

# **Java Server Pages (JSP)-II**

# JSP - Directives

- Directives provide directions and instructions to the container, telling it how to handle certain aspects of the JSP processing.
- The JSP directives are messages that tells the web container how to translate a JSP page into the corresponding servlet.
- It usually has the following form –

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

- Directives can have a number of attributes which you can list down as key-value pairs and separated by commas.
- The blanks between the @ symbol and the directive name, and between the last attribute and the closing %>, are optional.

S.No.	Three types of Directive
1	<b>&lt;%@ page ... %&gt;</b> Defines page-dependent attributes, such as scripting language, error page, and buffering requirements.
2	<b>&lt;%@ include ... %&gt;</b> Includes a file during the translation phase.
3	<b>&lt;%@ taglib ... %&gt;</b> Declares a tag library, containing custom actions, used in the page

# Page Directive

- The page directive is used to provide instructions to the container. These instructions pertain to the current JSP page.
- You may code page directives anywhere in your JSP page. By convention, page directives are coded at the top of the JSP page. Syntax is as follows:

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

# Page Directive cont..

## Attributes of Page Directives:

S.No.	Attribute & Purpose
1	<b>Buffer:</b> Specifies a buffering model for the output stream.
2	<b>autoFlush:</b> Controls the behavior of the servlet output buffer.
3	<b>contentType:</b> Defines the character encoding scheme.
4	<b>errorPage:</b> Defines the URL of another JSP that reports on Java unchecked runtime exceptions.
5	<b>isErrorPage:</b> Indicates if this JSP page is a URL specified by another JSP page's errorPage attribute.
6	<b>Extends:</b> Specifies a superclass that the generated servlet must extend.
7	<b>Import:</b> Specifies a list of packages or classes for use in the JSP as the Java import statement does for Java classes.
8	<b>Info:</b> Defines a string that can be accessed with the servlet's <b>getServletInfo()</b> method.
9	<b>isThreadSafe:</b> Defines the threading model for the generated servlet.
10	<b>Language:</b> Defines the programming language used in the JSP page.
11	<b>Session:</b> Specifies whether or not the JSP page participates in HTTP sessions
12	<b>isELIgnored:</b> Specifies whether or not the EL expression within the JSP page will be ignored.
13	<b>isScriptingEnabled:</b> Determines if the scripting elements are allowed for use.

# Page Directive cont..

## The buffer Attribute

- The buffer attribute specifies the buffering characteristics for the server output response object.
- You may code a value of "none" to specify no buffering so that the servlet output is immediately directed to the response object or you may code a maximum buffer size in kilobytes, which directs the servlet to write to the buffer before writing to the response object.
- To direct the servlet to write the output directly to the response output object, use the following –

```
<%@ page buffer = "none" %>
```

- Use the following to direct the servlet to write the output to a buffer of size not less than 8 kilobytes –

```
<%@ page buffer = "8kb" %>
```

# Page Directive cont..

## The autoFlush Attribute

- The autoFlush attribute specifies whether buffered output should be flushed automatically when the buffer is filled, or whether an exception should be raised to indicate the buffer overflow.
- A value of true (default) indicates automatic buffer flushing and a value of false throws an exception
- The following directive causes the servlet to throw an exception when the servlet's output buffer is full –

```
<%@ page autoFlush = "false" %>
```

- This directive causes the servlet to flush the output buffer when full –

```
<%@ page autoFlush = "true" %>
```

- Usually, the buffer and the autoFlush attributes are coded on a single page directive as follows –

```
<%@ page buffer = "16kb" autoflush = "true" %>
```

# Page Directive cont..

## The contentType Attribute

- The contentType attribute sets the character encoding for the JSP page and for the generated response page. The default content type is text/html, which is the standard content type for HTML pages.
- The following statement directs the browser to render the generated page as HTML –

```
<%@ page contentType = "text/html" %>
```

- The following directive sets the content type as a Microsoft Word document –

```
<%@ page contentType = "application/msword" %>
```

# Page Directive cont..

## The **errorPage** Attribute

- The **errorPage** attribute tells the JSP engine which page to display if there is an error while the current page runs. The value of the **errorPage** attribute is a relative URL.
- The following directive displays **MyErrorPage.jsp** when all uncaught exceptions are thrown –

```
<%@ page errorPage = "MyErrorPage.jsp" %>
```

## The **isErrorPage** Attribute

- The **isErrorPage** attribute indicates that the current JSP can be used as the error page for another JSP.
- The value of **isErrorPage** is either true or false. The default value of the **isErrorPage** attribute is false.

For example, the **handleError.jsp** sets the **isErrorPage** option to true because it is supposed to handle errors –

```
<%@ page isErrorPage = "true" %>
```



# Page Directive cont..

## The extends Attribute

- The extends attribute specifies a superclass that the generated servlet must extend.
- For example, the following directive directs the JSP translator to generate the servlet such that the servlet extends `somePackage.SomeClass` –

```
<%@ page extends = "somePackage.SomeClass" %>
```

## The import Attribute

- The import attribute serves the same function as, and behaves like, the Java import statement. The value for the import option is the name of the package you want to import.
- To import `java.sql.*`, use the following page directive –

```
<%@ page import = "java.sql.*" %>
```

- To import multiple packages, you can specify them separated by comma as follows –

```
<%@ page import = "java.sql.*, java.util.*" %>
```

- By default, a container automatically imports `java.lang.*`, `javax.servlet.*`, `javax.servlet.jsp.*`, and `javax.servlet.http.*`.

# Page Directive cont..

## The info Attribute

- The info attribute lets you provide a description of the JSP. The following is a coding example –

```
<%@ page info = "This JSP Page Written By XYZ" %>
```

## The isThreadSafe Attribute

- The isThreadSafe option marks a page as being thread-safe.
- By default, all JSPs are considered thread-safe. If you set the isThreadSafe option to false, the web container will serialize the multiple requests and the JSP engine makes sure that only one thread at a time is executing your JSP.
- The following page directive sets the isThreadSafe option to false –

```
<%@ page isThreadSafe = "false" %>
```

# Page Directive cont..

## The language Attribute

- The language attribute indicates the programming language used in scripting the JSP page.
- For example, because you usually use Java as the scripting language, your language option looks like this –

```
<%@ page language = "java" %>
```

## The session Attribute

- The session attribute indicates whether or not the JSP page uses HTTP sessions.
- A value of true means that the JSP page has access to a builtin session object and a value of false means that the JSP page cannot access the builtin session object.
- Following directive allows the JSP page to use any of the builtin object session methods such as session.getCreationTime() or session.getLastAccessTime() –

```
<%@ page session = "true" %>
```

# Page Directive cont..

## The isELIgnored Attribute

- The isELIgnored attribute gives you the ability to disable the evaluation of Expression Language (EL) expressions which has been introduced in JSP 2.0.
- The default value of the attribute is true, meaning that expressions, `${...}`, are evaluated as dictated by the JSP specification. If the attribute is set to false, then expressions are not evaluated but rather treated as static text.
- Following directive sets an expression not to be evaluated –

```
<%@ page isELIgnored = "false" %>
```

## The isScriptingEnabled Attribute

- The isScriptingEnabled attribute determines if the scripting elements are allowed for use.
- The default value (true) enables scriptlets, expressions, and declarations. If the attribute's value is set to false, a translation-time error will be raised if the JSP uses any scriptlets, expressions (non-EL), or declarations.
- The attribute's value can be set to false if you want to restrict the usage of scriptlets, expressions (non-EL), or declarations –

```
<%@ page isScriptingEnabled = "false" %>
```

# Include Directive

- The include directive is used to include a file during the translation phase.
- This directive tells the container to merge the content of other external files with the current JSP during the translation phase.
- You may code the include directives anywhere in your JSP page. The general usage form of this directive is as follows –

```
<%@ include file = "relative url" >
```

- The filename in the include directive is actually a relative URL. If you just specify a filename with no associated path, the JSP compiler assumes that the file is in the same directory as your JSP.
- The include directive includes the original content of the included resource at page translation time (the JSP page is translated only once so it will be better to include static resource).

# Include Directive cont..

**header.jsp -**

```
<%!  
int pageCount = 0;  
    void addCount() {  
        pageCount++;    }  
%>  
<% addCount(); %>  
<html> <head>  
<title> The include Directive Example </title> </head>  
    <body>    <center>  
        <h2>This is Include Directive Demo</h2>  
        <p>This has been executed<%= pageCount %> times.  
    </p>  
    </center>    <br/><br/>  
</body>  
</html>
```

**footer.jsp -**

```
<html> <head>  
<title> This is Footer page </title>  
</head>  
    <center>  
        <p>UPES</p>  
    </center>  
</body>  
</html>
```

**main.jsp -**

```
<html> <head>  
<meta charset="ISO-8859-1">  
<title>This is Main Page</title>  
</head>  
<body>  
<%@ include file = "header.jsp" %>  
    <center>  
        <p>This is OOP.</p>  
    </center>  
<%@ include file = "footer.jsp" %>  
</body> </html>
```

# References

- <https://docs.oracle.com/javaee/7/api/javax/servlet/jsp/package-summary.html>
- <https://www.tutorialspoint.com/jsp/index.htm>