



链滴

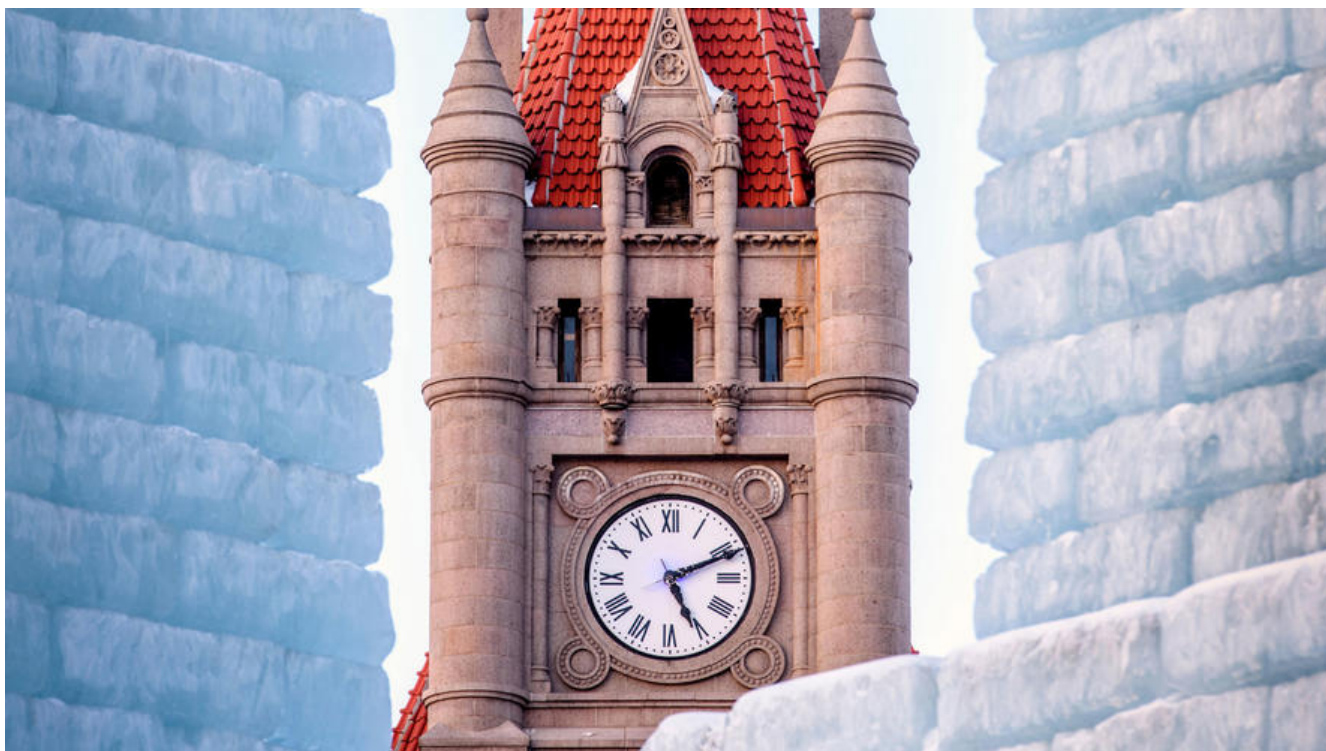
在 dotnet core (C#) 下的颜色渐变

作者: [hsxian](#)

原文链接: <https://ld246.com/article/1557416916720>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



直接使用等比例抽样算法，连同透明度一起计算。

```
public IList<Color> ShadeColors(Color c1, Color c2, int resultCount)
{
    var result = new List<Color>();
    for (int i = 0; i < resultCount; i++)
    {
        var a = c1.A + (int)((c2.A - c1.A) * i / resultCount);
        var r = c1.R + (int)((c2.R - c1.R) * i / resultCount);
        var g = c1.G + (int)((c2.G - c1.G) * i / resultCount);
        var b = c1.B + (int)((c2.B - c1.B) * i / resultCount);
        var c = Color.FromArgb(a, r, g, b);
        result.Add(c);
    }
    return result;
}
```

输出图片

如果要输出图片可能需要安装 [libgdplus](#)，如果缺少依赖的话,详见 [CoreCompat/CoreCompat](#)。在 [Ubuntu](#) 执行命令：

```
sudo apt-get update
sudo apt-get install libgdplus
```

在 [dockerfile](#)：

```
FROM microsoft/aspnetcore:2.0
# install System.Drawing native dependencies
RUN apt-get update \
    && apt-get install -y --allow-unauthenticated \
```

```
libc6-dev \  
libgdiplus \  
libx11-dev \  
&& rm -rf /var/lib/apt/lists/*
```

输出图片的代码:

```
static void Main(string[] args)  
{  
    int size = 300;  
    int h = 30;  
    var bmp = new Bitmap(size, h, PixelFormat.Format32bppArgb);  
    Color c1 = Color.FromArgb(0xff, Color.Green);  
    Color c2 = Color.FromArgb(0x4e, Color.Yellow);  
    var cs = ShadeColors(c1, c2, size);  
    for (int i = 0; i < size; i++)  
    {  
        for (int j = 0; j < h; j++)  
        {  
            bmp.SetPixel(i, j, cs[i]);  
        }  
    }  
    bmp.Save("shade.png", ImageFormat.Png);  
    Console.WriteLine("Hello World!");  
    // Console.ReadLine();  
}
```

最终输出的图片

