**Aspose.PDF for .NET使用指南**

Aspose.PDF for .NET是一种高PDF处理和解析API，用于在跨平台应用程序中执行文档管理和操作任务。API可以轻松用于生成、修改、转换、渲染、保护和打印PDF文档，而无需使用Adobe Acrobat。此外，API还提供PDF压缩选项，表格创建和操作，图形和图像功能，广泛的超链接功能，印章和水印任务，扩展的安全控制和自定义字体处理。

**使用附件**

附件可以包含各种各样的信息，并且可以是各种文件类型。想要向PDF文件添加附件只需两步：

使用要添加的文件和文件描述创建一个文件具体化对象。

使用集合的Add方法将Filspeciification对象添加到文档对象的EmbeddedFiles集合中。

该EmbeddedFiles集合包含PDF文件中的所有附件。以下代码段显示了如何在PDF文档中添加附件：

//文档目录的路径。

string dataDir = RunExamples。GetDataDir\_AsposePdf\_Attachments（）;

//打开文档

Document pdfDocument = new Document(dataDir + "AddAttachment.pdf" );

//设置要添加为附件的新文件

FileSpecification fileSpecification = new FileSpecification（dataDir + “ test.txt ”，“ Sample text file ”）;

//添加附件到文档的附件集合

pdfDocument.EmbeddedFiles.Add(fileSpecification);

dataDir = dataDir + "AddAttachment\_out.pdf" ;

//保存新输出

pdfDocument.Save(dataDir);

**从PDF文档获取所有附件**

使用Aspose.PDF，可以从PDF文档中获取所有附件。当想要将文档与PDF分开保存，或者需要剥离PDF附件时，这非常有用。

要从PDF文件中获取所有附件只需两步：

循环遍历文档对象的EmbeddedFiles集合。EmbeddedFiles集合包含所有附件。这个集合的每个元素都表示一个文件具体化对象。Foreach循环通过EmbeddedFiles集合的每次迭代都返回一个filspeciification对象。

一旦对象可用，检索所有附加文件的属性或文件本身。

以下代码段显示如何从PDF文档中获取所有附件：

//文档目录的路径

string dataDir = RunExamples.GetDataDir\_AsposePdf\_Attachments();

//打开文档

Document pdfDocument = new Document(dataDir + "GetAlltheAttachments.pdf" );

//获取嵌入式文件集合

EmbeddedFileCollection embeddedFiles = pdfDocument.EmbeddedFiles;

//获取嵌入文件的数量

Console.WriteLine( "Total files : {0}" , embeddedFiles.Count);

int count = 1 ;

//遍历集合以获取所有附件

foreach (FileSpecification fileSpecification in embeddedFiles)

{

 Console.WriteLine( "Name: {0}" , fileSpecification.Name);

 Console.WriteLine( "Description: {0}" ,

 fileSpecification.Description);

 Console.WriteLine( "Mime Type: {0}" , fileSpecification.MIMEType);

 //检查参数对象是否包含参数

 if (fileSpecification.Params != null )

 {

 Console.WriteLine( "CheckSum: {0}" ,

 fileSpecification.Params.CheckSum);

 Console.WriteLine( "Creation Date: {0}" ,

 fileSpecification.Params.CreationDate);

 Console.WriteLine( "Modification Date: {0}" ,

 fileSpecification.Params.ModDate);

 Console.WriteLine( "Size: {0}" , fileSpecification.Params.Size);

 }

 //获取附件并写入文件或流

 byte[] fileContent = new byte[fileSpecification.Contents.Length];

 fileSpecification.Contents.Read(fileContent, 0 ,

 fileContent.Length);

 FileStream fileStream = new FileStream(dataDir + count + "\_out" + ".txt" ,

 FileMode.Create);

 fileStream.Write(fileContent, 0 , fileContent.Length);

 fileStream.Close();

count+= 1 ;

**获得单独的附件**

为了获得单独的附件，我们可以在文档实例的EmbeddedFiles对象中指定附件索引。请尝试使用以下代码片段。

//文档目录的路径

string dataDir = RunExamples.GetDataDir\_AsposePdf\_Attachments();

//打开文档

Document pdfDocument = new Document(dataDir + "GetIndividualAttachment.pdf" );

//获取特定的嵌入文件

FileSpecification fileSpecification = pdfDocument.EmbeddedFiles[ 1 ];

//获取文件属性

Console.WriteLine( "Name: {0}" , fileSpecification.Name);

Console.WriteLine( "Description: {0}" , fileSpecification.Description);

Console.WriteLine( "Mime Type: {0}" , fileSpecification.MIMEType);

//检查参数对象是否包含参数

if (fileSpecification.Params != null )

{

 Console.WriteLine( "CheckSum: {0}" ,

 fileSpecification.Params.CheckSum);

 Console.WriteLine( "Creation Date: {0}" ,

 fileSpecification.Params.CreationDate);

 Console.WriteLine( "Modification Date: {0}" ,

 fileSpecification.Params.ModDate);

 Console.WriteLine( "Size: {0}" , fileSpecification.Params.Size);

}

//获取附件并写入文件或流

byte[] fileContent = new byte[fileSpecification.Contents.Length];

fileSpecification.Contents.Read(fileContent, 0 , fileContent.Length);

FileStream fileStream = new FileStream(dataDir + "test\_out" + ".txt" , FileMode.Create);

fileStream.Write(fileContent, 0 , fileContent.Length);

fileStream.Close();

**从PDF文档中删除所有附件**

Aspose.PDF可以从PDF文件中删除附件。PDF文档的附件保存在Document对象的EmbeddedFiles集合中。

要删除与PDF文件关联的所有附件需两步：

调用EmbeddedFiles集合的Delete方法。

使用Document对象的Save方法保存更新的文件。

//文档目录的路径

string dataDir = RunExamples.GetDataDir\_AsposePdf\_Attachments();

//打开文档

Document pdfDocument = new Document(dataDir + "DeleteAllAttachments.pdf" );

//删除所有附件

pdfDocument.EmbeddedFiles.Delete();

dataDir = dataDir + "DeleteAllAttachments\_out.pdf" ;

//保存更新的文件

pdfDocument.Save(dataDir);

**获取附件信息**

附件信息保存在filspeciification对象中，与文档对象的EmbeddedFiles集合中的其他附件一起收集。文件化对象提供了获取hteattchment信息的方法，例如:

Name - 文件名。

Description - 文件描述。

MIMEType - 文件的MIME类型。

Params-有关文件的参数信息。

要获取这些参数，请首先确保该Params属性不为null。EmbeddedFiles使用foreach循环遍历集合中的所有附件，或通过指定其索引值获取单个附件。以下代码段显示了如何获取有关特定附件的信息：

//文档目录的路径

string dataDir = RunExamples.GetDataDir\_AsposePdf\_Attachments();

//打开文档

Document pdfDocument = new Document(dataDir + "GetAttachmentInfo.pdf" );

//获取特定的嵌入文件

FileSpecification fileSpecification = pdfDocument.EmbeddedFiles[ 1 ];

//获取文件属性

Console.WriteLine( "Name: {0}" , fileSpecification.Name);

Console.WriteLine( "Description: {0}" , fileSpecification.Description);

Console.WriteLine( "Mime Type: {0}" , fileSpecification.MIMEType);

//检查参数对象是否包含参数

if (fileSpecification.Params != null )

{

 Console.WriteLine( "CheckSum: {0}" ,

 fileSpecification.Params.CheckSum);

 Console.WriteLine( "Creation Date: {0}" ,

 fileSpecification.Params.CreationDate);

 Console.WriteLine( "Modification Date: {0}" ,

 fileSpecification.Params.ModDate);

 Console.WriteLine( "Size: {0}" , fileSpecification.Params.Size);

}