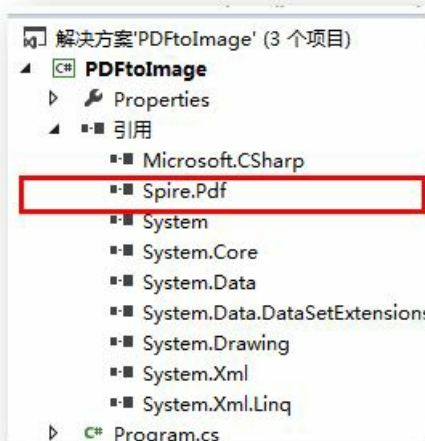


C#将PDF转为多种图像文件格式的方法 (Png/Bmp/Emf/Tiff)

里介绍将PDF转换多种不同格式的图像文件格式，如PNG，BMP，EMF，TIFF等，同时，转换文档也分为转换全部文档和转换部分文档为图片两种情况，本文也将作进一步介绍

PDF是一种在我们日常工作学习中最常用到的文档格式之一，但常常也会因为文档的不易编辑的特点，在遇到需要编辑PDF文档内容或者转换文件格式的情况时让人苦恼。通常对于开发者而言，可选择通过使用组件的方式来实现PDF文档的编辑或者格式转换，因此本文将介绍如何通过使用免费版的组件Free Spire.PDF for .NET来转换PDF文档。这里介绍将PDF转换多种不同格式的图像文件格式，如PNG，BMP，EMF，TIFF等，同时，转换文档也分为转换全部文档和转换部分文档为图片两种情况，本文也将作进一步介绍。下面是实现转换功能的详述，供参考。

提示：在下载安装该组件后，在项目中注意添加引用Spire.PDF.dll文件，如下图：



一、转换整个PDF文档为图片

(一) PDF转Png

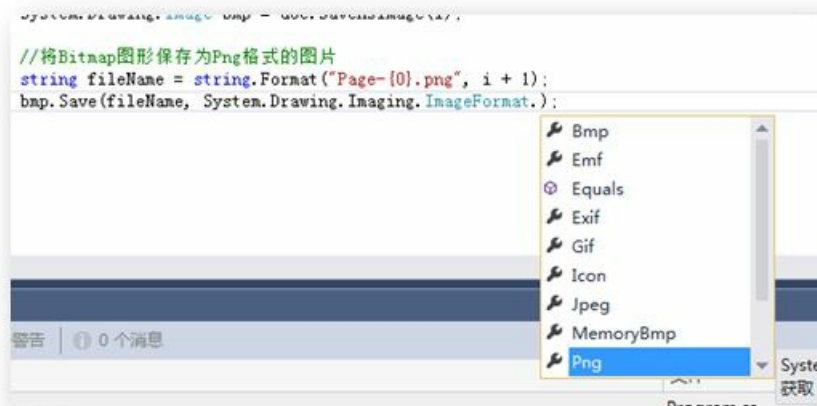
```
using Spire.Pdf;
using System.Drawing;
namespace PDFtoImage1
{
    class Program
    {
        static void Main(string[] args)
        {
            //初始化一个PdfDocument类实例,并加载PDF文档
            PdfDocument doc = new PdfDocument();
            doc.LoadFromFile(@"C:\Users\Administrator\Desktop\sample.pdf");
            //遍历PDF每一页
            for (int i = 0; i < doc.Pages.Count; i++)
            {
                //将PDF页转换成Bitmap图形
                System.Drawing.Image bmp = doc.SaveAsImage(i);
                //将Bitmap图形保存为Png格式的图片
                string fileName = string.Format("Page-{0}.png", i + 1);
                bmp.Save(fileName, System.Drawing.Imaging.ImageFormat.Png);
            }
        }
    }
}
```

调试运行程序，生成文档。

运行结果：

名称	修改日期	类型	大小
Page-1.png	2018/2/6 10:08	PNG 图像	99 KB
Page-2.png	2018/2/6 10:08	PNG 图像	189 KB
Page-3.png	2018/2/6 10:08	PNG 图像	181 KB
PDFtoImage1.exe	2018/2/6 10:10	应用程序	6 KB
PDFtoImage1.pdb	2018/2/6 10:10	Program Debug...	14 KB
PDFtoImage1.vshost.exe	2018/2/6 10:10	应用程序	23 KB
PDFtoImage1.vshost.exe.manifest	2013/3/18 17:00	MANIFEST 文件	1 KB
Spire.License.dll	2017/4/13 13:43	应用程序扩展	55 KB
Spire.License.xml	2017/4/13 13:43	XML 文档	5 KB
Spire.Pdf.dll	2017/5/10 16:36	应用程序扩展	12,980 KB
Spire.Pdf.xml	2017/4/13 13:43	XML 文档	1,111 KB

Spire.PDF支持将PDF文档转换为多种图像格式的文件，可根据需要选择相应的文件格式，这里以Png为例。



(二) PDF转TIFF

```
using System;
using System.Drawing;
using System.Drawing.Imaging;
using Spire.Pdf;
namespace SavePdfAsTiff
{
    class Program
    {
        static void Main(string[] args)
        {
            //创建一个PdfDocument类对象，并加载PDF文档
            PdfDocument document = new PdfDocument();
            document.LoadFromFile(@"C:\Users\Administrator\Desktop\sample.pdf");
            //调用方法SaveAsImage()将PDF文档保存为tiff格式
            JoinTiffImages(SaveAsImage(document), "result.tiff", EncoderValue.CompressionLZW);
            System.Diagnostics.Process.Start("result.tiff");
        }
        //自定义方法SaveAsImage()将PDF文档保存图像文件
        private static Image[] SaveAsImage(PdfDocument document)
        {
            Image[] images = new Image[document.Pages.Count];
            for (int i = 0; i < document.Pages.Count; i++)
            {
                images[i] = document.SaveAsImage(i);
            }
            return images;
        }
        private static ImageCodecInfo GetEncoderInfo(string mimeType)
        {
            ImageCodecInfo[] encoders = ImageCodecInfo.GetImageEncoders();
            for (int j = 0; j < encoders.Length; j++)
            {

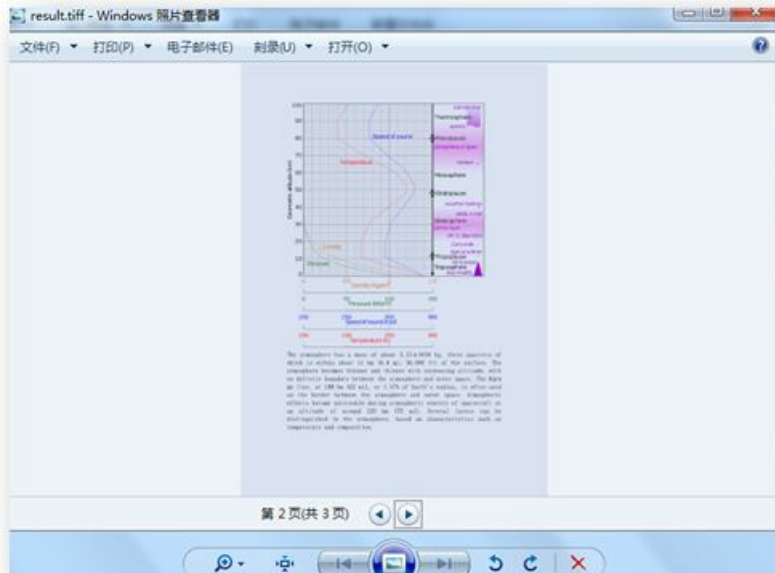
```

```

    if (encoders[j].MimeType == mimeType)
        return encoders[j];
    }
    throw new Exception(mimeType + " mime type not found in ImageCodeclInfo");
}
//自定义JoinTiffImages()方法,使用指定编码器和图像编码器参数将图像从pdf页面保存到tiff图像类型,。
public static void JoinTiffImages(Image[] images, string outFile, EncoderValue compressEncoder)
{
    Encoder enc = Encoder.SaveFlag;
    EncoderParameters ep = new EncoderParameters(2);
    ep.Param[0] = new EncoderParameter(enc, (long)EncoderValue.MultiFrame);
    ep.Param[1] = new EncoderParameter(Encoder.Compression, (long)compressEncoder);
    Image pages = images[0];
    int frame = 0;
    ImageCodeclInfo info = GetEncoderInfo("image/tiff");
    foreach (Image img in images)
    {
        if (frame == 0)
        {
            pages = img;
            pages.Save(outFile, info, ep);
        }
        else
        {
            ep.Param[0] = new EncoderParameter(enc, (long)EncoderValue.FrameDimensionPage);
            pages.SaveAdd(img, ep);
        }
        if (frame == images.Length - 1)
        {
            ep.Param[0] = new EncoderParameter(enc, (long)EncoderValue.Flush);
            pages.SaveAdd(ep);
        }
        frame++;
    }
}
}
}

```

运行结果:



二、转换PDF指定页为图片 (PDF转Png、Bmp、Emf)

```

using Spire.Pdf;
using System.Drawing;
using System.Drawing.Imaging;
namespace PdfToImage
{
    class Program
    {
        static void Main(string[] args)
        {
            //实例化一个PdfDocument类对象,并加载PDF文档
            PdfDocument doc = new PdfDocument();
            doc.LoadFromFile(@"C:\Users\Administrator\Desktop\sample.pdf");
        }
    }
}

```

```

//调用方法SaveAsImage()将PDF第二页保存为Bmp格式
Image bmp = doc.SaveAsImage(1);
//调用另一个SaveAsImage()方法, 并将指定页面保存保存为Emf、Png
Image emf = doc.SaveAsImage(0, Spire.Pdf.Graphics.PdfImageType.Metafile);
Image zoomImg = new Bitmap((int)(emf.Size.Width * 2), (int)(emf.Size.Height * 2));
using (Graphics g = Graphics.FromImage(zoomImg))
{
    g.ScaleTransform(2.0f, 2.0f);
    g.DrawImage(emf, new Rectangle(new Point(0, 0), emf.Size), new Rectangle(new Point(0, 0), emf.Size), GraphicsUnit.Pixel);
}
//命名保存的文件并打开
bmp.Save("convertToBmp.bmp", ImageFormat.Bmp);
System.Diagnostics.Process.Start("convertToBmp.bmp");
emf.Save("convertToEmf.emf", ImageFormat.Emf);
System.Diagnostics.Process.Start("convertToEmf.emf");
zoomImg.Save("convertToZoom.png", ImageFormat.Png);
System.Diagnostics.Process.Start("convertToZoom.png");
}
}
}

```

运行结果：

名称	修改日期	类型	大小
convertToBmp.bmp	2018/2/6 11:11	BMP 图像	3,484 KB
convertToEmf.emf	2018/2/6 11:11	EMF File	99 KB
convertToZoom.png	2018/2/6 11:11	PNG 图像	294 KB
PDFtoImage.exe	2018/2/6 11:10	应用程序	6 KB
PDFtoImage.pdb	2018/2/6 11:10	Program Debug...	14 KB
PDFtoImage.vshost.exe	2018/2/6 11:11	应用程序	23 KB
PDFtoImage.vshost.exe.manifest	2013/3/18 17:00	MANIFEST 文件	1 KB
Spire.License.dll	2017/4/13 13:43	应用程序扩展	55 KB
Spire.License.xml	2017/4/13 13:43	XML 文档	5 KB
Spire.Pdf.dll	2017/5/10 16:36	应用程序扩展	12,980 KB
Spire.Pdf.xml	2017/4/13 13:43	XML 文档	1,111 KB

总结

以上所述是小编给大家介绍的C#将PDF转为多种图像文件格式的方法（Png/Bmp/Emf/Tiff），希望对大家有所帮助，如果大家有任何疑问请给我留言，小编会及时回复大家的。在此也非常感谢大家对我们的支持！