



JSP

Action Tags

Table of Contents

- Jsp:forward
- Jsp:include
- Java Bean class
- Jsp:useBean
- set & getProperty

There are many JSP action tags or elements. Each JSP action tag is used to perform some specific tasks.

The action tags are used to control the flow between pages and to use Java Bean. The Jsp action tags are given below.

JSP Action Tags	Description
jsp:forward	forwards the request and response to another resource.
jsp:include	includes another resource.
jsp:useBean	creates or locates bean object.
jsp:setProperty	sets the value of property in bean object.
jsp:getProperty	prints the value of property of the bean.
jsp:plugin	embeds another components such as applet.
jsp:param	sets the parameter value. It is used in forward and include mostly.
jsp:fallback	can be used to print the message if plugin is working. It is used in jsp:plugin.

The jsp:useBean, jsp:setProperty and jsp:getProperty tags are used for bean development. So we will see these tags in bean development.

jsp:forward action tag

The jsp:forward action tag is used to forward the request to another resource it may be jsp, html or another resource.

Syntax of jsp:forward action tag without parameter

```
<jsp:forward page="relativeURL" | <%= expression %>" />
```

Syntax of jsp:forward action tag with parameter

```
<jsp:forward page="relativeURL" | <%= expression %>">  
<jsp:param name="parametername" value="parametervalue" | <%=expression%>" />  
</jsp:forward>
```

Example of jsp:forward action tag without parameter

In this example, we are simply forwarding the request to the printdate.jsp file.

index.jsp

```
<html>  
<body>  
<h2>this is index page</h2>
```

```
<jsp:forward page="printdate.jsp" />  
</body>  
</html>
```

printdate.jsp

```
<html>  
<body>  
<% out.print("Today is:" + java.util.Calendar.getInstance().getTime()); %>  
</body>
```

</html>

Example of jsp:forward action tag with parameter

In this example, we are forwarding the request to the printdate.jsp file with parameter and printdate.jsp file prints the parameter value with date and time.

index.jsp

```
<html>
<body>
<h2>this is index page</h2>

<jsp:forward page="printdate.jsp" >
<jsp:param name="name" value="javatpoint.com" />
</jsp:forward>
```

```
</body>
</html>
```

printdate.jsp

```
<html>
<body>

<% out.print("Today is:"+java.util.Calendar.getInstance().getTime()); %>
<%= request.getParameter("name") %>

</body>
</html>
```

jsp:include action tag

The jsp:include action tag is used to include the content of another resource it may be jsp, html or servlet. The jsp include action tag includes the resource at request time so it is better for dynamic pages because there might be changes in future.

The jsp:include tag can be used to include static as well as dynamic pages.

Advantage of jsp:include action tag

Code reusability : We can use a page many times such as including header and footer pages in all pages. So it saves a lot of time.

Difference between jsp include directive and include action

JSP include directive	JSP include action
includes resource at translation time.	includes resource at request time.
better for static pages.	better for dynamic pages.
includes the original content in the generated servlet.	calls the include method.

Syntax of jsp:include action tag without parameter

```
<jsp:include page="relativeURL" | <%= expression %>" />
```

Syntax of jsp:include action tag with parameter

```
<jsp:include page="relativeURL" | <%= expression %>">
<jsp:param name="parametername" value="parametervalue" | <%=expression%>" />
</jsp:include>
```

Example of jsp:include action tag without parameter

In this example, index.jsp file includes the content of the printdate.jsp file.

File: index.jsp

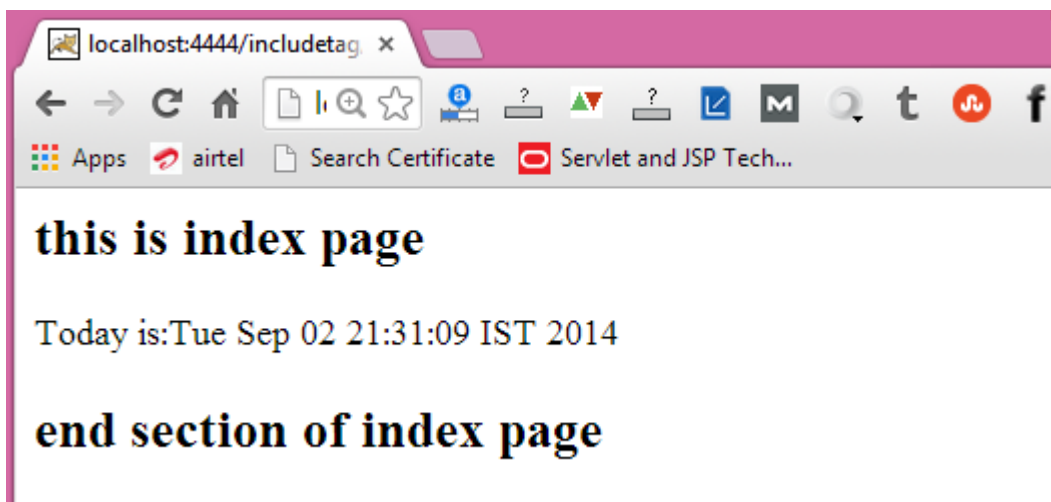
```
<h2>this is index page</h2>
```

```
<jsp:include page="printdate.jsp" />
```

```
<h2>end section of index page</h2>
```

File: printdate.jsp

```
<% out.print("Today is:"+java.util.Calendar.getInstance().getTime()); %>
```



Java Bean

A Java Bean is a java class that should follow following conventions:

It should have a no-arg constructor.

It should be Serializable.

It should provide methods to set and get the values of the properties, known as getter and setter methods.

Why use Java Bean?

According to Java white paper, it is a reusable software component. A bean encapsulates many objects into one object, so we can access this object from multiple places. Moreover, it provides the easy maintenance.

Simple example of java bean class

```
//Employee.java
```

```
package mypack;
```

```
public class Employee implements java.io.Serializable{
```

```
private int id;
```

```
private String name;
```

```
public Employee(){ }
```

```
public void setId(int id){ this.id=id;}
```

```
public int getId(){ return id;}
```

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

```
public String getName(){return name;}
```

```
}
```

How to access the java bean class?

To access the java bean class, we should use getter and setter methods.

```
package mypack;
```

```
public class Test{
```

```
public static void main(String args[]){
```

```
Employee e=new Employee();//object is created
```

```
e.setName("Arjun");//setting value to the object
```

```
System.out.println(e.getName());
```

```
}}
```

Note: There are two ways to provide values to the object, one way is by constructor and second is by setter method.

jsp:useBean action tag

The jsp:useBean action tag is used to locate or instantiate a bean class.

If bean object of the Bean class is already created, it doesn't create the bean depending on the scope. But if object of bean is not created, it instantiates the bean.

Syntax of jsp:useBean action tag

```
<jsp:useBean id= "instanceName" scope= "page | request | session | application"  
class= "packageName.className" type= "packageName.className"  
beanName="packageName.className | <%= expression >" >  
</jsp:useBean>
```

Attributes and Usage of jsp:useBean action tag

id: is used to identify the bean in the specified scope.

scope: represents the scope of the bean. It may be page, request, session or application. The default scope is page.

page: specifies that you can use this bean within the JSP page. The default scope is page.

request: specifies that you can use this bean from any JSP page that processes the same request. It has wider scope than page.

session: specifies that you can use this bean from any JSP page in the same session whether processes the same request or not. It has wider scope than request.

application: specifies that you can use this bean from any JSP page in the same application. It has wider scope than session.

class: instantiates the specified bean class (i.e. creates an object of the bean class) but it must have no-arg or no constructor and must not be abstract.

type: provides the bean a data type if the bean already exists in the scope. It is mainly used with class or beanName attribute. If you use it without class or beanName, no bean is instantiated.

beanName: instantiates the bean using the java.beans.Beans.instantiate() method.

Simple example of jsp:useBean action tag

In this example, we are simply invoking the method of the Bean class.

Calculator.java (a simple Bean class)

```
package com.kalibermind;  
public class Calculator{
```

```
    public int cube(int n)  
    {  
        return n*n*n;  
    }  
}
```

index.jsp file

```
<jsp:useBean id="obj" class="com.kalibermind.Calculator"/>
```

```
<%
```

```
int m=obj.cube(5);
```

```
out.print("cube of 5 is "+m);
```

```
%>
```



jsp:setProperty and jsp:getProperty action tags

The setProperty and getProperty action tags are used for developing web application with Java Bean. In web development, bean class is mostly used because it is a reusable software component that represents data.

The jsp:setProperty action tag sets a property value or values in a bean using the setter method.

Syntax of jsp:setProperty action tag

```
<jsp:setProperty name="instanceOfBean" property="*" |  
property="propertyName" param="parameterName" |  
property="propertyName" value="{ string | <%= expression %>}"  
>
```

Example of jsp:setProperty action tag if you have to set all the values of incoming request in the bean

```
<jsp:setProperty name="bean" property="*" />
```

Example of jsp:setProperty action tag if you have to set value of the incoming specific property

```
<jsp:setProperty name="bean" property="username" />
```

Example of jsp:setProperty action tag if you have to set a specific value in the property

```
<jsp:setProperty name="bean" property="username" value="Kumar" />
```

jsp:getProperty action tag

The jsp:getProperty action tag returns the value of the property.

Syntax of jsp:getProperty action tag

```
<jsp:getProperty name="instanceOfBean" property="propertyName" />
```

Simple example of jsp:getProperty action tag

```
<jsp:getProperty name="obj" property="name" />
```

Example of bean development in JSP

In this example there are 3 pages:

index.html for input of values

welcome.jsp file that sets the incoming values to the bean object and prints the one value

User.java bean class that have setter and getter methods

index.html

```
<form action="process.jsp" method="post">
Name:<input type="text" name="name"><br>
Password:<input type="password" name="password"><br>
Email:<input type="text" name="email"><br>
<input type="submit" value="register">
</form>
```

process.jsp

```
<jsp:useBean id="u" class="org.kalibermind.User"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>
```

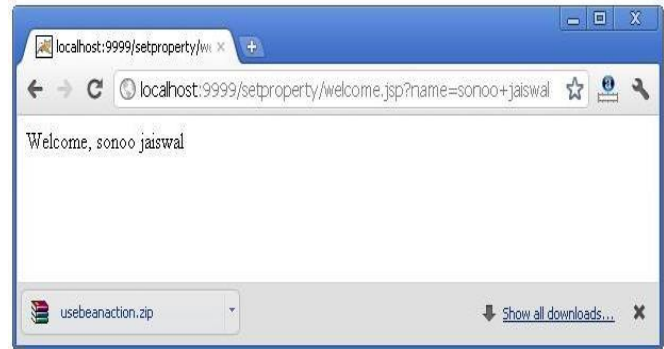
Record:


```
<jsp:getProperty property="name" name="u"/><br>
<jsp:getProperty property="password" name="u"/><br>
<jsp:getProperty property="email" name="u" /><br>
```

User.java

```
package org.kalibermind;
```

```
public class User {
private String name,password,email;
//setters and getters
}
```

Reusing Bean in Multiple Jsp Pages

Let's see the simple example, that prints the data of bean object in two jsp pages.

index.jsp

Same as above.

User.java

Same as above.

process.jsp

```
<jsp:useBean id="u" class="org.sssit.User" scope="session"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>
```

Record:


```
<jsp:getProperty property="name" name="u"/><br>
<jsp:getProperty property="password" name="u"/><br>
<jsp:getProperty property="email" name="u" /><br>
```

```
<a href="second.jsp">Visit Page</a>
```

second.jsp

```
<jsp:useBean id="u" class="org.kalibermind.User" scope="session"></jsp:useBean>
```

Record:


```
<jsp:getProperty property="name" name="u"/><br>
<jsp:getProperty property="password" name="u"/><br>
<jsp:getProperty property="email" name="u" /><br>
```

Using variable value in setProperty tag

In some case, you may get some value from the database, that is to be set in the bean object, in such case, you need to use expression tag. For example:

process.jsp

```
<jsp:useBean id="u" class="org.kalibermind.User"></jsp:useBean>
<%
String name="arjun";
%>
<jsp:setProperty property="name" name="u" value="<%=name %>"/>
```

Record:


```
<jsp:getProperty property="name" name="u"/><br>
```