SQL-Assignment-4

Name: PATEL DEV V.

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

QUERY:

```sql
DROP PROCEDURE  IF EXISTS spAvgFreightForCustomer
create procedure spAvgFreightForCustomer
@customer_id char(5),
@AvgFreight real output
as
begin
    select @AvgFreight = avg(freight)
    from orders
    where CustomerID = @customer_id;
end

Declare @AvgFreight real
EXECUTE spAvgFreightForCustomer 'ROMEY', @AvgFreight =
@AvgFreight OUTPUT
select @AvgFreight as [answer];



alter TRIGGER trCheckAvgVal
ON orders
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @AverageFreightValue real, @InsertedFreight
        real, @OldAverage real
```

```sql
DECLARE @CustomerId char(5), @Count int
DECLARE @errorText  NVARCHAR(100)
SELECT @CustomerId = CustomerID, @InsertedFreight =
      freight FROM inserted

DECLARE @DeletedId int, @OldFreight real
SELECT @DeletedId = OrderID, @OldFreight = freight
      FROM deleted


SELECT @Count = COUNT(OrderID) FROM orders WHERE
      CustomerID = @CustomerId

EXECUTE  spAvgFreightForCustomer  @CustomerId ,
      @AvgFreight = @AverageFreightValue OUTPUT

If @DeletedId IS NULL
BEGIN
    SET @OldAverage = ((@AverageFreightValue *
         @Count) - @InsertedFreight)/(@Count-1)
END
ELSE
BEGIN
    SET @OldAverage ((@AverageFreightValue*@Count -
         @InsertedFreight+@OldFreight)/(@Count)
END

IF @InsertedFreight > @OldAverage
BEGIN
    SET @errorText = 'Frieght Value cannot be
              greater than Average Fright value =
              '+CAST(@OldAverage AS NVARCHAR)
    RAISERROR(@errorText,16,1)
    ROLLBACK TRAN
END
END


insert into Orders
(EmployeeID,OrderDate,Freight,ShipCity,ShipCountry)
   values( 1, '1998-05-06',200,'Albuquerque','USA')
```

Messages

```
Msg 50000, Level 16, State 1, Procedure trCheckAvgVal, Line 32 [Batch Start Line 1889]
Frieght Value cannot be greater than Average Fright value = 113.169
Msg 3609, Level 16, State 1, Line 1891
The transaction ended in the trigger. The batch has been aborted.
```

p

830	11077	RATTC	1	1998-05-06	2	8.53	Albuquerque	USA
831	11078	RATTC	1	2023-02-08	2	16	San Cristóbal	USA
832	11079	RATTC	1	1998-05-06	2	15	Albuquerque	USA

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

QUERY:

```
CREATE PROCEDURE spRetrieveEmployeeSalesByCountry
@ShipCountry NVARCHAR(15),
@EmployeeID INT

AS

BEGIN

SELECT O.ShipCountry AS "Country",COUNT(E.EmployeeID) AS
"Total Sale Items" , SUM(OD.Quantity *(1-OD.Discount)*
OD.UnitPrice)  AS "Total sales amount"
FROM  Orders O INNER JOIN Employees E ON O.EmployeeID =
E.EmployeeID
INNER JOIN [Order Details] OD ON O.OrderID = OD.OrderID
WHERE ShipCountry = @ShipCountry AND E.EmployeeID =
@EmployeeID
GROUP BY ShipCountry

END

EXECUTE spRetrieveEmployeeSalesByCountry USA, 3
```
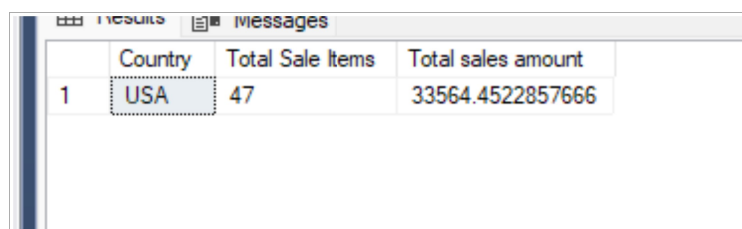
	Country	Total Sale Items	Total sales amount
1	USA	47	33564.4522857666

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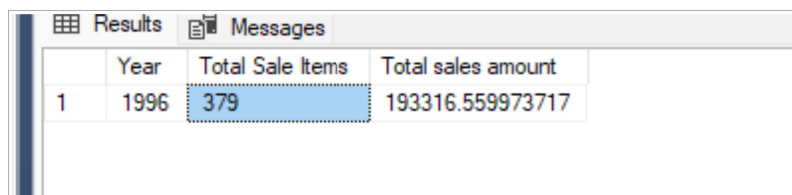
QUERY:

```
CREATE PROCEDURE spRetrieveSalesByYear
@Year NVARCHAR(60)

AS

BEGIN

SELECT YEAR(O.ShippedDate) AS "Year", COUNT
(YEAR(O.ShippedDate)) AS "Total Sale Items", SUM(OD.Quantity
*(1-OD.Discount)*  OD.UnitPrice)  AS "Total sales amount"
FROM Orders O INNER JOIN Employees E ON O.EmployeeID =
E.EmployeeID
INNER JOIN [Order Details] OD ON O.OrderID = OD.OrderID
WHERE YEAR(O.ShippedDate) = YEAR(@Year)
GROUP BY YEAR(O.ShippedDate)
END

EXECUTE spRetrieveSalesByYear 1996
```

	Year	Total Sale Items	Total sales amount
1	1996	379	193316.559973717

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

QUERY:

```
CREATE PROCEDURE spRetrieveSalesByCategories
@CategoryName VARCHAR(15)

AS

BEGIN

SELECT  C.CategoryName, COUNT(C.CategoryName) AS "Total Sale
items" , SUM(OD.Quantity *(1-OD.Discount)*  OD.UnitPrice)  AS
"Total sales amount"
FROM [Order Details] OD INNER JOIN Orders O ON OD.OrderID =
O.OrderID
INNER JOIN Products P ON OD.ProductID = P.ProductID
INNER JOIN Categories C ON P.CategoryID= C.CategoryID
WHERE C.CategoryName = @CategoryName
group by C.CategoryName
order by C.CategoryName ASC

END

EXECUTE spRetrieveSalesByCategories 'Confections'
```

	CategoryName	Total Sale items	Total sales amount
1	Confections	334	167357.224902153

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

QUERY:

```
CREATE PROCEDURE spTenMostExpensiveProduct

AS

BEGIN

SELECT TOP 10 *
FROM Products
ORDER BY UnitPrice DESC

END

EXECUTE spTenMostExpensiveProduct
```

	ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder	ReorderLevel	Discontinued
1	38	Côte de Blaye	18	1	12 - 75 cl bottles	263.50	17	0	15	0
2	29	Thüringer Rostbratwurst	12	6	50 bags x 30 sausgs.	123.79	0	0	0	1
3	9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.00	29	0	0	1
4	20	Sir Rodney's Marmalade	8	3	30 gift boxes	81.00	40	0	0	0
5	18	Carnarvon Tigers	7	8	16 kg pkg.	62.50	42	0	0	0
6	59	Raclette Courdavault	28	4	5 kg pkg.	55.00	79	0	0	0
7	51	Manjimup Dried Apples	24	7	50 - 300 g pkgs.	53.00	20	0	10	0
8	62	Tarte au sucre	29	3	48 pies	49.30	17	0	0	0
9	43	Ipoh Coffee	20	1	16 - 500 g tins	46.00	17	10	25	0
10	28	Rössle Sauerkraut	12	7	25 - 825 g cans	45.60	26	0	0	1

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

QUERY:

```sql
CREATE procedure spInsertCustomerOrderDetails

@CustomerId char(5),  @EmployeeId int, @ShipVia int ,
@FreightVal real,@ShipCity nvarchar(15) , @ShipCountry
nvarchar(15) , @product_id int , @quantity int ,
@discount real

as
begin

    declare @order_id int  , @unit_price real
    set @order_id = (select max(OrderID) from orders)+1
    set @unit_price = (select  UnitPrice  from products
    where ProductID =                   @product_id)


    begin try

    insert into orders

        (OrderID,CustomerID,EmployeeID,ShippedDate
        ,ShipVia,Freight,ShipCity,ShipCountry)
        values (@order_id , @CustomerId ,
        @EmployeeId,GETDATE(), @ShipVia, @FreightVal,
         @ShipCity, @ShipCountry)

    insert into [Order Details] values (@order_id ,
    @product_id ,  @unit_price , @quantity , @discount)

    end try

    begin catch
        RAISERROR ('ERROR OCCURS',16,1)
        ROLLBACK TRAN

    end catch
```
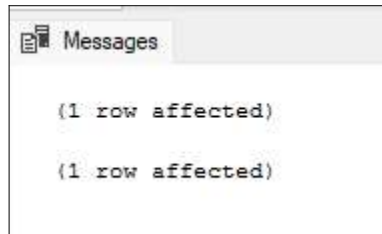
end

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

QUERY:

```
CREATE PROCEDURE spUpdateCustomerOrderDetails
@OrderId int, @CustomerId char(5), @EmployeeId int, @OrderDate
date,
@ShipVia int ,@Freight real, @ShipCountry nvarchar(15),
@ProductId int, @Quantity int, @Discount real , @ShipCity
nvarchar(15)
AS
BEGIN
DECLARE @UnitPrice real
BEGIN TRAN
set @UnitPrice = (select UnitPrice from products where ProductID =
@ProductId)

BEGIN TRY
UPDATE orders
SET
CustomerID = @CustomerId, EmployeeID = @EmployeeId, OrderDate =
@OrderDate, ShipVia = @ShipVia, freight = @Freight, ShipCity =
@ShipCity, ShipCountry = @ShipCountry WHERE OrderID = @OrderId

UPDATE [Order Details]
SET @ProductId = @ProductId, UnitPrice = @UnitPrice, quantity =
@Quantity, discount = @Discount WHERE OrderID = @OrderId

COMMIT TRAN
END TRY
BEGIN CATCH
SELECT ERROR_MESSAGE() as Error
ROLLBACK TRAN
END CATCH
END
```

| 830 | 11077 | RATTC | 1 | 1998-05-06 | 2 | 3 | USA |
| 831 | 11078 | RATTC | 1 | 1998-02-08 | 2 | 8 | UK |