```csharp
1  /*-----------------------------------------------------------------------------------------
2   * Author:      Dan Cassidy
3   * Date:        2015-06-23
4   * Assignment:  cView-P4
5   * Source File: Delete.aspx.cs
6   * Language:    C#
7   * Course:      CSCI-C 490, C# Programming, MoWe 08:00
8   * Purpose:     Code-behind file for Delete.aspx.  Controls the deletion of a single row of data (1
9   *              object) from a table.
10  -----------------------------------------------------------------------------------------*/
11
12 using System;
13 using System.Collections.Generic;
14 using System.Linq;
15 using System.Web;
16 using System.Web.UI;
17 using System.Web.UI.WebControls;
18
19 namespace cView_P4_DanCassidy
20 {
21     public partial class Delete : System.Web.UI.Page
22     {
23         /*-----------------------------------------------------------------------------------------
24          * Name:    btnDelete_Click
25          * Type:    Event Handler Method
26          * Purpose: Handles deleting an object from the database.
27          * Input:   object sender, holds a reference to the object that raised this event.
28          * Input:   EventArgs e, holds data related to this event.
29          * Output:  Nothing.
30          -----------------------------------------------------------------------------------------*/
31         protected void btnDelete_Click(object sender, EventArgs e)
32         {
33             // Dummy polymorphoc object.
34             object keyToDelete = null;
35
36             // Attempt to delete a record.
37             try
38             {
39                 using (CViewDataEntities database = new CViewDataEntities())
40                 {
41                     switch ((Global.Enums.ItemTypes)ddlItemType.SelectedIndex)
42                     {
43                         case Global.Enums.ItemTypes.Business:
44                             if (gViewBusiness.SelectedIndex == -1)
45                                 return;
46                             keyToDelete = gViewBusiness.SelectedDataKey.Value;
47                             gViewBusiness.SelectRow(-1);
48                             Business businessToDelete = database.Businesses.Find(keyToDelete);
49                             database.Businesses.Remove(businessToDelete);
50                             break;
51
52                         case Global.Enums.ItemTypes.Park:
53                             if (gViewPark.SelectedIndex == -1)
54                                 return;
55                             keyToDelete = gViewPark.SelectedDataKey.Value;
56                             gViewPark.SelectRow(-1);
57                             Park parkToDelete = database.Parks.Find(keyToDelete);
58                             database.Parks.Remove(parkToDelete);
59                             break;
60
61                         case Global.Enums.ItemTypes.PublicFacility:
62                             if (gViewPublicFacility.SelectedIndex == -1)
63                                 return;
64                             keyToDelete = gViewPublicFacility.SelectedDataKey.Value;
65                             gViewPublicFacility.SelectRow(-1);
66                             PublicFacility publicFacilityToDelete =
```

```
 67                             database.PublicFacilities.Find(keyToDelete);
 68                             database.PublicFacilities.Remove(publicFacilityToDelete);
 69                             break;

 71                         default:
 72                             // ... How?
 73                             return;
 74                     }

 76                     database.SaveChanges();
 77                     mViewData.DataBind();
 78                 }

 80                 lblResult.Text = "Row successfully deleted.";
 81                 lblResult.Visible = true;
 82                 lblError.Visible = false;
 83             }
 84             catch
 85             {
 86                 lblError.Text = "Error: Could not delete the specified row.";
 87                 lblError.Visible = true;
 88                 lblResult.Visible = false;
 89             }
 90         }

 92         /*-------------------------------------------------------------------------------------
 93          * Name:    ddlItemType_SelectedIndexChanged
 94          * Type:    Event Handler Method
 95          * Purpose: Handles showing and hiding the various controls to allow a user to choose an
 96          *          item to be deleted from the database.
 97          * Input:   object sender, holds a reference to the object that raised this event.
 98          * Input:   EventArgs e, holds data related to this event.
 99          * Output:  Nothing.
100          -------------------------------------------------------------------------------------*/
101         protected void ddlItemType_SelectedIndexChanged(object sender, EventArgs e)
102         {
103             // Hide things until needed.
104             lblError.Visible = false;
105             lblResult.Visible = false;

107             mViewData.ActiveViewIndex = -1;
108             btnDelete.Visible = false;

110             // Decide which controls should be shown to enable user to delete an item.
111             switch ((Global.Enums.ItemTypes)ddlItemType.SelectedIndex)
112             {
113                 case Global.Enums.ItemTypes.Business:
114                     mViewData.ActiveViewIndex = 0;
115                     btnDelete.Visible = true;
116                     break;

118                 case Global.Enums.ItemTypes.Park:
119                     mViewData.ActiveViewIndex = 1;
120                     btnDelete.Visible = true;
121                     break;

123                 case Global.Enums.ItemTypes.PublicFacility:
124                     mViewData.ActiveViewIndex = 2;
125                     btnDelete.Visible = true;
126                     break;

128                 default:
129                     break;
130             }
131         }
132
```

```
133     }
134 }
```