

Java Server Page(JSP) Elements

JSP Elements

- JSP Elements are used to add dynamic content to JSP page.
- In other words, JSP Elements allows us to write java code in HTML document.
- All JSP Elements may be written in one of two forms
 - The **XML** Form
 - The **<% %>** form
- There are several types of JSP Elements, those are
 - 1) Directive Elements
 - 2) Scripting Elements
 - 3) Action Elements

Directive Elements

- Directive elements in JSP provides the special information to the JSP engine. It gives the information about the JSP page.
- Each JSP page goes through the two phases i.e. translation phase and request processing phase.
- At the translation phase, the JSP engine translates the JSP page into a Servlet, directive elements are handled at the translation time.
- Directive Elements gives special instruction to Web Container at the time of page translation.

Syntax:

```
<%@ directive attribute="value" %>
```

Directive Elements

- JSP support 3 types of Directive Elements: **page**, **include** and **taglib**.

page directive:

- The page directive defines attributes (or) properties that apply to an entire JSP page. such as the size of the allocated buffer, imported packages, defining what type of page it is etc.
- Syntax of JSP page directive

`<%@ page attribute="value" %>`

- **Page** directive supports so many attributes ,some of them are,
- import
- contentType
- buffer
- session
- isErrorPage

Directive Elements

include directive:

- JSP include directive is used to include other files into the current jsp page. These files can be html files, other sp files etc.
- The advantage of using an include directive is that it allows code re-usability.
- The syntax of an include directive :
`<%@ include file = "file location"%>`

taglib directive:

- JSP taglib directive is used to define the tag library and custom tags, which we can use in JSP.
- **Syntax of taglib directive:**
`<%@ taglib uri="uri" prefix="value"%>`

Example: “dirvdemo.jsp”

```
<html>
```

```
<body>
```

```
<!-- JSP include directive -->
```

```
<%@include file="welcome.html" %>
```

```
<!-- JSP page directive -->
```

```
<%@page import="java.util.Date" %>
```

```
<%
```

```
Date d=new Date();
```

```
out.println("Today Date is :"+d);
```

```
%>
```

```
</body>
```

```
</html>
```

Example: “welcome.html”

```
<html>
```

```
<head>
```

```
<title>welocme</title>
```

```
</head>
```

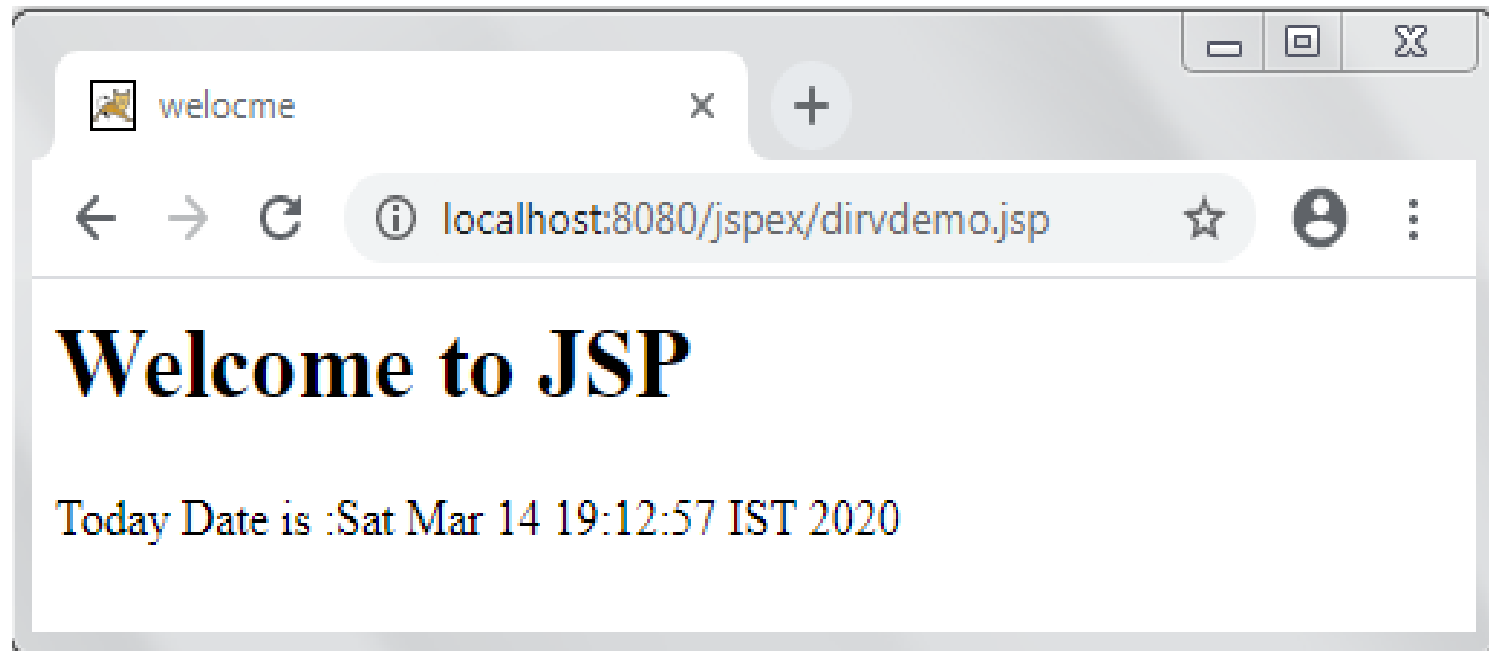
```
<body>
```

```
<h1> Welcome to JSP</h1>
```

```
</body>
```

```
</html>
```

Output:



Scripting Elements

- The scripting elements provides the ability to insert java code inside the jsp.
- JSP Scripting element are written inside `<% %>` tags. These code inside `<% %>` tags are processed by the JSP engine during translation of the JSP page.
- Any other text in the JSP page is considered as HTML code or plain text.
- There are five different types of scripting elements

Scripting Element	Example
Comment	<code><%-- comment --%></code>
Directive	<code><%@ directive %></code>
Declaration	<code><%! declarations %></code>
Expression	<code><%= expression %></code>
Scriptlet	<code><% scriptlets %></code>

Scripting Elements

Comment:

- JSP comments are used to provide descriptive information or hide code from JSP container.
- The syntax of JSP Comment :

<%-- comment --%>

Example:

<%-- this is comment line --%>

Declaration:

- JSP declarations are used to define variables or methods.
- The syntax of JSP Declaration :

<%! Declaration of variable or method %>

Example:

<%!

int a=10;

%>

Scripting Elements

Expression:

- The code placed within **expression tag** is written to the output stream of the response. So you need not write `out.println()` to write data. It is mainly used to print the values of variable or method.
- The syntax of JSP Expression:

<%= Expression %>

Example:

<%= (2*5) %>

Scriptlet:

- Scriptlet Tag allows you to write java code inside JSP page.
- The syntax of JSP Scriptlet :

<% Java Code %>

Example:

<%

`out.println("Welcome to JSP");`

%>

Example: “scriptdemo.jsp”

```
<%@page import="java.util.Date" %>
```

```
<html>
```

```
<body>
```

```
<!-- Comments-->
```

```
<%-- JSP Comments --%>
```

```
<!-- Declaration-->
```

```
<%!
```

```
int a=10;
```

```
int b=20;
```

```
%>
```

```
<!-- Expression-->
```

```
<%= "Sum is "+(a+b) %>
```

```
<br/><br/>
```

```
<!-- Scriptlet-->
```

```
<%
```

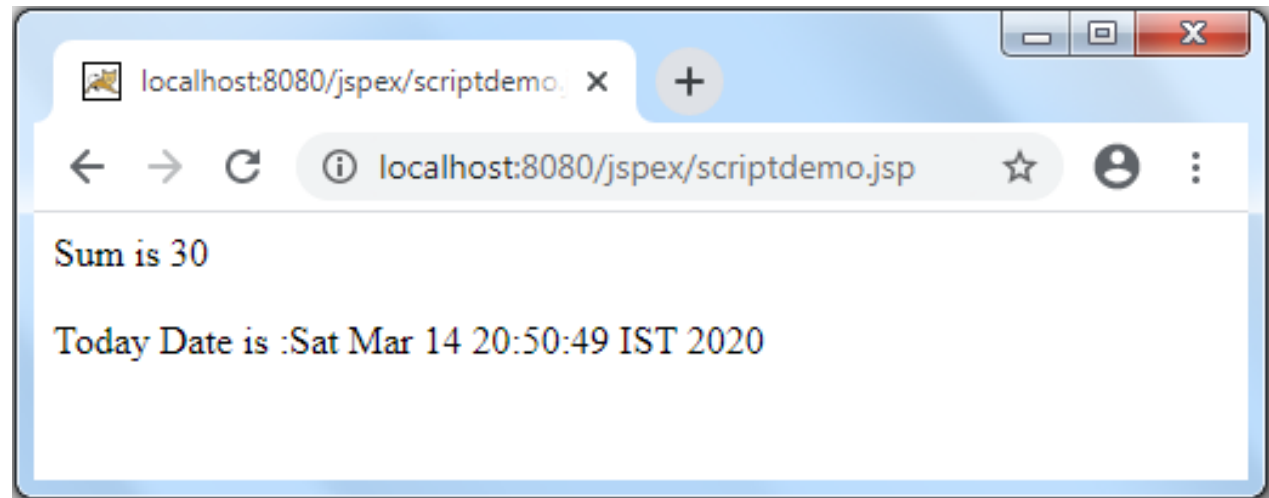
```
Date d=new Date();
```

```
out.println("Today Date is :"+d);
```

```
%>
```

```
</body>
```

```
</html>
```



Action Elements

- JSP provides a bunch of standard action elements or tags that we can use for specific tasks such as working with java bean objects, including other resources, forward the request to another resource etc.

Directives vs Actions

- Directives are used during translation phase while actions are used during request processing phase.
- Unlike Directives Actions are re-evaluated each time the page is accessed.

Syntax for the Action element

```
<jsp:action_name attribute = "value" />
```

Action Elements

- The list of standard JSP action elements are given below.

Action Element	Description
jsp:include	To include a resource at runtime, can be HTML, JSP or any other file
jsp:useBean	To create a new object of java bean.
jsp:forward	To forward the request to another resource.
jsp:text	To write template text in JSP page.
jsp:element	To define the XML elements dynamically.
jsp:attribute	To define the attributes for XML elements.
jsp:body	To define the body for XML element.

Action Elements

<jsp:include> Element :

- The **jsp:include action tag** is used to include the content of another resource it may be jsp or html.
- The jsp:include tag can be used to include static as well as dynamic pages.
- The jsp include action tag includes the resource at request time so it is **better for dynamic pages**.

Syntax:

```
<jsp:include page="relativeURL" />
```

Action Elements

Example: “include.jsp”

```
<html>
```

```
<body>
```

```
<h2>This is Start of Home page</h2>
```

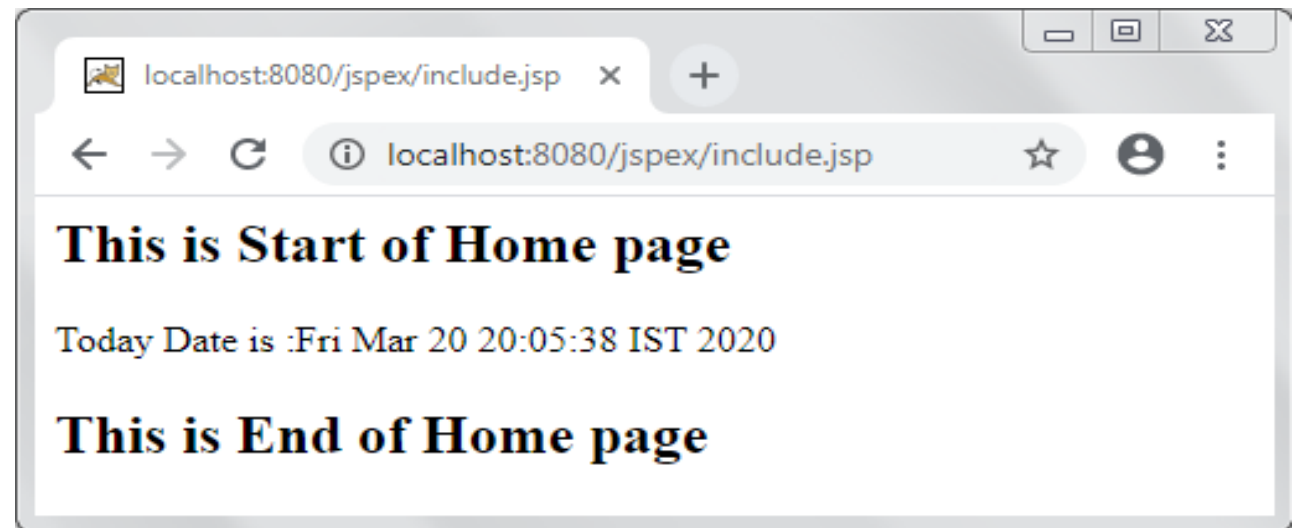
```
<jsp:include page="printdate.jsp" />
```

```
<h2>This is End of Home page</h2>
```

```
</body>
```

```
</html>
```

Output:



Action Elements

<jsp:forward> Element :

- The jsp:forward action tag is used to forward the request to another resource it may be jsp, html or another resource.
- The **forward** action terminates the action of the current page and forwards the request to another resource such as a static page, another JSP page, or a Java Servlet.
- <jsp:forward> is used for redirecting the request.

Syntax:

```
<jsp:forward page="URL of the another page" />
```

Action Elements

Example: “forward.jsp”

```
<html>
```

```
<body>
```

```
<h2>This is Start of Home page</h2>
```

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

```
<h2>This is End of Home page</h2>
```

```
</body>
```

```
</html>
```

Output:

