# How to Query in a DataSet with Multiple DataTables

## Introduction

We can use traditional SQL queries to get data from the data source, but we cannot use SQL to query in DataSet. In this sample, we will demonstrate how to use Expression to query in a DataSet.

In this sample, we will demonstrate how to use Expression to query in a DataSet:

1. Create a DataSet with two DataTables;

2. Create the constraints between the tables;

3. Use [DataTable.Select Method](http://msdn.microsoft.com/en-us/library/way3dy9w.aspx) to get rows from the tables;

4. Use [DataTable.Compute Method](http://msdn.microsoft.com/en-us/library/system.data.datatable.compute.aspx) to compute the specified rows;

5. Use [Expression](http://msdn.microsoft.com/en-us/library/system.data.datacolumn.expression.aspx) in the above methods to query.

## Running the Sample

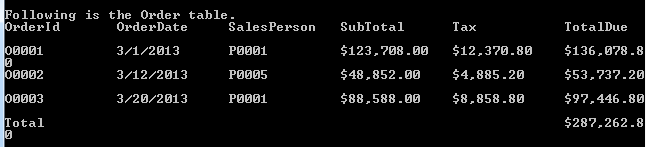
Press F5 to run the sample.

First, the application displays three tables that store data.

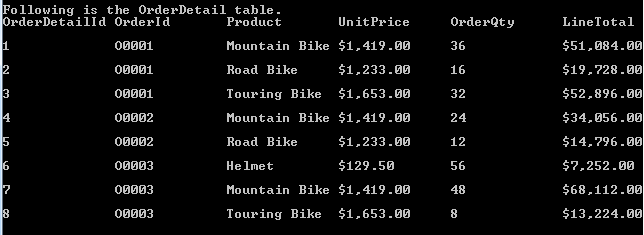
SalesPerson table:



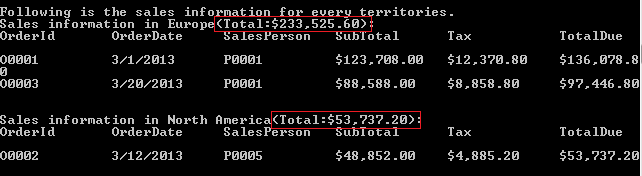
Order table:



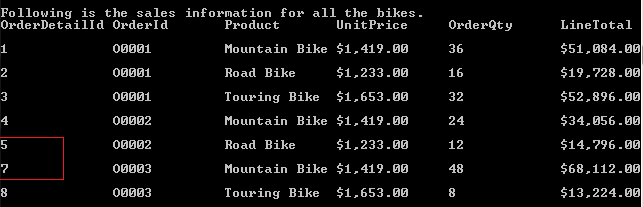
OrderDetail table:



Second, the application selects rows from Order table based on the territories in SalesPerson table and then computes the Total results:



At last, the application selects rows from OrderDetail table to get all the sales information of Bike:



The sixth row is the product Helmet, and you can’t find it now.

## Using the Code

1. Define the constraints between the tables.

|  |
| --- |
| -Code block start-  --C# code snippet start--  salesSet.Relations.Add("OrderOrderDetail",  orderTable.Columns["OrderId"], orderDetailTable.Columns["OrderId"], true);  salesSet.Relations.Add("SalesPersonOrder",  salesPersonTable.Columns["PersonId"], orderTable.Columns["SalesPerson"], true);  --C# code snippet end--  --VB code snippet start--  salesSet.Relations.Add("OrderOrderDetail", orderTable.Columns("OrderId"),  orderDetailTable.Columns("OrderId"), True)  salesSet.Relations.Add("SalesPersonOrder", salesPersonTable.Columns("PersonId"),  orderTable.Columns("SalesPerson"), True)  --VB code snippet end--  -Code block end- |

2. Create the expression columns

If you must perform an operation on two or more columns, you should create a [DataColumn](http://msdn.microsoft.com/en-us/library/system.data.datacolumn.aspx) instead of using DataTable.Compute method andset its Expression property to an appropriate expression.

|  |
| --- |
| -Code block start-  --C# code snippet start--  // Use the Aggregate-Sum on the child table column to get the result.  DataColumn colSub = new DataColumn("SubTotal", typeof(Decimal), "Sum(Child.LineTotal)");  orderTable.Columns.Add(colSub);  // Compute the tax by referencing the SubTotal expression column.  DataColumn colTax = new DataColumn("Tax", typeof(Decimal), "SubTotal\*0.1");  orderTable.Columns.Add(colTax);  // If the OrderId is 'Total', compute the due on all orders; or compute the due on this order.  DataColumn colTotal = new DataColumn("TotalDue", typeof(Decimal),  "IIF(OrderId='Total',Sum(SubTotal)+Sum(Tax),SubTotal+Tax)");  orderTable.Columns.Add(colTotal);  DataRow totalRow = orderTable.NewRow();  totalRow["OrderId"] = "Total";  orderTable.Rows.Add(totalRow);  --C# code snippet end--  --VB code snippet start--  ' Use the Aggregate-Sum on the child table column to get the result.  Dim colSub As New DataColumn("SubTotal", GetType(Decimal), "Sum(Child.LineTotal)")  orderTable.Columns.Add(colSub)  ' Compute the tax by referencing the SubTotal expression column.  Dim colTax As New DataColumn("Tax", GetType(Decimal), "SubTotal\*0.1")  orderTable.Columns.Add(colTax)  ' If the OrderId is 'Total', compute the due on all orders; or compute the due on this order.  Dim colTotal As New DataColumn("TotalDue", GetType(Decimal),  "IIF(OrderId='Total',Sum(SubTotal)+Sum(Tax),SubTotal+Tax)")  orderTable.Columns.Add(colTotal)  Dim totalRow As DataRow = orderTable.NewRow()  totalRow("OrderId") = "Total"  orderTable.Rows.Add(totalRow)  --VB code snippet end--  -Code block end- |

3. Use Select method and Compute method

Use Select method to get the rows based on the Territory in the SalesPerson Table. And also use Compute method to get the Total result.

|  |
| --- |
| -Code block start-  --C# code snippet start--  String[] territories = { "Europe", "North America"};  Console.WriteLine("Following is the sales information for every territories.");  foreach(String territory in territories)  {  String expression = String.Format("Parent.Territory='{0}'",  territory);  Object total = orderTable.Compute("Sum(TotalDue)", expression);  Console.WriteLine("Sales information in {0}(Total:{1:C}):", territory, total);  DataRow[] territoryRows = orderTable.Select(expression);  ShowRows(territoryRows);  }  --C# code snippet end--  --VB code snippet start--  Dim territories() As String = {"Europe", "North America"}  Console.WriteLine("Following is the sales information for every territories.")  For Each territory As String In territories  Dim expression As String = String.Format("Parent.Territory='{0}'", territory)  Dim total As Object = orderTable.Compute("Sum(TotalDue)", expression)  Console.WriteLine("Sales information in {0}(Total:{1:C}):", territory, total)  Dim territoryRows() As DataRow = orderTable.Select(expression)  ShowRows(territoryRows)  Next territory  --VB code snippet end--  -Code block end- |

4. Use Select method and Like

Use select method and Like to get all the Bikes’ sales information.

|  |
| --- |
| -Code block start-  --C# code snippet start--  Console.WriteLine("Following is the sales information for all the bikes.");  DataRow[] bikeRows = orderDetailTable.Select("Product like '\*Bike'");  ShowRows(bikeRows);  --C# code snippet end--  --VB code snippet start--  Console.WriteLine("Following is the sales information for all the bikes.")  Dim bikeRows() As DataRow = orderDetailTable.Select("Product like '\*Bike'")  ShowRows(bikeRows)  --VB code snippet end--  -Code block end- |

## More Information

[DataTable.Select Method (String, String)](http://msdn.microsoft.com/en-us/library/way3dy9w.aspx)

[DataTable.Compute Method](http://msdn.microsoft.com/en-us/library/system.data.datatable.compute.aspx)

[DataColumn.Expression Property](http://msdn.microsoft.com/en-us/library/system.data.datacolumn.expression.aspx)