**MySQL 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.

**DELIMITER //**

**CREATE TRIGGER query1\_1**

**BEFORE INSERT on orders for EACH ROW**

**BEGIN**

**declare avg\_Freight double;**

**select avg(Freight) into avg\_Freight from orders where CustomerID = NEW.CustomerID;**

**IF NEW.Freight > avg\_Freight then signal sqlstate '45000'**

**SET message\_text = "insert record error, your fright value is greater than**  **avg\_fright values";**

**END IF;**

**END //**



**DELIMITER //**

**CREATE TRIGGER query1\_2**

**BEFORE UPDATE on orders for EACH ROW**

**BEGIN**

**declare avg\_Freight double;**

**select avg(Freight) into avg\_Freight from orders where CustomerID = NEW.CustomerID;**

**IF NEW.Freight > avg\_Freight then signal sqlstate '45000'**

**SET MESSAGE\_TEXT = "update record error, your fright value is greater than**  **avg\_fright values";**

**END IF;**

**END //**



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

**DELIMITER &&**

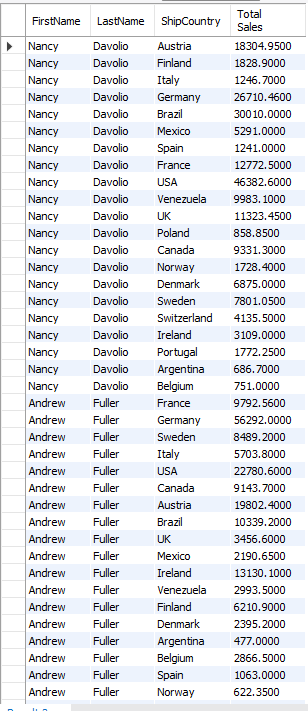
**CREATE PROCEDURE query2 ()**

**BEGIN**

**select e.FirstName, e.LastName, d.ShipCountry, d.`Total Sales` from employees e inner**  **join (select o.EmployeeID, o.ShipCountry, sum(UnitPrice\*Quantity) as "Total Sales" from**  **`order details` od inner join orders o on o.OrderID = od.OrderID group by**  **o.ShipCountry,o.EmployeeID) d on e.EmployeeID = d.EmployeeID;**

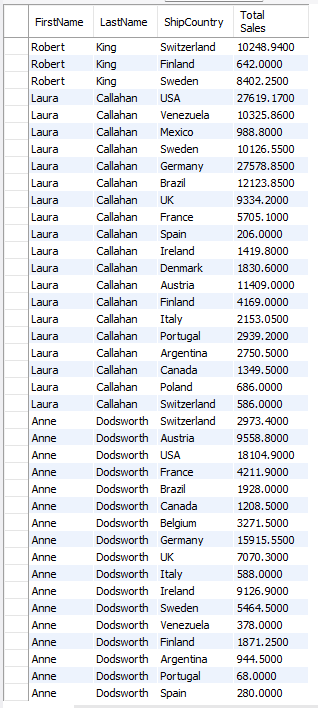
**END &&**

**call query2();**





…



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

**DELIMITER &&**

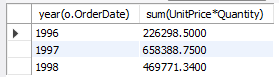
**CREATE PROCEDURE query3 ()**

**BEGIN**

**select year(o.OrderDate), sum(UnitPrice\*Quantity) from orders o inner join `order details`**  **od on o.OrderID = od.OrderID group by year(o.OrderDate);**

**END &&**

**CALL query3();**



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

**DELIMITER &&**

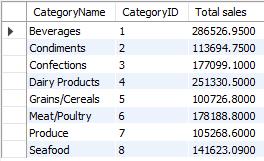
**CREATE PROCEDURE query4 ()**

**BEGIN**

**select c.CategoryName, d.CategoryID, d.`Total sales` from categories c inner join (select**  **p.CategoryID, sum(od.UnitPrice\*od.Quantity) as "Total sales" from products p inner join**  **`order details` od on p.ProductID = od.ProductID group by p.CategoryID) d on d.CategoryID**  **= c.CategoryID;**

**END &&**

**CALL query4();**



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

**DELIMITER &&**

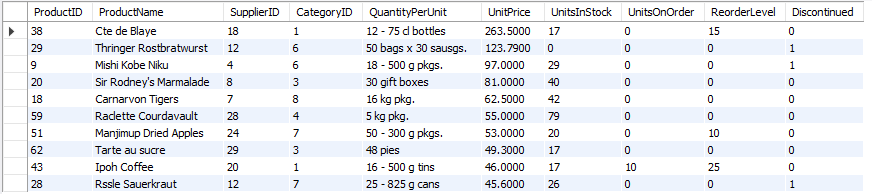
**CREATE PROCEDURE query5 ()**

**BEGIN**

**select \* from products order by UnitPrice desc limit 10;**

**END &&**

**call query5();**



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

**DELIMITER &&**

**DROP PROCEDURE IF EXISTS `add\_customer\_order`;**

**create procedure `add\_customer\_order` (in OrderID int, in CustomerID varchar(5), in EmployeeID int, in OrderDate datetime, in RequiredDate datetime, in ShippedDate datetime, in ShipVia int, in Freight double, in ShipName varchar(40), in ShipAddress varchar(60), in ShipCity varchar(15), in ShipRegion varchar(15), in ShipPostalCode varchar(10), in ShipCountry varchar(10), in ProductID int, in UnitPrice double, in Quantity smallint, in Discount double)**

**BEGIN**

**insert into orders values (OrderID,CustomerID,EmployeeID,OrderDate,RequiredDate,ShippedDate,ShipVia,Freight,ShipName,ShipAddress,ShipCity,ShipRegion,ShipPostalCode,ShipCountry);**

**insert into `order details` values(OrderID,ProductID,UnitPrice,Quantity,Discount);**

**END &&**

**call add\_customer\_order(11079, 'WARTH', 3, '1996-07-26 00:00:00', '1996-09-06 00:00:00', '1996-07-31 00:00:00', 3, 25.77, 'Wartian Herkku', 'Torikatu 38', 'Oulu', NULL, '90110', 'Finland', 11, 1000, 12, 0);**





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

**DELIMITER &&**

**DROP PROCEDURE IF EXISTS `change\_customer\_order`;**

**create procedure `change\_customer\_order` (in aOrderID int, in aCustomerID varchar(5), in aEmployeeID int, in aOrderDate datetime, in aRequiredDate datetime, in aShippedDate datetime, in aShipVia int, in aFreight double, in aShipName varchar(40), in aShipAddress varchar(60), in aShipCity varchar(15), in aShipRegion varchar(15), in aShipPostalCode varchar(10), in aShipCountry varchar(10))**

**BEGIN**

**update orders set CustomerID = aCustomerID, EmployeeID = aEmployeeID, OrderDate = aOrderDate, RequiredDate = aRequiredDate, ShippedDate = aShippedDate, ShipVia = aShipVia, Freight = aFreight, ShipName = aShipName, ShipAddress = aShipAddress, ShipCity = aShipCity, ShipRegion = aShipRegion, ShipPostalCode = aShipPostalCode, ShipCountry = aShipCountry where OrderId = aOrderId;**

**END &&**

**call change\_customer\_order(11079, 'WARTH', 4, '1996-07-26 00:00:00', '1996-09-06 00:00:00', '1996-07-31 00:00:00', 3, 50.77, 'Wartian Herkku', 'Torikatu 38', 'Oulu', NULL, '90110', 'Finland');**

