# 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.

Insert Trigger -

```
DROP TRIGGER IF EXISTS
`northwind`.`check_freight_before_insert`;


DELIMITER $$

CREATE TRIGGER `check_freight_before_insert` 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 = "Error Inserting Record, Your Average Freight
Value is Greater than Customer Freight Value";

    end if;



END$$

DELIMITER ;
```

```
INSERT INTO `northwind`.`orders` (`OrderID`, `CustomerID`,
`EmployeeID`, `OrderDate`, `RequiredDate`, `ShipVia`, `Freight`,
`ShipName`, `ShipAddress`, `ShipCity`, `ShipPostalCode`,
`ShipCountry`) VALUES ('11078', 'VINET', '4', '1998-05-06
00:00:00', '1998-05-06 00:00:00', '2', '100', 'Rattlesnake
Canyon Grocery', '12, rue des Bouchers', 'Marseille', '380007',
'India');
```

... Error Code: 1644. Error Inserting Record, Your Average Freight Value is Greater than Customer Freight Value

```
INSERT INTO `northwind`.`orders` (`OrderID`, `CustomerID`,
`EmployeeID`, `OrderDate`, `RequiredDate`, `ShipVia`, `Freight`,
`ShipName`, `ShipAddress`, `ShipCity`, `ShipPostalCode`,
`ShipCountry`) VALUES ('11078', 'VINET', '4', '1998-05-06
00:00:00', '1998-05-06 00:00:00', '2', '5.5', 'Rattlesnake
Canyon Grocery', '12, rue des Bouchers', 'Marseille', '380007',
'India');
```

,... 1 row(s) affected

## Update Trigger

```
DELIMITER $$

CREATE TRIGGER check_freight_before_update

    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 = "Error Updating Record, Your Average Freight
Value is Greater than Customer Freight Value";

    end if;

END$$


UPDATE `northwind`.`orders` SET `Freight` = '1000' WHERE
(`OrderID` = '10248');
```

Error Code: 1644. Error Updating Record, Your Average Freight Value is Greater than Customer Freight Value

```
UPDATE `northwind`.`orders` SET `Freight` = '1.5' WHERE
(`OrderID` = '10248');
```

1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

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

```
drop procedure if exists `employee_sales_by_country`;

create procedure `employee_sales_by_country` ()

begin

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

end $$


call employee_sales_by_country();
```

| FirstName | LastName | EmployeeID | ShipCountry | Total Sales |
|-----------|----------|------------|-------------|-------------|
| Nancy | Davolio | 1 | Austria | 18304.9500 |
| Nancy | Davolio | 1 | Finland | 1828.9000 |
| Nancy | Davolio | 1 | Italy | 1246.7000 |
| Nancy | Davolio | 1 | Germany | 26710.4600 |
| Nancy | Davolio | 1 | Brazil | 30010.0000 |
| Nancy | Davolio | 1 | Mexico | 5291.0000 |
| Nancy | Davolio | 1 | Spain | 1241.0000 |
| Nancy | Davolio | 1 | France | 12772.5000 |
| Nancy | Davolio | 1 | USA | 46382.6000 |
| Nancy | Davolio | 1 | Venezuela | 9983.1000 |
| Nancy | Davolio | 1 | UK | 11323.4500 |
| Nancy | Davolio | 1 | Poland | 858.8500 |
| Nancy | Davolio | 1 | Canada | 9331.3000 |
| Nancy | Davolio | 1 | Norway | 1728.4000 |
| Nancy | Davolio | 1 | Denmark | 6875.0000 |
| Nancy | Davolio | 1 | Sweden | 7801.0500 |
| Nancy | Davolio | 1 | Switzerland | 4135.5000 |
| Nancy | Davolio | 1 | Ireland | 3109.0000 |
| Nancy | Davolio | 1 | Portugal | 1772.2500 |
| Nancy | Davolio | 1 | Argentina | 686.7000 |
| Nancy | Davolio | 1 | Belgium | 751.0000 |
| Andrew | Fuller | 2 | France | 9792.5600 |
| Andrew | Fuller | 2 | Germany | 56292.0000 |
| Andrew | Fuller | 2 | Sweden | 8489.2000 |
| Andrew | Fuller | 2 | Italy | 5703.8000 |
| Andrew | Fuller | 2 | USA | 22780.6000 |
| Andrew | Fuller | 2 | Canada | 9143.7000 |
| Andrew | Fuller | 2 | Austria | 19802.4000 |
| Andrew | Fuller | 2 | Brazil | 10339.2000 |
| Andrew | Fuller | 2 | UK | 3456.6000 |
| Andrew | Fuller | 2 | Mexico | 2190.6500 |
| Andrew | Fuller | 2 | Ireland | 13130.1000 |
| Andrew | Fuller | 2 | Venezuela | 2993.5000 |
| Andrew | Fuller | 2 | Finland | 6210.9000 |
| Andrew | Fuller | 2 | Denmark | 2395.2000 |
| Andrew | Fuller | 2 | Argentina | 477.0000 |
| Andrew | Fuller | 2 | Belgium | 2866.5000 |
| Andrew | Fuller | 2 | Spain | 1063.0000 |
| Andrew | Fuller | 2 | Norway | 622.3500 |

| Andrew | Fuller | 2 | Belgium | 2866.5000 |
|---|---|---|---|---|
| Andrew | Fuller | 2 | Spain | 1063.0000 |
| Andrew | Fuller | 2 | Norway | 622.3500 |
| Janet | Leverling | 3 | France | 16118.2000 |
| Janet | Leverling | 3 | Brazil | 10087.2500 |
| Janet | Leverling | 3 | Finland | 976.1000 |
| Janet | Leverling | 3 | Germany | 48978.2600 |
| Janet | Leverling | 3 | Venezuela | 12999.9300 |
| Janet | Leverling | 3 | Ireland | 17029.6400 |
| Janet | Leverling | 3 | UK | 4723.3000 |
| Janet | Leverling | 3 | Canada | 13260.5000 |
| Janet | Leverling | 3 | USA | 34809.2300 |
| Janet | Leverling | 3 | Portugal | 1005.2000 |
| Janet | Leverling | 3 | Mexico | 3076.9000 |
| Janet | Leverling | 3 | Sweden | 11822.6500 |
| Janet | Leverling | 3 | Argentina | 319.2000 |
| Janet | Leverling | 3 | Austria | 24464.8500 |
| Janet | Leverling | 3 | Spain | 3375.2500 |
| Janet | Leverling | 3 | Switzerland | 5233.0600 |
| Janet | Leverling | 3 | Italy | 88.0000 |
| Janet | Leverling | 3 | Denmark | 1704.0000 |
| Janet | Leverling | 3 | Norway | 2684.4000 |
| Janet | Leverling | 3 | Belgium | 295.3800 |
| Margaret | Peacock | 4 | Brazil | 18818.3800 |
| Margaret | Peacock | 4 | Belgium | 14425.8000 |
| Margaret | Peacock | 4 | Venezuela | 8421.4000 |
| Margaret | Peacock | 4 | Mexico | 6921.7500 |
| Margaret | Peacock | 4 | Germany | 41285.7500 |
| Margaret | Peacock | 4 | Spain | 8738.8400 |
| Margaret | Peacock | 4 | Italy | 3993.4000 |
| Margaret | Peacock | 4 | USA | 50657.9400 |
| Margaret | Peacock | 4 | UK | 8871.1100 |
| Margaret | Peacock | 4 | Portugal | 4852.0000 |
| Margaret | Peacock | 4 | France | 25968.0000 |
| Margaret | Peacock | 4 | Ireland | 1708.0000 |
| Margaret | Peacock | 4 | Austria | 19171.6300 |
| Margaret | Peacock | 4 | Canada | 5495.8000 |
| Margaret | Peacock | 4 | Denmark | 18696.9500 |
| Margaret | Peacock | 4 | Switzerland | 5587.4000 |

...

...

| FirstName | LastName | EmployeeID | ShipCountry | Total Sales |
|---|---|---|---|---|
| Robert | King | 7 | Switzerland | 10248.9400 |
| Robert | King | 7 | Finland | 642.0000 |
| Robert | King | 7 | Sweden | 8402.2500 |
| Laura | Callahan | 8 | USA | 27619.1700 |
| Laura | Callahan | 8 | Venezuela | 10325.8600 |
| Laura | Callahan | 8 | Mexico | 988.8000 |
| Laura | Callahan | 8 | Sweden | 10126.5500 |
| Laura | Callahan | 8 | Germany | 27578.8500 |
| Laura | Callahan | 8 | Brazil | 12123.8500 |
| Laura | Callahan | 8 | UK | 9334.2000 |
| Laura | Callahan | 8 | France | 5705.1000 |
| Laura | Callahan | 8 | Spain | 206.0000 |
| Laura | Callahan | 8 | Ireland | 1419.8000 |
| Laura | Callahan | 8 | Denmark | 1830.6000 |
| Laura | Callahan | 8 | Austria | 11409.0000 |
| Laura | Callahan | 8 | Finland | 4169.0000 |
| Laura | Callahan | 8 | Italy | 2153.0500 |
| Laura | Callahan | 8 | Portugal | 2939.2000 |
| Laura | Callahan | 8 | Argentina | 2750.5000 |
| Laura | Callahan | 8 | Canada | 1349.5000 |
| Laura | Callahan | 8 | Poland | 686.0000 |
| Laura | Callahan | 8 | Switzerland | 586.0000 |
| Anne | Dodsworth | 9 | Switzerland | 2973.4000 |
| Anne | Dodsworth | 9 | Austria | 9558.8000 |
| Anne | Dodsworth | 9 | USA | 18104.9000 |
| Anne | Dodsworth | 9 | France | 4211.9000 |
| Anne | Dodsworth | 9 | Brazil | 1928.0000 |
| Anne | Dodsworth | 9 | Canada | 1208.5000 |
| Anne | Dodsworth | 9 | Belgium | 3271.5000 |
| Anne | Dodsworth | 9 | Germany | 15915.5500 |
| Anne | Dodsworth | 9 | UK | 7070.3000 |
| Anne | Dodsworth | 9 | Italy | 588.0000 |
| Anne | Dodsworth | 9 | Ireland | 9126.9000 |
| Anne | Dodsworth | 9 | Sweden | 5464.5000 |
| Anne | Dodsworth | 9 | Venezuela | 378.0000 |
| Anne | Dodsworth | 9 | Finland | 1871.2500 |
| Anne | Dodsworth | 9 | Argentina | 944.5000 |
| Anne | Dodsworth | 9 | Portugal | 68.0000 |
| Anne | Dodsworth | 9 | Spain | 280.0000 |

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

```
DELIMITER $$

drop procedure if exists `sales_by_year`;

create procedure `sales_by_year` ()

begin

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

end $$


call sales_by_year();
```

| | Year | Sales |
|---|------|---------------|
| ▶ | 1996 | 226298.5000 |
| | 1997 | 658388.7500 |
| | 1998 | 469771.3400 |

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

```
DELIMITER $$

drop procedure if exists `sales_by_category`;

create procedure `sales_by_category` ()

begin

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

end $$


call sales_by_category();
```

| | CategoryID | CategoryName | Total Sales |
|---|---|---|---|
| ▶ | 1 | Beverages | 286526.9500 |
| | 2 | Condiments | 113694.7500 |
| | 3 | Confections | 177099.1000 |
| | 4 | Dairy Products | 251330.5000 |
| | 5 | Grains/Cereals | 100726.8000 |
| | 6 | Meat/Poultry | 178188.8000 |
| | 7 | Produce | 105268.6000 |
| | 8 | Seafood | 141623.0900 |

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

```
drop procedure if exists `most_expensive_product`;

create procedure `most_expensive_product` (in max_limit int)

begin

select ProductID, ProductName, UnitPrice from products order by
UnitPrice desc limit max_limit;

end $$


call most_expensive_product(10);
```

| | ProductID | ProductName | UnitPrice |
|---|---|---|---|
| ▶ | 38 | Cte de Blaye | 263.5000 |
| | 29 | Thringer Rostbratwurst | 123.7900 |
| | 9 | Mishi Kobe Niku | 97.0000 |
| | 20 | Sir Rodney's Marmalade | 81.0000 |
| | 18 | Carnarvon Tigers | 62.5000 |
| | 59 | Raclette Courdavault | 55.0000 |
| | 51 | Manjimup Dried Apples | 53.0000 |
| | 62 | Tarte au sucre | 49.3000 |
| | 43 | Ipoh Coffee | 46.0000 |
| | 28 | Rssle Sauerkraut | 45.6000 |

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

```
DELIMITER $$

drop procedure if exists `insert_customer_order`;

create procedure `insert_customer_order` (in CustomerID
varchar(5), in EmployeeID int, in OrderDate datetime, in
RequiredDate 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(15),  in ProductID int, in
UnitPrice double, in Quantity int, in Discount int, in orderID
int, in addingProduct bool)

begin

DECLARE newOID int;

if addingProduct then

INSERT INTO `order details` (`OrderID`, `ProductID`,
`UnitPrice`, `Quantity`, `Discount`) VALUES (orderID, ProductID,
UnitPrice, Quantity, Discount);

else

INSERT INTO `orders` (`CustomerID`, `EmployeeID`, `OrderDate`,
`RequiredDate`, `ShipVia`, `Freight`, `ShipName`, `ShipAddress`,
`ShipCity`, `ShipRegion`, `ShipPostalCode`, `ShipCountry`)

    VALUES (CustomerID, EmployeeID, OrderDate, RequiredDate,
ShipVia, Freight, ShipName, ShipAddress, ShipCity, ShipRegion,
ShipPostalCode, ShipCountry);

    select o.oID into newOID from (select max(k.OrderID) oID
from orders k) o;

    select newOID as 'New Order ID';

    INSERT INTO `order details` (`OrderID`, `ProductID`,
`UnitPrice`, `Quantity`, `Discount`) VALUES (newOID, ProductID,
UnitPrice, Quantity, Discount);

end if;

end $$
```

```
DELIMITER ;
```

```
call insert_customer_order('RICSU', 4, '1998-05-06 00:00:00',
'1998-05-06 00:00:00', 2, 5.1, 'Rattlesnake Canyon Grocery',
'2817 Milton Dr.', 'Ahmedabad', 'North', '380007', 'India', 42,
25.5, 2, 0, null, false);
```

| newOID |
| --- |
| 11091 |

Record Inserted:

```
SELECT * FROM northwind.orders where OrderID=11091;
```

| OrderID | CustomerID | EmployeeID | OrderDate | RequiredDate | ShippedDate | ShipVia | Freight | ShipName | ShipAddress | ShipCity | ShipRegion | ShipPostalCode | ShipCountry |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11091 | RICSU | 4 | 1998-05-06 00:00:00 | 1998-05-06 00:00:00 | NULL | 2 | 5.1000 | Rattlesnake Canyon Grocery | 2817 Milton Dr. | Ahmedabad | North | 380007 | India |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Inserting Records Only in "Order Details"

```
call insert_customer_order(null, null, null, null, null, null,
null, null, null, null, null, null, 46, 20.5, 5, 0, 11091,
true);
```

Record Inserted:

```
SELECT * FROM northwind.`order details` where OrderId=11091;
```

| OrderID | ProductID | UnitPrice | Quantity | Discount |
| --- | --- | --- | --- | --- |
| 11091 | 42 | 25.5000 | 2 | 0 |
| 11091 | 46 | 20.5000 | 5 | 0 |
| NULL | NULL | NULL | NULL | NULL |

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

```
DELIMITER $$

drop procedure if exists `update_customer_order`;

create procedure `update_customer_order` (in iOrderID int, in
nCustomerID varchar(5), in nEmployeeID int, in nOrderDate
datetime, in nRequiredDate datetime, in nShipVia int, in
nFreight double, in nShipName varchar(40), in nShipAddress
varchar(60), in nShipCity varchar(15),in nShipRegion
varchar(15), in nShipPostalCode varchar(10), in nShipCountry
varchar(15))

begin

update orders set

    `CustomerID`=nCustomerID,`EmployeeID`=nEmployeeID,
`OrderDate`=nOrderDate, `RequiredDate`=nRequiredDate,
`ShipVia`=nShipVia,

    `Freight`=nFreight, `ShipName`=nShipName,
`ShipAddress`=nShipAddress, `ShipCity`=nShipCity,
`ShipRegion`=nShipRegion,

    `ShipPostalCode`=nShipPostalCode, `ShipCountry`=nShipCountry
where OrderID=iOrderID;

end $$

DELIMITER ;
```

```
call update_customer_order(11091, 'VINET', 2, '1998-05-06
00:00:00', '1998-05-06 00:00:00', 2, 1.1, 'Grocery Store',
'Paldi', 'Ahmedabad', 'West', '380007', 'India');
```

1 row(s) affected

```
SELECT * FROM northwind.orders where OrderID=11091;
```

| OrderID | CustomerID | EmployeeID | OrderDate | RequiredDate | ShippedDate | ShipVia | Freight | ShipName | ShipAddress | ShipCity | ShipRegion | ShipPostalCode | ShipCountry |
|---------|-----------|-----------|-----------|--------------|-------------|---------|---------|----------|-------------|----------|-----------|----------------|-------------|
| 11091 | VINET | 2 | 1998-05-06 00:00:00 | 1998-05-06 00:00:00 | NULL | 2 | 1.1000 | Grocery Store | Paldi | Ahmedabad | West | 380007 | India |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

| OrderID | CustomerID | EmployeeID | OrderDate | RequiredDate | ShippedDate | ShipVia | Freight | ShipName | ShipAddress | ShipCity | ShipRegion | ShipPostalCode | ShipCountry |
|---------|-----------|-----------|-----------|--------------|-------------|---------|---------|----------|-------------|----------|-----------|----------------|-------------|