

itext 生成 PDF(一)



Iconcise

关注

0.808

2018.06.21 19:59:45

字数 624

阅读 6,464

itext 生成 PDF(二)

官网: <http://itextsupport.com/apidocs/itext5/latest/>
博文: <https://blog.csdn.net/u010154380/article/details/78087663>
博文: <https://blog.csdn.net/u013129932/article/details/43889705>

iText是著名的开放源码的站点sourceforge一个项目，是用于生成PDF文档的一个java类库。通过iText不仅可以生成PDF或rtf的文档，而且可以将XML、Html文件转化为PDF文件。

项目要使用iText，必须引入jar包。才能使用，maven依赖如下：

```
1 <dependency>
2   <groupId>com.itextpdf</groupId>
3   <artifactId>itextpdf</artifactId>
4   <version>5.5.10</version>
5 </dependency>
```

输出中文，还要引入下面itext-asian.jar包：

```
1 <dependency>
2   <groupId>com.itextpdf</groupId>
3   <artifactId>itext-asian</artifactId>
4   <version>5.2.0</version>
5 </dependency>
```

设置pdf文件密码，还要引入下面bcprov-jdk15on.jar包：

```
1 <dependency>
2   <groupId>org.bouncycastle</groupId>
3   <artifactId>bcprov-jdk15on</artifactId>
4   <version>1.54</version>
5 </dependency>
```

iText常用类

- com.itextpdf.text.Document：这是iText库中最常用的类，它代表了一个pdf实例。如果你需要从零开始生成一个PDF文件，你需要使用这个Document类。首先创建（new）该实例，然后打开（open）它，并添加（add）内容，最后关闭（close）该实例，即可生成一个pdf文件。
- com.itextpdf.text.Paragraph：表示一个缩进的文本段落，在段落中，你可以设置对齐方式，缩进，段落前后间隔等
- com.itextpdf.text.Chapter：表示PDF的一个章节，他通过一个Paragraph类型的标题和整形章数创建
- com.itextpdf.text.Font：这个类包含了所有规范好的字体，包括family of font，大小，样式和颜色，所有这些字体都被声明为静态常量
- com.itextpdf.text.List：表示一个列表；
- com.itextpdf.text.Anchor：表示一个锚，类似于HTML页面的链接。

广告



Iconcise

关注

总资产79 (约7.88元)

- SpringSecurity OAuth2 自定义令牌配置(JWT)
- 阅读 154
- 学习Dockerfile[转载]
- 阅读 5
- SpringCloud Eureka服务治理(二)
- 阅读 16

- 推荐阅读
- 李敖前妻胡因梦：遇到一个渣男，她整整消化了40年
- 阅读 16,381
- 中元 | 世间最好的相遇，是久别重逢
- 阅读 923

广告

写下你的评论...

评论0 赞18 ...

iText使用

创建一个简单的pdf文件，如下：

```
1 package com.hd.pdf;
2
3 import java.io.FileNotFoundException;
4 import java.io.FileOutputStream;
5
6 import com.itextpdf.text.Document;
7 import com.itextpdf.text.DocumentException;
8 import com.itextpdf.text.Paragraph;
9 import com.itextpdf.text.pdf.PdfWriter;
10
11 public class TestPDFDemo1 {
12
13     public static void main(String[] args) throws FileNotFoundException, DocumentException {
14
15         // 1.新建document对象
16         Document document = new Document();
17
18         // 2.建立一个书写器(Writer)与document对象关联，通过书写器(Writer)可以将文档写入到磁盘中。
19         // 创建 PdfWriter 对象 第一个参数是对文档对象的引用，第二个参数是文件的实际名称，在该名称中还会
20         PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
21
22         // 3.打开文档
23         document.open();
24
25         // 4.添加一个内容段落
26         document.add(new Paragraph("Hello World!"));
27
28         // 5.关闭文档
29         document.close();
30     }
31 }
32
33 }
```

打开文件



851491-20161209165247147-746087588.png

PDF中创建表格

```
1 public static void main(String[] args) throws DocumentException, FileNotFoundException {
2     //创建文件
3     Document document = new Document();
4     //建立一个书写器
5     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
6     //打开文件
7     document.open();
8     //添加内容
9     document.add(new Paragraph("HD content here"));
10
11     // 3列的表.
```

写下你的评论...

评论0

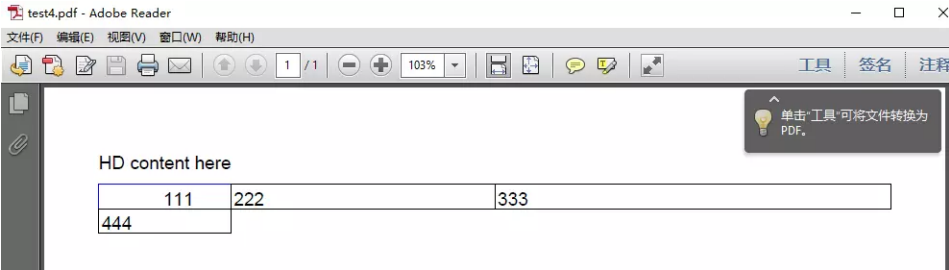
赞18

...

```
17 List<PdfPRow> listRow = table.getRows();
18 //设置列宽
19 float[] columnWidths = { 1f, 2f, 3f };
20 table.setWidths(columnWidths);
21
22 //行1
23 PdfPCell cells1[] = new PdfPCell[3];
24 PdfPRow row1 = new PdfPRow(cells1);
25
26 //单元格
27 cells1[0] = new PdfPCell(new Paragraph("111")); //单元格内容
28 cells1[0].setBorderColor(BaseColor.BLUE); //边框验证
29 cells1[0].setPaddingLeft(20); //左填充20
30 cells1[0].setHorizontalAlignment(Element.ALIGN_CENTER); //水平居中
31 cells1[0].setVerticalAlignment(Element.ALIGN_MIDDLE); //垂直居中
32
33 cells1[1] = new PdfPCell(new Paragraph("222"));
34 cells1[2] = new PdfPCell(new Paragraph("333"));
35
36 //行2
37 PdfPCell cells2[] = new PdfPCell[3];
38 PdfPRow row2 = new PdfPRow(cells2);
39 cells2[0] = new PdfPCell(new Paragraph("444"));
40
41 //把第一行添加到集合
42 listRow.add(row1);
43 listRow.add(row2);
44 //把表格添加到文件中
45 document.add(table);
46
47 //关闭文档
48 document.close();
49 //关闭书写器
50 writer.close();
51 }
```



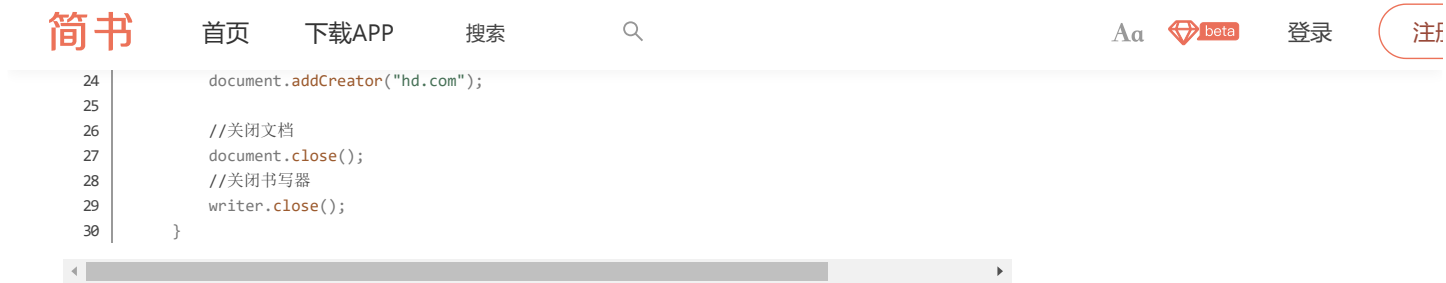
打开图片



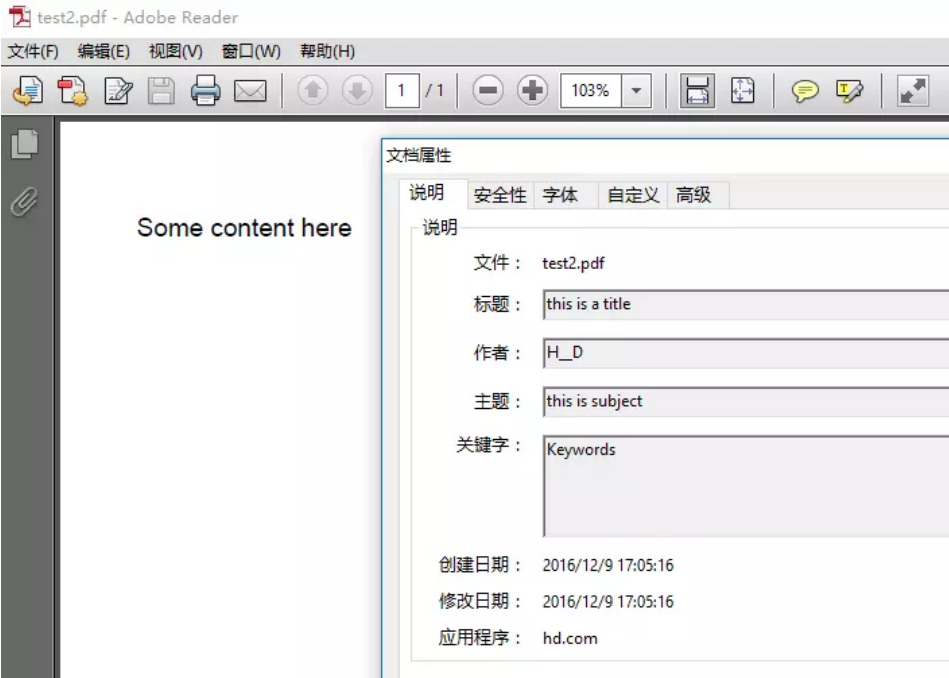
851491-20161209165247147-746087588.png

给PDF文件设置文件属性，例如：

```
1 public static void main(String[] args) throws FileNotFoundException, DocumentException {
2
3     //创建文件
4     Document document = new Document();
5     //建立一个书写器
6     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
7 //打开文件
8 document.open();
9 //添加内容
10 document.add(new Paragraph("Some content here"));
11
12 //设置属性
13 //标题
14 document.addTitle("this is a title");
15 //作者
16 document.addAuthor("H_D");
17 //主题
```

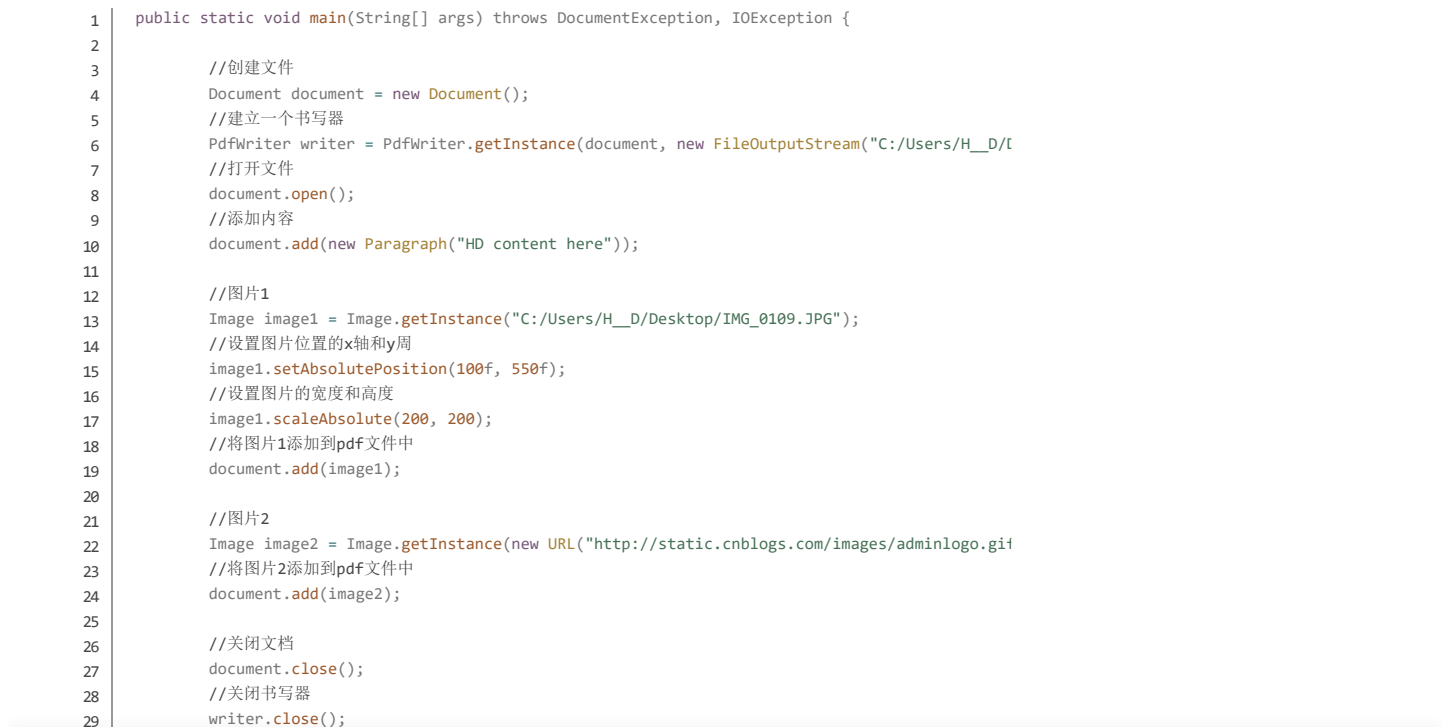


打开文件



851491-20161209165247147-746087588.png

PDF中添加图片

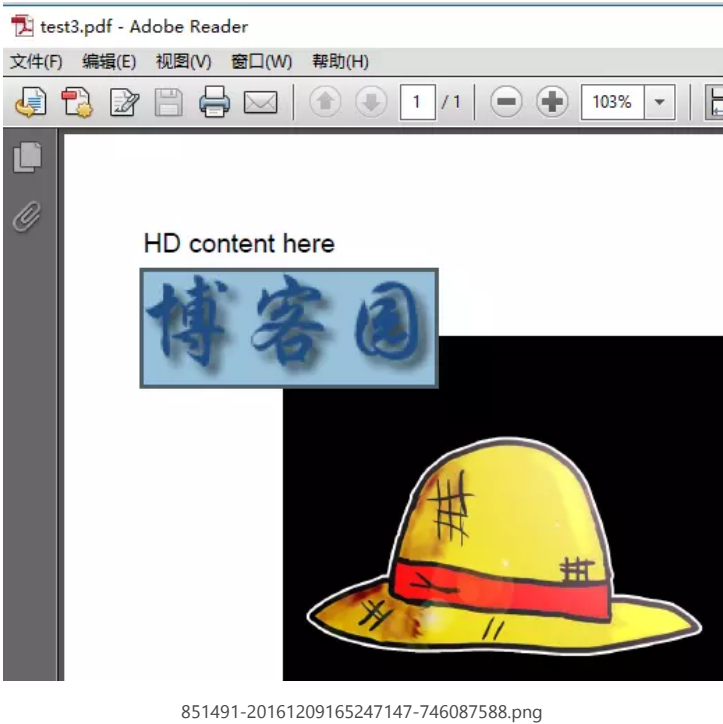


写下你的评论...

评论0

赞18

...



PDF中创建列表

```
1 public static void main(String[] args) throws DocumentException, FileNotFoundException {
2     //创建文件
3     Document document = new Document();
4     //建立一个书写器
5     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
6     //打开文件
7     document.open();
8     //添加内容
9     document.add(new Paragraph("HD content here"));
10
11     //添加有序列表
12     List orderedList = new List(List.ORDERED);
13     orderedList.add(new ListItem("Item one"));
14     orderedList.add(new ListItem("Item two"));
15     orderedList.add(new ListItem("Item three"));
16     document.add(orderedList);
17
18     //关闭文档
19     document.close();
20     //关闭书写器
21     writer.close();
22 }
```

打开文件

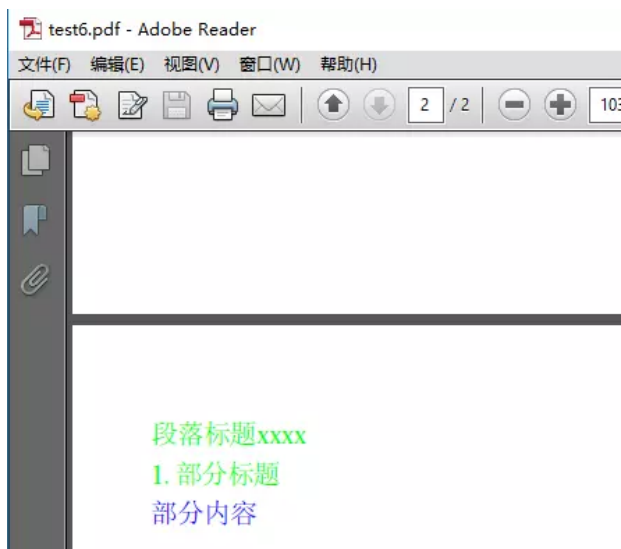


PDF中设置样式/格式化输出，输出中文内容，必须引入itext-asian.jar

```
1 public static void main(String[] args) throws DocumentException, IOException {
2     //创建文件
3     Document document = new Document();
4     //建立一个书写器
5     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
6     //打开文件
7     document.open();
8
9     //中文字体,解决中文不能显示问题
10    BaseFont bfChinese = BaseFont.createFont("STSong-Light", "UniGB-UCS2-H", BaseFont.NOT_EMBE
11
12    //蓝色字体
13    Font blueFont = new Font(bfChinese);
14    blueFont.setColor(BaseColor.BLUE);
15    //段落文本
16    Paragraph paragraphBlue = new Paragraph("paragraphOne blue front", blueFont);
17    document.add(paragraphBlue);
18
19    //绿色字体
20    Font greenFont = new Font(bfChinese);
21    greenFont.setColor(BaseColor.GREEN);
22    //创建章节
23    Paragraph chapterTitle = new Paragraph("段落标题xxxx", greenFont);
24    Chapter chapter1 = new Chapter(chapterTitle, 1);
25    chapter1.setNumberDepth(0);
26
27    Paragraph sectionTitle = new Paragraph("部分标题", greenFont);
28    Section section1 = chapter1.addSection(sectionTitle);
29
30    Paragraph sectionContent = new Paragraph("部分内容", blueFont);
31    section1.add(sectionContent);
32
33    //将章节添加到文章中
34    document.add(chapter1);
35
36    //关闭文档
37    document.close();
38    //关闭书写器
39    writer.close();
40 }
```

打开图片

![]



851491-20161209180029726-1168732515.png

写下你的评论...

评论0

赞18

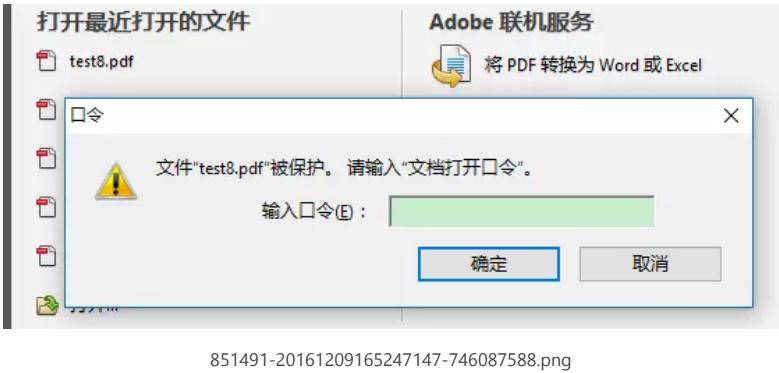
...



给PDF文件设置密码，需要引入bcprov-jdk15on.jar包：

```
1 public static void main(String[] args) throws DocumentException, IOException {
2     // 创建文件
3     Document document = new Document();
4     // 建立一个书写器
5     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
6
7     //用户密码
8     String userPassword = "123456";
9     //拥有者密码
10    String ownerPassword = "hd";
11    writer.setEncryption(userPassword.getBytes(), ownerPassword.getBytes(), PdfWriter.ALLOW_
12        PdfWriter.ENCRYPTION_AES_128);
13
14    // 打开文件
15    document.open();
16
17    //添加内容
18    document.add(new Paragraph("password !!!!"));
19
20    // 关闭文档
21    document.close();
22    // 关闭书写器
23    writer.close();
24 }
```

打开图片



给PDF文件设置权限

```
1 public static void main(String[] args) throws DocumentException, IOException {
2     // 创建文件
3     Document document = new Document();
4     // 建立一个书写器
5     PdfWriter writer = PdfWriter.getInstance(document, new FileOutputStream("C:/Users/H_D/I
6
7     // 只读权限
8     writer.setEncryption("", "", PdfWriter.ALLOW_PRINTING, PdfWriter.F
9
10    // 打开文件
11    document.open();
```

```
18 // 关闭书写器
19 writer.close();
20 }
```

读取/修改已有的PDF文件

```
1 public static void main(String[] args) throws DocumentException, IOException {
2
3     //读取pdf文件
4     PdfReader pdfReader = new PdfReader("C:/Users/H__D/Desktop/test1.pdf");
5
6     //修改器
7     PdfStamper pdfStamper = new PdfStamper(pdfReader, new FileOutputStream("C:/Users/H__D/Desktop/test1.pdf"));
8
9     Image image = Image.getInstance("C:/Users/H__D/Desktop/IMG_0109.JPG");
10    image.scaleAbsolute(50, 50);
11    image.setAbsolutePosition(0, 700);
12
13    for(int i=1; i<= pdfReader.getNumberOfPages(); i++)
14    {
15        PdfContentByte content = pdfStamper.getUnderContent(i);
16        content.addImage(image);
17    }
18
19    pdfStamper.close();
20 }
```

itext 生成 PDF(二)

👍 18人点赞 >

👎

📖 JAVA功底 ...

 Iconcise
总资产79 (约7.88元)

关注

"小礼物走一走，来简书关注我"

赞赏


广告

写下你的评论...

全部评论 0 只看作者

按时间倒序 按时间正序

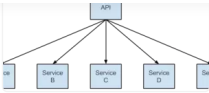
被以下专题收入，发现更多相似内容

 我爱编程

推荐阅读

更多精彩内容 >

下下下2017



Spring Boot集成Freemarker和iText生成PDF文档

PDF格式文档导出，是信息系统中非常实用的一种功能，用于各种报表和文档的到处。最近正好有空，用之前项目中使用过的i...

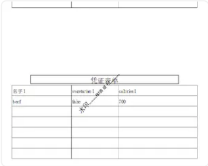
liuwill



JAVA中iText生成PDF的基本使用

iText是一个能够快速产生PDF文件的java类库。iText的java类对于那些要产生包含文本，表格，图形的只...

fly_he



纵观全局 投资自己

1.大处着眼 小处着手 看到这个就想到小时候班级又两种人，一种是每天认真听讲，埋头苦读最后成绩还是平平的学生，一种...

芒果佳子Mango

梦

今天做了一个超级美的梦 不知道是抖音看多了还是怎样 之前谈恋爱的时候最喜欢小孩子 分手之后再也不喜欢了 突然做梦自...

侯小刀