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.

```
CREATE PROCEDURE que1
AS
	SELECT CustomerID, AVG(Freight) as AvgFreight
	FROM Orders
	GROUP BY CustomerID

GO
```

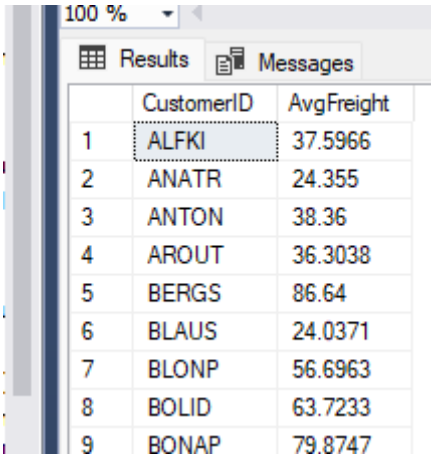
Insert :

```
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,ShipAddre
ss,ShipCity,ShipRegion,ShipPostalCode,ShipCountry)
			SELECT
CustomerID,EmployeeID,OrderDate,RequiredDate,ShippedDate,ShipVia,Freight,ShipName,ShipAddres
s,ShipCity,ShipRegion,ShipPostalCode,ShipCountry
```

```
                From Inserted
            END

            Delete from #TempTable where OrderID = @OrderID

        End
END
```

NOW we check the out put:
if we add Freight value above avg(Freight) value the not add else add.

```
exec que1
```



100 %

Results | Messages

	CustomerID	AvgFreight
1	ALFKI	37.5966
2	ANATR	24.355
3	ANTON	38.36
4	AROUT	36.3038
5	BERGS	86.64
6	BLAUS	24.0371
7	BLONP	56.6963
8	BOLID	63.7233
9	BONAP	79.8747

now insert greater value

```
insert into orders values ( 'AROUT', 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')
```

```
(89 rows affected)

(1 row affected)
Msg 50000, Level 16, State 1, Procedure tr_que1_insert, Line 24 [Batch Start Line 30]
ABOVE AVERAGE

(1 row affected)

(1 row affected)
```

now insert less value

```
insert into orders values ( 'AROUT', 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')
```

```
(89 rows affected)

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)
```

Update:

```
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
```

NOW we check the out put:
if we update and add Freight value above avg(Freight) value the not add else add.

```
exec que1
```

Results | Messages

	CustomerID	AvgFreight
57	PICCO	118.611
58	PRINI	72.972
59	QUEDE	36.3944
60	QUEEN	152.5153
61	QUICK	200.201
62	RANCH	43.836
63	RATTC	113.7877
64	REGGC	26.63
65	RICAR	57.5409
66	RICSU	100.129
67	ROMEY	12.894

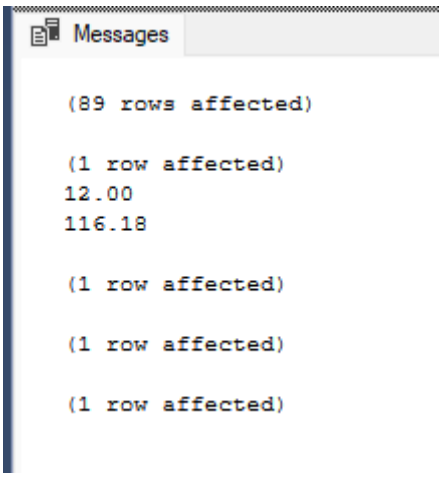
now insert greater value

```
UPDATE Orders SET Freight=120 WHERE OrderID = 10272
```

```
(89 rows affected)

(1 row affected)
120.00
116.18
Msg 50000, Level 16, State 1, Procedure tr_que1_update, Line 28 [Batch Start Line 34]
ABOVE AVERAGE

(1 row affected)

(1 row affected)
```

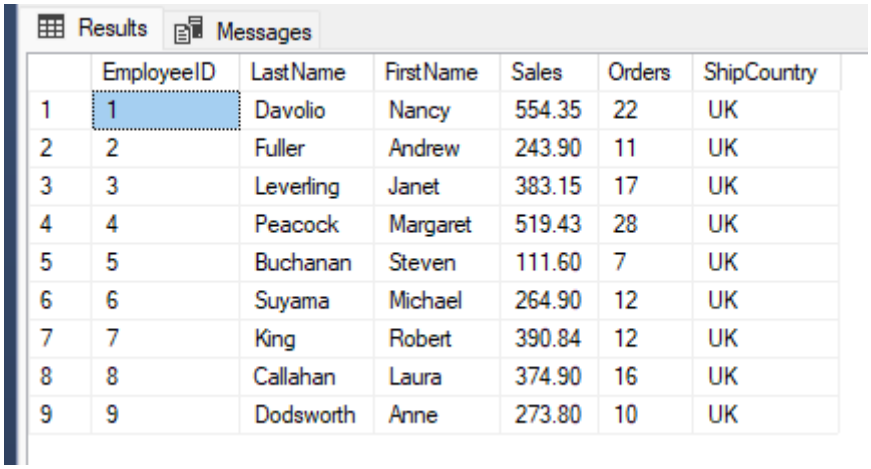
now insert less value

```
UPDATE Orders SET Freight=12 WHERE OrderID = 10272
```



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

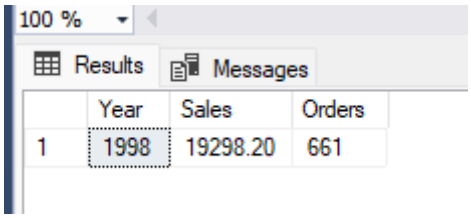
```
ALTER proc Proc_Employee_sales_bycountry
	@ShipCountry nvarchar(10)
	as
	begin
		SELECT
Employees.EmployeeID,Employees.[LastName],Employees.[FirstName],sum(UnitPrice) as
Sales,COUNT(Orders.EmployeeID) as [Orders],Orders.ShipCountry
		FROM Employees
		INNER JOIN Orders
		ON Orders.EmployeeID = Employees.EmployeeID
		INNER JOIN [Order Details]
		ON [Order Details].orderID = Orders.orderID
		Where ShipCountry = @ShipCountry
		GROUP BY Employees.EmployeeID,
Orders.ShipCountry,Employees.[LastName],Employees.[FirstName]
	end

Execute Proc_Employee_sales_bycountry 'UK'
```


Results Messages

	EmployeeID	LastName	FirstName	Sales	Orders	ShipCountry
1	1	Davolio	Nancy	554.35	22	UK
2	2	Fuller	Andrew	243.90	11	UK
3	3	Leverling	Janet	383.15	17	UK
4	4	Peacock	Margaret	519.43	28	UK
5	5	Buchanan	Steven	111.60	7	UK
6	6	Suyama	Michael	264.90	12	UK
7	7	King	Robert	390.84	12	UK
8	8	Callahan	Laura	374.90	16	UK
9	9	Dodsworth	Anne	273.80	10	UK

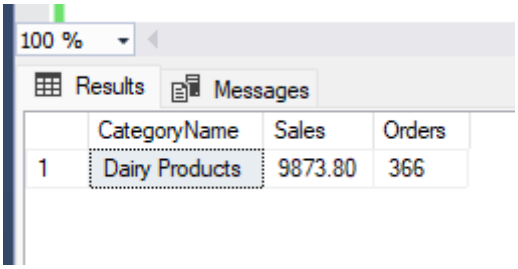
3. write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales by Year

```
ALTER PROC Proc_sales_by_year
	@year int
	as
	begin
		SELECT year(ShippedDate) as Year ,sum(UnitPrice) as Sales,
COUNT(Orders.EmployeeID) as Orders
		FROM Orders
		INNER JOIN [Order Details]
		ON [Order Details].orderID = Orders.orderID
	/*	LEFT JOIN Employees
		ON Orders.EmployeeID = Employees.EmployeeID */
		Where year(ShippedDate) = @year
		GROUP BY year(ShippedDate)
	end

execute sp_rename 'Proc_Employee_sales_by' , 'Proc_sales_by_year';

execute Proc_sales_by_year '1998'
```



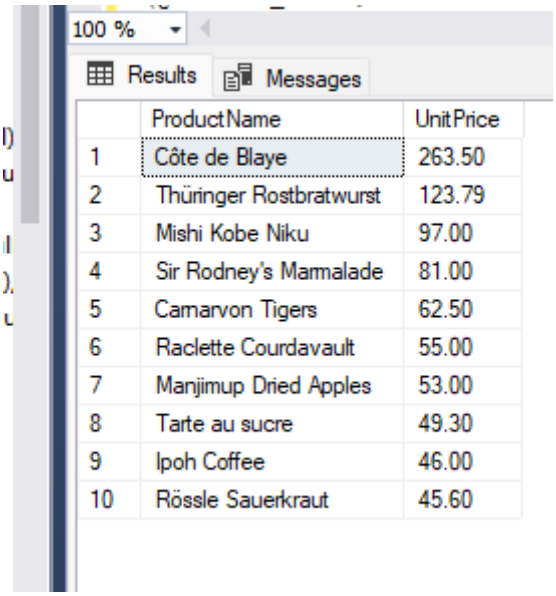
4. write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales By Category

```
ALTER PROC Proc_sales_by_Categories
	@categoriename varchar(50)
	as
	begin
		SELECT CategoryName ,sum([Order Details].UnitPrice) as Sales,
COUNT(Orders.EmployeeID) as Orders
		FROM Orders , [Order Details], Products,Categories
		Where CategoryName = @categoriename
		and [Order Details].ProductID = Products.ProductID
		and [Order Details].OrderID = Orders.OrderID
		and Products.CategoryID = Categories.CategoryID
		GROUP BY CategoryName
	end

execute Proc_sales_by_Categories 'Dairy Products'
```



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

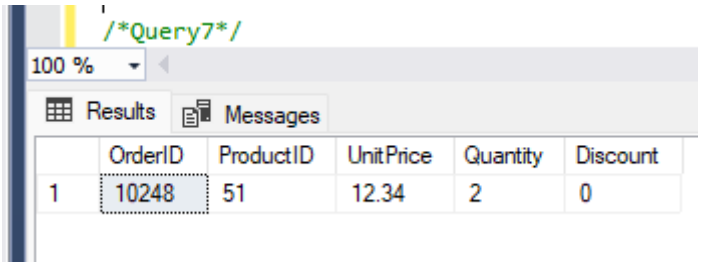
```
ALTER PROC Proc_10_most_expensive_products
	as
	begin
		SELECT TOP(10) ProductName , UnitPrice
		FROM Products
		Order BY UnitPrice desc
	end

execute Proc_10_most_expensive_products
```



	ProductName	UnitPrice
1	Côte de Blaye	263.50
2	Thüringer Rostbratwurst	123.79
3	Mishi Kobe Niku	97.00
4	Sir Rodney's Marmalade	81.00
5	Carnarvon Tigers	62.50
6	Raclette Courdavault	55.00
7	Manjimup Dried Apples	53.00
8	Tarte au sucre	49.30
9	Ipoh Coffee	46.00
10	Rössle Sauerkraut	45.60

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

```
ALTER PROCEDURE [insert_Order Details_1]
(@OrderID_1 int,
@ProductID_2 int,
@UnitPrice_3 money = NULL,
@Quantity_4 smallint,
@Discount_5 real = 0)
AS
INSERT INTO [Northwind].[dbo].[Order Details]
( [OrderID], [ProductID], [UnitPrice], [Quantity], [Discount])
VALUES ( @OrderID_1, @ProductID_2, @UnitPrice_3, @Quantity_4, @Discount_5)

execute [insert_Order Details_1] 10248,51,12.34,2,0
SELECT * FROM [Order Details]
WHERE OrderID = 10248 and ProductID = 51
```



```
/*Query7*/
```

	OrderID	ProductID	UnitPrice	Quantity	Discount
1	10248	51	12.34	2	0

7. write a SQL query to Create Stored procedure in the Northwind databaseto update Customer Order Details

```
CREATE PROCEDURE [update_Order Details_1]
(
@OrderID_1 int,
@ProductID_2 int,
@NewQuantity_4 smallint= NULL,
@NewUnitPrice_3 money = NULL,
@NewDiscount_5 real = NULL
)
AS
UPDATE [Northwind].[dbo].[Order Details]
SET [Quantity] = @NewQuantity_4, [UnitPrice] = @NewUnitPrice_3, [Discount] = @NewDiscount_5
WHERE ( [OrderID] = @OrderID_1 AND
              [ProductID] = @ProductID_2 )

execute [update_Order Details_1] 10248,11,12.00,12,0
SELECT * FROM [Order Details]
WHERE OrderID = 10248 and ProductID = 11
```