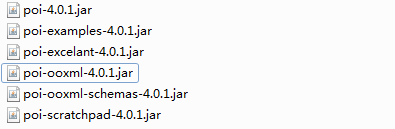
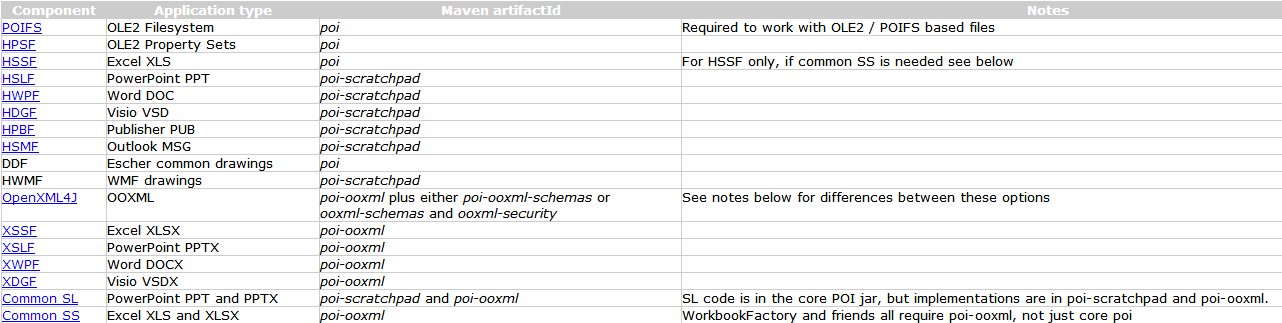
# Jar分类





# HWPF And XWPF

HWPF和XWPF提供了相似的功能，但是它们之间没有通用的接口。

HWPF只对旧的Word 6 和Word 95文件格式提供只读的功能

HWPF包含再poi-scratchpad-xxx.jar中

XWPF包含再poi-ooxml-xxx.jar中。

XWPF对.docx文件的读和写，提供了核心的API，但是它并不完全。对于某些东西，我们可能需要深入研究底层的XMLBeans对象，去操作ooxml。

## HWPF

HWPF入口类是：HWPFDocument。

## XWPF

XWPF的入口类是：XWPFDocument。

# HSSF And XSSF

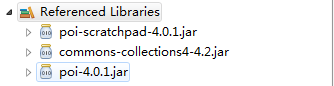
## HSSF

## XSSF

# Word

## doc

需要的jar包：



1. 读取word文件内容，但是只能读取纯文字，无法读取word中的图片和表格，

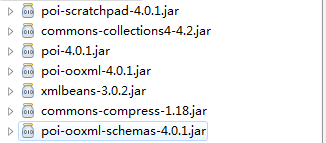
|  |
| --- |
| **import** java.io.FileInputStream;  **import** java.io.FileNotFoundException;  **import** java.io.IOException;  **import** java.io.InputStream;  **import** org.apache.poi.hwpf.extractor.WordExtractor;  **public** **class** Demo2  {  **public** **static** **void** main(String[] args)  {  **try**(InputStream input = **new** FileInputStream("F:\\Program Files\\demo.doc"))  {  WordExtractor we = **new** WordExtractor(input) ;  System.***out***.println(we.getText());  } **catch** (FileNotFoundException e) {  e.printStackTrace();  } **catch** (IOException e) {  e.printStackTrace();  }  }  } |

1. 把图片从word中读出，保存为jpg格式或其他格式。

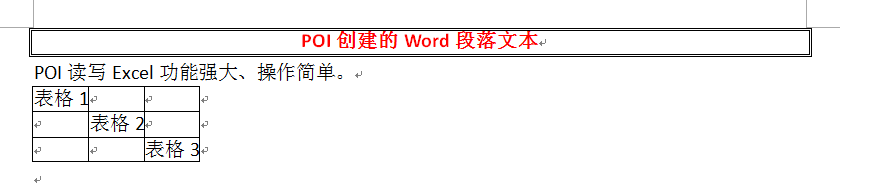
|  |
| --- |
| **import** org.apache.poi.hwpf.HWPFDocument;  **import** org.apache.poi.hwpf.model.PicturesTable;  **import** org.apache.poi.hwpf.usermodel.Picture;  **public** **class** Demo2  {  **public** **static** **void** main(String[] args)  {  **try**(InputStream input = **new** FileInputStream("F:\\Program Files\\demo.doc"))  {  HWPFDocument doc = **new** HWPFDocument(input);  PicturesTable pt = doc.getPicturesTable() ;  List<Picture> pictures = pt.getAllPictures();  **for**(Picture p : pictures)  {  p.writeImageContent(**new** FileOutputStream(**new** File("F:\\Program Files\\hao.png")));  }  } **catch** (FileNotFoundException e) {  e.printStackTrace();  } **catch** (IOException e) {  e.printStackTrace();  }  }  } |

1. 创建doc文件。

依赖的jar包：



|  |
| --- |
| **import** java.io.FileOutputStream;  **import** java.io.IOException;  **import** org.apache.poi.xwpf.usermodel.Borders;  **import** org.apache.poi.xwpf.usermodel.ParagraphAlignment;  **import** org.apache.poi.xwpf.usermodel.XWPFDocument;  **import** org.apache.poi.xwpf.usermodel.XWPFParagraph;  **import** org.apache.poi.xwpf.usermodel.XWPFRun;  **import** org.apache.poi.xwpf.usermodel.XWPFTable;  **public** **class** Demo1  {  **public** **static** **void** main(String[] args)  {  //创建一个word文件  XWPFDocument doc = **new** XWPFDocument() ;  //设置段落  XWPFParagraph p = doc.createParagraph() ;  // 设置段落的对齐方式  p.setAlignment(ParagraphAlignment.***CENTER***);  //设置下边框  p.setBorderBottom(Borders.***DOUBLE***);  //设置上边框  p.setBorderTop(Borders.***DOUBLE***);  //设置右边框  p.setBorderRight(Borders.***DOUBLE***);  //设置左边框  p.setBorderLeft(Borders.***DOUBLE***);  //创建段落文本  XWPFRun r = p.createRun();  r.setText("POI创建的Word段落文本");  r.setBold(**true**);//设置为粗体  r.setColor("FF0000");//设置颜色  p = doc.createParagraph();// 新建一个段落  r = p.createRun();  r.setText("POI读写Excel功能强大、操作简单。");  XWPFTable table= doc.createTable(3, 3);//创建一个表格  table.getRow(0).getCell(0).setText("表格1");  table.getRow(1).getCell(1).setText("表格2");  table.getRow(2).getCell(2).setText("表格3");  **try**(FileOutputStream out = **new** FileOutputStream("demo1.doc"))  {  doc.write(out);  } **catch** (IOException e) {  e.printStackTrace();  }  }  } |



## docx