

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Text.RegularExpressions;
6 using System.Threading.Tasks;
7 using System.Windows;
8 using System.Windows.Controls;
9
10 namespace QuiddichV2._0
11 {
12     public static class Validator
13     {
14         private static string title = "Entry Error";
15
16         public static string Title
17         {
18             get
19             {
20                 return title;
21             }
22             set
23             {
24                 title = value;
25             }
26         }
27
28         public static bool IsPresent(TextBox textBox)
29         {
30             if (textBox.Text == "")
31             {
32                 MessageBox.Show(textBox.Tag + " is a required field.", Title);
33                 textBox.Focus();
34                 return false;
35             }
36             return true;
37         }
38
39         public static bool IsPresent(PasswordBox passwordBox)
40         {
41             if (passwordBox.Password == "")
42             {
43                 MessageBox.Show(passwordBox.Tag + " is a required field.", Title);
44                 passwordBox.Focus();
45                 return false;
46             }
47             return true;
48         }
49
50         public static bool IsDecimal(TextBox textBox)
51         {
52             decimal number = 0m;
53             if (Decimal.TryParse(textBox.Text, out number))
54             {
55                 return true;
56             }
57             else
58             {
59                 MessageBox.Show(textBox.Tag + " must be a decimal value.", Title);
60                 textBox.Focus();
61                 return false;
62             }
63         }
64
65         // The IsInt32 and IsWithinRange methods were omitted from figure 12-15.
66         public static bool IsInt32(TextBox textBox)
```

```
67     {
68         int number = 0;
69         if (Int32.TryParse(textBox.Text, out number))
70         {
71             return true;
72         }
73         else
74         {
75             MessageBox.Show(textBox.Tag + " must be an integer.", Title);
76             textBox.Focus();
77             return false;
78         }
79     }
80
81     public static bool IsWithinRange(TextBox textBox, decimal min, decimal max)
82     {
83         decimal number = Convert.ToDecimal(textBox.Text);
84         if (number < min || number > max)
85         {
86             MessageBox.Show(textBox.Tag + " must be between " + min
87                 + " and " + max + ".", Title);
88             textBox.Focus();
89             return false;
90         }
91         return true;
92     }
93
94     public static bool IsValidZip(TextBox textBox)
95     {
96         bool valid = false;
97
98         string pattern = @"^\d{5}(-\d{4})?$";
99         Regex reg = new Regex(pattern);
100
101         if(reg.Match(textBox.Text).Success)
102         {
103             valid = true;
104         }
105         else
106         {
107             MessageBox.Show(textBox.Tag + "is not a valid zip code." +
108                 "\n Valid zip codes are in the format XXXXX or XXXXX-XXXX",
109                 "Invalid Zip Code", MessageBoxButtons.OK, MessageBoxIcon.Error);
110         }
111
112         return valid;
113     }
114 }
115 }
116
```