**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.**

For Insert:

CREATE DEFINER=`root`@`localhost` TRIGGER `orders\_BEFORE\_INSERT` BEFORE INSERT ON `orders` FOR EACH ROW BEGIN

DECLARE averageF DOUBLE;

SET averageF = (SELECT avg(Freight) from orders where customerID = new.CustomerID);

IF averageF < new.Freight and averageF > 0 then SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Freight Error';

END IF;

END

For Update:

CREATE DEFINER=`root`@`localhost` TRIGGER `orders\_BEFORE\_UPDATE` BEFORE UPDATE ON `orders` FOR EACH ROW BEGIN

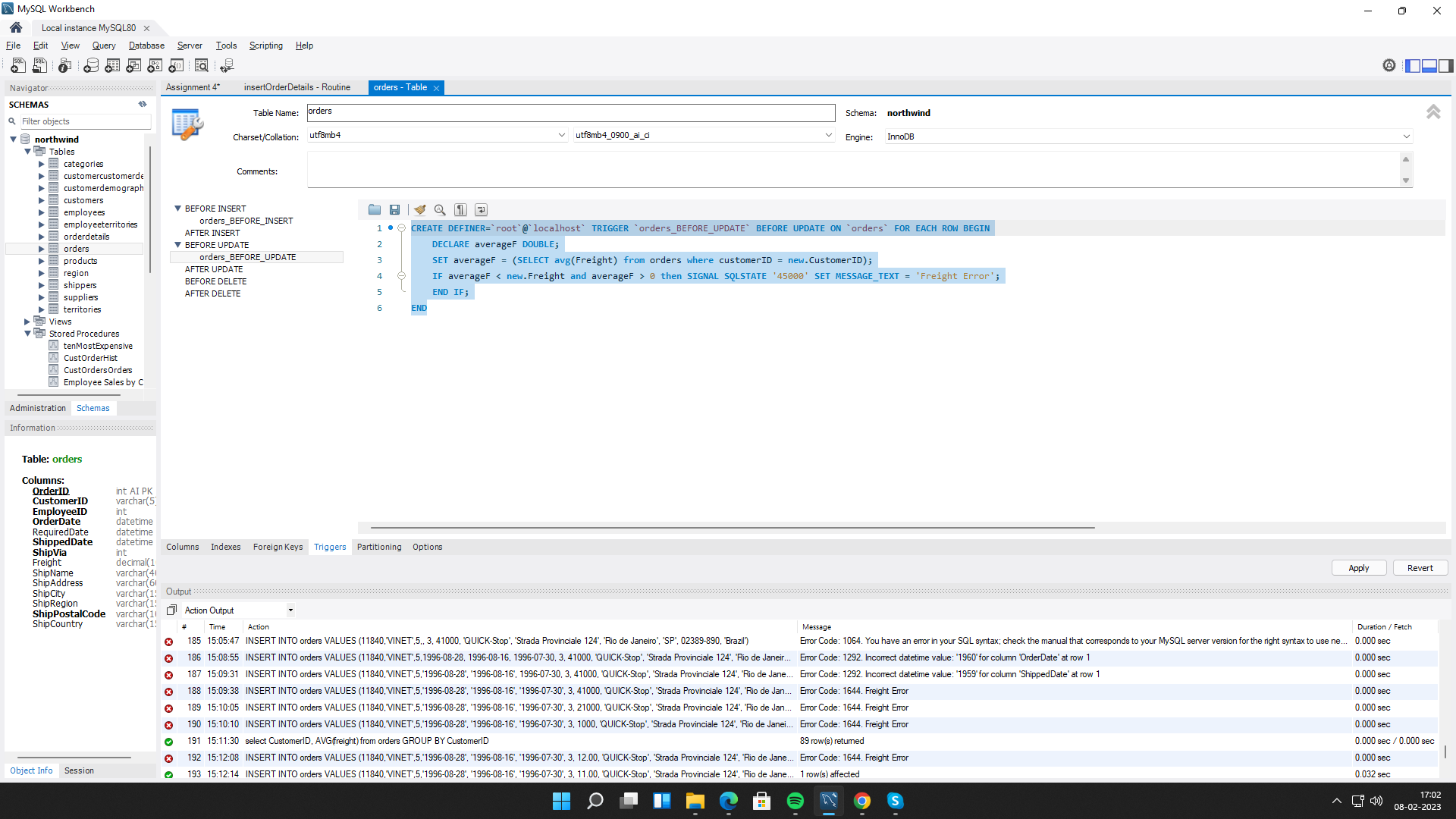
DECLARE averageF DOUBLE;

SET averageF = (SELECT avg(Freight) from orders where customerID = new.CustomerID);

IF averageF < new.Freight and averageF > 0 then SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Freight Error';

END IF;

END



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

CREATE DEFINER=`root`@`localhost` PROCEDURE `salesByCountry`()

BEGIN

select orderDet.\*, e.firstName, e.LastName from employees e

JOIN (select o.employeeID,o.ShipCountry, sum(od.UnitPrice \* od.Quantity) sales

from orders o INNER JOIN orderDetails od

ON o.orderid = od.orderId

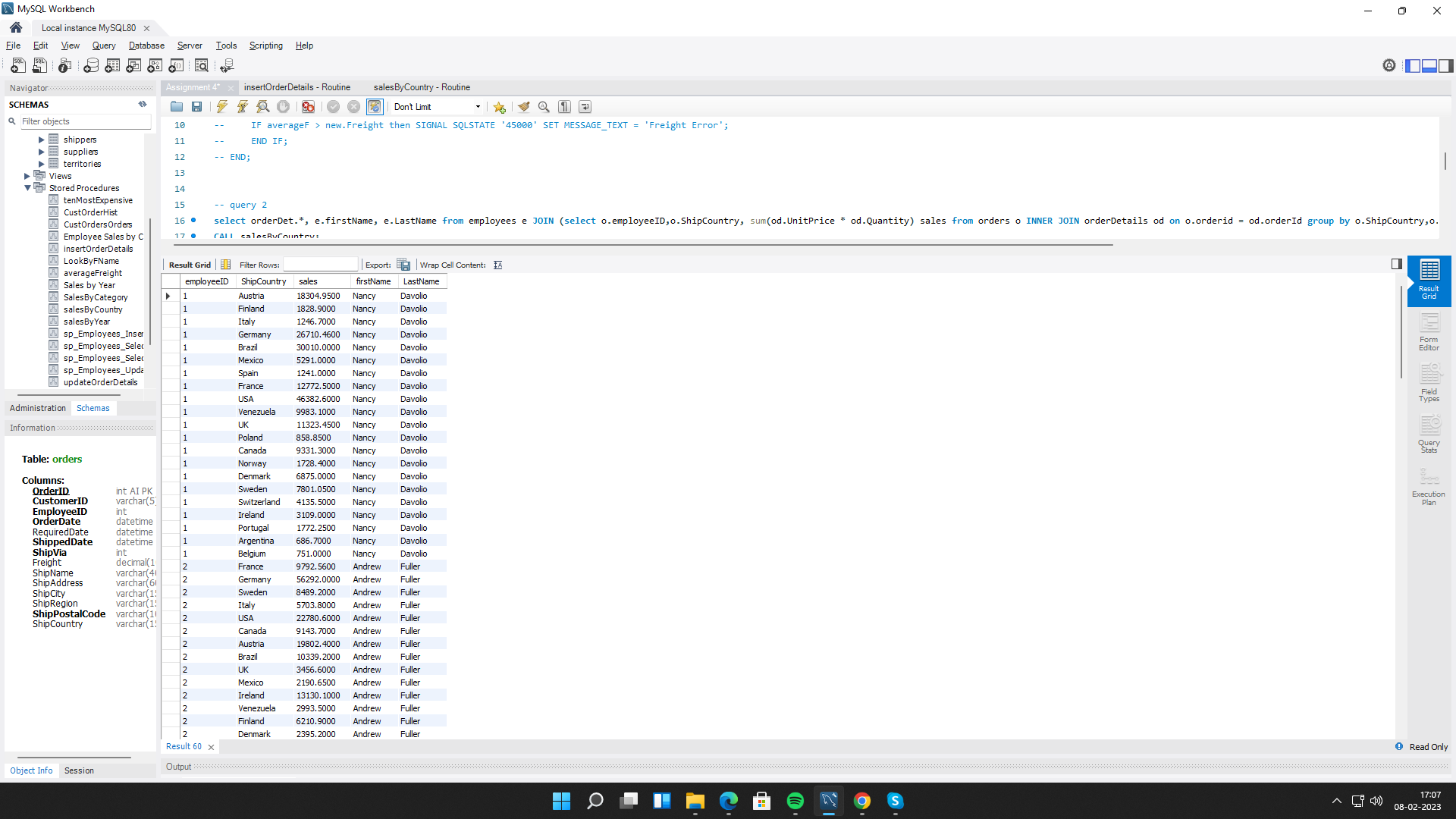
group by o.ShipCountry,o.EmployeeID

)

AS orderDet

ON e.employeeID = orderDet.employeeID;

END



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

CREATE DEFINER=`root`@`localhost` PROCEDURE `salesByYear`()

BEGIN

select YEAR(o.orderdate) year, SUM(od.UnitPrice \* od.Quantity) sales

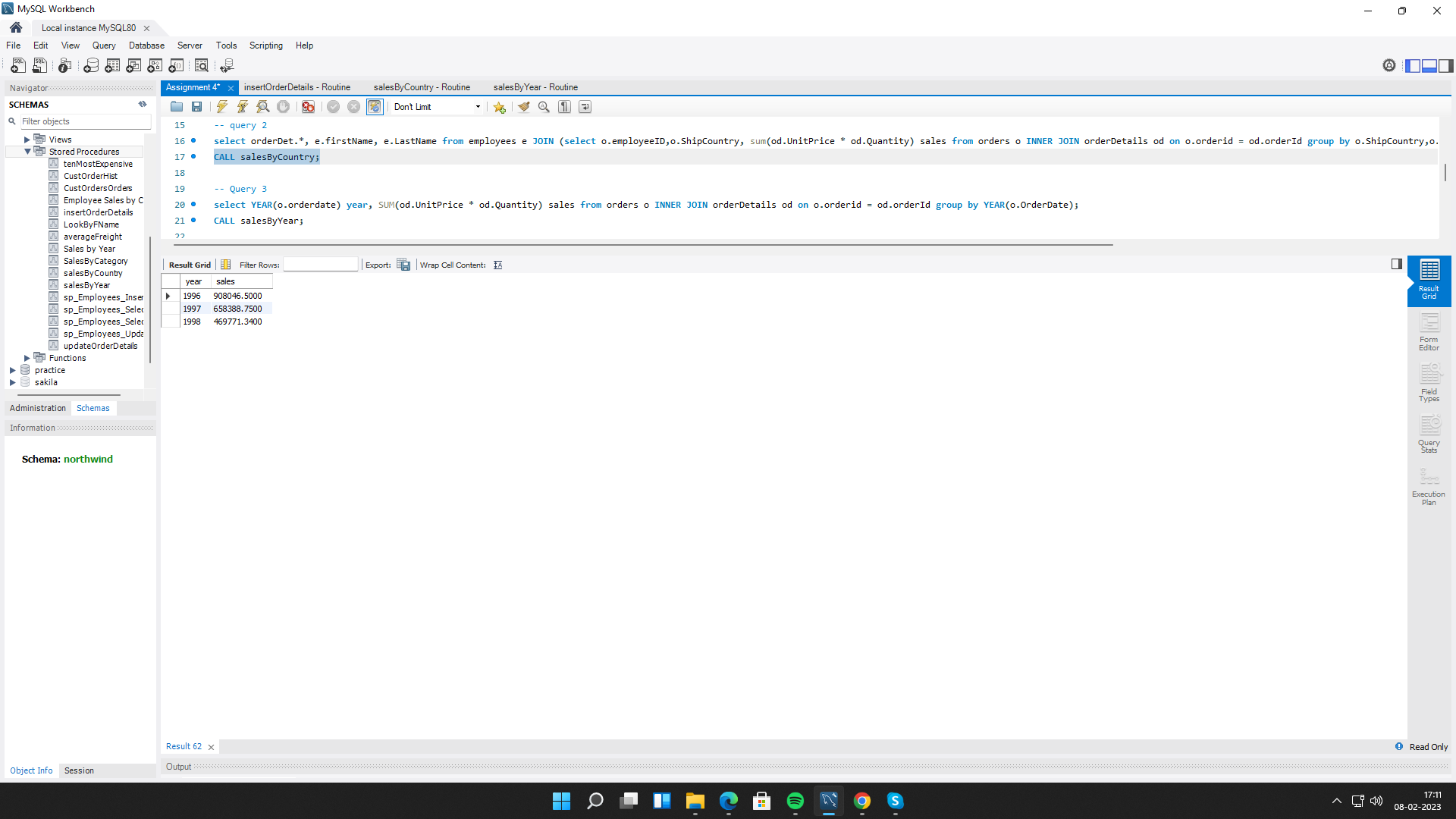
from orders o INNER JOIN orderDetails od

ON o.orderid = od.orderId

group by YEAR(o.OrderDate);

END

CALL salesByCountry;



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

CREATE DEFINER=`root`@`localhost` PROCEDURE `SalesByCategory`()

BEGIN

select p.CategoryID, SUM(od.unitprice\* od.Quantity) as sales

from products p

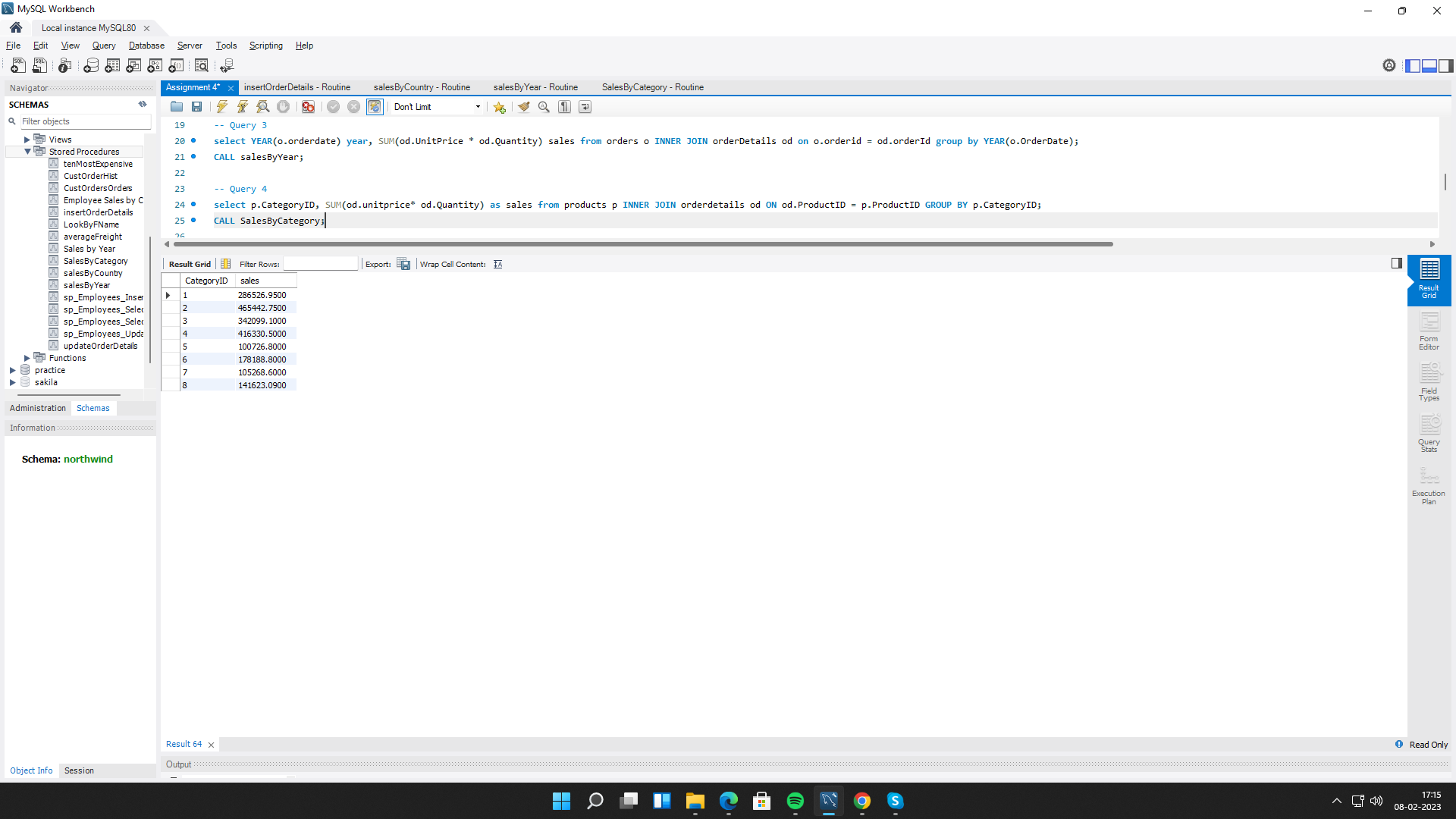
INNER JOIN orderdetails od

ON od.ProductID = p.ProductID

GROUP BY p.CategoryID;

END

CALL SalesByCategory;



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

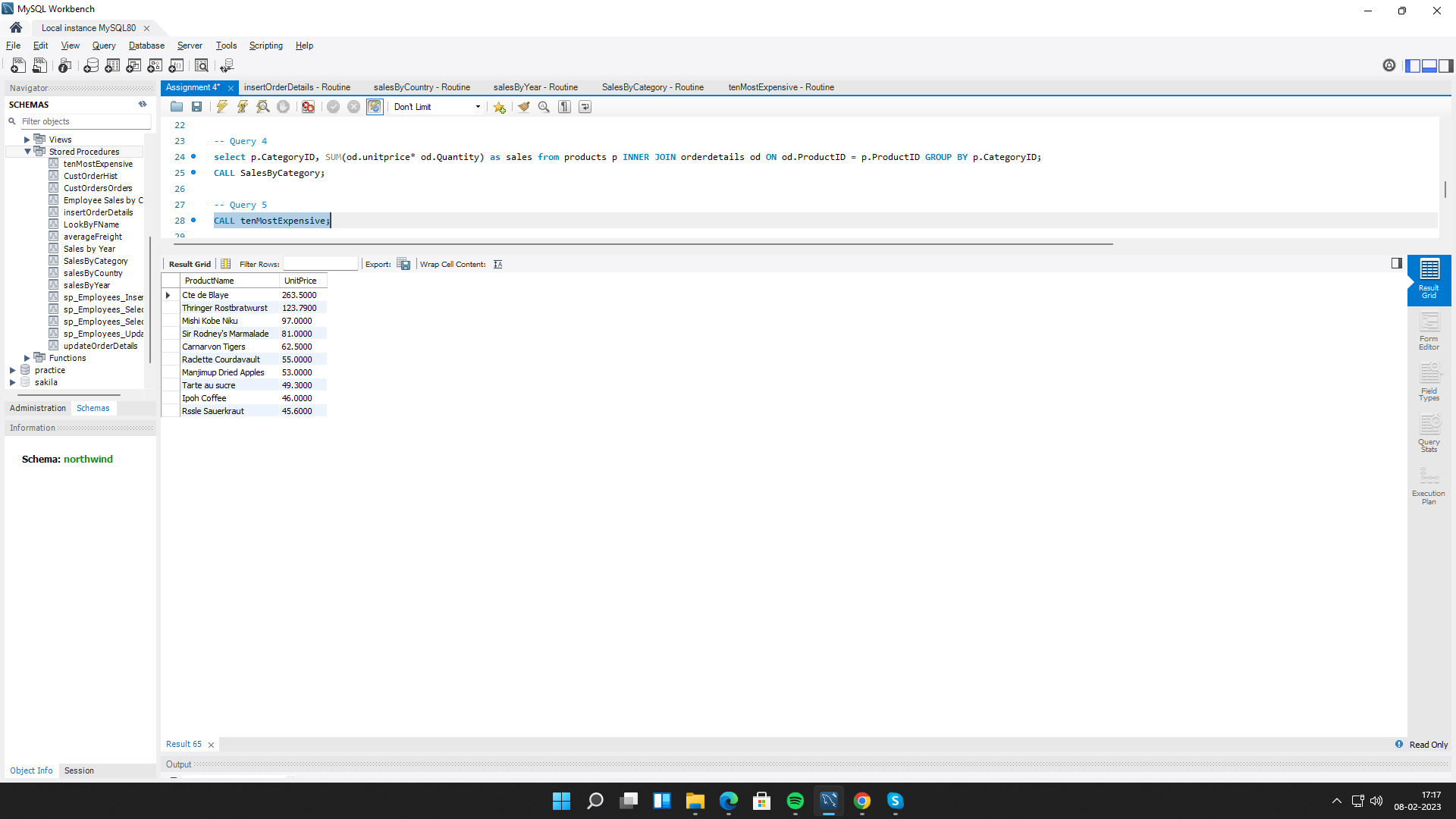
CREATE DEFINER=`root`@`localhost` PROCEDURE `tenMostExpensive`()

BEGIN

select ProductName, UnitPrice from products order by UnitPrice desc limit 10;

END

CALL tenMostExpensive;



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

**Insert:**

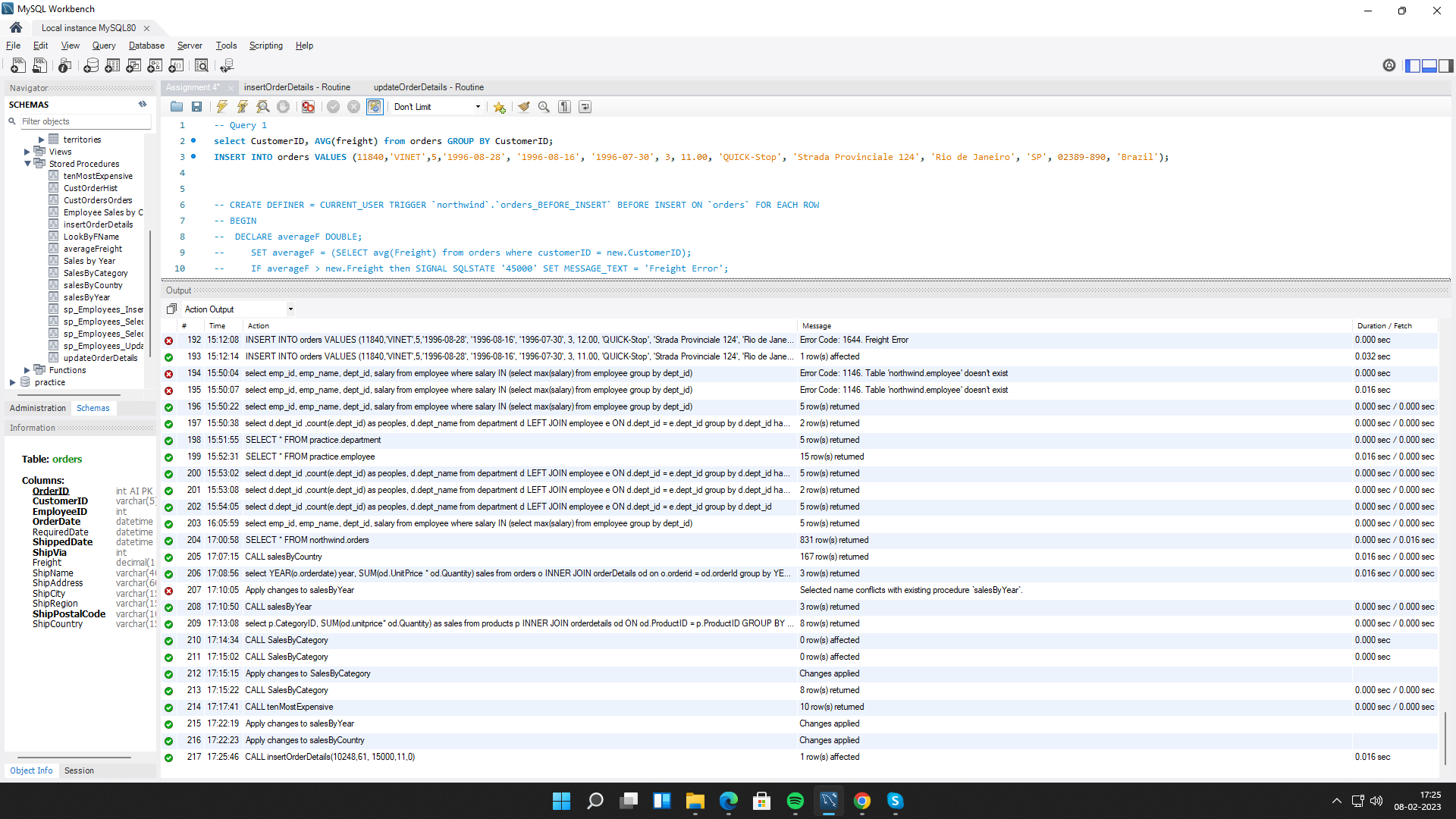
CREATE DEFINER=`root`@`localhost` PROCEDURE `insertOrderDetails`(IN odrid INT,IN proid INT, IN uprice DECIMAL(10,4), IN quan SMALLINT, IN dis DOUBLE(8,0))

BEGIN

insert into orderdetails values (odrid,proid, uprice, quan, dis);

END

CALL insertOrderDetails(10248,60, 15000,11,0);



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

**Update:**

CREATE DEFINER=`root`@`localhost` PROCEDURE `updateOrderDetails`(

In ordid INT,

In proid INT,

In uprice DECIMAL(10,4),

In quan SMALLINT,

In dis DOUBLE(8,0)

)

BEGIN

update orderdetails SET UnitPrice = uprice, Quantity = quan, Discount = dis where OrderID = ordid AND ProductID = proid;

END

CALL updateOrderDetails(10250,65,16000,22,0);

