**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.**

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

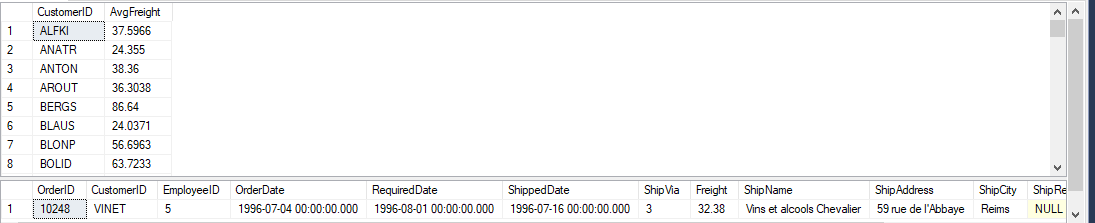
From Inserted

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].[spEmployeeSalesby\_Country]

@countryname varchar(10)

AS

Begin

SELECT Employees.Country, Employees.LastName, Employees.FirstName,

Orders.ShippedDate, Orders.OrderID, [Order Details].ProductID,

[Order Details].UnitPrice, [Order Details].Discount, [Order Details].Quantity

FROM Employees

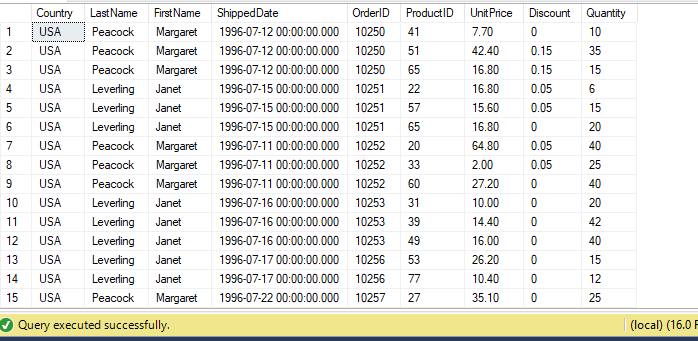
INNER JOIN (Orders INNER JOIN "Order Details" ON "Order Details".OrderID = Orders.OrderID)

ON Employees.EmployeeID = Orders.EmployeeID

WHERE Employees.Country = @countryname

End

[spEmployeeSalesby\_Country] 'USA'



**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]

@year int

AS

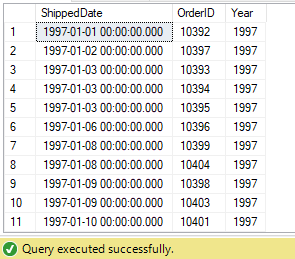
SELECT Orders.ShippedDate, Orders.OrderID , DATENAME(yy,ShippedDate) AS Year

FROM Orders

WHERE Year(ShippedDate) = @year

GO

[Sales by Year] 1996



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

Ans.

create PROCEDURE [dbo].[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

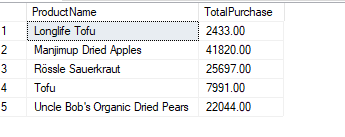
AND C.CategoryName = @CategoryName

GROUP BY ProductName

ORDER BY ProductName

END

[SalesByCategory] 'Produce'



**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

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

AS

BEGIN

UPDATE [Order Details]

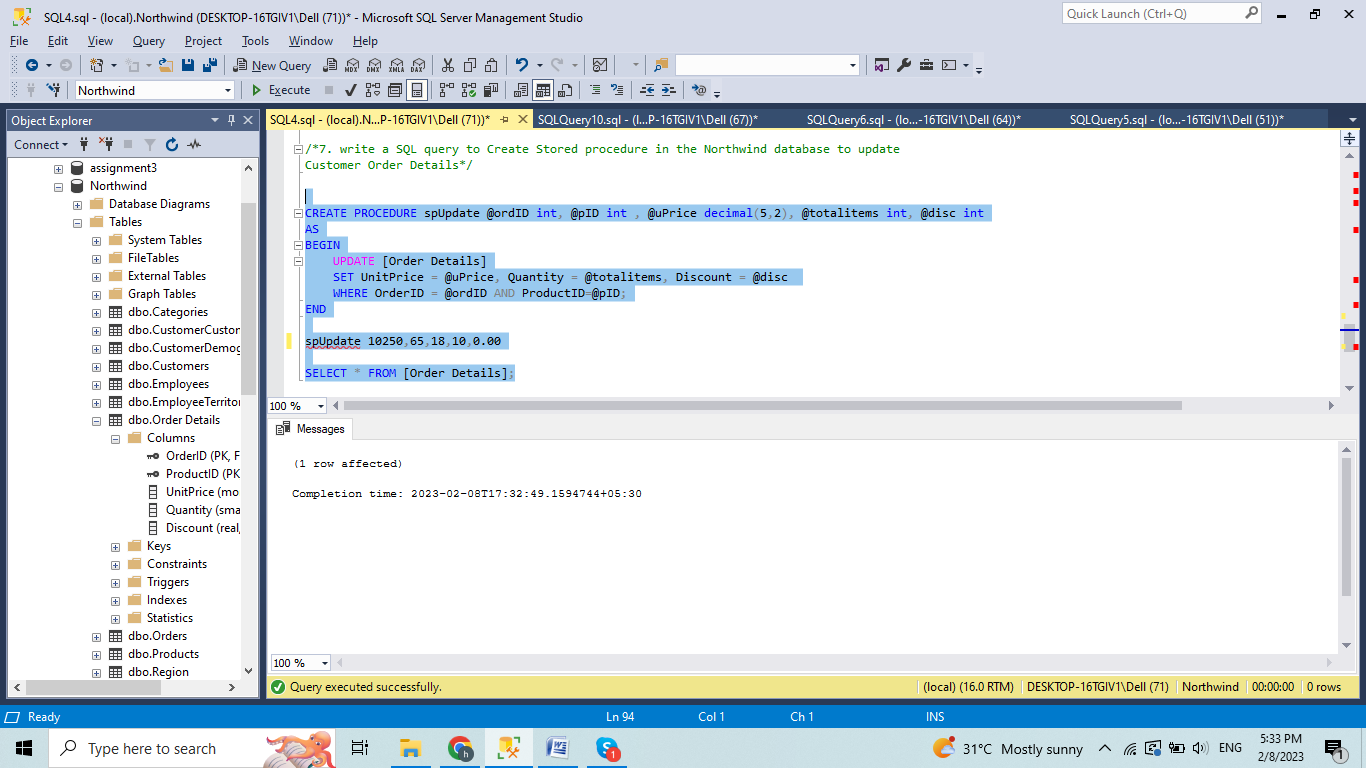
SET UnitPrice = @uPrice, Quantity = @totalitems, Discount = @disc

WHERE OrderID = @ordID AND ProductID=@pID;

END

spUpdate 10250,65,18,10,0.00

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 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 10251,42,1,10,0

