

Unidade 4: Introdução à Tecnologia Servlets

Como Construir um Servlet Básico

Prof. Daniel Caetano

Objetivo: Preparar o aluno para construir Servlets para compor aplicações Web.

Bibliografia: QIAN, 2007; DEITEL, 2005.

INTRODUÇÃO

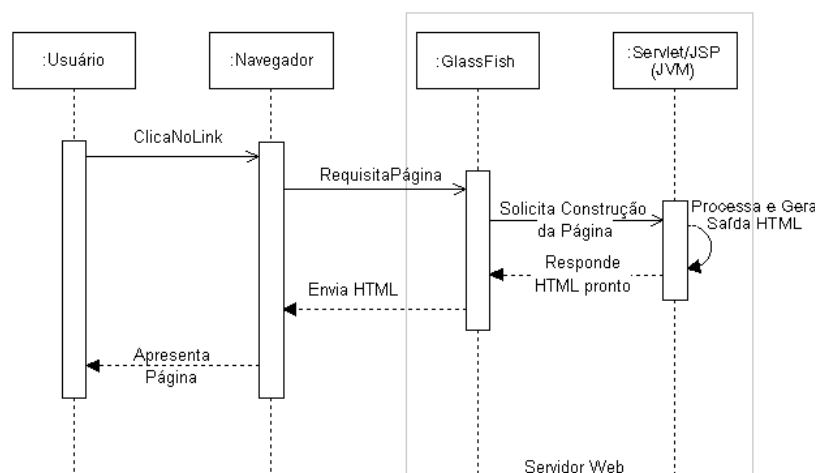
Como este curso tem o objetivo de apresentar brevemente a linguagem Java em um contexto de desenvolvimento para a Web, faz-se necessário estudar um pouco dos componentes que tornam essa apresentação possível.

Neste curso iremos utilizar um ambiente baseado no servidor de aplicações GlassFish, que fornece recursos similares ao uso combinado do TomCat com o JBoss, isto é, combina um servidor de páginas Web básico com os recursos avançados do Java Enterprise Edition 5 e 6 (JEE5 e JEE6). Nesta aula serão apresentados:

- 1) O que é um servlet, como ele funciona e seu ciclo de vida
- 2) Configurando uma Web Application com o NetBeans
- 3) Construindo uma Servlet com o NetBeans
- 4) Adicionando código à Servlet

1. O QUE É UM SERVLET

- O que é um Servlet.
- Como o Servlet é acionado:



- Ciclo de Vida de um Servlet:
 - Invocação => init() => service() => destroy()
- O Init pode ocorrer logo que o servidor de aplicações se inicia...
 - Ou apenas quando o Servlet for executado a primeira vez!
- O Servlet fica a maior parte do tempo em modo "service"

2. CONFIGURANDO UMA WEB APPLICATION

- Clicar em Criar Projeto
- Selecionar "Java Web" e "Aplicação Web"
- Clicar em Próximo.
- Dar nome ao projeto "WProjeto1"
- Clicar em Próximo.
- Clicar em Finalizar.

- web.xml : arquivo de configuração do GlassFish
 - Configuração pela Interface
 - Configuração pelo XML

3. CONSTRUINDO UMA SERVLET

```
<title>Cálculo de Índice de Massa Corporal</title>
<h1>Digite seus dados:</h1>
```

- Construir formulário HTML com:
 - action para "Imc" método "post"
 - peso e altura, dentro de tabela
 - botão submit
- Executar: erro => não existe o servlet!

- Clicar com botão direito em "Pacotes de Código Fonte"
- Selecionar **Novo > Pacote Java**
- Criar pacote com nome **imc**, clicando depois em Finalizar.
- Clicar com botão direito no pacote "imc".
- Selecionar **Novo > Servlet** (se não existir, procure na opção "Outros...")
- Dê o nome **Imc** para o servlet e clique em **Próximo**.
- Marque "**Adicionar informação ao descritor de implementação (web.xml)**" e, depois, clique em **Finalizar**.

- web.xml : arquivo de configuração das servlets disponíveis
 - Aba "servlets" permite configurar descrições e ordem de inicialização
 - Inicializar na criação do GlassFish / Na Execução
 - Edição direto no XML

- Executar: funciona => mas não faz nada (não programamos!)

- Mudar o conteúdo do texto de saída da Servlet para:
`<title>Cálculo de Índice de Massa Corporal: Resultados</title>`
`<h1>Índice de Massa Corporal: </h1>`
- Adicionar atributo "double imc = 18.0;"
- Adicionar a impressão do atributo no texto de resultado.

4. ADICIONANDO CÓDIGO À SERVLET

- Dentro do processRequest:
 - Pegar os parâmetros com
`String pesoT = request.getParameter("peso");`
`String alturaT = request.getParameter("altura");`
 - Formato String x formato numérico: conversão
`double peso = Double.valueOf(pesoT);`
`double altura = Double.valueOf(alturaT);`
 - Calcular imc com:
`imc = peso / (altura * altura);`
- Executar: funciona! => Erro em números com vírgula!
- Corrigir conversão:
`pesoT = pesoT.replaceAll(",", ".");`
- Impressão do número está ruim! Formatar saída:
`out.printf` com uso do parâmetro **%3.1f**
- Acrescentar **ifs** para mostrar as descrições:

<code>imc < 18.5</code>	=> Cuidado! Você está muito abaixo do peso!
<code>18.5 <= imc < 25.0</code>	=> Parabéns! Você está em seu peso ideal!
<code>25.0 <= imc < 30.0</code>	=> Atenção! Você está acima de seu peso ideal!
<code>30.0 <= imc < 35.0</code>	=> Atenção! Obesidade grau 1!
<code>35.0 <= imc < 40.0</code>	=> Cuidado! Obesidade grau 2!
<code>40.0 <= imc</code>	=> Cuidado! Obesidade grau 3!

5. BIBLIOGRAFIA

DEITEL, H.M; DEITEL, P.J. **Java: como programar** - Sexta edição. São Paulo: Pearson-Prentice Hall, 2005.

QIAN, K; ALLEN, R; GAN, M; BROWN, R. **Desenvolvimento Web Java**. Rio de Janeiro: LTC, 2007.