[用poi把ppt或pptx转为图片](http://blog.csdn.net/emoven/article/details/52683215)

标签： [java](http://www.csdn.net/tag/java)[poi](http://www.csdn.net/tag/poi)

2016-09-27 18:12 7300人阅读 [评论](http://blog.csdn.net/emoven/article/details/52683215?locationNum=2&fps=1#comments)(1) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/emoven/article/details/52683215?locationNum=2&fps=1#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

java（27） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为博主原创文章，未经博主允许不得转载。

代码主要参考官网：<http://poi.apache.org/slideshow/how-to-shapes.html>

需要jar下载地址：<http://download.csdn.net/detail/emoven/9657787>

**[html]** [view plain](http://blog.csdn.net/emoven/article/details/52683215?locationNum=2&fps=1) [copy](http://blog.csdn.net/emoven/article/details/52683215?locationNum=2&fps=1)

[在CODE上查看代码片](https://code.csdn.net/snippets/1903478)

1. **<pre** name="code" class="html"**>**import java.awt.Color;
2. import java.awt.Dimension;
3. import java.awt.Graphics2D;
4. import java.awt.Image;
5. import java.awt.geom.Rectangle2D;
6. import java.awt.image.BufferedImage;
7. import java.io.File;
8. import java.io.FileInputStream;
9. import java.io.FileOutputStream;
10. import java.io.IOException;
11. import javax.imageio.ImageIO;
13. import org.apache.poi.hslf.usermodel.HSLFShape;
14. import org.apache.poi.hslf.usermodel.HSLFSlideShow;
15. import org.apache.poi.hslf.usermodel.HSLFSlideShowImpl;
16. import org.apache.poi.hslf.usermodel.HSLFTextParagraph;
17. import org.apache.poi.hslf.usermodel.HSLFTextRun;
18. import org.apache.poi.hslf.usermodel.HSLFTextShape;
19. import org.apache.poi.xslf.usermodel.XMLSlideShow;
20. import org.apache.poi.xslf.usermodel.XSLFShape;
21. import org.apache.poi.xslf.usermodel.XSLFTextParagraph;
22. import org.apache.poi.xslf.usermodel.XSLFTextRun;
23. import org.apache.poi.xslf.usermodel.XSLFTextShape;
25. /\*
26. \* ppt或pptx 转为图片
27. \*/
28. public class PPTtoImage {
29. public static void main(String[] args) throws Exception {
30. toImage2007();
31. toImage2003();

34. }
35. public static void toImage2007() throws Exception{
36. FileInputStream is = new FileInputStream("D:/demo/11.pptx");
37. XMLSlideShow ppt = new XMLSlideShow(is);
38. is.close();
40. Dimension pgsize = ppt.getPageSize();
41. System.out.println(pgsize.width+"--"+pgsize.height);
43. for (int i = 0; i **<** **ppt.getSlides**().size(); i++) {
44. try {
45. //防止中文乱码
46. for(XSLFShape shape : ppt.getSlides().get(i).getShapes()){
47. if(shape instanceof XSLFTextShape) {
48. XSLFTextShape tsh = (XSLFTextShape)shape;
49. for(XSLFTextParagraph p : tsh){
50. for(XSLFTextRun r : p){
51. r.setFontFamily("宋体");
52. }
53. }
54. }
55. }
56. BufferedImage img = new BufferedImage(pgsize.width, pgsize.height, BufferedImage.TYPE\_INT\_RGB);
57. Graphics2D graphics = img.createGraphics();
58. // clear the drawing area
59. graphics.setPaint(Color.white);
60. graphics.fill(new Rectangle2D.Float(0, 0, pgsize.width, pgsize.height));
62. // render
63. ppt.getSlides().get(i).draw(graphics);
65. // save the output
66. String filename = "D:/demo/07-" + (i+1) + ".jpg";
67. System.out.println(filename);
68. FileOutputStream out = new FileOutputStream(filename);
69. javax.imageio.ImageIO.write(img, "png", out);
70. out.close();
71. } catch (Exception e) {
72. System.out.println("第"+i+"张ppt转换出错");
73. }
74. }
75. System.out.println("7success");
76. }
78. public static void toImage2003(){
79. try {
80. HSLFSlideShow ppt = new HSLFSlideShow(new HSLFSlideShowImpl("D:/demo/22.ppt"));
82. Dimension pgsize = ppt.getPageSize();
83. for (int i = 0; i **<** **ppt.getSlides**().size(); i++) {
84. //防止中文乱码
85. for(HSLFShape shape : ppt.getSlides().get(i).getShapes()){
86. if(shape instanceof HSLFTextShape) {
87. HSLFTextShape tsh = (HSLFTextShape)shape;
88. for(HSLFTextParagraph p : tsh){
89. for(HSLFTextRun r : p){
90. r.setFontFamily("宋体");
91. }
92. }
93. }
94. }
95. BufferedImage img = new BufferedImage(pgsize.width, pgsize.height, BufferedImage.TYPE\_INT\_RGB);
96. Graphics2D graphics = img.createGraphics();
97. // clear the drawing area
98. graphics.setPaint(Color.white);
99. graphics.fill(new Rectangle2D.Float(0, 0, pgsize.width, pgsize.height));
101. // render
102. ppt.getSlides().get(i).draw(graphics);
104. // save the output
105. String filename = "D:/demo/03-760-" + (i+1) + ".jpg";
106. System.out.println(filename);
107. FileOutputStream out = new FileOutputStream(filename);
108. javax.imageio.ImageIO.write(img, "png", out);
109. out.close();
110. //              resizeImage(filename, filename, width, height);
112. }
113. System.out.println("3success");
114. } catch (Exception e) {
115. // TODO: handle exception
116. }
117. }
118. /\*\*\*
119. \* 功能 :调整图片大小
120. \* @param srcImgPath 原图片路径
121. \* @param distImgPath  转换大小后图片路径
122. \* @param width   转换后图片宽度
123. \* @param height  转换后图片高度
124. \*/
125. public static void resizeImage(String srcImgPath, String distImgPath,
126. int width, int height) throws IOException {
128. File srcFile = new File(srcImgPath);
129. Image srcImg = ImageIO.read(srcFile);
130. BufferedImage buffImg = null;
131. buffImg = new BufferedImage(width, height, BufferedImage.TYPE\_INT\_RGB);
132. buffImg.getGraphics().drawImage(
133. srcImg.getScaledInstance(width, height, Image.SCALE\_SMOOTH), 0,
134. 0, null);
136. ImageIO.write(buffImg, "JPEG", new File(distImgPath));
138. }
139. }

[http://static.blog.csdn.net/images/save_snippets.png](javascript:;)