

# STRUTS 2 - THE INCLUDE TAG

[http://www.tutorialspoint.com/struts\\_2/struts\\_include\\_tag.htm](http://www.tutorialspoint.com/struts_2/struts_include_tag.htm)

Copyright © tutorialspoint.com

The struts **include** tag is very similar to the jsp **include** tag and it is rarely used. We have seen how to include the output of a struts action into a jsp using the `<s:action>` tags. The `<s:include>` tag is slightly different. It allows you to include the output of a jsp, servlet or any other resource *somethingotherthanastrutsaction* into a jsp. Behind the scenes it is exactly similar to the `<jsp:include>`, but it allows you to pass parameters to the included file and it is also part of Struts framework.

Following example shows how we will include the output of HelloWorld.jsp into the employee.jsp. In this case, the action method in HelloWorldAction.java will not be invoked, as we are directly including the jsp.

## 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>

<h2>Example of Generator Tag</h2>
<h3>The colours of rainbow:</h3>

<s:generator val="%{'Violet,Indigo,Blue,
    Green,Yellow,Orange,Red '}" count="7"
    separator=",">
    <s:iterator>
        <s:property /><br/>
    </s:iterator>
</s:generator>

</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/employee.action>. This will give you following screen:

Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js