

# 1.3.0 – LINQ Advanced

## Introduction

In lesson 1.2.0 you were introduced to creating LINQ queries in LINQPad. In this lesson, you will create some advance LINQ queries in LINQPad. These queries will return data that contains a collection of data, and other advanced LINQ functions.

## Example #1

Consider the following portion of the eStore ERD:

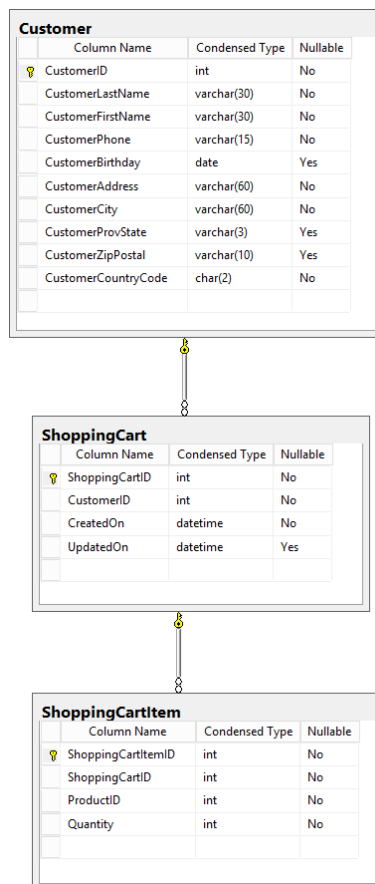


Figure 1: Customer - ShoppingCart - ShoppingCartItem

If we wanted to return the ShoppingCarts that belong to a selected Customer, we could write the following query:

Listing 1: Customer Shopping Carts (1)

```
var results = from x in Customers
               where x.CustomerID == 11
               orderby x.LastName, x.FirstName
               select new
               {
```

```

        Name = x.LastName + ", " + x.FirstName,
        Phone = x.Phone,
        Carts = x.ShoppingCarts
    };

results.Dump();

```

The result of this query is:

IOrderedQueryable<> (1 item)						
Name	Phone	Carts				
Bain, Paige	7804560007	EntitySet<ShoppingCart> (1 item)				
ShoppingCartID	CustomerID	CreatedOn	UpdatedOn	Customer	ShoppingCartItems	
4	11	4/2/2017 9:34:23 AM	4/2/2017 9:44:50 AM	Customer	ShoppingCartItems	

Figure 2: Customer Shopping Carts Output (1)

The output displays an EntitySet<ShoppingCart>, which has [ShoppingCartItems](#). If we expand this, we see:

IOrderedQueryable<> (1 item)						
Name	Phone	Carts				
Bain, Paige	7804560007	EntitySet<ShoppingCart> (1 item)				
ShoppingCartID	CustomerID	CreatedOn	UpdatedOn	Customer	ShoppingCartItems	
4	11	4/2/2017 9:34:23 AM	4/2/2017 9:44:50 AM	Customer	EntitySet<ShoppingCartItem> (4 items)	
ShoppingCartItemID	ShoppingCartID	ProductID	Quantity	Product	ShoppingCart	
9	4	58	1	Product	ShoppingCart	
10	4	62	1	Product	ShoppingCart	
11	4	63	1	Product	ShoppingCart	
12	4	61	1	Product	ShoppingCart	
				4		

Figure 3: Expanded ShoppingCartItems

Also included in the output is [Customer](#), which we do not need. We will need [Product](#) in some part of the query.

The results shown are a result of LINQPad using the navigation properties, which are created from the Primary Key – Foreign Key relationships between the database tables.

## Example #2

Let us refine the query from the previous example to show just the relevant information. Looking at the output below, what would be the query?

^ IOrderedQueryable<> (1 item)																																			
Name	Phone	Carts																																	
Bain, Paige	7804560007	^ (1 item) <table> <tr> <th>Created</th><th>Updated</th><th colspan="2">Items</th></tr> <tr> <td>4/2/2017 9:34:23 AM</td><td>4/2/2017 9:44:50 AM</td><td colspan="2">               ^ (4 items)               <table> <tr> <th>SKU</th><th>Name</th><th>Price</th><th>Qty</th></tr> <tr> <td>C-LTide-02</td><td>Tide</td><td>7.15</td><td>1</td></tr> <tr> <td>C-Bleac-01</td><td>Bleach</td><td>2.45</td><td>1</td></tr> <tr> <td>C-Scrbs-01</td><td>Scrubbies</td><td>4.10</td><td>1</td></tr> <tr> <td>C-DsDwn-01</td><td>Dawn</td><td>6.10</td><td>1</td></tr> <tr> <td></td><td></td><td>19.80</td><td>4</td></tr> </table> </td></tr> </table>		Created	Updated	Items		4/2/2017 9:34:23 AM	4/2/2017 9:44:50 AM	^ (4 items) <table> <tr> <th>SKU</th><th>Name</th><th>Price</th><th>Qty</th></tr> <tr> <td>C-LTide-02</td><td>Tide</td><td>7.15</td><td>1</td></tr> <tr> <td>C-Bleac-01</td><td>Bleach</td><td>2.45</td><td>1</td></tr> <tr> <td>C-Scrbs-01</td><td>Scrubbies</td><td>4.10</td><td>1</td></tr> <tr> <td>C-DsDwn-01</td><td>Dawn</td><td>6.10</td><td>1</td></tr> <tr> <td></td><td></td><td>19.80</td><td>4</td></tr> </table>		SKU	Name	Price	Qty	C-LTide-02	Tide	7.15	1	C-Bleac-01	Bleach	2.45	1	C-Scrbs-01	Scrubbies	4.10	1	C-DsDwn-01	Dawn	6.10	1			19.80	4
Created	Updated	Items																																	
4/2/2017 9:34:23 AM	4/2/2017 9:44:50 AM	^ (4 items) <table> <tr> <th>SKU</th><th>Name</th><th>Price</th><th>Qty</th></tr> <tr> <td>C-LTide-02</td><td>Tide</td><td>7.15</td><td>1</td></tr> <tr> <td>C-Bleac-01</td><td>Bleach</td><td>2.45</td><td>1</td></tr> <tr> <td>C-Scrbs-01</td><td>Scrubbies</td><td>4.10</td><td>1</td></tr> <tr> <td>C-DsDwn-01</td><td>Dawn</td><td>6.10</td><td>1</td></tr> <tr> <td></td><td></td><td>19.80</td><td>4</td></tr> </table>		SKU	Name	Price	Qty	C-LTide-02	Tide	7.15	1	C-Bleac-01	Bleach	2.45	1	C-Scrbs-01	Scrubbies	4.10	1	C-DsDwn-01	Dawn	6.10	1			19.80	4								
SKU	Name	Price	Qty																																
C-LTide-02	Tide	7.15	1																																
C-Bleac-01	Bleach	2.45	1																																
C-Scrbs-01	Scrubbies	4.10	1																																
C-DsDwn-01	Dawn	6.10	1																																
		19.80	4																																

Figure 4: Example #2 Output

The query has two nested queries; one query for the ShoppingCart for the Customer, and the second for the ShoppingCartItems that are in the ShoppingCart:

Listing 2: Customer Shopping Carts (2)

```
var results = from x in Customers
              where x.CustomerID == 11
              orderby x.LastName, x.FirstName
              select new
              {
                  Name = x.LastName + ", " + x.FirstName,
                  Phone = x.Phone,
                  Carts = from y in x.ShoppingCarts
                          orderby y.UpdatedOn
                          select new
                          {
                              Created = y.CreatedOn,
                              Updated = y.UpdatedOn,
                              Items = from z in y.ShoppingCartItems
                                      select new
                                      {
                                          SKU = z.Product.SKU,
                                          Name = z.Product.ProductName,
                                          Price = z.Product.SellingPrice,
                                          Qty = z.Quantity
                                      }
                              }
              };
results.Dump();
```

Nested LINQ Query