1. 需要下载pdf依赖的jar包（以最新的为准）

<dependency>

<groupId>com.itextpdf</groupId>

<artifactId>itextpdf</artifactId>

<version>5.5.10</version>

</dependency>

<dependency>

<groupId>com.itextpdf</groupId>

<artifactId>itext-asian</artifactId>

<version>5.2.0</version>

</dependency>

1. 根据模板生成对应的pdf，需要先下载Adobe Acrobat DC
2. 编辑模板表单

**public** **static** **void** fillTemplate() {

// 模板路径

String templatePath = "d:/test.pdf";

// 生成的新文件路径

String newPDFPath = "d:/ceshi.pdf";

PdfReader reader;

FileOutputStream out;

ByteArrayOutputStream bos;

PdfStamper stamper;

**try** {

out = **new** FileOutputStream(newPDFPath);// 输出流

reader = **new** PdfReader(templatePath);// 读取pdf模板

bos = **new** ByteArrayOutputStream();

stamper = **new** PdfStamper(reader, bos);

AcroFields form = stamper.getAcroFields();

String[] str = { "123456789", "TOP\_\_ONE", "男", "1991-01-01", "130222111133338888", "河北省保定市" };

**int** i = 0;

java.util.Iterator<String> it = form.getFields().keySet().iterator();

**while** (it.hasNext()) {

String name = it.next().toString();

System.*out*.println(name);

form.setField(name, "zhangsan");

}

stamper.setFormFlattening(**true**);// 如果为false那么生成的PDF文件还能编辑，一定要设为true

stamper.close();

Document doc = **new** Document();

PdfCopy copy = **new** PdfCopy(doc, out);

doc.open();

PdfImportedPage importPage = copy.getImportedPage(**new** PdfReader(bos.toByteArray()), 1);

copy.addPage(importPage);

doc.close();

} **catch** (IOException e) {

System.*out*.println(1);

} **catch** (DocumentException e) {

System.*out*.println(2);

}

1. }
2. 手动生成pdf模板
3. package com.davidstudio.gbp.core.util;
4. import java.io.ByteArrayOutputStream;
5. import java.io.IOException;
6. import java.io.OutputStream;
7. import java.text.SimpleDateFormat;
8. import java.util.Date;
9. import java.util.List;
10. import java.util.Map;
11. import javax.servlet.ServletOutputStream;
12. import javax.servlet.http.HttpServletRequest;
13. import javax.servlet.http.HttpServletResponse;
14. import xggl.djgl.dj\_dycjb.vo.Xs\_djdkcj\_view;
15. import xggl.djgl.dj\_sxhbb.vo.Dj\_sxhbb;
16. import xggl.djgl.dj\_xsdydnjcb.vo.Xs\_djdnjc\_view;
17. import xggl.djgl.dj\_xsdyhdqk.vo.Dj\_xsdyhdqk;
18. import xggl.djgl.dj\_xsdyrdjsb.vo.Dj\_xsdyrdjsb;
19. import xggl.djgl.dj\_xsdysjb.vo.Xs\_djdysj\_view;
20. import xggl.djgl.dmwh.vo.Code\_dnzwb;
21. import xggl.djgl.dmwh.vo.Code\_dzbb;
22. import xggl.djgl.dmwh.vo.Code\_zzdmb;
23. import xggl.djgl.zzhd.vo.Dj\_xsdyzzsqb\_view;
24. import xggl.xj.xjgl.vo.Xs\_xjb;
25. import xggl.xj.xjgl.vo.Xs\_xjb\_view;
26. import com.itextpdf.text.Document;
27. import com.itextpdf.text.DocumentException;
28. import com.itextpdf.text.Element;
29. import com.itextpdf.text.Font;
30. import com.itextpdf.text.Paragraph;
31. import com.itextpdf.text.Rectangle;
32. import com.itextpdf.text.pdf.BaseFont;
33. import com.itextpdf.text.pdf.PdfPCell;
34. import com.itextpdf.text.pdf.PdfPTable;
35. import com.itextpdf.text.pdf.PdfWriter;
36. public class ItextUtil {
37. private static String str= new SimpleDateFormat("yyyyMMdd").format(new Date());
39. /\*\*
40. \* 下载生成好的pdf
41. \* @param request
42. \* @param response
43. \* @param map
44. \*/
45. public static void generPdf(HttpServletRequest request,HttpServletResponse response,Map<String, Object> map){
46. ByteArrayOutputStream out = null;
47. Document document=null;
48. try {
49. out= new ByteArrayOutputStream();
50. // 创建文Pdf文挡50, 50, 50,50左右上下距离
51. document = new Document(new Rectangle(1500, 2000), 50, 50, 50, 50);
52. PdfWriter.getInstance(document, out);
53. document.open();
54. document= genDy(document,map);
55. document.close();
56. } catch (IOException e) {
57. // TODO Auto-generated catch block
58. e.printStackTrace();
59. } catch (DocumentException e) {
60. // TODO Auto-generated catch block
61. e.printStackTrace();
62. } catch (Exception e) {
63. // TODO Auto-generated catch block
64. e.printStackTrace();
65. }finally{
66. response.reset();
67. response.setContentType("application/pdf");
68. response.setHeader("content-disposition", "attachment; filename=党员信息.pdf");
69. ServletOutputStream outputStream;
70. try {
71. outputStream = response.getOutputStream();
72. response.setContentLength(out.size());
73. out.writeTo(outputStream);
74. outputStream.flush();
75. outputStream.close();
76. } catch (IOException e) {
77. // TODO Auto-generated catch block
78. e.printStackTrace();
79. }
81. }
82. }
84. public static Document genDy(Document document,Map<String, Object> map) throws Exception, IOException{
85. //获取相关的信息
86. Xs\_xjb\_view xs\_xjb=(Xs\_xjb\_view) map.get("xs\_xjb");
87. Xs\_djdysj\_view xs\_djdysj\_view=(Xs\_djdysj\_view) map.get("xs\_djdysj\_view");
88. Xs\_djdkcj\_view xs\_djdkcj\_view=(Xs\_djdkcj\_view)map.get("xs\_djdkcj\_view");
89. List<Dj\_xsdyrdjsb> xs\_djrdjsr\_view1=(List<Dj\_xsdyrdjsb>)map.get("xs\_djrdjsr\_view1");
90. List<Xs\_djdnjc\_view> xs\_djdnjc\_view=(List<Xs\_djdnjc\_view>) map.get("xs\_djdnjc\_view1");
91. List<Dj\_xsdyhdqk> dj\_xsdyhdqks=(List<Dj\_xsdyhdqk>) map.get("dj\_xsdyhdqks");
92. List<Dj\_xsdyzzsqb\_view> dyzzs=(List<Dj\_xsdyzzsqb\_view>)map.get("dyzzs");
93. List<Dj\_sxhbb> yjfks=(List<Dj\_sxhbb>)map.get("yjfks");
94. List<Dj\_sxhbb> xxxds=(List<Dj\_sxhbb>)map.get("xxxds");
95. List<Code\_dzbb> code\_dzbbs= (List<Code\_dzbb>)map.get("code\_dzbb");
96. List<Code\_dnzwb> code\_dnzwbs=(List<Code\_dnzwb>)map.get("code\_dnzwb");
97. List<Code\_zzdmb> code\_zzdmbs=(List<Code\_zzdmb>) map.get("code\_zzdmb");
98. //设置标题的字样
99. Font titleFont =getFont(50);
100. Paragraph title = new Paragraph("山东大学本科生党员信息表",titleFont);
101. title.setAlignment(Element.ALIGN\_CENTER);
102. title.setSpacingAfter(20);
103. document.add(title);
105. //设置第一行
106. Font context = getFont(25);
107. title = new Paragraph("学院："+xs\_xjb.getXsm()+" 党支部： "+(code\_dzbbs.size()>0?code\_dzbbs.get(0).getDzbmc():"")+" 导出时间： "+str.substring(0, 4)+" 年 "+str.subSequence(4, 6)+"月 "+str.subSequence(6, 8)+"日",context);
108. title.setAlignment(Element.ALIGN\_LEFT);
109. title.setIndentationLeft(140);
110. title.setSpacingAfter(20);
111. document.add(title);
113. Font fontChinese = getFont(20);
114. // 创建有colNumber(6)列的表格
115. PdfPTable datatable = getTable(10, new int [] { 2,3,2,3,2,3,2,3,2,3 });
116. // 表格的宽度百分比
117. datatable.setWidthPercentage(80);
118. // 添加表头元素
119. PdfPCell cell2=getCell("姓名", fontChinese);
120. datatable.addCell(cell2);
121. cell2=getCell(xs\_xjb.getXm(), fontChinese);
122. datatable.addCell(cell2);
123. cell2=getCell("学号", fontChinese);
124. datatable.addCell(cell2);
125. cell2=getCell(xs\_xjb.getId(), fontChinese);
126. datatable.addCell(cell2);
127. cell2=getCell("性别", fontChinese);
128. datatable.addCell(cell2);
129. cell2=getCell(xs\_xjb.getXb(), fontChinese);
130. datatable.addCell(cell2);
131. cell2=getCell("出生年月", fontChinese);
132. datatable.addCell(cell2);
133. cell2=getCell(xs\_xjb.getCsrq(), fontChinese);
134. datatable.addCell(cell2);
135. cell2=getCell("班级", fontChinese);
136. datatable.addCell(cell2);
137. cell2=getCell(xs\_xjb.getBm(), fontChinese);
138. datatable.addCell(cell2);
139. cell2=getCell("政治面貌", fontChinese);
140. datatable.addCell(cell2);
141. cell2=getCell(xs\_xjb.getZzmmmc(), fontChinese);
142. datatable.addCell(cell2);
143. cell2=getCell("入党时间", fontChinese);
144. datatable.addCell(cell2);
145. boolean flag1=(xs\_djdysj\_view==null);
146. cell2=getCell((!flag1?xs\_djdysj\_view.getRdsj():""), fontChinese);
147. datatable.addCell(cell2);
148. cell2=getCell("转正时间", fontChinese);
149. datatable.addCell(cell2);
150. cell2=getCell((!flag1?xs\_djdysj\_view.getZzsj():""), fontChinese);
151. datatable.addCell(cell2);
152. cell2=getCell("提交入党申请时间", fontChinese);
153. datatable.addCell(cell2);
154. cell2=getCell((!flag1?xs\_djdysj\_view.getTjrdsqsj():""), fontChinese);
155. cell2.setColspan(3);
156. datatable.addCell(cell2);
157. boolean flag2=(xs\_djdkcj\_view==null);
158. cell2=getCell("党课成绩", fontChinese);
159. datatable.addCell(cell2);
160. cell2=getCell((!flag2?xs\_djdkcj\_view.getDkcj()+"":""), fontChinese);
161. datatable.addCell(cell2);
162. cell2=getCell("通过党课考试时间", fontChinese);
163. datatable.addCell(cell2);
164. cell2=getCell((!flag2?xs\_djdkcj\_view.getKstgsj():""), fontChinese);
165. datatable.addCell(cell2);
166. cell2=getCell("党内职务", fontChinese);
167. datatable.addCell(cell2);
168. cell2=getCell((!flag1?xs\_djdysj\_view.getDzbzw():""), fontChinese);
169. datatable.addCell(cell2);
170. cell2=getCell("其他任职", fontChinese);
171. datatable.addCell(cell2);
172. cell2=getCell((!flag1?xs\_djdysj\_view.getDnzwbz():""), fontChinese);
173. datatable.addCell(cell2);
174. cell2=getCell("入党介绍人", fontChinese);
175. datatable.addCell(cell2);
176. cell2=getCell((xs\_djrdjsr\_view1.size()>0?xs\_djrdjsr\_view1.get(0).getRdjsr():""), fontChinese);
177. datatable.addCell(cell2);
178. PdfPCell cell3=getCell("", fontChinese);
179. cell3.setColspan(10);
180. //
181. PdfPTable datatable1=getTable(5,new int [] {2,4,4,4,4});
182. datatable1.setWidthPercentage(100);
183. datatable1.getDefaultCell().setBorderWidthTop(0);
184. datatable1.getDefaultCell().setBorderWidthLeft(0);
185. datatable1.getDefaultCell().setBorderWidthRight(0);
186. datatable1.getDefaultCell().setVerticalAlignment(Element.ALIGN\_MIDDLE);
187. datatable1.getDefaultCell().setHorizontalAlignment(Element.ALIGN\_CENTER);
188. cell2=getCell("党内奖惩情况", fontChinese);
189. cell2.setBorderWidthTop(0);
190. cell2.setBorderWidthLeft(0);
191. cell2.setRowspan(xs\_djdnjc\_view.size()+1);
192. datatable1.addCell(cell2);
193. cell2=getCell("奖惩时间", fontChinese);
194. datatable1.addCell(cell2);
195. cell2=getCell("奖惩名称", fontChinese);
196. datatable1.addCell(cell2);
197. cell2=getCell("奖惩级别", fontChinese);
198. datatable1.addCell(cell2);
199. cell2=getCell("发布单位", fontChinese);
200. cell2.setBorderWidthRight(0);
201. datatable1.addCell(cell2);
202. if(xs\_djdnjc\_view.size()>0){
203. for (Xs\_djdnjc\_view jc : xs\_djdnjc\_view) {
204. cell2=getCell(jc.getJcsj(), fontChinese);
205. datatable1.addCell(cell2);
206. cell2=getCell(jc.getJcmc(), fontChinese);
207. datatable1.addCell(cell2);
208. cell2=getCell(jc.getJb(), fontChinese);
209. datatable1.addCell(cell2);
210. cell2=getCell(jc.getFbdw(), fontChinese);
211. cell2.setBorderWidthRight(0);
212. datatable1.addCell(cell2);
213. }
214. }
215. cell3.addElement(datatable1);
216. //
217. PdfPTable datatable2=getTable(6, new int[]{2,4,4,4,4,4});
218. datatable2.setWidthPercentage(100);
219. datatable2.getDefaultCell().setBorderWidthLeft(0);
220. datatable2.getDefaultCell().setBorderWidthRight(0);
221. datatable2.getDefaultCell().setVerticalAlignment(Element.ALIGN\_MIDDLE);
222. datatable2.getDefaultCell().setHorizontalAlignment(Element.ALIGN\_CENTER);
223. cell2=getCell("参加党员活动情况", fontChinese);
224. cell2.setBorderWidthTop(0);
225. cell2.setBorderWidthLeft(0);
226. cell2.setRowspan(dj\_xsdyhdqks.size()+1);
227. datatable2.addCell(cell2);
228. cell2=getCell("活动日期", fontChinese);
229. datatable2.addCell(cell2);
230. cell2=getCell("活动名称", fontChinese);
231. datatable2.addCell(cell2);
232. cell2=getCell("活动地点", fontChinese);
233. datatable2.addCell(cell2);
234. cell2=getCell("活动级别", fontChinese);
235. datatable2.addCell(cell2);
236. cell2=getCell("发起单位", fontChinese);
237. cell2.setBorderWidthRight(0);
238. datatable2.addCell(cell2);
239. if(dj\_xsdyhdqks.size()>0){
240. for (Dj\_xsdyhdqk hd : dj\_xsdyhdqks) {
241. cell2=getCell(hd.getHdrq(), fontChinese);
242. datatable2.addCell(cell2);
243. cell2=getCell(hd.getHdmc(), fontChinese);
244. datatable2.addCell(cell2);
245. cell2=getCell(hd.getHddd(), fontChinese);
246. datatable2.addCell(cell2);
247. cell2=getCell(hd.getHdjb(), fontChinese);
248. datatable2.addCell(cell2);
249. cell2=getCell(hd.getSqbs(), fontChinese);
250. cell2.setBorderWidthRight(0);
251. datatable2.addCell(cell2);
252. }
253. }
254. cell3.addElement(datatable2);
255. //
256. PdfPTable datatable3=getTable(5, new int[] {2,4,4,4,4});
257. datatable3.setWidthPercentage(100);
258. datatable3.getDefaultCell().setBorderWidthLeft(0);
259. datatable3.getDefaultCell().setBorderWidthRight(0);
260. datatable3.getDefaultCell().setVerticalAlignment(Element.ALIGN\_MIDDLE);
261. datatable3.getDefaultCell().setHorizontalAlignment(Element.ALIGN\_CENTER);
262. cell2=getCell("参加党员组织情况", fontChinese);
263. cell2.setBorderWidthTop(0);
264. cell2.setBorderWidthLeft(0);
265. cell2.setRowspan(dyzzs.size()+1);
266. datatable3.addCell(cell2);
267. cell2=getCell("组织名称", fontChinese);
268. datatable3.addCell(cell2);
269. cell2=getCell("组织级别", fontChinese);
270. datatable3.addCell(cell2);
271. cell2=getCell("成立时间", fontChinese);
272. datatable3.addCell(cell2);
273. cell2=getCell("发起单位", fontChinese);
274. cell2.setBorderWidthRight(0);
275. datatable3.addCell(cell2);
276. if(dyzzs.size()>0){
277. for (Dj\_xsdyzzsqb\_view zzb : dyzzs) {
278. cell2=getCell(zzb.getZzmc(), fontChinese);
279. datatable3.addCell(cell2);
280. cell2=getCell(zzb.getZzjb(), fontChinese);
281. datatable3.addCell(cell2);
282. cell2=getCell(zzb.getClsj(), fontChinese);
283. datatable3.addCell(cell2);
284. cell2=getCell(zzb.getSqbs(), fontChinese);
285. cell2.setBorderWidthRight(0);
286. datatable3.addCell(cell2);
287. }
288. }
289. cell3.addElement(datatable3);
290. //
291. PdfPTable datatable4=getTable(3, new int[] {2,4,4});
292. datatable4.setWidthPercentage(100);
293. datatable4.getDefaultCell().setBorderWidthLeft(0);
294. datatable4.getDefaultCell().setBorderWidthRight(0);
295. datatable4.getDefaultCell().setVerticalAlignment(Element.ALIGN\_MIDDLE);
296. datatable4.getDefaultCell().setHorizontalAlignment(Element.ALIGN\_CENTER);
297. cell2=getCell("学习心得", fontChinese);
298. cell2.setBorderWidthTop(0);
299. cell2.setBorderWidthLeft(0);
300. cell2.setRowspan(xxxds.size()+1);
301. datatable4.addCell(cell2);
302. cell2=getCell("时间", fontChinese);
303. datatable4.addCell(cell2);
304. cell2=getCell("主题", fontChinese);
305. cell2.setBorderWidthRight(0);
306. datatable4.addCell(cell2);
307. if(xxxds.size()>0){
308. for (Dj\_sxhbb hbb : xxxds) {
309. cell2=getCell(hbb.getSj(), fontChinese);
310. datatable4.addCell(cell2);
311. cell2=getCell(hbb.getHbzt(), fontChinese);
312. cell2.setBorderWidthRight(0);
313. datatable4.addCell(cell2);
314. }
315. }
316. cell3.addElement(datatable3);
317. datatable.addCell(cell3);
318. document.add(datatable);
319. return document;
320. }
322. /\*\*
323. \* 获取单元格根据传递的参数
324. \* @param column
325. \* @param font
326. \* @return
327. \*/
328. public static PdfPCell getCell(String column,Font font){
329. PdfPCell pdfPCell= new PdfPCell();
330. pdfPCell.setHorizontalAlignment(Element.ALIGN\_CENTER);
331. pdfPCell.setVerticalAlignment(Element.ALIGN\_MIDDLE);
332. pdfPCell.setPhrase(new Paragraph(column,font));
333. return pdfPCell;
334. }
336. /\*\*
337. \* 根据指定的列指定的宽度创建表格
338. \* @param col
339. \* @param widths
340. \* @return
341. \*/
342. public static PdfPTable getTable(int col,int [] widths){
343. PdfPTable pdfPTable=new PdfPTable(col);
344. try {
345. pdfPTable.setWidths(widths);
346. } catch (DocumentException e) {
347. // TODO Auto-generated catch block
348. e.printStackTrace();
349. }
350. pdfPTable.setHorizontalAlignment(Element.ALIGN\_CENTER);
351. return pdfPTable;
352. }
354. /\*\*
355. \* 根据字体大小获取字体
356. \* @param size
357. \* @return
358. \*/
359. public static Font getFont(int size){
360. Font font=null;
361. // 中文字体
362. try {
363. BaseFont bfChinese =BaseFont.createFont("STSong-Light",
364. "UniGB-UCS2-H",BaseFont.NOT\_EMBEDDED);
365. font=new Font(bfChinese, size, Font.NORMAL);
366. } catch (DocumentException e) {
367. // TODO Auto-generated catch block
368. e.printStackTrace();
369. } catch (IOException e) {
370. // TODO Auto-generated catch block
371. e.printStackTrace();
372. }
373. return font;
374. }
375. }