

[< VOLTAR](#)

Primeiros passos em JSP

Capacitar o aluno a dar os primeiros passos em JSP, trabalhando com delimitadores (scriptlets, declarações, expressões e comentários), tipos de dados, variáveis, constantes e expressões. Criar aplicações web utilizando os softwares NetBeans, GlassFish e Java EE.

NESTE TÓPICO

- › Introdução
- › Scriptlet
- › Comentários
- › Tipos de dados
- › Variáveis
- › Constantes



Introdução

JavaServer Pages ou JSP é mais uma tecnologia da plataforma Java EE para o desenvolvimento de aplicações dinâmicas de forma simplificada para WEB. (GONÇALVES, 2007, p.429; TERUEL, 2009, P.151; SANTOS, 2007, p.76).

Como as páginas JSP são convertidas em servlets, não há os métodos doGet() e doPost(), porque os comandos são resolvidos na própria página JSP. No entanto, há os métodos jspInit() e jspDestroy() que podem ser programados na inicialização ou finalização de uma página JSP. (GONÇALVES, 2007, p.118-119).

O JSP trabalha com arquivos com a extensão jsp. no entanto, seu conteúdo é formado de TAGs HTML, com trechos de código escritos em Java, identificados por delimitadores que passamos a identificar.

Scriptlet

Para que o scriptlet seja entendido pelo servidor web, o código Java deve estar entre as TAGs `<% e %>`.

Para melhor entendermos o que é um scriptlet, vejamos o código de um exemplo básico de um arquivo JSP:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>JSP Page</title>
5.     </head>
6.     <body>
7.         <h3>Teste JSP - scriptlet</h3>
8.         <%
9.             String nome = "Antônio Andrade";
10.            byte idade = 33;
11.            out.println("O " + nome + " tem " + idade + " anos.");
12.        %>
13.     </body>
14. </html>
```

No exemplo, foi criada uma variável do tipo string para guardar um nome e outra do tipo byte para armazenar a idade. Em seguida, com `out.println()`, foi solicitado uma saída de dados. Vejamos o resultado:

Teste JSP - scriptlet

O Antônio Andrade tem 33 anos.

Interessante vermos também o código HTML que o cliente receberá:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>JSP Page</title>
5.     </head>
6.     <body>
7.         <h3>Teste JSP - scriptlet</h3>
8.         O Antônio Andrade tem 33 anos.
9.     </body>
10. </html>
```



Além dos scriptlets, existem outros elementos (ou TAGs) que podem ser utilizados. A seguir, vejamos as mais comuns.

Comentários

Os comentários devem ficar entre as TAGs: `<%--` e `--%>`.

Exemplo:

```
1. <%-- Isto é um comentário em JSP. Não há retorno ao cliente via HTML --%>
```

Caso seja necessário fazer comentários dentro de um scriptlet, lembre-se de que você está programando em Java.

A qualquer ponto do trecho de código poderá ser utilizado `//` para comentários de uma linha e `/* ... */` para comentários de múltiplas linhas.

Vejamos o código a seguir que nos mostra um exemplo com os três tipos de comentários:

```

1.  <!DOCTYPE html>
2.  <html>
3.      <head>
4.          <title>JSP Page</title>
5.      </head>
6.      <body>
7.          <!-- Programa que retorna informações do lado servidor -->
8.          <h3>Teste JSP - scriptlet e comentários</h3>
9.          <%
10.              // Declaração das variáveis que receberão os dados:
11.              String diretorio, protocolo;
12.              /* O diretório do site e o nome do protocolo de navegação,
13.               * serão atribuídos inicialmente às variáveis que são
14.               * do tipo String. Depois estes dados serão retornado para
15.               * o cliente HTML através do método println da objeto out.
16.               */
17.              diretorio = request.getContextPath();
18.              protocolo = request.getProtocol();
19.              out.print("<pre>");
20.              out.println("Path do contexto (diretório do site): " + diretorio);
21.              out.println("Protocolo utilizado para navegacao..: " + protocolo);
22.              out.print("</pre>");
23.          %>
24.      </body>
25. </html>

```

- Leia atentamente os comentários, eles estão explicando o código. Vale lembrar que, no dia a dia, se faz comentários com critérios sobre o que realmente é necessário. Temos, inclusive, ferramentas (por exemplo, o javadoc) para extrair comentários feitos em um código, para que sirva de documentação.
- Comentários óbvios, como os colocados acima, valem apenas em tempo de aprendizagem. Em tela, o código apresenta:



Teste JSP - scriptlet e comentários

```

Path do contexto (diretório do site): /07_introducao_JSP
Protocolo utilizado para navegacao..: HTTP/1.1

```

Vejamos também o código que o usuário conseguirá ver no navegador:

```

1.  <!DOCTYPE html>
2.  <html>
3.      <head>
4.          <title>JSP Page</title>
5.      </head>
6.      <body>
7.          <h3>Teste JSP - scriptlet e comentários</h3>
8.          <pre>
9.              Path do contexto (diretório do site): /07_introducao_JSP
10.             Protocolo utilizado para navegacao..: HTTP/1.1
11.          </pre>
12.      </body>
13. </html>

```

Tipos de dados

Os principais tipos de dados no JSP são boolean, byte, double, int e String.

Variáveis

As variáveis nos servem de referência aos dados que precisamos armazenar na memória do computador durante um determinado tempo. Às vezes, dentro de um método, de uma classe, de uma página, uma sessão, ou ainda, durante todo ciclo de vida de uma aplicação.

Vale a pena lembrarmos que, em Java, temos também várias classes para conversão de tipos de dados, conforme a necessidade.

Exemplo:

```
1. boolean jsp = true;
2. byte idade = 33;
3. double saldoDasContas = 98779871897.67;
4. int qtdeContasSP = 483648;
5. String texto = "Este exemplo serve apenas para fins didáticos!!!";
```

Constantes

Caso seja necessário definir uma constante, utiliza-se a palavra reservada final antes do tipo da variável.

Exemplo:

```
1. final String AUTOR_DO_SISTEMA = "AAS";
```



Declarações e expressões

Outros elementos (ou TAGs) em JSP importantíssimos são as declarações e as expressões.

Declarações

As declarações devem ficar entre as TAGs: `<%! e %>`.

Com esse elemento, pode-se declarar variáveis ou métodos para a página JSP. Lembre-se de que nosso JSP transforma-se em um servlet, que é uma classe em Java. Assim, a variável seria de classe e o método seria o método da classe que será criada no servidor pelo JSP acionado.

Expressões

As expressões são comumente utilizadas para substituir método `out.print()`, que retorna ao navegador do usuário um valor convertido em uma string. Devem ficar entre as TAGs `<%= e %>`.

Vejamos um exemplo que envolve declarações e expressões:

```
1.  <!DOCTYPE html>
2.  <html>
3.      <head>
4.          <title>JSP Page</title>
5.      </head>
6.      <body>
7.          <h3>Declarações e Expressões:</h3>
8.          <%!
9.              // Exemplo de Declaração de um método que calcula a média:
10.             private float calcMedia(float n1, float n2, float n3) {
11.                 float m;
12.                 m = (n1 + n2 + n3) / 3;
13.                 return m;
14.             }
15.             // Exemplo de três variáveis de classe:
16.             // O F colocado após os valores, indica que esses valores são Float.
17.             float n1 = 10;
18.             float n2 = 8.5F;
19.             float n3 = 4.5F;
20.         %>
21.         <pre>
22.             Nota 1: <%=n1%>
23.             Nota 1: <%=n1%>
24.             Nota 1: <%=n1%>
25.             Média.: <%=calcMedia(n1,n2,n3)%>
26.         </pre>
27.     </body>
28. </html>
```

Veja que o código tem um método para calcular a média e três variáveis de classe.



Referências

GONÇALVES, Edson. *Desenvolvendo aplicações web com JSP, servlets, Java Server Faces, Hibernate, EJB 3 Persistence e Ajax*. São Paulo: Ciência Moderna, 2007.

SANTOS, Rui Rossi dos. *Java na web: programando sites dinâmicos*. Rio de Janeiro: Axcel Books, 2007.

TERUEL, Evandro Carlos. *Web total: desenvolva sites com tecnologias de uso livre: prático e avançado*. São Paulo: Érica, 2009.



Avalie este tópico



ANTERIOR

Ambiente de desenvolvimento

Biblioteca

(<https://www.uninove.br/conhec>

a-

uninove/biblioteca/sobre-

a-

biblioteca/apresentacao/)

Portal Uninove

(<http://www.uninove.br>)

Mapa do Site



Índice

Estruturas de desvio condicional e de loops

ção em

© Todos os direitos reservados

Ajuda?

(<https://ava.un>

idcurso=)

