

2009-06-04

## [利用POI生成Excel并通过Servlet下载示例](#)

关键字: **java**

Java代码

```
1. package com.ljz;  
2.  
3. import java.io.BufferedInputStream;  
4. import java.io.BufferedOutputStream;  
5. import java.io.ByteArrayInputStream;  
6. import java.io.ByteArrayOutputStream;  
7. import java.io.IOException;  
8. import java.io.InputStream;  
9.  
10. import javax.servlet.ServletException;  
11. import javax.servlet.ServletOutputStream;  
12. import javax.servlet.http.HttpServlet;  
13. import javax.servlet.http.HttpServletRequest;  
14. import javax.servlet.http.HttpServletResponse;  
15.  
16. import org.apache.poi.hssf.usermodel.HSSFCell;  
17. import org.apache.poi.hssf.usermodel.HSSFRow;  
18. import org.apache.poi.hssf.usermodel.HSSFSheet;  
19. import org.apache.poi.hssf.usermodel.HSSFWorkbook;  
20.  
21. public class downloadExcelServlet extends HttpServlet {  
22.  
23.     private static final long serialVersionUID = 1L;  
24.  
25.     public downloadExcelServlet() {  
26.         super();  
27.     }  
28.  
29.     public void destroy() {
```

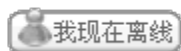


**ljzcome**

浏览: 336 次

性别:

来自: 广州



[详细资料](#)

[留言簿](#)

搜索本博客

最近访客  
客

[>>更多访客](#)



[coolhorse168](#) [唯一ID](#)



[znass](#)



[tangshd](#)

博客分类

- [全部博客 \(1\)](#)

其他分类

- [我的收藏 \(1\)](#)
- [我的论坛帖子 \(0\)](#)
- [我的精华良好贴 \(0\)](#)

最近加入圈子

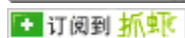
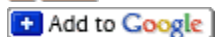
存档

- [2009-06 \(1\)](#)
- [更多存档...](#)

最新评论

评论排行榜

- [利用POI生成Excel并通  
过Servlet下载示例](#)



[\[什么是RSS?\]](#)

```
30.         super.destroy();
31.     }
32.
33.     private InputStream getInputStream() {
34.         HSSFWorkbook wb = new HSSFWorkbook();
35.         HSSFSheet sheet = wb.createSheet("sheet1");
36.
37.         HSSFRow row = sheet.createRow(0);
38.
39.         HSSFCell cell = row.createCell((short) 0);
40.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
41.         cell.setCellValue("序号");
42.
43.         cell = row.createCell((short) 1);
44.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
45.         cell.setCellValue("姓");
46.
47.         cell = row.createCell((short) 2);
48.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
49.         cell.setCellValue("名");
50.
51.         cell = row.createCell((short) 3);
52.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
53.         cell.setCellValue("年龄");
54.
55.         // 创建一行记录
56.         row = sheet.createRow(1);
57.
58.         cell = row.createCell((short) 0);
59.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
60.         cell.setCellValue("1");
61.
62.         cell = row.createCell((short) 1);
63.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
64.         cell.setCellValue("刘");
65.
66.         cell = row.createCell((short) 2);
67.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
68.         cell.setCellValue("继忠");
69.
70.         cell = row.createCell((short) 3);
71.         cell.setEncoding(HSSFCell.ENCODING_UTF_16);
72.         cell.setCellValue("25");
73.
```

```

74.         // List<User> list = this.findAll();
75.         //
76.         // for (int i = 0; i < list.size(); ++i)
77.         // {
78.         //     User user = list.get(i);
79.         //
80.         //     row = sheet.createRow(i + 1);
81.         //
82.         //     cell = row.createCell((short) 0);
83.         //     cell.setEncoding(HSSFCell.ENCODING_UTF_16);
84.         //     cell.setCellValue(i + 1);
85.         //
86.         //     cell = row.createCell((short) 1);
87.         //     cell.setEncoding(HSSFCell.ENCODING_UTF_16);
88.         //     cell.setCellValue(user.getFirstname());
89.         //
90.         //     cell = row.createCell((short) 2);
91.         //     cell.setEncoding(HSSFCell.ENCODING_UTF_16);
92.         //     cell.setCellValue(user.getLastname());
93.         //
94.         //     cell = row.createCell((short) 3);
95.         //     cell.setEncoding(HSSFCell.ENCODING_UTF_16);
96.         //     cell.setCellValue(user.getAge());
97.         // }
98.
99.         ByteArrayOutputStream os = new ByteArrayOutputStream();
100.
101.         try {
102.             wb.write(os);
103.         } catch (IOException e) {
104.             e.printStackTrace();
105.         }
106.
107.         byte[] content = os.toByteArray();
108.         InputStream is = new ByteArrayInputStream(content);
109.         return is;
110.     }
111.
112.     public void doGet(HttpServletRequest request, HttpServletResponse response)
113.         throws ServletException, IOException {
114.         response.reset();
115.         response.setContentType("application/vnd.ms-excel;charset=utf-8");
116.         response.setHeader("Content-Disposition", "attachment;filename="

```

```
117.         + new String("系统日志.xls".getBytes(), "iso-8859-1"));
118.
119.         ServletOutputStream out = response.getOutputStream();
120.
121.         BufferedInputStream bis = null;
122.         BufferedOutputStream bos = null;
123.
124.         try {
125.
126.             bis = new BufferedInputStream(this.getInputStream());
127.             bos = new BufferedOutputStream(out);
128.
129.             byte[] buff = new byte[2048];
130.             int bytesRead;
131.
132.             // Simple read/write loop.
133.             while (-1 != (bytesRead = bis.read(buff, 0, buff.length))) {
134.                 bos.write(buff, 0, bytesRead);
135.             }
136.
137.         } catch (final IOException e) {
138.             System.out.println("IOException.");
139.             throw e;
140.         } finally {
141.             if (bis != null)
142.                 bis.close();
143.             if (bos != null)
144.                 bos.close();
145.         }
146.     }
147.
148.     public void doPost(HttpServletRequest request, HttpServletResponse response)
149.         throws ServletException, IOException {
150.
151.     }
152.
153.     public void init() throws ServletException {
154.         // Put your code here
155.     }
156.
157. }
```

[DownloadExcel.rar](#) (732.9 KB)  
下载次数: 27

[EZ JCom: Excel Java API](#) [鎢將垂寮€鎢愁ava鎢 鈥 鎢學eb鎢 鈥](#)  
Complete Java API for MS Excel via the Excel寮€鎢愁eb鎢 鈥 涓嶇灏�Hibernate, 涓嶇灏�鎶€  
COM Automation interface. 棧QL Grid琛 Group琛 Cross琛 紝鎢將垂

鍔將垂寮€鑒熬ava鐳ヨ" 鐳學eb鐳ヨ"

寮€鎼€eb鐳ヨ〃涓€嶃€€Hibernate,涓€嶃€€鐳€  
 棣�QL Grid琛 Group琛 Cross琛 绾€€鐳€鐳€

1

踩





## 评论





表情图标





字体颜色:





字体大小:





对齐:





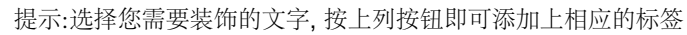






提示:选择您需要装饰的文字, 按上列按钮即可添加上相应的标签

字体颜色:  字体大小:  对齐:



您还没有登录,请[登录](#)后发表评论(快捷键 Alt+S / Ctrl+Enter)

---

声明:JavaEye 文章版权属于作者,受法律保护。没有作者书面许可不得转载。若作者同意转载,必须以超链接形式标明文章原始出处和作者。

© 2003-2009 JavaEye.com. All rights reserved. 上海炯耐计算机软件有限公司 [ 沪ICP备05023328号 ]