**POI报表**

--Jakata Poi HSSF：纯java的Excel解决方案

在我们实际的开发中，表现层的解决方案虽然有多样，但是IE浏览器已成为最多人使用的浏览器，因为大家都用Windows。在企业办公系统中，常常有客户这样子要求：你要把我们的报表直接用Excel打开(电信系统、银行系统)。或者是：我们已经习惯用Excel打印。

Apache的Jakata项目的POI子项目，目前比较成熟的是HSSF接口，处理MSExcel对象。它不象我们仅仅是用csv生成的没有格式的可以由Excel转换的东西，而是真正的Excel对象，你可以控制一些属性如sheet,cell等等。

首先，**理解一下一个Excel的文件的组织形式，一个Excel文件对应于一个workbook(HSSFWorkbook)，一个workbook可以有多个sheet（HSSFSheet）组成，一个sheet是由多个row（HSSFRow）组成，一个row是由多个cell（HSSFCell）组成。**

POI可以到www.apache.org下载到。实际运行时，需要有poi包就可以了。HSSF提供给用户使用的对象在rg.apache.poi.hssf.usermodel包中,主要部分包括Excel对象，样式和格式，还有辅助操作。

有以下几种对象：

HSSFWorkbook       excel的文档对象

HSSFSheet         excel的表单

HSSFRow         excel的行

HSSFCell         excel的格子单元

HSSFFont         excel字体

HSSFDataFormat       日期格式

在poi1.7中才有以下2项：

HSSFHeader         sheet头

HSSFFooter         sheet尾（只有打印的时候才能看到效果） 和这个样式

HSSFCellStyle        cell样式 辅助操作包括

HSSFDateUtil        日期

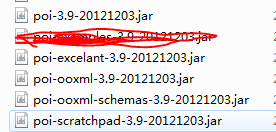
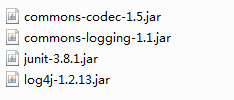
HSSFPrintSetup       打印

HSSFErrorConstants      错误信息表

最简单的例子：

第一步导包：

在下载成的poi包中可以找到以下包

 这些包需要就可以导

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** org.apache.poi.hssf.usermodel.HSSFCell;

**import** org.apache.poi.hssf.usermodel.HSSFRow;

**import** org.apache.poi.hssf.usermodel.HSSFSheet;

**import** org.apache.poi.hssf.usermodel.HSSFWorkbook;

**public** **class** Test {

**public** **static** **void** main(String[] args){

HSSFWorkbook workBook = **new** HSSFWorkbook();

//新建一个空工作簿

HSSFSheet sheet1 = workBook.createSheet();

HSSFSheet sheet2 = workBook.createSheet();

//创建工作单元 1 2

HSSFRow row1 = sheet1.createRow(0);

HSSFRow row4 = sheet2.createRow(0);

//创建一行

HSSFCell cell00 = row1.createCell(0);

HSSFCell cell01 = row1.createCell(1);

HSSFCell cell02 = row1.createCell(2);

//创建一格

cell00.setCellValue("编号");

cell01.setCellValue("学生姓名");

cell02.setCellValue("性别");

HSSFCell cell10 = row4.createCell(0);

HSSFCell cell11 = row4.createCell(1);

HSSFCell cell12 = row4.createCell(2);

cell10.setCellValue("编号");

cell11.setCellValue("姓名");

cell12.setCellValue("工资");//给每一格赋值

FileOutputStream fos = **null**;

**try** {//FileOutputStream创建文件存储文件

fos = **new** FileOutputStream("D:\\aa.xls");

workBook.write(fos);//将流写入进去

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}**finally**{

**try** {

fos.flush();

fos.close();

} **catch** (IOException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

运行后结果：



对于Struts2与poi处理

由于struts2中action是一个普通的类，没有像servlet中outputStream（response）

OutputStream se = response.getOutputStream();

Response本身是java的类，主要负责服务器的响应发送给服务（html代码）

Servlet是一个和response是一个不同的类