Assignment-4

Name: Parmar Bhavik J.

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:

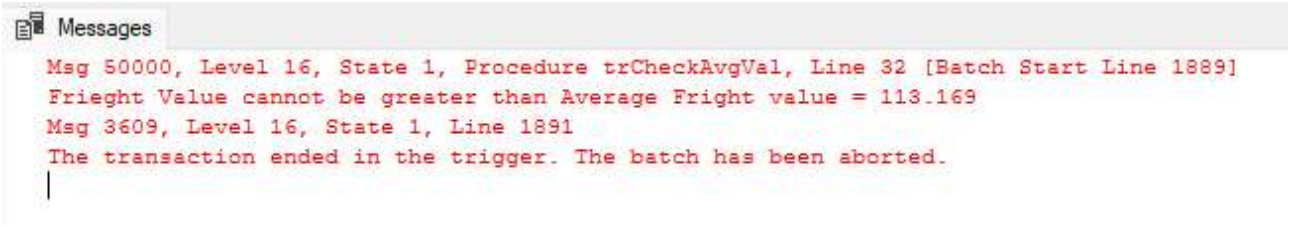
```
create procedure spAvgFreightForCustomer
@customer_id char(5),
@AvgFreight real output
as
begin
     select @AvgFreight = avg(freight)
     from orders
     where customer_id = @customer_id;
end

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

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

```
      SELECT @Count = COUNT(order_id) FROM orders WHERE customer_id
= @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
```

QUERY:

```
insert into orders values (11079 , 'RATTC' , 1, '1998-05-06' , 2 ,
200 , 'Albuquerque' , 'USA' )
```




```
insert into orders values (11079 , 'RATTC' , 1, '1998-05-06' , 2 ,
15 , '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 spEmployeeSalesByCountry
@country nvarchar(50)
as
begin
     SELECT e.first_name + ' ' + e.last_name as 'Employee Name'
,ship_country AS [Country], count(o.order_id) as [Total
Orders],SUM((d.unit_price*d.quantity*(1-d.discount))) AS [Total
Sale Amount] FROM orders o
     JOIN order_details d ON o.order_id = d.order_id
     JOIN employees e ON o.employee_id = e.employee_id
     where o.ship_country = @country
     GROUP BY ship_country, e.employee_id, e.first_name ,
e.last_name
end

exec spEmployeeSalesByCountry 'Brazil'
```

	Employee Name	Country	Total Orders	Total Sale Amount
1	Nancy Davolio	Brazil	1	1296
2	Andrew Fuller	Brazil	5	1088.10997390747
3	Janet Leverling	Brazil	9	3670.43990421295
4	Margaret Peacock	Brazil	18	5737.42002105713
5	Steven Buchanan	Brazil	4	9210.89996337891
6	Michael Suyama	Brazil	5	1517.51998138428
7	Robert King	Brazil	6	2054.78004455566
8	Laura Callahan	Brazil	14	5713.35001564026
9	Anne Dodsworth	Brazil	2	166

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

QUERY:

```
alter procedure spSalesByYear
as
begin
```

```
     SELECT DATEPART(YEAR,order_date) AS [Year] , count(o.order_id)
as [Total Orders],  SUM((d.unit_price*d.quantity*(1-d.discount)))
AS [Total Sale Amount] FROM orders o
     JOIN order_details d ON o.order_id = d.order_id
     GROUP BY DATEPART(YEAR,order_date)
     ORDER BY DATEPART(YEAR,order_date)

end

exec spSalesByYear
```

Results | Messages

	Year	Total Orders	Total Sale Amount
1	1996	405	208083.969679832
2	1997	229	133754.005244255
3	2023	1	249.900009155273

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

Query:

```
create procedure spSalesByCategory
as
begin
     SELECT c.category_name , count(p.product_id) as [Total
Products] , SUM((o.unit_price*o.quantity*(1-o.discount))) AS [Total
Sale Amount] FROM products p
JOIN order_details o ON p.product_id = o.product_id
JOIN categories c ON p.category_id = c.category_id
GROUP BY p.category_id, c.category_name
ORDER BY p.category_id;

end

exec spSalesByCategory
```

	category_name	Total Products	Total Sale Amount
1	Beverages	115	77174.2500133514
2	Condiments	59	31537.9499073029
3	Confections	103	48590.7650442123
4	Dairy Products	117	59902.2398300171
5	Grains/Cereals	55	22110.6199588776
6	Meat/Poultry	60	48478.2199134827
7	Produce	40	27145.5003128052
8	Seafood	86	27148.3299531937

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

Query:

```
alter procedure spMostExpensiveProducts
as
begin
     select TOP 10 product_id , product_name , unit_price
     from products
     order by unit_price desc
end

exec spMostExpensiveProducts
```

Results | Messages

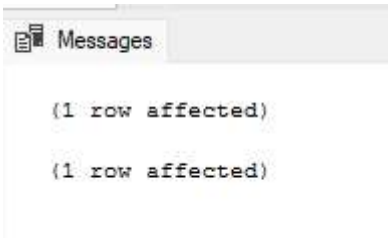
	product_id	product_name	unit_price
1	38	Côte de Blaye	263.5
2	29	Thüringer Rostbratwurst	123.79
3	9	Mishi Kobe Niku	97
4	20	Sir Rodney's Marmalade	81
5	18	Carnarvon Tigers	62.5
6	59	Raclette Courdavault	55
7	51	Manjimup Dried Apples	53
8	62	Tarte au sucre	49.3
9	43	Ipoh Coffee	46
10	28	Rössle Sauerkraut	45.6

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

Query:

```
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(order_id) from orders) + 1
     set @unit_price = (select  unit_price  from products where
product_id = @product_id)
```

```
	begin try

	insert into orders 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

EXEC spInsertCustomerOrderDetails 'RATTC', 1, 2,13,'San Cristóbal'
,'USA',11,14,0.15
```



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

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 unit_price from products where product_id
= @ProductId)

BEGIN TRY
UPDATE orders
SET
customer_id = @CustomerId, employee_id = @EmployeeId, order_date =
@OrderDate, ship_via = @ShipVia, freight = @Freight, ship_city =
@ShipCity, ship_country = @ShipCountry WHERE order_id = @OrderId

UPDATE order_details
SET product_id = @ProductId, unit_price = @UnitPrice, quantity =
@Quantity, discount = @Discount WHERE order_id = @OrderId

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

EXEC spUpdateCustomerOrderDetails 11078,'RATTC', 1,'1998-02-08',
2,8, 'UK',11,5,0.15
```

Order Table

831	11078	RATTC	1	1998-02-08	2	8	UK

Order_details

1711	11078	11	21	5	0.15