Override ToString() in your Classes

One of the reasons I love teaching is because of the questions that I get from attendees. I was giving a presentation at DevConnections and was showing a collection of Product objects. When I hovered over the variable that contained the collection, it looked like Figure 2. As you can see in the collection, I have actual product names of my videos from www.pdsa.com/videos being displayed. To get your data to appear in the data tips you must override the ToString() method in your class.

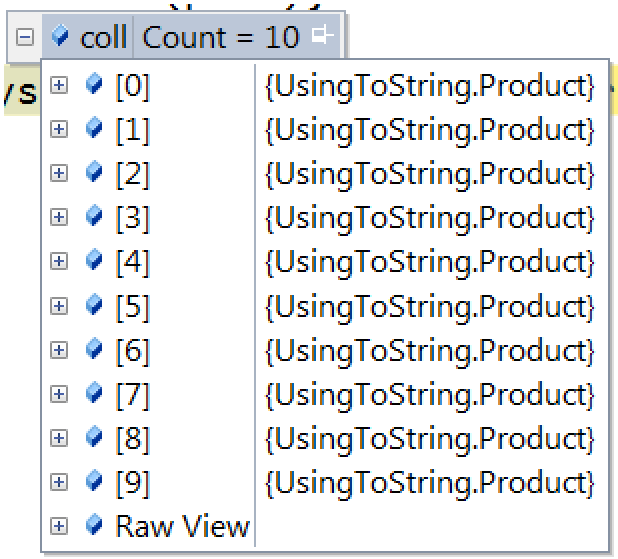
To illustrate this, take the following simple Product class shown below:

```
public class Product
{
  public string ProductName { get; set; }
  public int ProductId { get; set; }
}
```

This class does not have an override of the ToString() method so if you create a collection of Product objects you will end up with data tips that look like Figure 1. Below is the code I used to create a collection of Product objects. I have shortened the code in this blog, but you can get the full source code for this sample by following the instructions at the bottom of this blog entry.

```
List<Product> coll = new List<Product>();
Product prod;

prod = new Product()
  { ProductName = "From Zero to HTML 5 in 60 Minutes",
    ProductId = 1 };
coll.Add(prod);
prod = new Product()
  { ProductName = "Architecting Applications …",
    ProductId = 2 };
coll.Add(prod);
prod = new Product()
  { ProductName = "Introduction to Windows Phone Development",
    ProductId = 3 };
coll.Add(prod);
prod = new Product()
  { ProductName = "Architecting a Business  …",
    ProductId = 4 };
coll.Add(prod);
...
...
```



```
coll Count = 10
  [0]        {UsingToString.Product}
  [1]        {UsingToString.Product}
  [2]        {UsingToString.Product}
  [3]        {UsingToString.Product}
  [4]        {UsingToString.Product}
  [5]        {UsingToString.Product}
  [6]        {UsingToString.Product}
  [7]        {UsingToString.Product}
  [8]        {UsingToString.Product}
  [9]        {UsingToString.Product}
  Raw View
```

Figure 1: Class without overriding ToString()

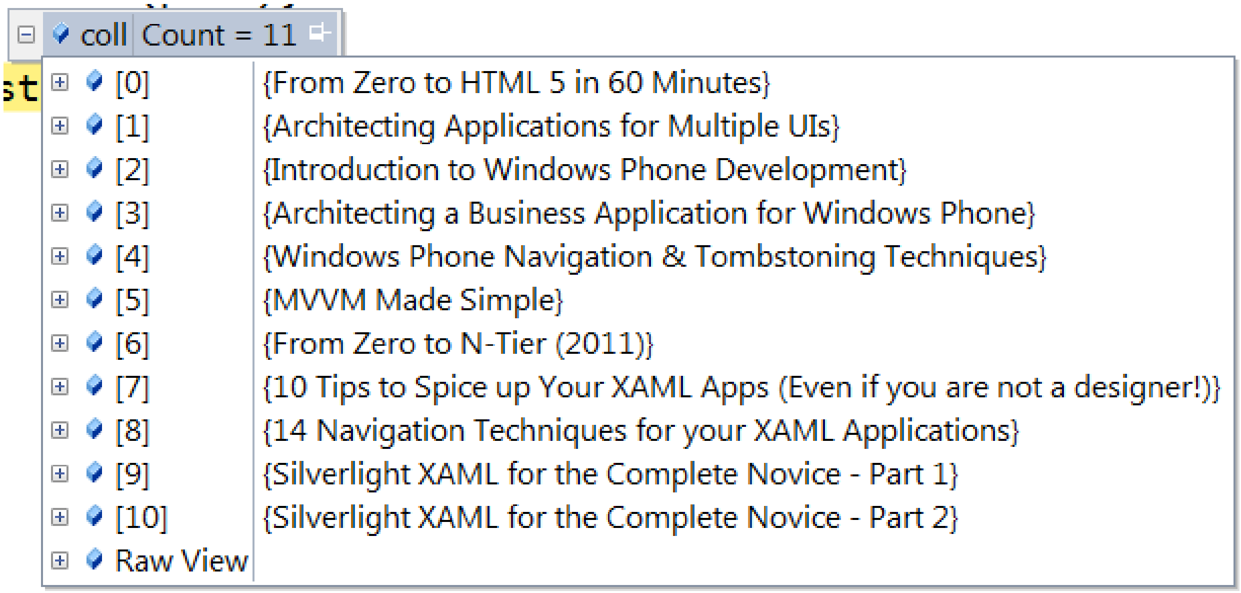
Now, go back to the Product class and add an override of the ToString() method as shown in the code listed below:

```
public class Product
{
  public string ProductName { get; set; }
  public int ProductId { get; set; }

  public override string ToString()
  {
    return ProductName;
  }
}
```

In this simple sample, I am just returning the ProductName property. However, you can create a whole string of information if you wish to display more data in your data tips. Just concatenate any properties you want from your class and return that string.

When you now run the application and hover over the collection object you will now see something that looks like Figure 2.



```
coll  Count = 11
[0]        {From Zero to HTML 5 in 60 Minutes}
[1]        {Architecting Applications for Multiple UIs}
[2]        {Introduction to Windows Phone Development}
[3]        {Architecting a Business Application for Windows Phone}
[4]        {Windows Phone Navigation & Tombstoning Techniques}
[5]        {MVVM Made Simple}
[6]        {From Zero to N-Tier (2011)}
[7]        {10 Tips to Spice up Your XAML Apps (Even if you are not a designer!)}
[8]        {14 Navigation Techniques for your XAML Applications}
[9]        {Silverlight XAML for the Complete Novice - Part 1}
[10]       {Silverlight XAML for the Complete Novice - Part 2}
Raw View
```

Figure 2: Overriding ToString in Your Class

Another place the ToString() override comes in handy is if you forget to use a DisplayMemberPath in your ListBox or ComboBox. The ToString() method is called automatically when a class is bound to a list control.

Summary

You should always override the ToString() method in your classes as this will help you when debugging your application. Seeing relevant data immediately in the data tip without having to drill down one more layer and maybe scroll through a complete list of properties should help speed up your development process.

NOTE: You can download the sample code for this article by visiting my website at http://www.pdsa.com/downloads. Select “Tips & Tricks”, then select “Override ToString” from the drop down list.