

Struts 标签 —— Bean 标签

JSP 开发之中的要求，程序中尽可能少的出现的 Scriptlet 代码。主要是因为可维护性相对较好。提供了大量的自定义的标签库，通过标签库可以简化 JSP 的页面代码开发。

Bean 标签：主要与 JavaBean 有关

回顾：

```
<jsp:useBean id="对象名称" scope="保存范围（四种属性范围）" class="包.类"/>
<jsp:setProperty name="id（jsp:useBean 中使用）" property="属性"/>
<jsp:getProperty name="id" property="属性"/>
```

只要是通过标签设置的对象，则自动将对象保存在一个固定的范围之中，由 scope 指定一般对象都使用 “id” 进行表示
如果要使用对象，在标签通过 “name” 属性完成的

1、Bean 标签

在 Struts 中提供了一系列的与 JavaBean 有关的操作标签，称为 Bean 标签。

1.1、<bean:define>标签

定义或复制一个对象的

定义对象一般为 String 类型。

复制对象 → 迭代标签

```
<bean:define id="str" value="Hello MLDN..."/>
<h1>${str}</h1>
```

如果可以通过 EL 进行访问，则表示 “str” 保存在了四种属性范围之中，等同与下面的语句：

```
String str = "Hello MLDN" ;
pageContext.setAttribute("str",str) ;
```

如果不使用 EL 该如何输出呢？通过 Struts 中的 <bean:write> 进行打印

```
<bean:write name="str"/>
```

```
<%@ page language="java" pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>
  <body>
    <bean:define id="str" value="Hello MLDN..."/>
```

```
<h1>${str}</h1>
<h1><bean:write name="str"/></h1>
</body>
</html:html>
```

强调:

- **id**: 就表示一个存放在四种属性范围之中的名称
- **name**: 表示使用一个存放在四种属性范围中的对象

1.2、<bean:size>标签

- 求出长度: 数组、Collection、Map

标签肯定数据存放在四种属性范围之中。

输出 Map 的长度:

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>

  <body>
    <%
      Map m = new HashMap() ;
      m.put("one","1");
      m.put("two","2");
      m.put("three","3");

      // 将 Map 对象保存在四种属性范围之中
      request.setAttribute("namemap",m);
    %>
    <bean:size id="len" name="namemap" scope="request"/>
    <h1>长度是: ${len}</h1>
  </body>
</html:html>
```

输出 Collection 的长度:

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
```

```
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>

  <body>
    <%
      Collection col = new ArrayList();
      col.add("MLDN");
      col.add("lxh");
      col.add("mole");

      // 将 Map 对象保存在四种属性范围之中
      request.setAttribute("namecol",col);
    %>
    <bean:size id="len" name="namecol" scope="request"/>
    <h1>长度是: ${len}</h1>
  </body>
</html:html>
```

求长度的时候，即可以是 Map 也可以是 Collection，但是对于 JSP 调用代码是不变的。

1.3、<bean:write>标签

打印对象，也可以打印对象中的属性。

在 JSP 2.0 中可以使用 EL 代替 Struts 中的<bean:write>标签

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ page import="org.lxh.tag.*"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>

  <body>
    <%
```

```
SimpleBean simple = new SimpleBean() ;
simple.setName("魔乐");
simple.setPassword("www.mldn.cn");
// 将此对象保存在属性范围之中
request.setAttribute("sim",simple);
%>
<h1>使用 EL: </h1>
<h2>姓名: ${sim.name}</h2>
<h2>密码: ${sim.password}</h2>
<hr>
<h1>使用 Bean 标签: </h1>
<h2>姓名: <bean:write name="sim" property="name" scope="request"/></h2>
<h2>密码: <bean:write name="sim" property="password" scope="request"/></h2>
</body>
</html:html>
```

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ page import="org.lxh.tag.*"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>

  <body>
    <jsp:useBean id="sim" class="org.lxh.tag.SimpleBean" scope="request"/>
    <jsp:setProperty name="sim" property="name" value="魔乐"/>
    <jsp:setProperty name="sim" property="password" value="www.mldn.cn"/>

    <h1>使用 EL: </h1>
    <h2>姓名: ${sim.name}</h2>
    <h2>密码: ${sim.password}</h2>
    <hr>
    <h1>使用 Bean 标签: </h1>
    <h2>姓名: <bean:write name="sim" property="name" scope="request"/></h2>
    <h2>密码: <bean:write name="sim" property="password" scope="request"/></h2>
  </body>
</html:html>
```

结论：只要是标签，在程序开发中就可以互相调用，所有的标签基本上都是对四种属性范围的操作。

1.4、<bean:message>标签

Struts 国际化

Struts 中存在一个资源文件（用于保存所有的错误消息）

有些时候希望将所有的显示内容保存在文件之中，

```
# Resources for parameter 'org.lxx.struts.ApplicationResources'
```

```
# Project TagDemo
```

```
welcome
```

=

```
<h1>\u6b22\u8fce\u5149\u4e34\u9b54\u4e50\u5728\u7ebf\u5728\u7ebf\u5728\u7ebf</h1>
```

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ page import="org.lxx.tag.*"%>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>
  <body>
    <bean:message key="welcome"/>
  </body>
</html:html>
```

新的要求？

希望显示如下效果：

欢迎 Xxx 光临魔乐在线!!!

在<bean:message>标签中提供了一个占位功能，在输出的文件中占着一位，这一位的数据等待标签填写。

```
welcome = <h1>\u6b22\u8fce {0} \u5149\u4e34\u9b54\u4e50\u5728\u7ebf\u5728\u7ebf\u5728\u7ebf</h1>
```

```
<%@ page language="java" pageEncoding="GB2312"%>
<%@ page import="java.util.*"%>
<%@ page import="org.lxx.tag.*"%>
```

```
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<%@ taglib uri="http://struts.apache.org/tags-tiles" prefix="tiles" %>
<html:html lang="true">
  <head>
    <title>BeanDemo01.jsp</title>
  </head>

  <body>
    <bean:message key="welcome" arg0="李兴华"/>
  </body>
</html:html>
```

引发了一个问题？

例如以下一种情况：

有一套程序，此程序完成之后要同时供 7 个国家的用户访问，那么该怎么做？

- 针对每一个国家的语言完成一套代码
- 将所有国家的语言单独保存在一个文件中，用户使用程序会根据自己的国家自动选择出语言，

而程序代码只有一套。（推荐）

Struts 国际化？

所有的内容通过<bean:message>输出，之后所有的语言，单独形成一个*.properties 文件。

根据浏览器的不同，显示的语言也不同，因为所有显示的内容在*.properties 中已经完成的配置了。

ApplicationResources.properties: 显示中文

```
# Resources for parameter 'org.lxx.struts.ApplicationResources'
# Project TagDemo
welcome = <h1>\u6b22\u8fce {0}
\u5149\u4e34\u9b54\u4e50\u5728\u7ebf\u5f00\u5f00\u5f00</h1>
name = \u7528\u6237\u540d\u5f1a
```

ApplicationResources_en_US.properties: 显示英语

```
# Resources for parameter 'org.lxx.struts.ApplicationResources'
# Project TagDemo
welcome = <h1>\u6b22\u8fce {0}
\u5149\u4e34\u9b54\u4e50\u5728\u7ebf\u5f00\u5f00\u5f00</h1>
name = User Name:
```

```
<h1><bean:message key="name"/></h1>
```

会根据浏览器的不同，自动进行语言的显示，自动去调用相应的属性文件