1. **Expression Tag** [ <%= %> ]

* This tag helps us to print Dynamic Value in the Browser
* Code placed with in this tag should not end with Semicolon.
* Functionality of this tag is similler to printing the dynamic data using out.print() in case of Servlets.

1. **Scriptlet Tag [ <% %> ]**

* In Servlets we write business logic (Java Code) inside service() method
* We also know that, JSP get tanslated to Servlet at runtime & home “we can also write business logic (Java Code) inside JSP using Scriptlet Tag”
* Code placed with in this tag becomes part of \_jspService() Method at runtime and get extended for every request.
* If we declare a variable using this tag then it becomes “Method Level Variable” and accessable by only within \_jspService() Method.

Example for Declaration, Expression & Scriptlet Tags:

<%!

Private int age = 100;

Public String name = “RajniKant”;

Public int getAge(){

Return age;

}

Public String getName(){

Return name;

}

Public String getName(String givenNM){

Return “Given name is – “+givenNM;

}

%>

<html>

<body>

<h1>JSP Tags Example</h1>

Name 1: <%= name %>

Name 2: <%= getName() %>

Name 3: <%= get Name() %>

<%

Int j = 100;

For(int I = 0;i<5;i++){

%>

Name: <%= name %>

<%

}

%>

</body>

</html>

1. **Action Tag [ <jsp:actiion\_name action\_attributes />**

Action Tag helps us to perform following actions in JSP

1. Forward Action
2. Include Action
3. UseBean Action
4. Forward Action: [ <jsp: forward page=”relativeUrl” /> ]

This tag helps us to Forward the Request to an another Internal Resouece (static/ Dynamic Servle or JSP)

Example:

<%-- Forward: Static Resource --%>

<jsp: forward page=”index.html”/>

<%-- Forward : Dynamic Resource – Servlet --%>

**<jsp:forward page=”currentDate” />**

**<jsp:forward page = “currentDate?fnam=ABC&lname=XYZ”/>**

**<jsp: forward page= “currentDate”>**

**<jsp:param name=”fname” value=”123”/>**

**<jsp:param name=”lname” value=”456”/>**

**</jsp:forward>**

Note:

* We can Forward the request to only one resource at a time
* Only in case of Action Tag we can make use of other JSP Tags.

For Example:

<%

String url = “currentDate”;

%>

<jsp:forward page =”<%=url%> />

1. **Include Action [< jsp:inlcude page=”relativeURL” /> ]**

Theis tag helps us to Include the Response of an another Internal Resource (static /dynamid i.e Servlet or JSP)

Example:

1111111111111

**<jsp: include page=”index.html” />**

2222222222222

**<jsp:inlude page=”currentDate?fname=AAA&lname=BBB”/>**

33333333333333

**<jsp:inlcude page=”currentDate”>**

**<jsp:param name = “fname” value=”xxx” />**

**<jsp:param name=”lname” value =”yyy”/>**

**</jsp:include>**

444444444444444

**<jsp:inlcude page=”currentDateJsp”>**

**<jsp:param name=”fname” value=”123” />**

**<jsp:param name=”lname” value=”456” />**

**</jsp: include>**

55555555555555555555

1. **UseBean** Action

**Syntax:**

<jsp:useBean

Id=”referenceNM”

class =”pkgNM.classNM”

scope=”page|request|session|application”/>

* This tag helps us to create Explicit Objects in JSP
* Container first search for an Existing object by using “id” & “scope” attribute present in this tag
* If found, then it makes the use of that object
* Otherwise,

\*it creates a “New Object” using “Public Default Constructor” &

\* set that object to the corresponding scope

* Since web application is Multi Threaded in Nature, the above logic is “synchronized” in Thranslated Servlet.
* We know that , Java Bean helps us to transfer the data in the form of Object from one Java program to other.
* Hence using this tag “we can Only create an instance of JavaBean “ & for normal Java Classes we should use “Scriptlet Tag”

Example:

<jsp:useBean

Id=”reg”

Class =”com.jspiders.studentsapp.dao.StudentInfoBean”

Scope=”application”

/>

Reg. no. Value is:

<jsp:getProperty name=”ref” property=”regno” />

Changing the Value of Reg. no.

<jsp:setProperty name=”ref” property=”regno” vallue=”10” />

New reg no value is:

<jsp:getProperty name= “ref” property=”regno”/>

5- **Directive Tag [ <%@directive\_name directive\_attribute %> ]**

**There are 3 Types of Directive Tags**

1. **Include directive**
2. **TagLib Directive**
3. **Page Directive**
4. Include Directive- [ <%@include file=”resourceFileNM”%>

* Include Directive includes the content of Physical File OR Static Resource into JSP at the time of translation
* Hence compared to Include Action (for static Resource Include Directive is Faster in nature

**Note:** Use include Action for Dynamic Resources & Use Include Directive for static Resources

Example:

11111111111111111

[%@include file = “index.html” %](mailto:%25@include%20file%20=%20)

2222222222222222222

<%--

Not Possible !!! We Cannot include Dynamic Resource using Include Directive

--%>

<%-- <%@include file=”currentDate” %> --%>

<% -- <%@include file=”currentDateJsp” %> --%>

ii. **Taglib Directive**

**Syntax:**

<%@ taglib uri=”URI of the Tag Library” prefix=”prefix of the Tag Library” %>

* Taglib Directive helps us to make use of “Custom Tags” in JSP
* The functionality of these custom tags are defined in “Tag Library Descriptor (TLD)” file
* Tag Library Description file will have “.tld” file extension & it will present inside WEB-INF folder.

**Page Directive [ <%@page attributes%> ]**

* Page Directive has 11 Optional Attribute that provides some processing information to Servlet Container.

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Syntax** | **Description** |
| Info | <%@page info=”My First JSP” %> | * This attribute is used to provide documentation information for a JSP * Details Such as Author, Version, Copyright, Date, etc., are placed inside this tag. |
| Language | <%@page language=”java” %> | * Language denotes the “Scripting Language” used in Declaration, Expression & Scriptlet Tags. * Default language is “Java” |
| **contentType** | <%A page ContentType=”text/html” %>  Or  <%@ page contentType=”text/html; charset=ISO-8859-1”%> | * This attribute specifies the Content Type and Character Set for JSP Response. * Default Content Type is “text/html” &Default Character Set is”ISO-8859-1” |
| Import | <%@page import=”pkg1.class1, pkg2.\*,.. “%>  Or  <[%@page import=”pkg1.class1”%](mailto:%25@page%20import=)>  <%@page import=”pkg2.\*”%> | * Functionality is similar to Java “import” statement. |
| Extends | <%@page extends=”pkgNM.classNM”%> | * Functionality is similler to Java “extends” statement |
| Buffer | <%@page buffer=”none | sizeInKB” %> | * This attribute set the Buffer Size in KB * The default size is 8KB * To turn off the buffered output the value should be “none”. |
| autoFlush | <%Apage autoFlush=”true | false”%> | * This attribute controls the behaviour of Buffer * If “true” (it’s default), the buffer will be fluses automativalyy when it’s full * If “faluse” then an exception is thrown when the buffer is full. |
| Session | <%@page session=”true” | false %> | * If “true” (it’s default), then “session” implicit object is available in JSP % it refers to Current/ New Session * If “false”, then “session” implicit object will NOT be available in JSP |
| isThreadSafe | <%@page isThreadSafe =”true” | false”%> | * If “true” (it’s default), then JSP becomes “Multi Threaded” in nature * If “jalse”, then JSP becomes “Single Threaded” in nature. |
| isErroPage | <%@page isErrorPage=”true” |”false” %> | * If “true”, then “exception” implicit object is available in JSP * If “true” , then “exception” implicit object is availble in JSP * If False (it’s default), then “excepiton” implicit object will Not be availble in JSP |
| errorPage | <%@page errorPage=”relativeURL”%> | * This attribute defines the “Error Page” for a JSP * While processing the request for a JSP , if an exception occurs then requestautomatically get forwarded to the “Specified Error Page” * The value of “errorPage” attribute is URL of the Static/ Dynamic Resource. |

* Info
* Language
* Content type
* Buffer
* Autoflush
* isThread
* isError
* Error Page
* Import
* Info
* session

Code words: LI4C BASE2

NOTE:

1. We can have Zero / More Page Directive in JSP & they can be present anywhere inside a JSP (Preferred is to keep at the beginning of the JSP)
2. We can club multiple Page Directive Attributes into one page directive as shown below

<[%@page errorPage=”JSPErrorPage.jsp” import=”java.util.Date”%](mailto:%25@page%20errorPage=)>

1. We can have Zero / More Page Directive in JSP & they can be preseint anywhere inside a JSP (Preferred is to kepp at the beginning of the JSP)
2. We can club multiple Page Directive Attributes into one page directive as shown below.

<%@ page errorPage=”JSPErrorPage.jsp” import=”java.util.Date” %>

Which is equal to

<%@page errorPage=”JSPErrorPage.jsp”%>

<%@page import=”java.util.Date” %>

1. Declaration Tag & Page Dirctive Tag Can be present at the beging of the JSP (preffered ). So Between these two, page directive tag should be present first folloeed by Declaration Tag.

Exception Handling in JSP

In JSP we can handle exception scnerios with the help of “errorPage” & “isErrorPage” page directive attributes.

Example:

MyJSP.jsp

<%@page errorPage=”JSPErrorPage.jsp”%>

<html>

<body>

Exception Simulation…

<%

Int I = 100/0;

%>

</body>

</html>

JSPErrorPage.jsp

<%@page isErrorPage=”true”%>

<html>

<body>

Exception Occurred!!! Exception is:

<font color=”red”>

<%

Incomplete….

Single Thread JSP:

We can create JSP, single threaded in nature with the help of “isThreadSafe” page Directive attribute.

Example:

<%@page isThreadSafe=”false”%>

<html>

<body>

<h1>My Single Threaded JSP</h1>

</body>

</html>

<%-- Don’t Generate the Response, Instead Redirect the Request =%>

<%

String url = “http://www.gmail.com”;

Response.sendRedirect(url);

%>