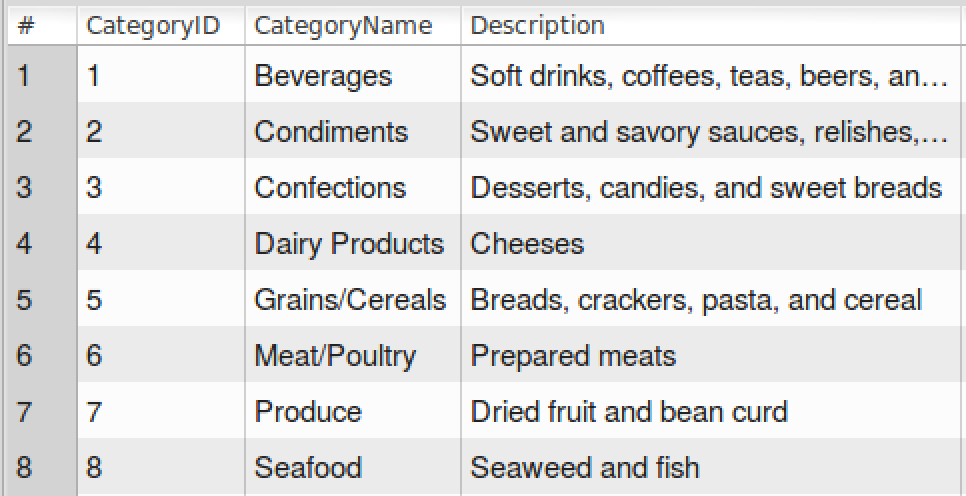
CSCI3287 Spring 2018 HW2

Chen Hao Cheng(Johnny)

1. SELECT \* FROM nwCategories;



1. SELECT CompanyName, Address, City, Region, PostalCode, Phone

FROM nwSuppliers

WHERE Country = 'France'

OR Country = 'Germany';



1. SELECT ContactName, ContactTitle

FROM nwSuppliers

WHERE SupplierID

BETWEEN 5 AND 20

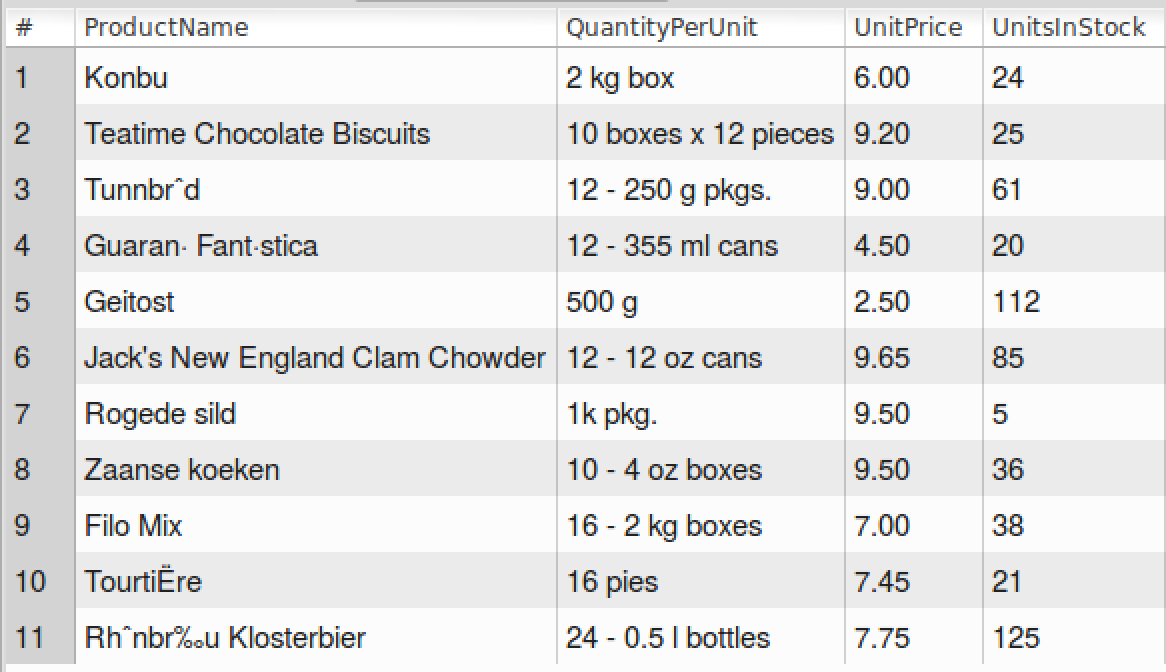
ORDER BY ContactName;



1. SELECT ProductName, QuantityPerUnit, UnitPrice, UnitsInStock

FROM nwProducts

WHERE UnitPrice < 10.00;

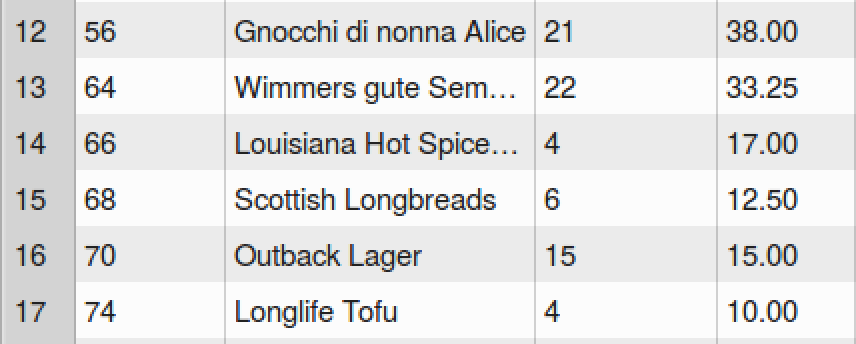
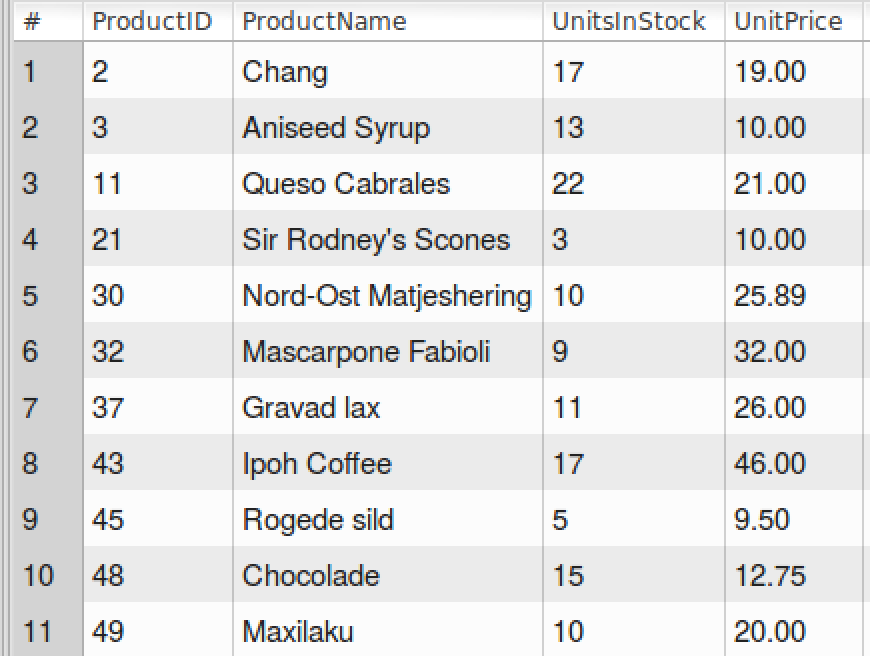


1. SELECT ProductID, ProductName, UnitsInStock, UnitPrice

FROM nwProducts

WHERE UnitsInStock > 0

AND UnitsInStock <= ReorderLevel



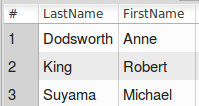
1. SELECT LastName, FirstName

FROM nwEmployees

WHERE DATEDIFF(CURDATE(), HireDate) / 365 > 5

AND Country != 'USA'

ORDER BY LastName;



1. SELECT ProductName, UnitPrice

FROM nwProducts

WHERE UnitPrice = (

SELECT MAX(UnitPrice)

FROM nwProducts );



1. SELECT ProductID, ProductName, (UnitsInStock \* UnitPrice) as 'Total Inventory Value'

FROM nwProducts

WHERE (UnitsInStock \* UnitPrice) > 2000

ORDER BY 'Total Inventory Value' DESC;

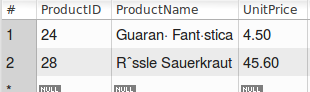


1. SELECT ProductID, ProductName, UnitPrice

FROM nwProducts

WHERE QuantityPerUnit

LIKE '%cans' AND Discontinued = 1;



1. SELECT distinct(ShipCountry) FROM nwOrders

WHERE ShipCountry != 'USA'

AND MONTH(ShippedDate) = 9

AND YEAR(ShippedDate) = 2013;

 Use distinct can separate the overlap country

1. SELECT AVG(UnitPrice) FROM nwProducts;



1. SELECT count(\*)

FROM nwCustomers

WHERE Country = 'France';



1. SELECT CustomerID, CompanyName

FROM nwCustomers

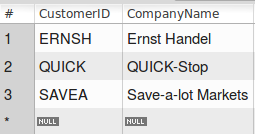
WHERE CustomerID = ANY(

SELECT CustomerID

FROM nwOrders

GROUP BY CustomerID

HAVING count(8) > 20);



When used with a subquery, the word IN is an alias for = ANY

IN and = ANY are not synonyms when used with an expression list. IN can take an expression list, but = ANY cannot.

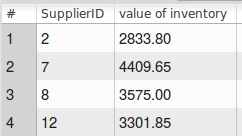
1. SELECT SupplierID, SUM(UnitsInStock \* UnitPrice)

AS 'value of inventory'

FROM nwProducts

GROUP BY SupplierID

HAVING count(\*) > 3;



1. SELECT CompanyName, ProductName, UnitPrice

FROM nwSuppliers, nwProducts

WHERE nwSuppliers.Country = 'USA'

AND nwSuppliers.SupplierID = nwProducts.SupplierID;



1. SELECT LastName, FirstName, Title, Extension

FROM nwEmployees

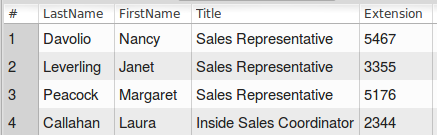
WHERE EmployeeID = ANY(

SELECT EmployeeID

FROM nwOrders

GROUP BY EmployeeID

HAVING count(\*) > 100);



1. SELECT CustomerID, CompanyName

FROM nwCustomers

WHERE CustomerID != ALL(

SELECT CustomerID

FROM nwOrders);



1. SELECT CompanyName, ContactName, CategoryName, Description, ProductName, UnitsOnOrder FROM nwProducts, nwCategories, nwSuppliers

WHERE UnitsInStock = 0

AND nwSuppliers.SupplierID = nwProducts.SupplierID

AND nwProducts.CategoryID = nwCategories.CategoryID;



1. SELECT ProductName, ContactName, Country UnitInStock

FROM nwProducts, nwSuppliers

WHERE QuantityPerUnit

LIKE '%bottle%'

AND nwProducts.SupplierID = nwSuppliers.SupplierID;



1. SELECT nwCustomers.CompanyName, nwCustomers.Country, round(sum(nwOrderDetails.Quantity \* nwOrderDetails.UnitPrice \* (1 - nwOrderDetails.discount)), 2)

AS netvalue from nwCustomers

JOIN nwOrders on nwOrders.CustomerID = nwCustomers.CustomerID

JOIN nwOrderDetails

ON nwOrderDetails.OrderID = nwOrders.OrderID

GROUP BY nwOrders.CustomerID

HAVING SUM(nwOrderDetails.Quantity \* nwOrderDetails.UnitPrice \* (1 - nwOrderDetails.discount)) $>$ 30000

ORDER BY netvalue DESC;



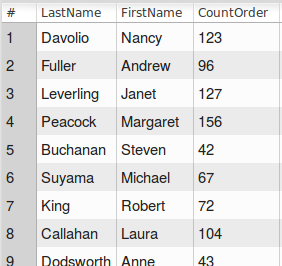
1. SELECT LastName, FirstName,

(SELECT count(nwOrders.EmployeeID)

FROM nwOrders WHERE nwEmployees.EmployeeID = nwOrders.EmployeeID )

AS CountOrder

FROM nwEmployees;



1. SELECT LastName, FirstName, (

SELECT count(nwOrders.EmployeeID)

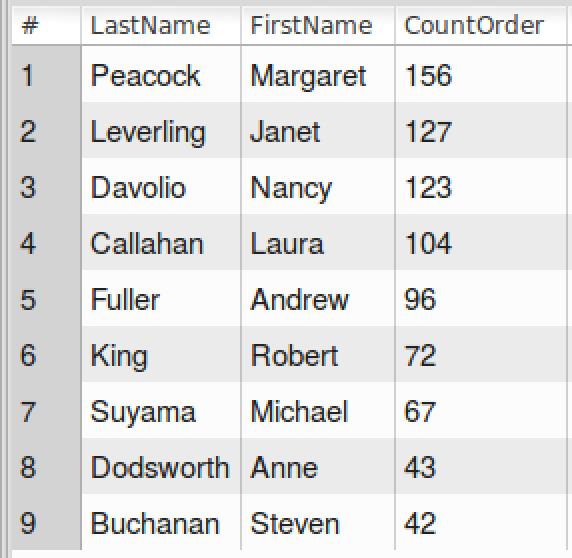
FROM nwOrders

WHERE nwEmployees.EmployeeID = nwOrders.EmployeeID)

AS CountOrder

FROM nwEmployees

ORDER BY CountOrder DESC;



1. CREATE TABLE Top\_Items (

ItemID int NOT NULL,

ItemCode int NOT NULL,

ItemName varchar(40) NOT NULL,

InventoryDate DATE NOT NULL,

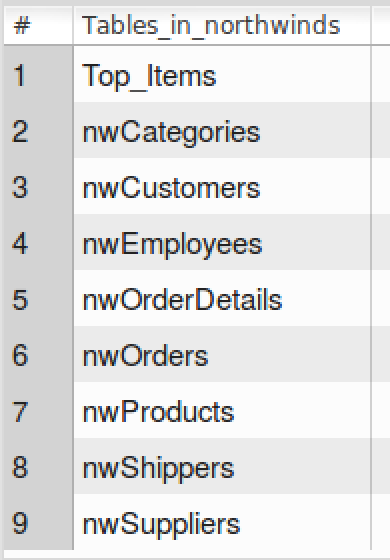
SupplierID int NOT NULL,

ItemQuantity int NOT NULL,

ItemPrice decimal(9,2) NOT NULL,

PRIMARY KEY (ItemID)

) ;

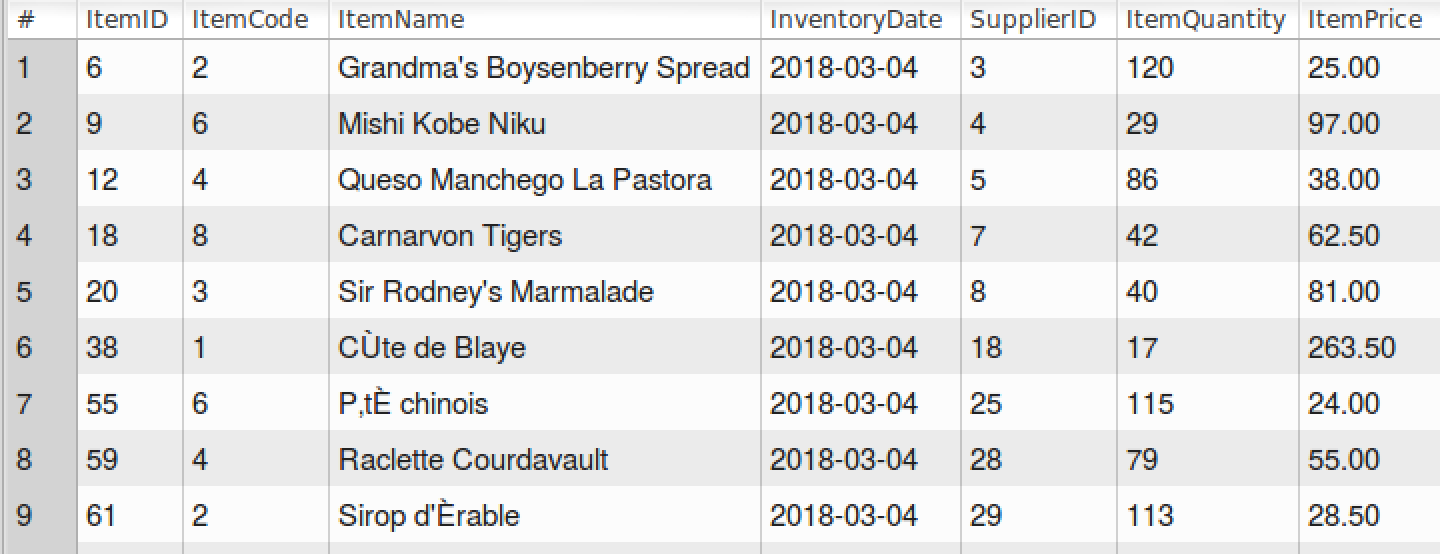


1. INSERT INTO Top\_Items(ItemID, ItemCode, ItemName, InventoryDate, ItemQuantity, ItemPrice, SupplierID)

SELECT ProductID, CategoryID, ProductName, CURDATE(), UnitsInStock, UnitPrice, SupplierID

FROM nwProducts

WHERE (UnitPrice \* UnitsInStock) > 2500;



1. DELETE FROM Top\_Items

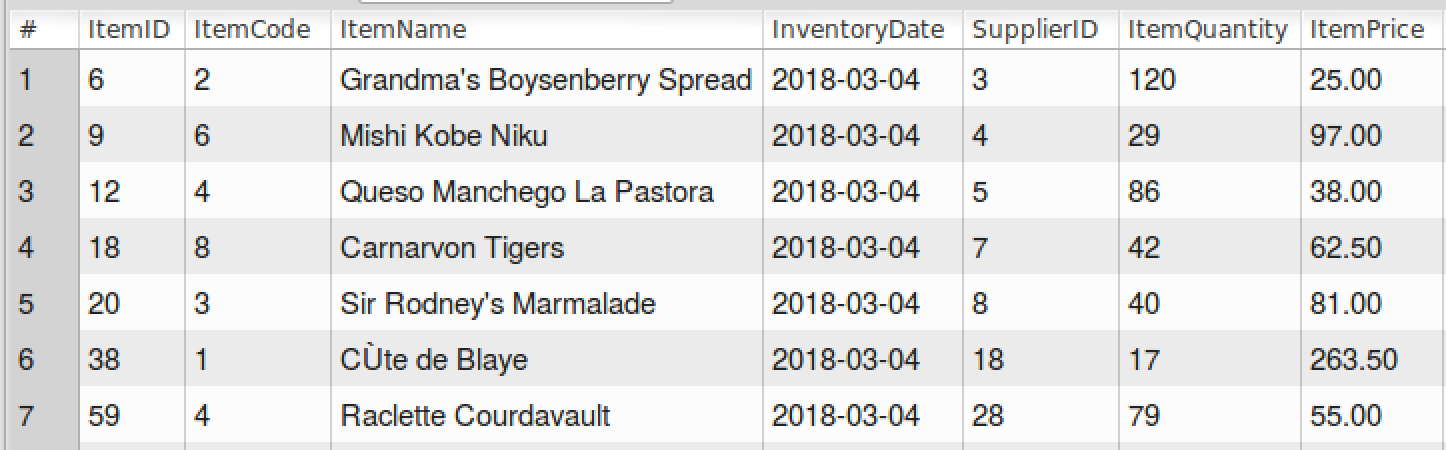
WHERE SupplierID = ANY (

SELECT SupplierID

FROM nwSuppliers

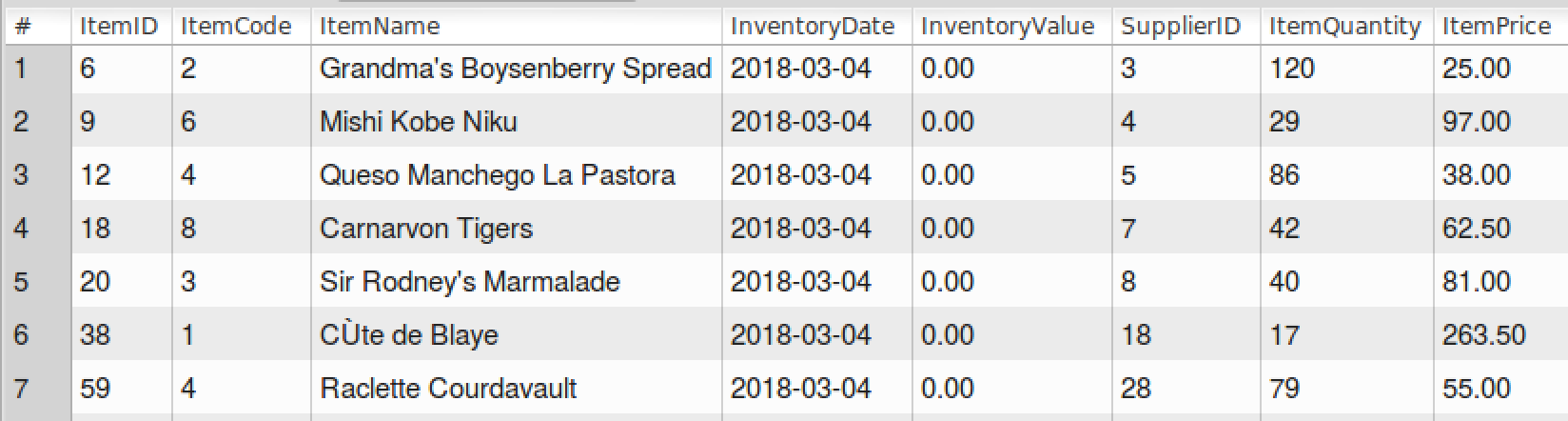
WHERE Country = 'Canada'

);



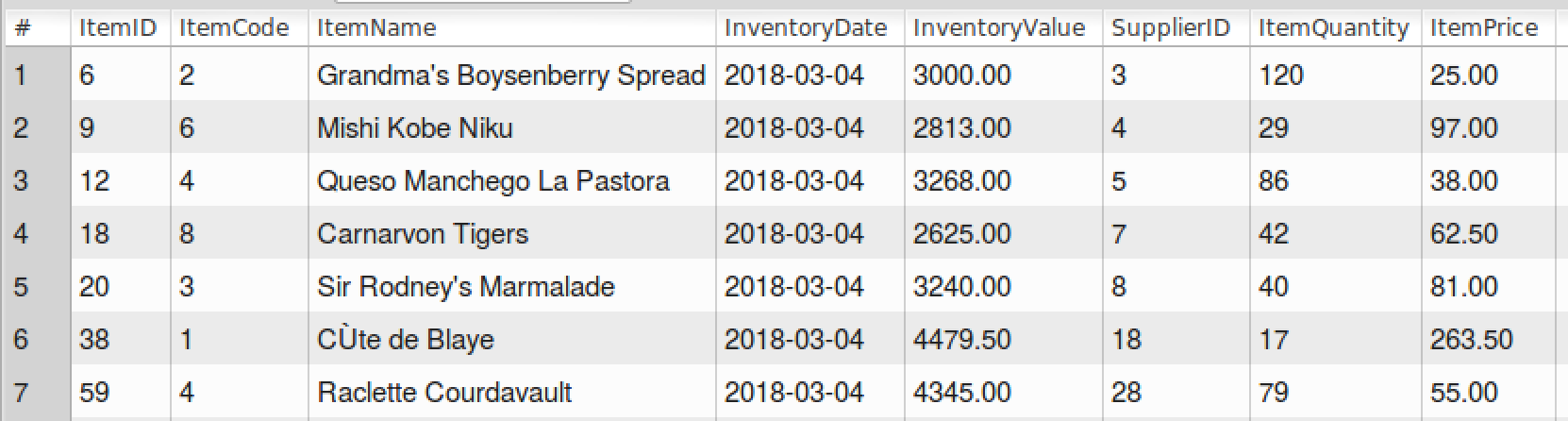
1. ALTER TABLE Top\_Items

ADD InventoryValue decimal(9,2) NOT NULL AFTER InventoryDate;

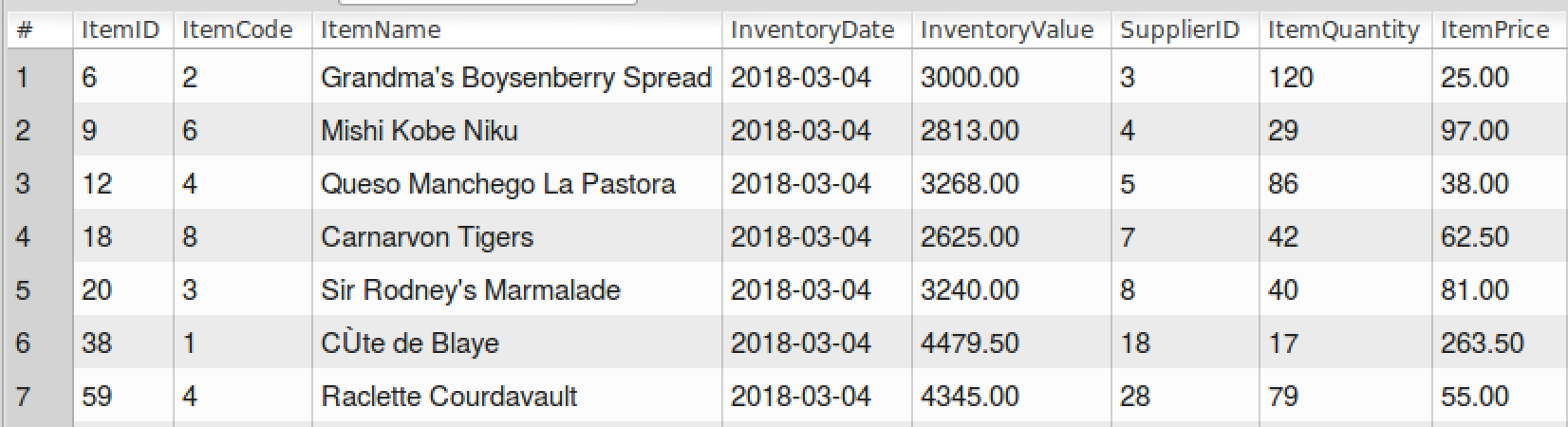


1. UPDATE Top\_Items

SET InventoryValue = (ItemPrice \* ItemQuantity);



1. SELECT \* FROM Top\_Items;



1. DROP TABLE Top\_Items;

SHOW TABLES;

