



链滴

XLSransformer + 模板 导出 EXCAL

作者: [zhaohaoxin](#)

原文链接: <https://ld246.com/article/1597853701443>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



@[TOC](XLSransformer + 模板 导出 EXCAL)

官网链接 : <http://jxls.sourceforge.net>

准备一个EXCAL模板

新建一个EXCAL模板文件,在模板文件中写好相应的列头,也可以使用组织的参数循环生成,模板中可以置字文本框,字体等样式(使用和正常操作EXCAL相同)

	A	B	C
1	id	name	age
2	<code><jx:forEach items="{数据Object}" var="item" ></code>		
3	<code>{item.id}</code>	<code>{item.name}</code>	<code>{item.age}</code>
4	<code></jx:forEach></code>		
5			
6			

模板标签

jx: if标签

<jx:if> 标签的典型用法如下

jx: if标记可用于根据某些条件从导出中排除某些列。如果将打开和关闭jx:if标签放在同一行中,则jXL将根据测试条件处理或除去标签主体中包含的列。

```
<jx:if test="{department.chief.payment > 2000.0}">
  Chief Name: ${department.chief.name}
</jx:if>
```

jx: outline标签

<jx:outline> 标记可让您对特定的行集进行分组。例如有

```
<jx:outline>
  //any rows can go here
</jx:outline>
```

jx: group标记正文中的所有excel行将被分组。

<jx:outline> 标记具有一个可选的布尔值属性，用于detail指示分组行的初始状态-将其展开显示还是叠。默认值是false 指分组的行将被折叠（或隐藏）。

jx: out标签

jx: out标签看起来像这样

```
<jx:out expr="expression" />
```

该标记可以在表达式可以正常使用的任何地方使用，只需将JEXL表达式作为属性值即可。

jx: forEach标签

使用jx: forEach标签对数据进行分组

jx:forEach标记允许您通过基础bean属性之一对收集数据进行分组。这可以通过groupBy和groupOrder属性来完成，这些属性指定应使用哪个属性对数据进行分组以及应如何对组进行排序。例如

```
//可以forEach标签也可以嵌套使用
<jx:forEach items="{employees}" groupBy="age">
  Age: ${group.item.age}
  <jx:forEach items="{group.items}" var="employee">
    ${employee.name} | ${employee.payment} | ${employee.bonus}
  </jx:forEach>
</jx:forEach>
```

筛选收集项目

您可以使用jx: forEach标记的'select'属性来控制哪些项目包含在循环中。例如，如果我们只想包含款超过2000的员工，则可以使用下一个语法

```
<jx:forEach items="{employees}" var="employee" select="{employee.payment > 2000}">
  ${employee.name} | ${employee.payment} | ${employee.bonus}
</jx:forEach>
```

varStatus属性

jax:forEach标记支持varStatus属性, 该属性定义循环状态对象的名称, 该名称将为每次迭代传递到bean上下文中。循环状态对象是LoopStatus类的实例, 该类具有单个“index”属性, 用于标识要迭的集合中当前项目的索引。

```
<jx:forEach items="{employees}" var="employee" varStatus="status">
  | ${status.index} | ${employee.name} | ${employee.payment} | ${employee.bonus}
</jx:forEach>
```

通过模板导出EXCAL

```
HttpServletRequest request;
HttpServletResponse response;
XLSTransformer transformer = new XLSTransformer();

//使用1 导出模板保存在指定位置
transformer.transformXLS("模板地址",数据Object,"保存地址");
return;

//使用2(推荐) 通过模板生Workbook 通过流下载
Workbook workbook = transformer.transformXLS(new FileInputStream("模板地址"),数据Object);
String realName = "下载的EXCAL名称";
String userAgent = request.getHeader("User-Agent");
if(StringUtils.isNotEmpty(userAgent)){
    userAgent = userAgent.toLowerCase();
    if(userAgent.indexOf("msie") != -1){
        realName = URLEncoder.encode(realName,"UTF-8");
    }else{
        realName = new String(realName.getBytes("UTF-8"),"ISO8859-1");
    }
}
response.setContentType = "application/octet-stream;charset=UTF-8";
response.setHeader("Content-disposition","attachment; filename=" + realName);
workbook.write(response.getOutputStream());
workbook.close();
return;
```

作者邮箱:1903327874@qq.com