

STRUTS 2 - THE ACTION TAG

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

Copyright © tutorialspoint.com

The action tag allows the programmers to execute an action from the view page. They can achieve this by specifying the action name. They can set the "executeResult" parameter to "true" to render the result directly in the view. Or, they can set this parameter to "false", but make use of the request attributes exposed by the action method.

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** to demonstrate the use of the **generator** tag:

```
<%@ 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>
    <s:action name="hello" executeResult="true">
```

```
        Output from Hello: <br />
    </s:action>
</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:

As you can see in this example, we have specified the value of executeResult to "true". Therefore the outcome of the hello.action is rendered directly in the page. The HelloWorld.jsp prints the colors of the rainbow - which is now rendered within employee.jsp

Now, let us modify the HelloWorldAction.java slightly:

```
package com.tutorialspoint.struts2;

import java.util.ArrayList;
import java.util.List;

import org.apache.struts2.ServletActionContext;

public class HelloWorldAction{
    private String name;
    public String execute()
    {
        List names = new ArrayList();
        names.add("Robert");
        names.add("Page");
        names.add("Kate");
        ServletActionContext.getRequest().setAttribute("names", names);
        return "success";
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

Finally, modify the employee.jsp as follows:

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

    <s:action name="hello" executeResult="false">
        Output from Hello: <br />
    </s:action>
    <s:iterator value="#attr.names">
        <s:property /><br />
    </s:iterator>

</body>
</html>
```

Again, 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: