**如何用JAVA编写将PDF转换为WORD稳当 给个思路**

2010-06-30 09:21 [ljqenzqq](http://www.baidu.com/p/ljqenzqq?from=zhidao) | 分类：[办公软件](http://zhidao.baidu.com/browse/864) | 浏览3067次

我知道有软件能转换 所以不用告诉我去下什么软件 谢谢   
只需要个思路

分享到：

2010-06-30 09:32

提问者采纳

需要用到插件jacob，自己去下载吧。  
import com.jacob.activeX.ActiveXComponent;  
import com.jacob.com.ComThread;  
import com.jacob.com.Dispatch;  
import com.jacob.com.Variant;  
 public class D2P {  
 private ActiveXComponent wordCom = null;  
  
 private Object wordDoc = null;  
  
 private final Variant False = new Variant(false);  
  
 private final Variant True = new Variant(true);  
  
 /\*\* \*//\*\* \*//\*\* \*//\*\*  
 \* 打开word文档  
 \*   
 \* @param filePath  
 \* word文档  
 \* @return 返回word文档对象  
 \*/  
 public boolean openWord(String filePath) {  
 //建立ActiveX部件  
 wordCom = new ActiveXComponent("Word.Application");  
  
 try {  
 //返回wrdCom.Documents的Dispatch  
 Dispatch wrdDocs = wordCom.getProperty("Documents").toDispatch();  
 //调用wrdCom.Documents.Open方法打开指定的word文档，返回wordDoc  
 wordDoc = Dispatch.invoke(wrdDocs, "Open", Dispatch.Method,  
 new Object[] { filePath }, new int[1]).toDispatch();  
 return true;  
 } catch (Exception ex) {  
 ex.printStackTrace();  
 }  
 return false;  
 }  
  
 /\*\* \*//\*\* \*//\*\* \*//\*\*  
 \* 关闭word文档  
 \*/  
 public void closeWord() {  
 //关闭word文件  
 wordCom.invoke("Quit", new Variant[] {});  
 }  
  
 /\*\* \*//\*\* \*//\*\* \*//\*\*  
 \* \* 将word文档打印为PS文件后，使用Distiller将PS文件转换为PDF文件 \*  
 \*   
 \* @param sourceFilePath  
 \* 源文件路径 \*  
 \* @param destinPSFilePath  
 \* 首先生成的PS文件路径 \*  
 \* @param destinPDFFilePath  
 \* 生成PDF文件路径  
 \*/  
 public void docToPDF(String sourceFilePath, String destinPSFilePath,  
 String destinPDFFilePath) {  
 if (!openWord(sourceFilePath)) {  
 closeWord();  
 return;  
 }  
 //建立Adobe Distiller的com对象  
 ActiveXComponent distiller = new ActiveXComponent(  
 "PDFDistiller.PDFDistiller.1");  
 try {  
 //设置当前使用的打印机，我的Adobe Distiller打印机名字为"Adobe PDF"  
 wordCom.setProperty("ActivePrinter", new Variant("MS Publisher Color Printer"));  
 //设置printout的参数，将word文档打印为postscript文档。目前只使用了前5个参数，如果要使用更多的话可以参考MSDN的office开发相关api  
 //是否在后台运行  
 Variant Background = False;  
 //是否追加打印  
 Variant Append = False;  
 //打印所有文档  
 int wdPrintAllDocument = 0;  
 Variant Range = new Variant(wdPrintAllDocument);  
 //输出的postscript文件的路径  
 Variant OutputFileName = new Variant(destinPSFilePath);  
  
 Dispatch.callN((Dispatch) wordDoc, "PrintOut", new Variant[] {  
 Background, Append, Range, OutputFileName });  
 System.out.println("由word文档打印为ps文档成功！");  
 //调用Distiller对象的FileToPDF方法所用的参数，详细内容参考Distiller Api手册  
 //作为输入的ps文档路径  
 Variant inputPostScriptFilePath = new Variant(destinPSFilePath);  
 //作为输出的pdf文档的路径  
 Variant outputPDFFilePath = new Variant(destinPDFFilePath);  
 //定义FileToPDF方法要使用adobe pdf设置文件的路径，在这里没有赋值表示并不使用pdf配置文件  
 Variant PDFOption = new Variant("");  
 //调用FileToPDF方法将ps文档转换为pdf文档  
 Dispatch.callN(distiller, "FileToPDF", new Variant[] {  
 inputPostScriptFilePath, outputPDFFilePath, PDFOption });  
 System.out.println("由ps文档转换为pdf文档成功！");  
 } catch (Exception ex) {  
 ex.printStackTrace();  
 } finally {  
 closeWord();  
 wordCom=null;   
 //释放在程序线程中引用的其它com，比如Adobe PDFDistiller   
 ComThread.Release();   
 }  
 }  
  
 public static void main(String[] argv) {  
 D2P d2p = new D2P();  
 d2p.docToPDF("d:/12.doc", "d:/1p.ps", "d:/1p.pdf");  
 //这里是你建一个叫12.doc的word文档，生成的文档将在D盘下  
 //1p.ps和1p.pdf（这是我们要的）  
 }  
}