

### Conceitos básicos

Primeira coisa. Um JSP, no final, vira um Servlet. O JSP é apenas uma maneira para se escrever código Java dentro de um HTML, e não HTML dentro do código Java.

Existem 4 tipos diferentes de expressões em um JSP:

- **Scriptlet**      `<%            %>`
- **Diretiva**      `<%@        %>`
- **Expressão**    `<%=        %>`
- **Declarações** `<%!        %>`

Em um Scriptlet você escreve código Java, as diretivas você usa para fazer os imports. As expressões são parecidas com os scriptlets, mas não precisam do `out.println()`, e as declarações servem para criar métodos e variáveis de instância. Veja os exemplos abaixo:

### Scriptlet e Diretiva

```
<%@page import="utilidades.*" %>
<html>
<body>
O contador de páginas é:
<% out.println("Contador.getContagem()); %>
</body>
</html>
```

### Expressão e Diretiva

```
<%@page import="utilidades.*" %>
<html>
<body>
O contador de páginas é:
<%= Contador.getContagem() %>
</body>
</html>
```

Note que não se usa ponto e vírgula em expressões. Você também não pode usar uma classe feita por você em um JSP se ela não estiver em uma package e esta package não houver sido informada ao JSP por meio de uma diretiva import.

## Declaração

```
<html>
<body>
<%! int dobraContador() {
    contador = contador*2;
    return contador;
} %>
<% int contador=1; %>
O contador de páginas é:
<%= dobraContador() %>
</body>
</html>
```

O método acima define o método *int dobraContador()* e a variável de instância *int Contador*.

Quando o JSP é transformado em servlet, as expressões e os scriptlets são colocados dentro de um método do servlet gerado. Já as declarações são colocadas fora deste método principal, pois são outros métodos. E as diretivas, é claro, são colocadas fora da classe, local onde os imports são colocados.

Os JSP tem alguns objetos implícitos, isto é, objetos que não é você que instancia mas que estão lá para você usar. São eles:

API	Objeto Implícito
JspWriter	out
HttpServletRequest	request
HttpServletResponse	response
HttpSession	session
ServletContext	application
ServletConfig	config
JspException	exception
PageContext	pageContext
Object	page

**Exemplo:** Deitel&Deitel, Java Como Programar, 6a ed, pg 961

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 27.1: clock.jsp -->

<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv = "refresh" content = "60" />
    <title>A Simple JSP Example</title>
    <style type = "text/css">
      .big { font-family: helvetica, arial, sans-serif;
            font-weight: bold;
            font-size: 2em; }
    </style>
  </head>
  <body>
    <p class = "big">Simple JSP Example</p>
    <table style = "border: 6px outset;">
      <tr>
        <td style = "background-color: black;">
          <p class = "big" style = "color: cyan;">

              <!-- JSP expression to insert date/time -->
              <%= new java.util.Date() %>

            </p>
          </td>
        </tr>
      </table>
    </body>
  </html>
```

### Exercício Prático

Crie um novo projeto Web dinâmico - API 3.0 - copiando o projeto da Aula 02.  
Transforme o Servlet em JSP.

### Bibliografia:

DEITEL, Harvey M.; DEITEL, Paul J; FURMANKIEWICZ, Edson. **Java: como programar**. 6. ed. atual. São Paulo: Pearson, 2005. xl, 1110 p. ISBN 8576050196 (Broch.)

*nota: as outras edições não abordam JSP*

BASHAM, Bryan; SIERRA, Kathy; BATES, Bert. **Use a cabeça!: Servlets & JSP**. 2. ed. Rio de Janeiro: Alta Books, 2008-2010. xxxii, 879 p. ISBN 9788576082941 (broch.)