This new route will simply fire the Show action in the customer controller.

Now we create the controller class, `CustomerController`, as:

```
using NTier.BL;
using NTier.Common;

namespace Chapter05.MVC.Controllers
{
    public class CustomerController:Controller
    {
        #region ViewData
        /// <summary>
        /// Class used for transferring data to the View
        /// </summary>
        public class CustomerViewData
        {
            public CustomerViewData() { }

            public CustomerViewData(Collection<Customer> customers)
            {
                this.customers = customers;
            }

            public Collection<Customer> customers;

            public Customer customer;

        }
        #endregion

        public ActionResult Show()
        {
            CustomerViewData customerViewData = new CustomerViewData();
            CustomerCollection customers = new CustomerCollection();
            customerViewData.customers
            = customers.FindAll(LoadStatus.Loaded);

            return View("Show", customerViewData);
        }
    }//end class
}//end namespace
```

In this class, we first create a subclass called `CustomerViewData`, which is a wrapper to hold the `Customer` and `CustomerCollection` business objects. We will transfer this class object to the actual view through the controller as:

```
public ActionResult Show()
        {
                CustomerViewData customerViewData = new CustomerViewData();
                CustomerCollection customers = new CustomerCollection();
                customerViewData.customers
                = customers.FindAll(LoadStatus.Loaded);

                return View("Show", customerViewData);
        }
```

In this controller action, we simply get the list of customers from the database and pass it to the `Show.aspx` page using the `View()` method.

To create the `aspx` pages, we will create a folder named `Customers` under the `View`, and add an `aspx` file named `Show.aspx` that will display a list of all the customers.

```
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent"
runat="server">
    <asp:Repeater ID="rptCustomer" runat="server">
        <ItemTemplate>
            <%# Eval("Name")%>
        </ItemTemplate>
    </asp:Repeater>
</asp:Content>
```

In the code-behind, we simply bind the data as:

```
using System.Web.Mvc;
namespace Chapter05.MVC.Views.Customer
    {
        public partial class Show :
         ViewPage <Controllers.CustomerController.CustomerViewData>
        {
            protected void Page_Load(object sender, EventArgs e)
            {
            rptCustomer.DataSource = ViewData.Model.customers;
            rptCustomer.DataBind();
            }
        }
    }
```