

Colors and Brushes



```
//File:Window.xaml.cs
using System;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Shapes;
using System.Reflection;
using System.Collections.Generic;

namespace WpfApplication1
{
    public partial class ColorExample : Window
    {
        private Color color;
        SolidColorBrush colorBrush = new SolidColorBrush();
        public ColorExample()
        {
            InitializeComponent();
            Type colorsType = typeof(Colors);
            foreach (PropertyInfo property in colorsType.GetProperties())
```

```

{
listBox1.Items.Add(property.Name);
color = Colors.AliceBlue;
listBox1.SelectedIndex = 0;
ColorInfo();
}
}
private void listBox1SelectionChanged(object sender, EventArgs
e)
{
string colorString = listBox1.SelectedItem.ToString();
color = (Color)ColorConverter.ConvertFromString(colorString);
float opacity = Convert.ToSingle(textBox.Text);
if (opacity > 1.0f)
opacity = 1.0f;
else if (opacity < 0.0f) opacity = 0.0f; color.ScA = opacity;
ColorInfo(); } private void ColorInfo() { rect1.Fill = new
SolidColorBrush(color); Console.WriteLine("Alpha = " +
color.A.ToString()); Console.WriteLine("Red = " +
color.R.ToString()); Console.WriteLine("Green = " +
color.G.ToString()); Console.WriteLine("Blue = " +
color.B.ToString()); string rgbHex =
string.Format("{0:X2}{1:X2}{2:X2}{3:X2}", color.A, color.R,
color.G, color.B); Console.WriteLine("ARGB = #" + rgbHex);
Console.WriteLine("ScA = " + color.ScA.ToString());
Console.WriteLine("ScR = " + color.ScR.ToString());
Console.WriteLine("ScG = " + color.ScG.ToString());
Console.WriteLine("ScB = " + color.ScB.ToString()); } } }
[/csharp]

```