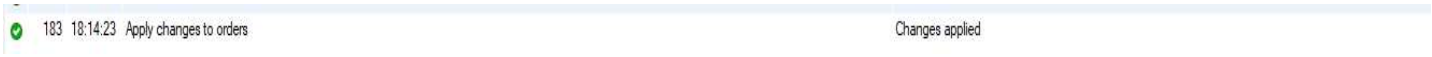
SQL Assignment 4

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.

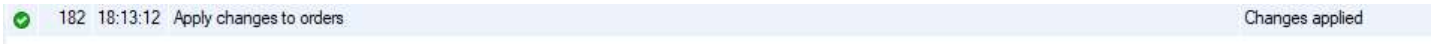
Before Insert Trigger :

```
CREATE DEFINER=`root`@`localhost` TRIGGER `check_freight_before_insert` BEFORE INSERT ON `orders` FOR EACH ROW BEGIN
DECLARE avgf decimal(10,4);
select avg(Freight) into avgf from orders where
CustomerID=new.CustomerID;
if new.Freight > avgf then
SIGNAL SQLSTATE '45000'
SET MESSAGE_TEXT = "Freight value exceeding from avaerage";
end if;
end
```



Before Update Trigger :

```
CREATE DEFINER=`root`@`localhost` TRIGGER `check_freight_before_update` BEFORE UPDATE ON `orders` FOR EACH ROW BEGIN
DECLARE avgf decimal(10,4);
select avg(Freight) into avgf from orders where
CustomerID=new.CustomerID;
if new.Freight > avgf then
SIGNAL SQLSTATE '45000'
SET MESSAGE_TEXT = "Updated Freight Value Exceed the Limit Of Average Freight";
end if;
END
```



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

call procedure :

```
call employeeSalesByCountry();/* Q-2 */
```

store procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `employeeSalesByCountry`()
  BEGIN
select e.EmployeeID,e.FirstName,o.ShipCountry,sum(od.Quantity*od.UnitPrice) as 'Amount
  Of Sale' from employees as e left join orders as o on o.EmployeeID=e.EmployeeID join
  `order details` as od on od.OrderID=o.OrderID group by e.EmployeeID,o.ShipCountry;
  END
```

EmployeeID	FirstName	ShipCountry	Amount Of Sale
1	Nancy	Austria	18304.9500
1	Nancy	Finland	1828.9000
1	Nancy	Italy	1246.7000
1	Nancy	Germany	26710.4600
1	Nancy	Brazil	30010.0000
1	Nancy	Mexico	5291.0000
1	Nancy	Spain	1241.0000
1	Nancy	France	12772.5000
1	Nancy	USA	46382.6000
1	Nancy	Venezuela	9983.1000
1	Nancy	UK	11323.4500
1	Nancy	Poland	858.8500

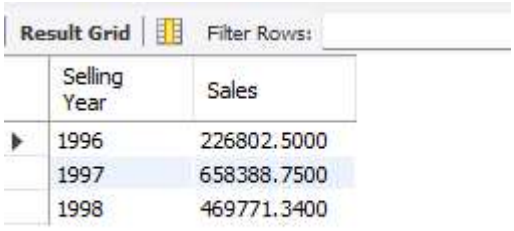
3. write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales by Year

call procedure :

```
call SalesbyYear();/* Q -3 */
```

stored procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `SalesbyYear`()
  BEGIN
  select year(orders.OrderDate) 'Selling Year', sum(`order details`.UnitPrice * `order
  details`.Quantity) Sales from `order details` inner join orders on `order
  details`.OrderID=orders.OrderID group by year(orders.OrderDate);
  END
```



Result Grid | Filter Rows:

Selling Year	Sales
1996	226802.5000
1997	658388.7500
1998	469771.3400

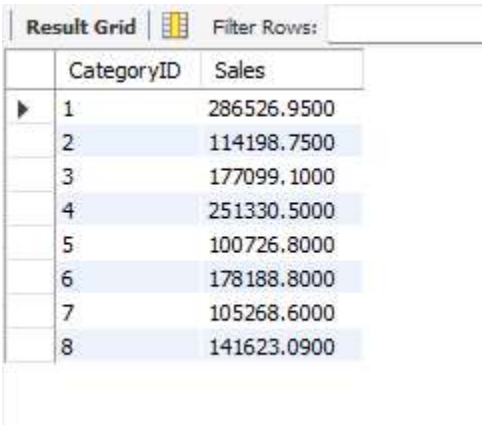
4. write a SQL query to Create Stored procedure in the Northwind database to retrieve Sales By Category

call procedure :

```
call salesByCategories();
```

Stored Procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `salesByCategories`()
  BEGIN
  select categories.CategoryID,sum(`order details`.Quantity* `order details`.UnitPrice) as
  'Sales' from categories left join products on products.CategoryID=categories.CategoryID
  join `order details` on `order details`.ProductID=products.ProductID group by
  categories.CategoryID;
  END
```



CategoryID	Sales
1	286526.9500
2	114198.7500
3	177099.1000
4	251330.5000
5	100726.8000
6	178188.8000
7	105268.6000
8	141623.0900

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

Call Procedure :

```
call topTen();
```

Stored Procedure:

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `topTen`()
  BEGIN
  select * from products order by UnitPrice desc limit 10;
  END
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder	ReorderLevel	Discontinued
38	Cte de Blaye	18	1	12 - 75 cl bottles	263.5000	17	0	15	0
29	Thringer Rostbratwurst	12	6	50 bags x 30 sausgs.	123.7900	0	0	0	1
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.0000	29	0	0	1
20	Sir Rodney's Marmalade	8	3	30 gift boxes	81.0000	40	0	0	0
18	Carnarvon Tigers	7	8	16 kg pkg.	62.5000	42	0	0	0
59	Raclette Courdavault	28	4	5 kg pkg.	55.0000	79	0	0	0
51	Manjimup Dried Apples	24	7	50 - 300 g pkgs.	53.0000	20	0	10	0
62	Tarte au sucre	29	3	48 pies	49.3000	17	0	0	0
43	Ipoh Coffee	20	1	16 - 500 g tins	46.0000	17	10	25	0
28	Rssle Sauerkraut	12	7	25 - 825 g cans	45.6000	26	0	0	1

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

Call Procedure :

```
call insertOrderDetails(10250,8,42,12,2.2);
```

Stored Procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `insertOrderDetails`(
  in oid int,
  in pd int,
  in up decimal(10,4),
  in qty smallint ,
  in dsc double(8,0))
  BEGIN
  INSERT INTO `northwind`.`order details`
  (`OrderID`,
  `ProductID`,
  `UnitPrice`,
  `Quantity`,
  `Discount`)
  VALUES
  (oid,
  pd,
  up,
  qty,
  dsc);
  END
```

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

Call Procedure:

```
call updateOrderDetails(10248,11,14,12,1);
```

Stored Procedure :

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `updateOrderDetails`(
  in oid int,
  in pd int,
  in up decimal(10,4),
  in qty smallint ,
  in dsc double(8,0))
  BEGIN
  update northwind.`order details` set
  UnitPrice=up,
  Quantity=qty,
  Discount=dsc where OrderID=oid and productID=pd;
  END
```