

JSP 标准标签库 (JSTL)

JSP标准标签库 (JSTL) 是一个JSP标签集合，它封装了JSP应用的通用核心功能。

JSTL支持通用的、结构化的任务，比如迭代，条件判断，XML文档操作，国际化标签，SQL标签。除了这些，它还提供了一个框架来使用集成JSTL的自定义标签。

根据JSTL标签所提供的功能，可以将其分为5个类别。

- 核心标签
- 格式化标签
- SQL 标签
- XML 标签
- JSTL 函数

JSTL 库安装

Apache Tomcat安装JSTL 库步骤如下：

从Apache的标准标签库中下载的二进包(jakarta-taglibs-standard-current.zip)。

- 官方下载地址：<http://archive.apache.org/dist/jakarta/taglibs/standard/binaries/>
- 本站下载地址：[jakarta-taglibs-standard-1.1.2.zip](#)

下载 **jakarta-taglibs-standard-1.1.2.zip** 包并解压，将 **jakarta-taglibs-standard-1.1.2/lib/** 下的两个 jar 文件：**standard.jar** 和 **jstl.jar** 文件拷贝到 **/WEB-INF/lib/** 下。

将 **tld** 下的需要引入的 **tld** 文件复制到 **WEB-INF** 目录下。

接下来我们在 **web.xml** 文件中添加以下配置：

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
  xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
    http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <jsp-config>
    <taglib>
      <taglib-uri>http://java.sun.com/jsp/jstl/fmt</taglib-uri>
      <taglib-location>/WEB-INF/fmt.tld</taglib-location>
    </taglib>
    <taglib>
      <taglib-uri>http://java.sun.com/jsp/jstl/fmt-rt</taglib-uri>
      <taglib-location>/WEB-INF/fmt-rt.tld</taglib-location>
    </taglib>
```

```
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/core</taglib-uri>
<taglib-location>/WEB-INF/c.tld</taglib-location>
</taglib>
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/core-rt</taglib-uri>
<taglib-location>/WEB-INF/c-rt.tld</taglib-location>
</taglib>
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/sql</taglib-uri>
<taglib-location>/WEB-INF/sql.tld</taglib-location>
</taglib>
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/sql-rt</taglib-uri>
<taglib-location>/WEB-INF/sql-rt.tld</taglib-location>
</taglib>
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/x</taglib-uri>
<taglib-location>/WEB-INF/x.tld</taglib-location>
</taglib>
<taglib>
<taglib-uri>http://java.sun.com/jsp/jstl/x-rt</taglib-uri>
<taglib-location>/WEB-INF/x-rt.tld</taglib-location>
</taglib>
</jsp-config>
</web-app>
```

使用任何库，你必须在每个 JSP 文件中的头部包含 `<taglib>` 标签。

核心标签

核心标签是最常用的 JSTL 标签。引用核心标签库的语法如下：

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
```

标签	描述
<code><c:out></code>	用于在JSP中显示数据，就像<%= ... >
<code><c:set></code>	用于保存数据
<code><c:remove></code>	用于删除数据
<code><c:catch></code>	用来处理产生错误的异常状况，并且将错误信息储存起来
<code><c:if></code>	与我们在一般程序中用的if一样

<c:choose>	本身只当做<c:when>和<c:otherwise>的父标签
<c:when>	<c:choose>的子标签，用来判断条件是否成立
<c:otherwise>	<c:choose>的子标签，接在<c:when>标签后，当<c:when>标签判断为false时被执行
<c:import>	检索一个绝对或相对 URL，然后将其内容暴露给页面
<c:forEach>	基础迭代标签，接受多种集合类型
<c:forTokens>	根据指定的分隔符来分隔内容并迭代输出
<c:param>	用来给包含或重定向的页面传递参数
<c:redirect>	重定向至一个新的URL.
<c:url>	使用可选的查询参数来创建一个URL

格式化标签

JSTL 格式化标签用来格式化并输出文本、日期、时间、数字。引用格式化标签库的语法如下：

```
<%@ taglib prefix="fmt"
    uri="http://java.sun.com/jsp/jstl/fmt" %>
```

标签	描述
<fmt:formatNumber>	使用指定的格式或精度格式化数字
<fmt:parseNumber>	解析一个代表着数字，货币或百分比的字符串
<fmt:formatDate>	使用指定的风格或模式格式化日期和时间
<fmt:parseDate>	解析一个代表着日期或时间的字符串
<fmt:bundle>	绑定资源
<fmt:setLocale>	指定地区
<fmt:setBundle>	绑定资源
<fmt:timeZone>	指定时区
<fmt:setTimeZone>	指定时区
<fmt:message>	显示资源配置文件信息

<fmt:requestEncoding>

设置request的字符编码

SQL 标签

JSTL SQL 标签库提供了与关系型数据库（ Oracle ， MySQL ， SQL Server 等等 ）进行交互的标签。引用SQL标签库的语法如下：

```
<%@ taglib prefix="sql"
    uri="http://java.sun.com/jsp/jstl/sql" %>
```

标签	描述
<sql:setDataSource>	指定数据源
<sql:query>	运行SQL查询语句
<sql:update>	运行SQL更新语句
<sql:param>	将SQL语句中的参数设为指定值
<sql:dateParam>	将SQL语句中的日期参数设为指定的java.util.Date 对象值
<sql:transaction>	在共享数据库连接中提供嵌套的数据库行为元素，将所有语句以一个事务的形式来运行

XML 标签

JSTL XML 标签库提供了创建和操作XML文档的标签。引用XML标签库的语法如下：

```
<%@ taglib prefix="x"
    uri="http://java.sun.com/jsp/jstl/xml" %>
```

在使用xml标签前，你必须将XML 和 XPath 的相关包拷贝至你的<Tomcat 安装目录>\lib下：

- **XercesImpl.jar**
下载地址：<http://www.apache.org/dist/xerces/j/>
- **xalan.jar**
下载地址：<http://xml.apache.org/xalan-j/index.html>

标签	描述
<x:out>	与<%= ... >,类似，不过只用于XPath表达式
<x:parse>	解析 XML 数据
<x:set>	设置XPath表达式

<x:if>	判断XPath表达式，若为真，则执行本体中的内容，否则跳过本体
<x:forEach>	迭代XML文档中的节点
<x:choose>	<x:when>和<x:otherwise>的父标签
<x:when>	<x:choose>的子标签，用来进行条件判断
<x:otherwise>	<x:choose>的子标签，当<x:when>判断为false时被执行
<x:transform>	将XSL转换应用在XML文档中
<x:param>	与<x:transform>共同使用，用于设置XSL样式表

JSTL函数

JSTL包含一系列标准函数，大部分是通用的字符串处理函数。引用JSTL函数库的语法如下：

```
<%@ taglib prefix="fn"
    uri="http://java.sun.com/jsp/jstl/functions" %>
```

函数	描述
fn:contains()	测试输入的字符串是否包含指定的子串
fn:containsIgnoreCase()	测试输入的字符串是否包含指定的子串，大小写不敏感
fn:endsWith()	测试输入的字符串是否以指定的后缀结尾
fn:escapeXml()	跳过可以作为XML标记的字符
fn:indexOf()	返回指定字符串在输入字符串中出现的位置
fn:join()	将数组中的元素合成一个字符串然后输出
fn:length()	返回字符串长度
fn:replace()	将输入字符串中指定的位置替换为指定的字符串然后返回
fn:split()	将字符串用指定的分隔符分隔然后组成一个子字符串数组并返回
fn:startsWith()	测试输入字符串是否以指定的前缀开始
fn:substring()	返回字符串的子集
fn:substringAfter()	返回字符串在指定子串之后的子集

<code>fn:substringBefore()</code>	返回字符串在指定子串之前的子集
<code>fn:toLowerCase()</code>	将字符串中的字符转为小写
<code>fn:toUpperCase()</code>	将字符串中的字符转为大写
<code>fn:trim()</code>	移除首位的空白符



1 篇笔记

 写笔记



JSTL 1.1 与 JSTL 1.2 之间的区别？如何下载 JSTL 1.2？

JSTL 1.2 中不要求 standard.jar 包。

您可以在 Maven 中央仓库中找到它们。

<http://repo2.maven.org/maven2/javax/servlet/jstl/>

<http://repo2.maven.org/maven2/taglibs/standard/>

由于JSTL 1.1已经过时，Apache已将其置于**存档中**。选择jakarta-taglibs-standard-current.zip文件。但是，如果您正在运行Servlet 2.5兼容容器并且web.xml声明为至少Servlet 2.5，那么您应该能够使用新的**JSTL 1.2**。需要注意的是JSTL 1.2并没有要求standard.jar。

stackoverflow 6个月前 (09-19)