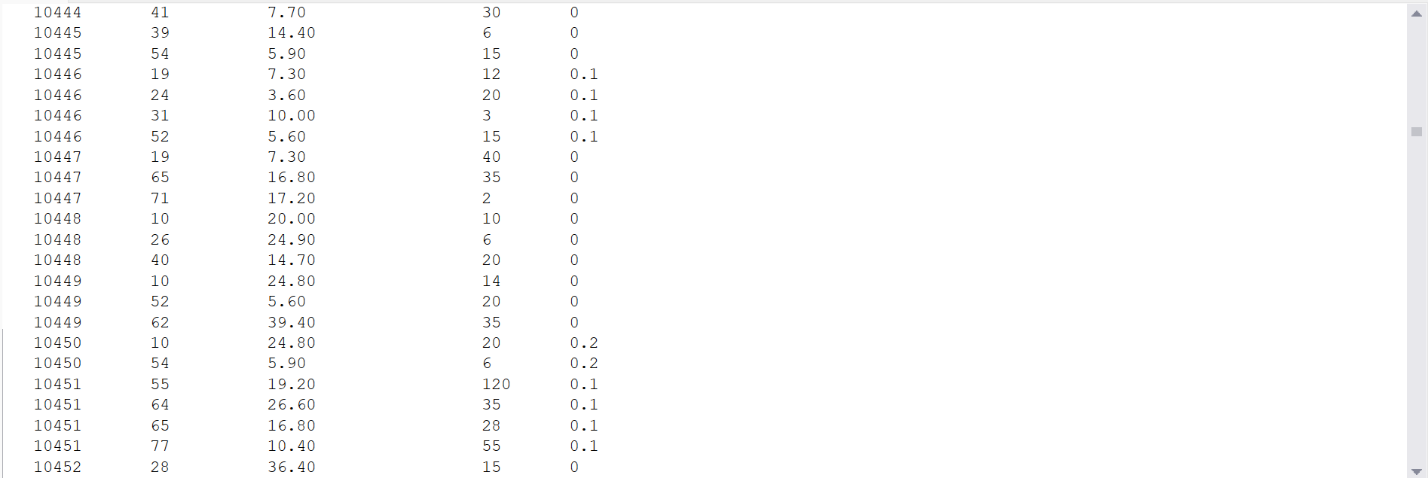
@OrderID=10448,

@ProductID=10,

@UnitPrice=20.00,

@Quantity=10,

@Discount=0

****