```csharp
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using System.Windows;
7 using System.Windows.Controls;
8 using System.Windows.Data;
9 using System.Windows.Documents;
10 using System.Windows.Input;
11 using System.Windows.Media;
12 using System.Windows.Media.Imaging;
13 using System.Windows.Navigation;
14 using System.Windows.Shapes;
15
16 namespace QuiddichV2._0
17 {
18     /// <summary>
19     /// Interaction logic for PlayerRoster.xaml
20     /// </summary>
21     public partial class PlayerRoster : Page
22     {
23         private List<Player> players = null;
24         private int teamID;
25
26         /// <summary>
27         /// Default constructor
28         /// </summary>
29         public PlayerRoster()
30         {
31             InitializeComponent();
32         }
33
34         /// <summary>
35         /// Customized construtor
36         /// </summary>
37         /// <param name="teamID"></param>
38         public PlayerRoster(int teamID)
39         {
40             InitializeComponent();
41             players = PlayerDB.GetPlayers(teamID);
42             this.teamID = teamID;
43             fillListOfPlayers();
44
45         }
46
47         /// <summary>
48         /// Fill the list box with players
49         /// </summary>
50         ///
51         private void fillListOfPlayers()
52         {
53             lstPlayers.Items.Clear();
54
55             foreach (Player p in players)
56             {
57                 lstPlayers.Items.Add(p.GetDisplayText("\t"));
58             }
59         }
60
61         /// <summary>
62         /// Opens a form to add a player
63         /// </summary>
64         /// <param name="sender"></param>
65         /// <param name="e"></param>
66         ///
```

```csharp
67          private void btnAddPlayer_Click(object sender, RoutedEventArgs e)
68          {
69              AddPlayer addplayer = new AddPlayer(teamID);
70              addplayer.addPlayer = true;
71              NavigationService.Navigate(addplayer);
72
73          }
74
75          /// <summary>
76          /// Opens a form to modify a player
77          /// </summary>
78          /// <param name="sender"></param>
79          /// <param name="e"></param>
80          ///
81          private void btnModifyPlayer_Click(object sender, RoutedEventArgs e)
82          {
83              if(playerSelected())
84              {
85                  int i = lstPlayers.SelectedIndex;
86
87                  AddPlayer addplayer = new AddPlayer(players[i].TeamID,
88                                                      players[i].FirstName,
89                                                      players[i].LastName);
90                  addplayer.addPlayer = false;
91                  NavigationService.Navigate(addplayer);
92              }
93              else
94              {
95                  MessageBox.Show("Please select a player.", "No Player Selected",
96                              MessageBoxButton.OK, MessageBoxImage.Error);
97              }
98
99          }
100
101         private bool playerSelected()
102         {
103             bool valid = false;
104
105             if (lstPlayers.SelectedIndex >= 0)
106             {
107                 valid = true;
108             }
109
110             return valid;
111         }
112
113         private void btnDeletePlayer_Click(object sender, RoutedEventArgs e)
114         {
115             int i = lstPlayers.SelectedIndex;
116             if (i != -1)
117             {
118                 Player player = (Player)players[i];
119                 string message = "Are you sure you want to delete the player "
120                     + player.FirstName.Trim() + " " + player.LastName.Trim() + "?";
121                 MessageBoxResult button =
122                     MessageBox.Show(message, "Confirm Delete",
123                     MessageBoxButton.YesNo);
124                 if (button == MessageBoxResult.Yes)
125                 {
126                     players.Remove(player);
127                     QuiddichDB.DeletePlayer(player);
128                     fillListOfPlayers();
129                 }
130             }
131         }
132
```

```
133          private void btnBack_Click(object sender, RoutedEventArgs e)
134          {
135              ManageTeams manageTeams = new ManageTeams();
136              NavigationService.Navigate(manageTeams);
137          }
138      }
139 }
140
```