

STRUTS 2 - THE PARAM TAG

http://www.tutorialspoint.com/struts_2/struts_param_tag.htm

Copyright © tutorialspoint.com

The **param** tag can be used to parameterize other tags. The include tag and bean tag are examples of such tags. Let us take same example which we have discussed while discussing **bean** tag.

Create action class

```
package com.tutorialspoint.struts2;

public class HelloWorldAction{
    private String name;

    public String execute() throws Exception {
        return "success";
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

Create views

Let us have **HelloWorld.jsp** with the following content:

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<title>Hello World</title>
</head>
<body>

<s:bean name="org.apache.struts2.util.Counter" var="counter">
    <s:param name="first" value="20"/>
    <s:param name="last" value="25" />
</s:bean>
<ul>
    <s:iterator value="#counter">
        <li><s:property /></li>
    </s:iterator>
</ul>

</body>
</html>
```

Next let us have **employees.jsp** with the following content:

```
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
<html>
<head>
<title>Employees</title>
</head>
<body>
    <p>An example of the include tag: </p>
    <s:include value="HelloWorld.jsp"/>
</body>
```

```
</html>
```

Configuration Files

Your **struts.xml** should look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">
<struts>
  <constant name="struts.devMode" value="true" />
  <package name="helloworld" extends="struts-default">

    <action name="hello"

      method="execute">
        <result name="success">/HelloWorld.jsp</result>
      </action>
    <action name="employee"

      method="execute">
        <result name="success">/employee.jsp</result>
      </action>

  </package>
</struts>
```

Your **web.xml** should look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
  >

  <display-name>Struts 2</display-name>
  <welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>
  <filter>
    <filter-name>struts2</filter-name>
    <filter-class>
      org.apache.struts2.dispatcher.FilterDispatcher
    </filter-class>
  </filter>

  <filter-mapping>
    <filter-name>struts2</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
</web-app>
```

Right click on the project name and click **Export > WAR File** to create a War file. Then deploy this WAR in the Tomcat's webapps directory. Finally, start Tomcat server and try to access URL <http://localhost:8080/HelloWorldStruts2/hello.action>. This will give you following screen:

In this example, we are instantiating a new instance of the `org.apache.struts2.util.Counter` bean. We then set the first property to 20 and the last property to 25. This means that the counter will have the values 20,21,22,23,24 and 25. We give the bean a name "counter". The struts bean tag instantiates the bean and puts it in the value stack. We can now use the iterator to go through the Counter bean and print out the value of the counter.