6. Diretivas em JSP

As diretivas JSP são instruções especiais ao container JSP que **não geram saída HTML** diretamente, mas afetam a **forma como a página é processada e compilada**. Elas são tratadas **no momento da tradução** (pré-compilação da JSP em Servlet).

★ Sintaxe básica:

< @ diretiva atributo="valor" %>

6.1 Diretiva page

A diretiva page define configurações relacionadas à própria página JSP, como:

- Imports de pacotes Java
- Página de erro associada
- Charset da resposta
- Tipo MIME (ex: text/html, application/xml)
- Escopo da linguagem

Exemplos:

Importando pacotes:

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

Permite usar classes como Connection, PreparedStatement, etc., sem precisar importar manualmente em cada scriptlet.

Definindo página de erro:

<%@ page errorPage="erroGeral.jsp" %>

Encaminha erros não tratados desta página para erroGeral.jsp.

✓ Página preparada para capturar o erro:

< @ page is Error Page = "true" %>

Necessária na página que **recebe** o erro, pois libera acesso à variável exception.

Definindo o tipo de conteúdo e charset:

<@ page contentType="text/html; charset=UTF-8" %>

✓ Definindo linguagem:

<@ page language="java" %> <!-- padrão -->

6.2 Diretiva include

Inclui **conteúdo estático** (HTML, JSP, texto) **durante a tradução da página**. Funciona como uma cópia e cola no momento da compilação.

Exemplo:

< @ include file="cabecalho.jsp" %>

Todo o conteúdo de cabecalho. j sp será incluído **antes da tradução**, sendo considerado parte da página final compilada.

🛕 Obs:

- Diferente de <jsp:include />, que é avaliado em tempo de execução.
- Use include para headers, menus, rodapés fixos.

6.3 Diretiva taglib

Permite o uso de **bibliotecas de tags personalizadas** (taglibs), como JSTL ou tags customizadas definidas por você.

Exemplo com JSTL:

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

Com isso, você pode usar tags JSTL como <c:if>, <c:forEach>, etc.



Diretiva	Finalidade
page	Configura a própria página (import, charset, página de erro, etc.)
includ e	Inclui conteúdo estático de outro arquivo na tradução da JSP
taglib	Habilita o uso de bibliotecas de tags personalizadas (ex: JSTL)

Estrutura de arquivos

cabecalho. jsp (será incluído no topo de todas as páginas)

```
<!-- cabecalho.jsp -->
<h2>Minha Aplicação JSP</h2>
<hr>
```

```
rodape.jsp (será incluído no final de todas as páginas)
<!-- rodape.jsp -->
<hr>
<hr>
<small>&copy; 2025 - Todos os direitos reservados.</small>

index.jsp (página principal)
<%@ page contentType="text/html; charset=UTF-8" %>
```

```
< @ page language="java" %>
< @ include file="cabecalho.jsp" %>

    4%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<%
  // Definindo atributo de requisição (alternativa: fazer isso em um Servlet)
  request.setAttribute("usuarioLogado", "Patrick");
%>
<html>
<head>
  <title>Exemplo de Diretivas JSP</title>
</head>
<body>
  >Bem-vindo à aplicação JSP.
```

```
<!-- Exibindo condicional com JSTL -->

<c:if test="${not empty usuarioLogado}">

Usuário: <strong>${usuarioLogado}</strong>
</c:if>

</body>
</html>

</@ include file="rodape.jsp" %>
```

O que esse exemplo mostra:

Recurso usado	Onde aparece	Função
page	<%@ page contentType="" %>	Define o tipo de conteúdo e o charset da página.
include	<%@ include file="" %>	Inclui arquivos HTML fixos (header e footer).
taglib	<pre><%@ taglib prefix="c" %></pre>	Ativa a biblioteca JSTL para usar <c:if> e outras tags.</c:if>
JSTL + EL	\${usuarioLogado}	Exibe variáveis definidas no request (Expressões JSP).

Scriptlet Java <%
request.setAttribute(
...) %>

Define valor diretamente no JSP (prática comum em estudos, mas não recomendada em produção).

🧠 Observação

Em projetos reais, é altamente recomendável **evitar lógica Java dentro de páginas JSP**. Prefira Servlets, Controllers ou Beans para manipular lógica, deixando a JSP apenas com a camada de visualização.