

## JSP Standard Actions:

JavaServer Pages (JSP) is a technology used to create dynamic web content by embedding Java code within HTML pages. JSP Standard Actions are a set of predefined tags that simplify the integration of Java code into JSP pages. These actions allow you to perform common tasks such as controlling flow, looping, conditionals, and including other resources within your JSP pages without writing extensive Java code.

Here, I'll explain two common JSP Standard Actions: ``<jsp:include>`` and ``<jsp:useBean>`` with code examples.

### 1. ``<jsp:include>``:

`<jsp:include>` is used to include the content of another resource (JSP page or HTML) within the current JSP page. This action is useful for reusing content across multiple pages.

Example 1 - Including Another JSP Page:

Suppose you have two JSP pages, `header.jsp` and `content.jsp`, and you want to include the header in the content page.

`header.jsp`:

```
```.jsp
<html>
<head>
  <title>Header Page</title>
</head>
<body>
  <h1>Header Content</h1>
</body>
</html>
...`
```

`content.jsp`:

```
```.jsp
<html>
<head>
  <title>Content Page</title>
</head>
<body>
  <jsp:include page="header.jsp" />
  <h2>Content Page Content</h2>
</body>
</html>
...`
```

In this example, when you access `content.jsp`, it will include the content of `header.jsp` within it.

## 2. ``<jsp:useBean>``:

`<jsp:useBean>` is used to create and manage JavaBeans (Java objects) within JSP pages. JavaBeans are reusable Java objects with getter and setter methods.

### Example 2 - Using ``<jsp:useBean>`` to Work with a JavaBean:

Suppose you have a JavaBean called `Product` with attributes `name` and `price`. You can use ``<jsp:useBean>`` to work with this bean.

`Product.java` (JavaBean):

```
```java
public class Product {
    private String name;
    private double price;

    // Getter and Setter methods
    public String getName() {
        return name;
    }

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

    public double getPrice() {
        return price;
    }

    public void setPrice(double price) {
        this.price = price;
    }
}
```
```

`product.jsp` (JSP Page):

```
```jsp
<jsp:useBean id="productBean" class="com.example.Product" scope="request" />
<jsp:setProperty name="productBean" property="name" value="Laptop" />
<jsp:setProperty name="productBean" property="price" value="999.99" />

<html>
<head>
```

```
<title>Product Page</title>
</head>
<body>
  <h1>Product Information</h1>
  <p>Name: <jsp:getProperty name="productBean" property="name" /></p>
  <p>Price: <jsp:getProperty name="productBean" property="price" /></p>
</body>
</html>
...
```

In this example, we use ``<jsp:useBean>`` to create a `productBean` object of the `Product` class. Then, we set its properties (name and price) using ``<jsp:setProperty>``. Finally, we retrieve and display these properties using ``<jsp:getProperty>``.

JSP Standard Actions simplify common tasks and promote code reusability in JSP pages. You can use them to include content, work with JavaBeans, and perform other tasks without writing extensive Java code within your JSP pages.