

# Linguagem Técnica de Programação Web

## **AULA 3 – Configuração do ambiente de desenvolvimento de uma aplicação Web**

Prof. João Paulo Pimentel  
[joao.pimentel@projecao.br](mailto:joao.pimentel@projecao.br)

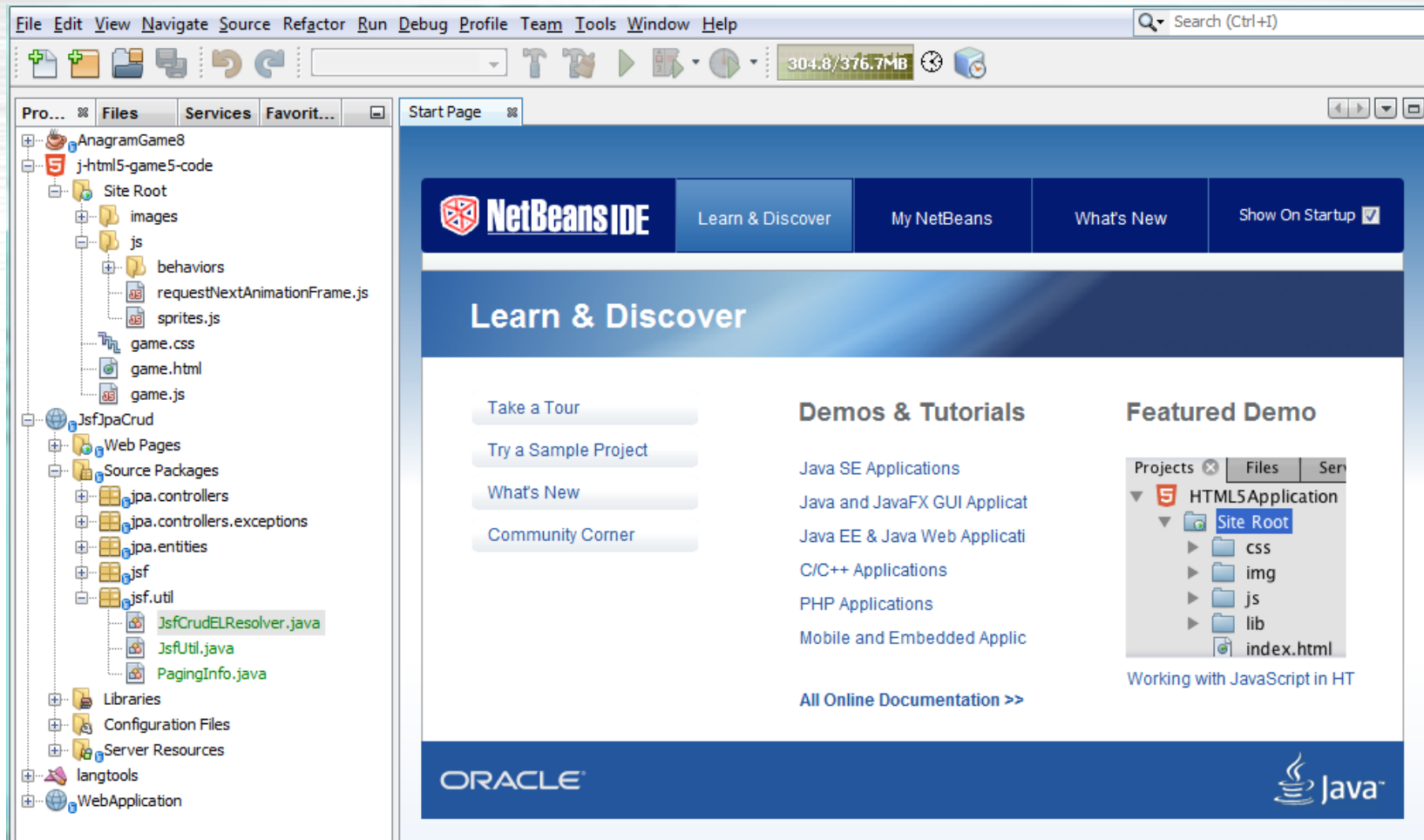
---

# Roteiro da Aula

- Ferramentas necessárias:
    - IDE Netbeans
    - JDK
    - Apache Tomcat
    - GlassFish
  - Criando um Projeto Web no Netbeans
  - Desenvolvendo Servlet
  - Exercícios práticos
-

# Ferramentas para desenvolvimento Web

- IDE Netbeans





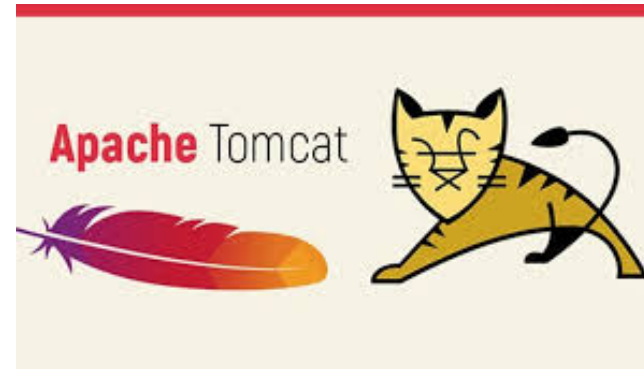
# Ferramentas para desenvolvimento Web



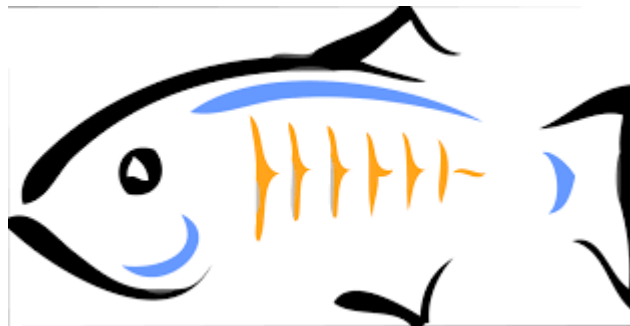
## JDK



## Apache Tomcat



## GlassFish

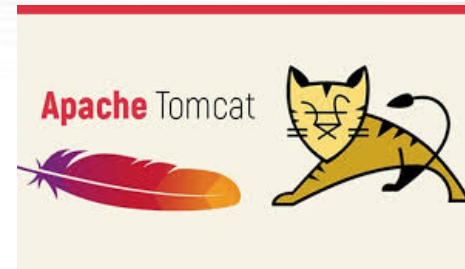




# Ferramentas para desenvolvimento Web



- **Sobre o Apache Tomcat**



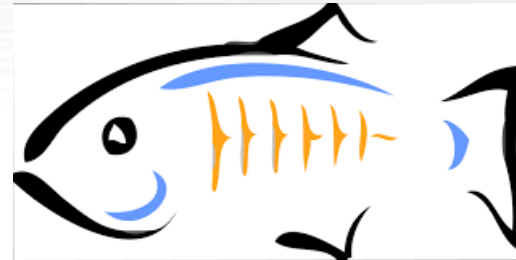
- O Tomcat é simplesmente um servidor HTTP e um contêiner de servlet Java.
  - É executado pela comunidade Apache.
  - É de código aberto e gratuito.
  - Suporta as tecnologias Java EE.
  - O Tomcat possui uma área de memória mais leve (~ 60-70 MB).
  - Servidor para Java Servlet e JavaServer Pages.
-



# Ferramentas para desenvolvimento Web

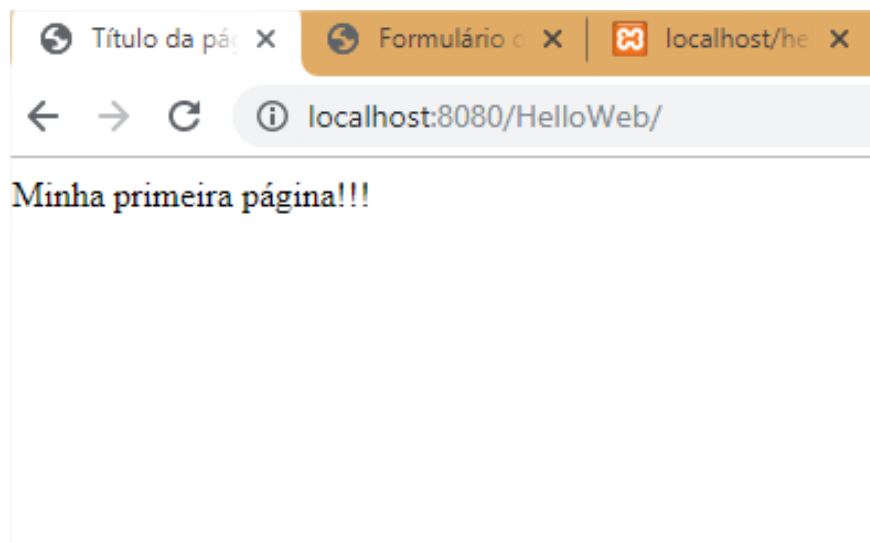


- **Sobre o GlassFish**

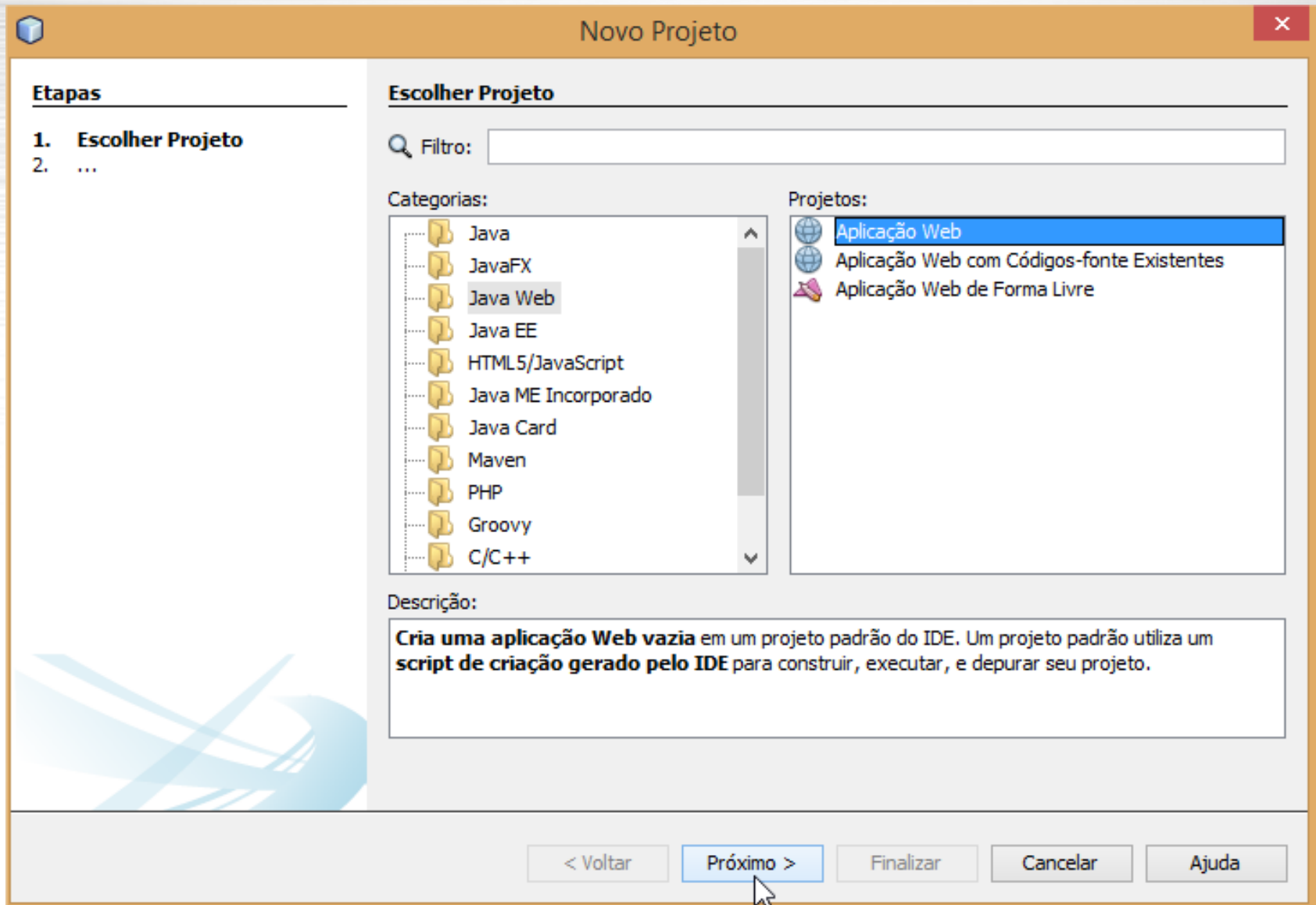


- O Glassfish é um servidor de aplicativos Java EE completo, incluindo um contêiner EJB e todos os outros recursos dessa pilha (API de persistência Java (JPA), JSF, Serviço de Mensagens Java (JMS), API Java para serviços de rede RESTfull (JAX-RS), API Java para processamento de JSON (JSON-P), ...
  - Ele vem da Oracle e, portanto, cada nova especificação será testada e implementada primeiro com o GlassFish.
-

- Vamos desenvolver o HelloWeb?
- Selecione Arquivo → Novo Projeto (Ctrl-Shift-N) no menu principal. Em Categorias, selecione **Java Web**. Em Projetos, selecione **Aplicação Web** e clique em **Próximo**.




# Criando um Projeto Web no Netbeans





# Criando um Projeto Web no Netbeans



Novo Aplicação Web

**Etapas**

1. Escolher Projeto
- 2. Nome e Localização**
3. Servidor e Definições
4. Frameworks

**Nome e Localização**

Nome do Projeto: HelloWeb

Localização do Projeto: C:\Users\JoãoPaulo\Documents\NetBeansProjects

Procurar...

Pasta do Projeto: C:\Users\JoãoPaulo\Documents\NetBeansProjects\HelloWeb

☐ Usar pasta dedicada para armazenar bibliotecas

Pasta Bibliotecas: 

Procurar...

Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).

< Voltar

Próximo >

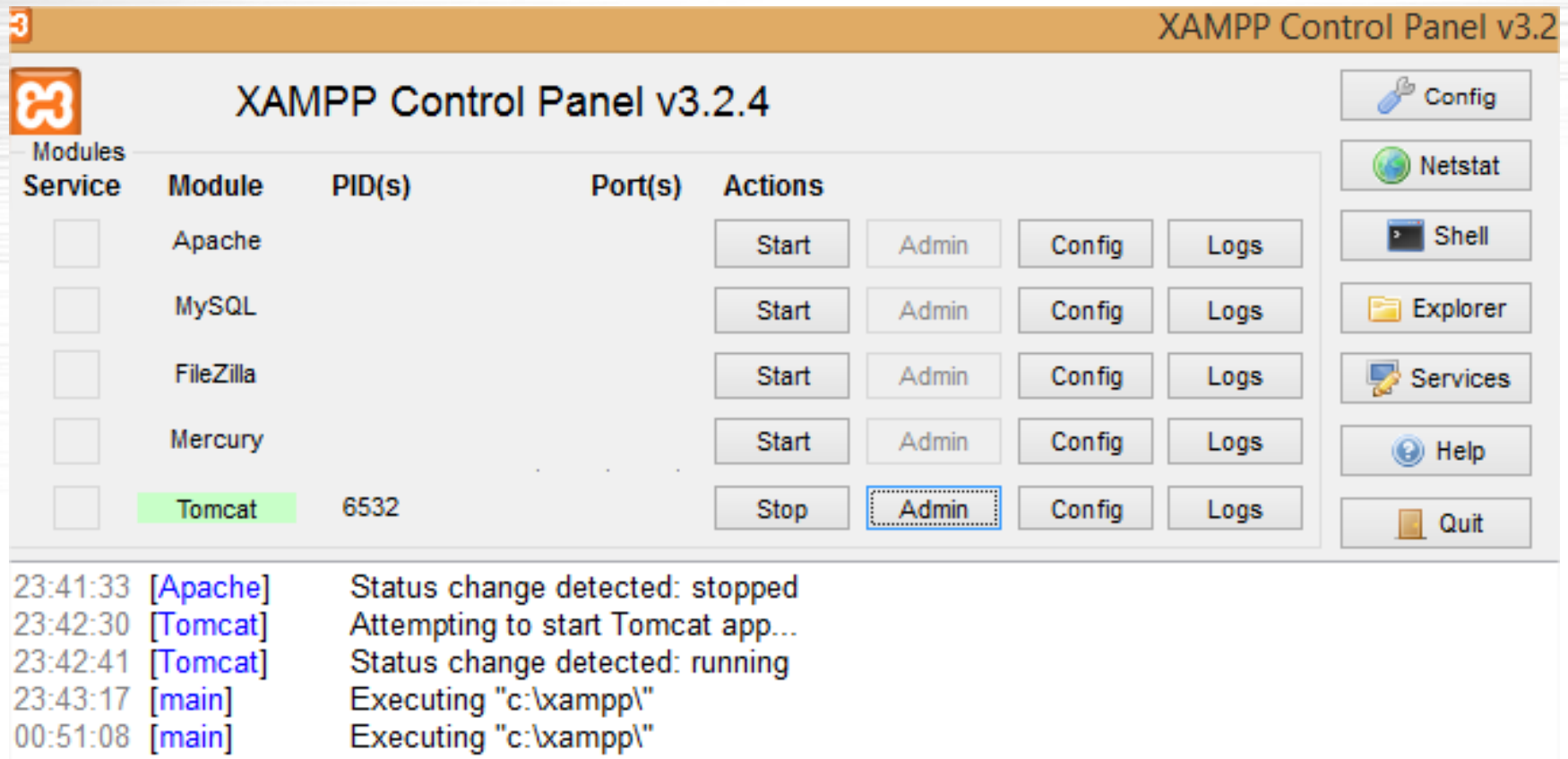
Finalizar

Cancelar

Ajuda

# Criando um Projeto Web no Netbeans

- C:\xampp\tomcat\webapps




**XAMPP Control Panel v3.2.4**

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache			Start Admin Config Logs
<input type="checkbox"/>	MySQL			Start Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat	6532		Stop Admin Config Logs

23:41:33 [Apache] Status change detected: stopped  
23:42:30 [Tomcat] Attempting to start Tomcat app...  
23:42:41 [Tomcat] Status change detected: running  
23:43:17 [main] Executing "c:\xampp\  
00:51:08 [main] Executing "c:\xampp\  
...

# Criando um Projeto Web no Netbeans



Novo Aplicação Web

**Etapas**

1. Escolher Projeto
- 2. Nome e Localização**
3. Servidor e Definições
4. Frameworks

**Nome e Localização**

Nome do Projeto:

HelloWeb

Localização do Projeto:

C:\xampp\tomcat\webapps

Procurar...

Pasta do Projeto:

C:\xampp\tomcat\webapps\HelloWeb

☐ Usar pasta dedicada para armazenar bibliotecas

Pasta Bibliotecas:

Procurar...

Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).

< Voltar


Próximo >

Finalizar

Cancelar

Ajuda

# Criando um Projeto Web no Netbeans



Adicionar Instância do Servidor

**Etapas**

- Escolher Servidor**
- ...

**Escolher Servidor**

Servidor:

Amazon Beanstalk

Apache Tomcat ou TomEE

GlassFish Server

Oracle WebLogic Server

Servidor de Aplicações JBoss

WildFly Application Server

Nome:

Apache Tomcat ou TomEE

< Voltar



Próximo >

Finalizar

Cancelar

Ajuda


# Criando um Projeto Web no Netbeans


 Novo Aplicação Web 


**Etapas**

1. Escolher Projeto
2. Nome e Localização
- 3. Servidor e Definições**
4. Frameworks

**Servidor e Definições**

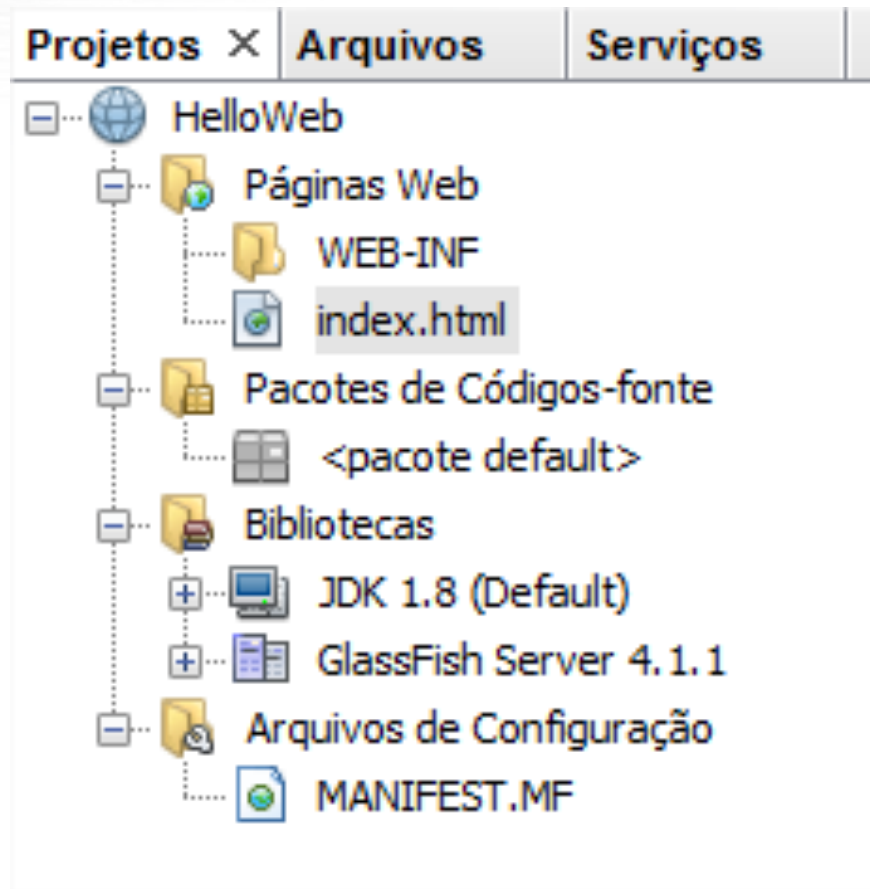
Adicionar à aplicação corporativa: <Nenhum(a)> 

Servidor:  

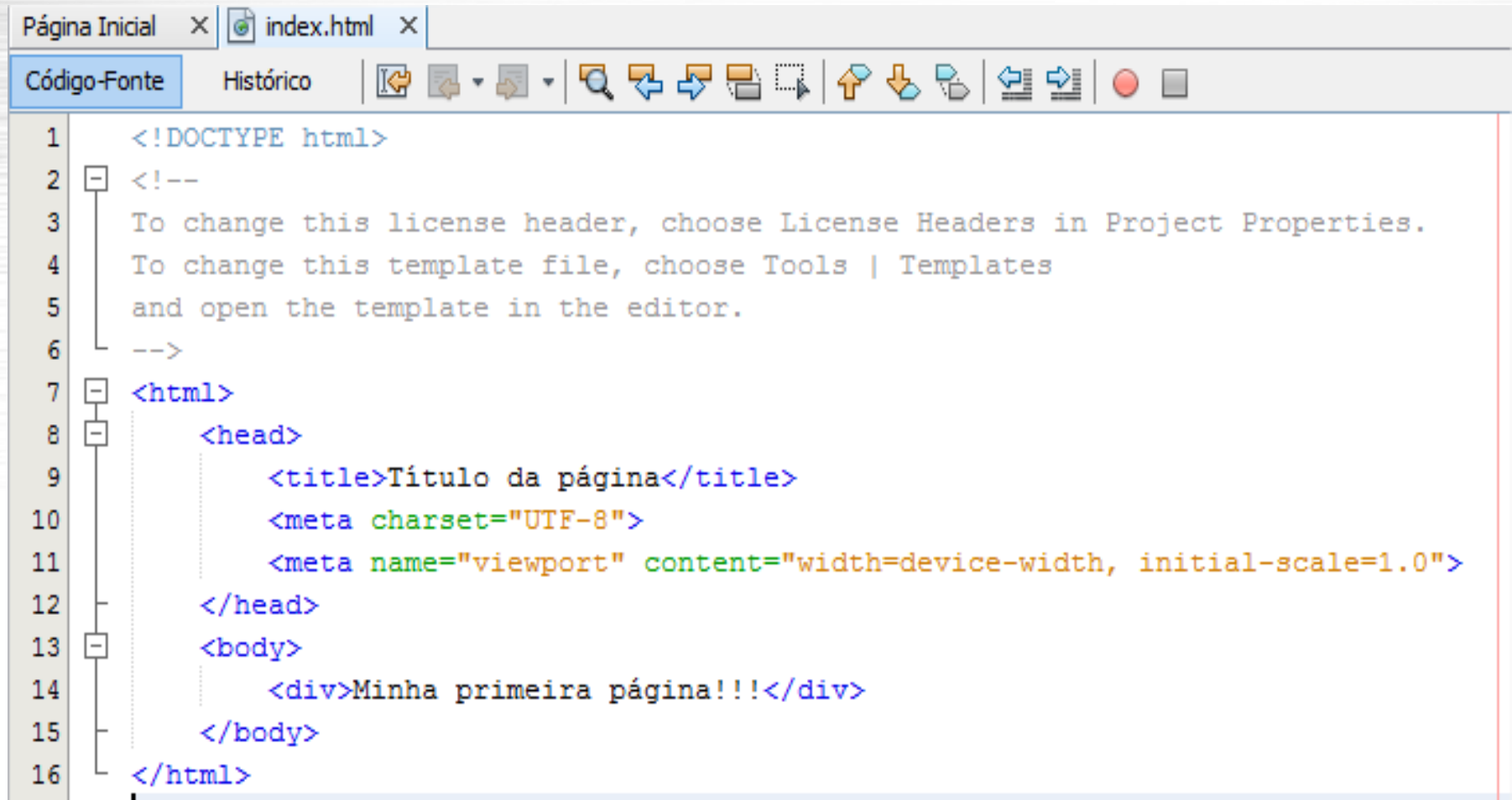
Versão do Java EE:  

Caminho do Contexto:

- O projeto **HelloWeb**:

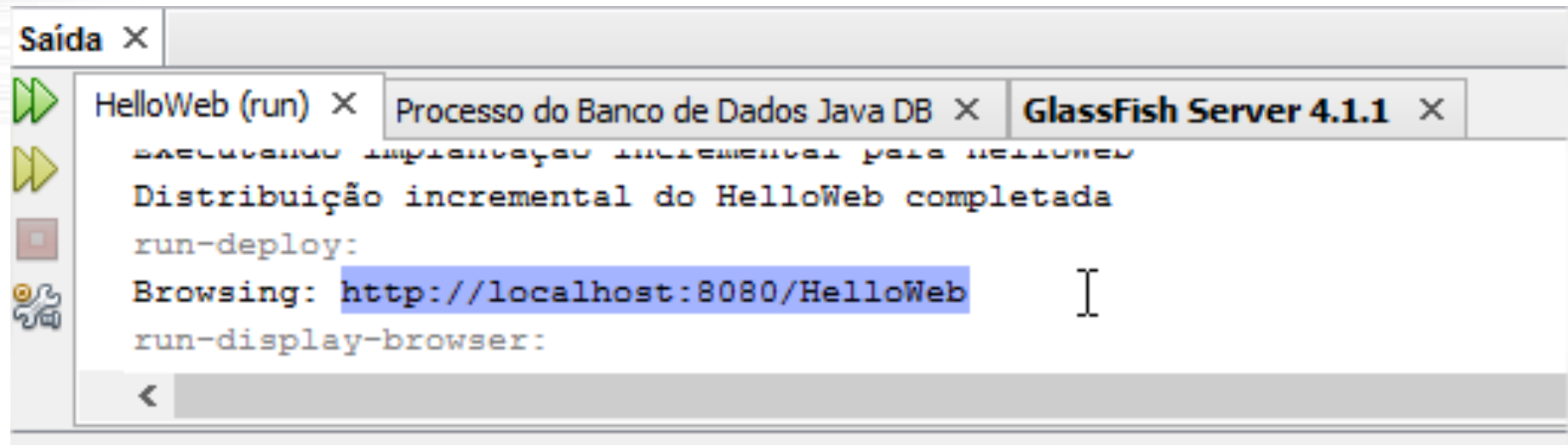


- O index.html



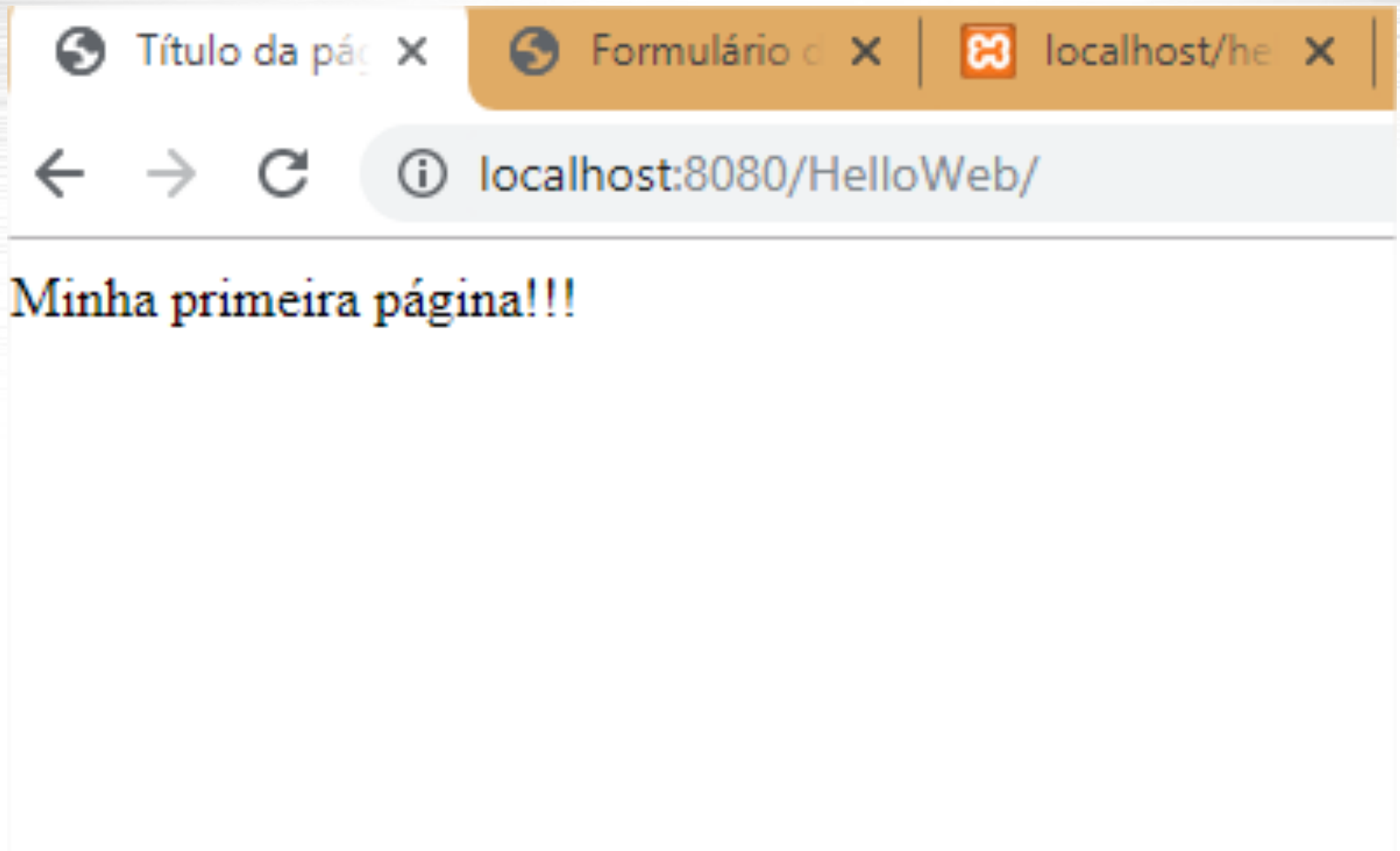
```
1  <!DOCTYPE html>
2  <!--
3  To change this license header, choose License Headers in Project Properties.
4  To change this template file, choose Tools | Templates
5  and open the template in the editor.
6  -->
7  <html>
8  <head>
9      <title>Título da página</title>
10     <meta charset="UTF-8">
11     <meta name="viewport" content="width=device-width, initial-scale=1.0">
12 </head>
13 <body>
14     <div>Minha primeira página!!!</div>
15 </body>
16 </html>
```

- Executando no **Netbeans**





- Executando no browser:



- Desenvolvendo Servlet
- Já que criamos nosso projeto Web, vamos agora desenvolver a primeira Servlet.
- O que são Servlets?

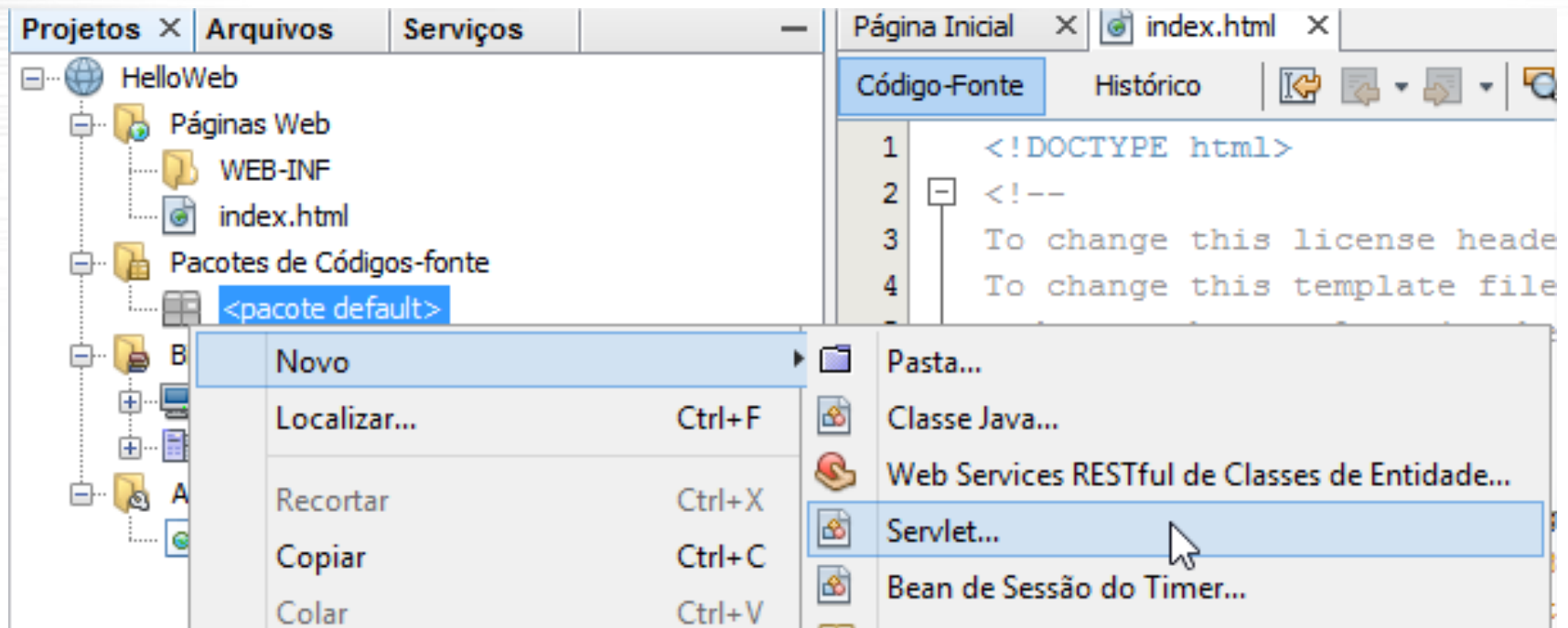
**Servlets** são classes Java, desenvolvidas de acordo com uma estrutura bem definida que quando instaladas e configuradas em um Servidor que implemente um Servlet Container, podem tratar requisições recebidas de clientes Web, como por exemplo os Browsers (Google Chrome®, Mozilla Firefox®, Internet Explorer® , ... )

---

- Mais sobre Servlets
  - As **Servlets** são a primeira forma que veremos de criar páginas dinâmicas com Java. Usaremos a própria linguagem Java para isso, criando uma classe que terá capacidade de gerar conteúdo HTML.
  - O nome "servlet" vem da ideia de um pequeno servidor cujo objetivo é receber chamadas HTTP, processá-las e devolver uma resposta ao cliente.
  - Cada servlet é, portanto, um objeto Java que recebe tais requisições (request) e produz algo (response), como uma página HTML dinamicamente gerada.
-

# Criando um Projeto Web no Netbeans

- Voltando ao nosso Projeto no Netbeans (HelloWeb), vamos criar a nossa primeira Servlet.



# Criando um Projeto Web no Netbeans

- Altere o nome da Classe para **MeuServlet**

**New Servlet**

**Etapas**

1. Escolher Tipo de Arquivo
- 2. Nome e Localização**
3. Configurar Implantação do Servlet

**Nome e Localização**

Nome da Classe:

Projeto:

Localização:

Pacote:

Arquivo Criado:

⚠ Advertência: é altamente recomendado que você não coloque classes Java no pacote default.

< Voltar   **Próximo >**   Finalizar   Cancelar   Ajuda

# Criando um Projeto Web no Netbeans

- Clique em **next**. Agora o Netbeans mostra o nome que vai ser seu Servlet.

**New Servlet**

**Etapas**

1. Escolher Tipo de Arquivo
2. Nome e Localização
3. **Configurar Implantação do Servlet**

**Configurar Implantação do Servlet**

Registre o Servlet na aplicação atribuindo ao Servlet um nome interno (Nome do Servlet). Depois especifique os padrões que identifiquem as URLs que invocam o Servlet. Separar múltiplos padrões com vírgulas.

☐ Adicionar informações ao descritor de implantação (web.xml)

Nome da Classe: MeuServlet

Nome do Servlet: MeuServlet

Padrão(ões) de URL: /MeuServlet

Parâmetros de Inicialização:

Nome	Valor
------	-------

Novo

Editar...

Excluir

< Voltar   Próximo >   **Finalizar**   Cancelar   Ajuda

# Criando um Projeto Web no Netbeans

- Após clicar em Finalizar ...

**Etapas**

1. Escolher Tipo de Arquivo
2. Nome e Localização
3. **Configurar Implantação do Servlet**

**Configurar Implantação do Servlet**

Registre o Servlet na aplicação atribuindo ao Servlet um nome interno (Nome do Servlet). Depois especifique os padrões que identifiquem as URLs que invocam o Servlet. Separar múltiplos padrões com vírgulas.

☐ Adicionar informações ao descritor de implantação (web.xml)

Nome da Classe:

Nome do Servlet:

Padrão(ões) de URL:

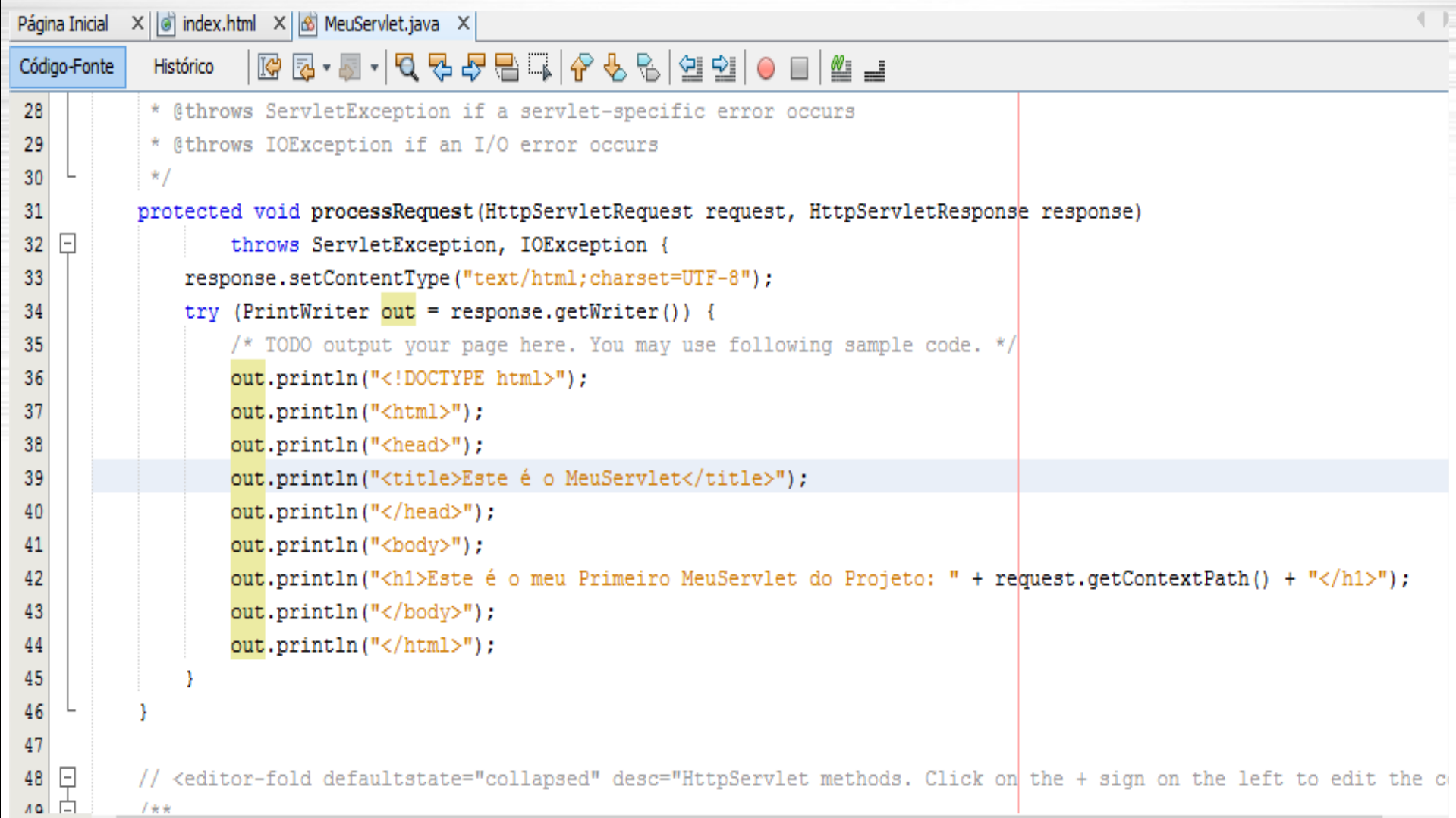
Parâmetros de Inicialização:

Nome	Valor
------	-------

Aguarde.

# Criando um Projeto Web no Netbeans

- Servlet criado, vamos fazer algumas alterações?



```
Página Inicial x index.html x MeuServlet.java x
Código-Fonte Histórico
28 * @throws ServletException if a servlet-specific error occurs
29 * @throws IOException if an I/O error occurs
30 */
31 protected void processRequest(HttpServletRequest request, HttpServletResponse response)
32     throws ServletException, IOException {
33     response.setContentType("text/html;charset=UTF-8");
34     try (PrintWriter out = response.getWriter()) {
35         /* TODO output your page here. You may use following sample code. */
36         out.println("<!DOCTYPE html>");
37         out.println("<html>");
38         out.println("<head>");
39         out.println("<title>Este é o MeuServlet</title>");
40         out.println("</head>");
41         out.println("<body>");
42         out.println("<h1>Este é o meu Primeiro MeuServlet do Projeto: " + request.getContextPath() + "</h1>");
43         out.println("</body>");
44         out.println("</html>");
45     }
46 }
47
48 // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the c
49 /**
```



- Executando MeuServlet no browser:
- <http://localhost:8080/HelloWeb/MeuServlet>



**Este é o meu Primeiro MeuServlet do Projeto: /HelloWeb**

- Onde estão os arquivos dessa aplicação?
  - Os arquivos .java vão ficar dentro do diretório Pacote de Códigos-Fonte.
-

- Mais algumas alterações no nosso Servlet:

```
39 out.println("<title>Este é o MeuServlet</title>");
40 out.println("</head>");
41 out.println("<body>");
42 out.println("<h1>Este é o meu Primeiro MeuServlet do Projeto: " + request.getContextPath() + "</h1>");
43 out.println("<h1>Nome do Servidor: " + request.getServerName());
44 out.println("<h1>Local da Porta do Servidor: " + request.getLocalPort());
45 out.println("</body>");
46 out.println("</html>");
47 }
```

`request.getServerName()` → trás o nome do servidor

`request.getLocalPort()` → trás a porta do servidor

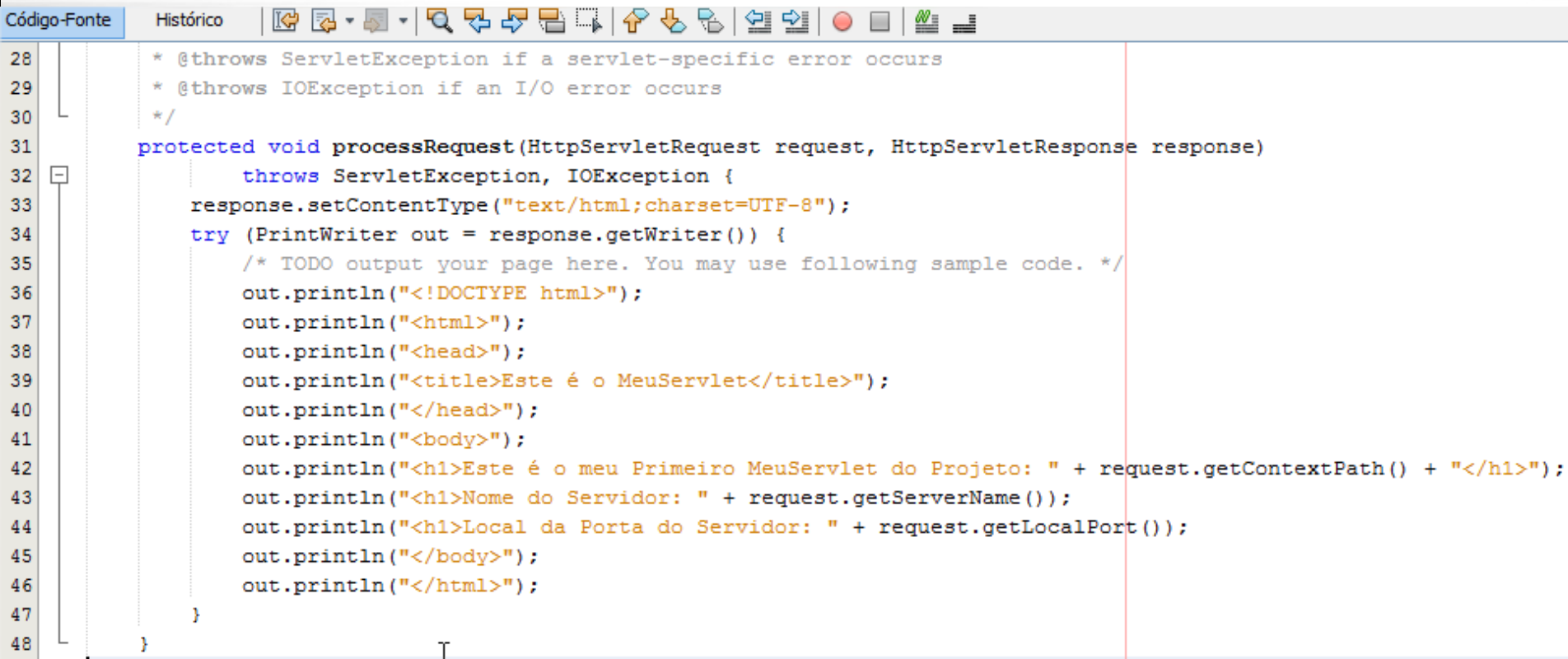
---

- Atualizando no browser:



# Criando um Projeto Web no Netbeans

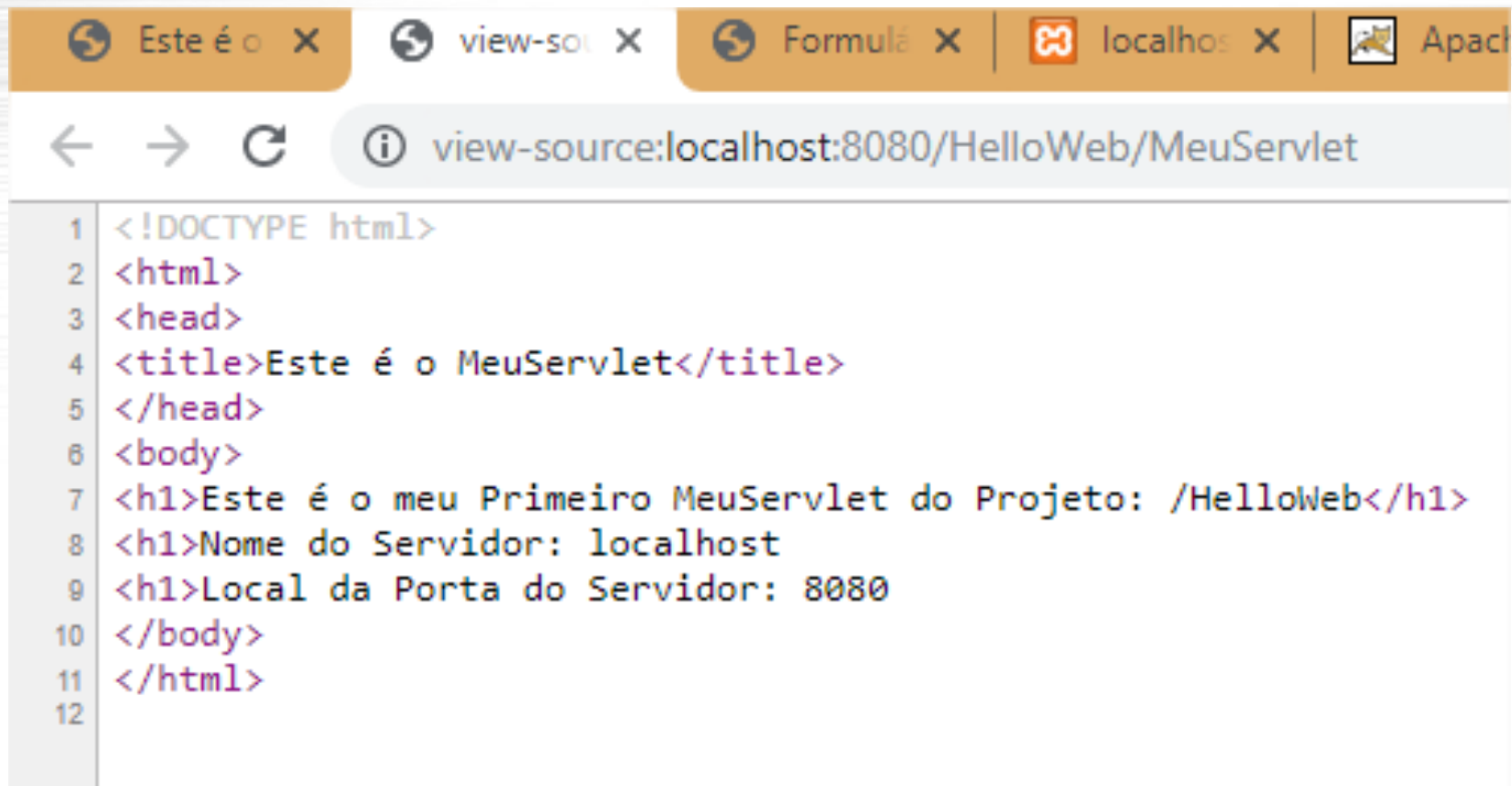
- Vejam que no servidor ficam os códigos Java:



```
Código-Fonte  Histórico  [Icons]

28  * @throws ServletException if a servlet-specific error occurs
29  * @throws IOException if an I/O error occurs
30  */
31  protected void processRequest(HttpServletRequest request, HttpServletResponse response)
32      throws ServletException, IOException {
33      response.setContentType("text/html;charset=UTF-8");
34      try (PrintWriter out = response.getWriter()) {
35          /* TODO output your page here. You may use following sample code. */
36          out.println("<!DOCTYPE html>");
37          out.println("<html>");
38          out.println("<head>");
39          out.println("<title>Este é o MeuServlet</title>");
40          out.println("</head>");
41          out.println("<body>");
42          out.println("<h1>Este é o meu Primeiro MeuServlet do Projeto: " + request.getContextPath() + "</h1>");
43          out.println("<h1>Nome do Servidor: " + request.getServerName());
44          out.println("<h1>Local da Porta do Servidor: " + request.getLocalPort());
45          out.println("</body>");
46          out.println("</html>");
47      }
48  }
```

- Já no cliente (browser), somente o HTML da aplicação:



The screenshot shows a web browser window with multiple tabs. The active tab is titled 'view-source:localhost:8080/HelloWeb/MeuServlet'. The address bar shows the URL 'view-source:localhost:8080/HelloWeb/MeuServlet'. The main content area displays the HTML source code of the application, which is a simple web page with a title and three paragraphs.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Este é o MeuServlet</title>
5 </head>
6 <body>
7 <h1>Este é o meu Primeiro MeuServlet do Projeto: /HelloWeb</h1>
8 <h1>Nome do Servidor: localhost
9 <h1>Local da Porta do Servidor: 8080
10 </body>
11 </html>
12
```

- Vamos desenvolver mais no nosso Projeto HelloWeb?
- Agora vamos incluir no **index.html** dois campos (**nome** e **email**) e um botão **Enviar** (submit).
- Estas informações serão enviadas ao nosso Java Servlet (MeuServlet.java) e nosso Java Servlet mostrará em tela o nome e e-mail enviado.

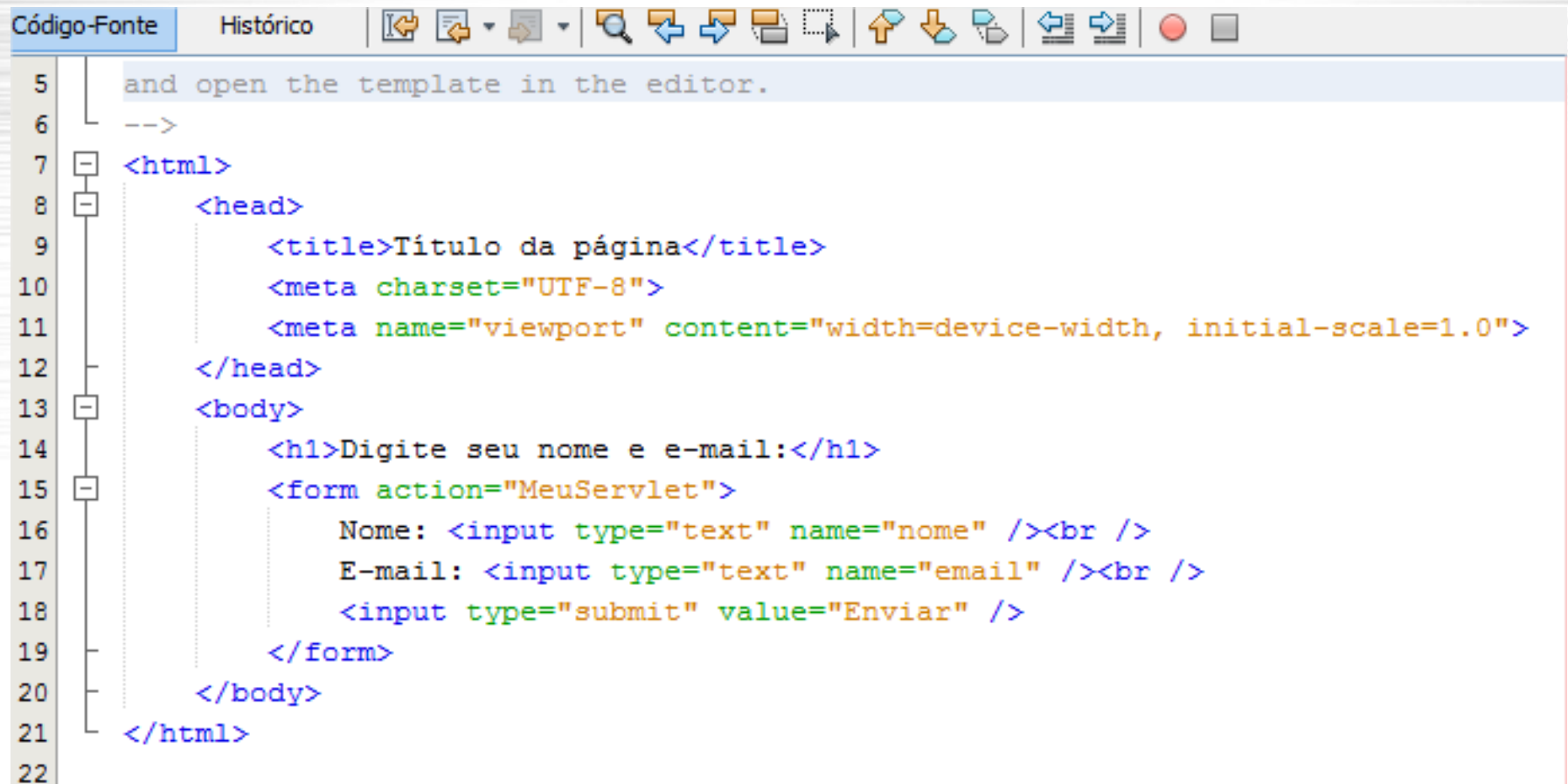
**Digite seu nome e e-mail:**

Nome:

E-mail:

---

- Nosso **index.html** ficará assim:



```
Código-Fonte  Histórico  [Icons]
5  and open the template in the editor.
6  -->
7  <html>
8  <head>
9      <title>Título da página</title>
10     <meta charset="UTF-8">
11     <meta name="viewport" content="width=device-width, initial-scale=1.0">
12 </head>
13 <body>
14     <h1>Digite seu nome e e-mail:</h1>
15     <form action="MeuServlet">
16         Nome: <input type="text" name="nome" /><br />
17         E-mail: <input type="text" name="email" /><br />
18         <input type="submit" value="Enviar" />
19     </form>
20 </body>
21 </html>
22
```



- Pegando os parâmetros da requisição no nosso Servlet:
  - Dentro do método **service** da nossa **Servlet** para adição de contatos, vamos buscar os dados que foram enviados na **requisição**.
  - Para buscarmos esses dados, precisamos utilizar o parâmetro **request** do método **service** chamando o método **getParameter("nomeDoParametro")**, onde o nome do parâmetro é o mesmo nome do **input** que você quer buscar o valor. Isso vai retornar uma **String** com o valor do parâmetro. Caso não exista o parâmetro, será retornado **null**:
-



- Exemplo do request.getParameter:

```
String valorDoParametro =  
request.getParameter("nomeDoParametro");
```

Acrescente o código abaixo no **MeuServlet.java**

```
out.println("<br><br><h1>Dados enviados do index.html: ");  
out.println("<h1>Nome: " + request.getParameter("nome"));  
out.println("<h1>E-mail: " + request.getParameter("email"));
```

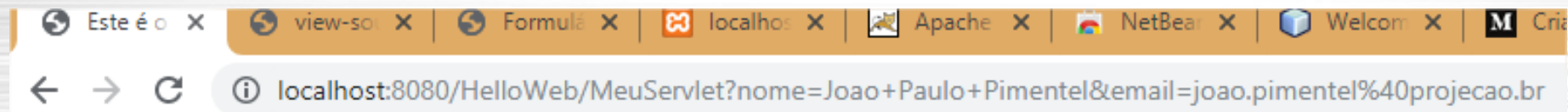
---

- Executando o Projeto:
- <http://localhost:8080/HelloWeb/>



The screenshot shows a web browser window with the address bar displaying `localhost:8080/HelloWeb/`. The main content area contains the text **Digite seu nome e e-mail:** followed by two input fields. The first field is labeled 'Nome:' and contains the text 'Joao Paulo Pimentel'. The second field is labeled 'E-mail:' and contains the text 'joao.pimentel@projecao.br'. Below these fields is a button labeled 'Enviar'.

- Após clicar em enviar:



**Este é o meu Primeiro MeuServlet do Projeto: /HelloWeb**

**Nome do Servidor: localhost**

**Local da Porta do Servidor: 8080**

**Dados enviados do index.html:**

**Nome: Joao Paulo Pimentel**

**E-mail: joao.pimentel@projecao.br**

# Exercícios

- Enviar os dados dos nossos formulários (Cadastro e Login) para servlets:

## Cadastro

Seu nome

Seu e-mail

Sua senha

**Cadastrar**

Já tem conta? [Ir para Login](#)

## Login

Seu e-mail

Sua senha

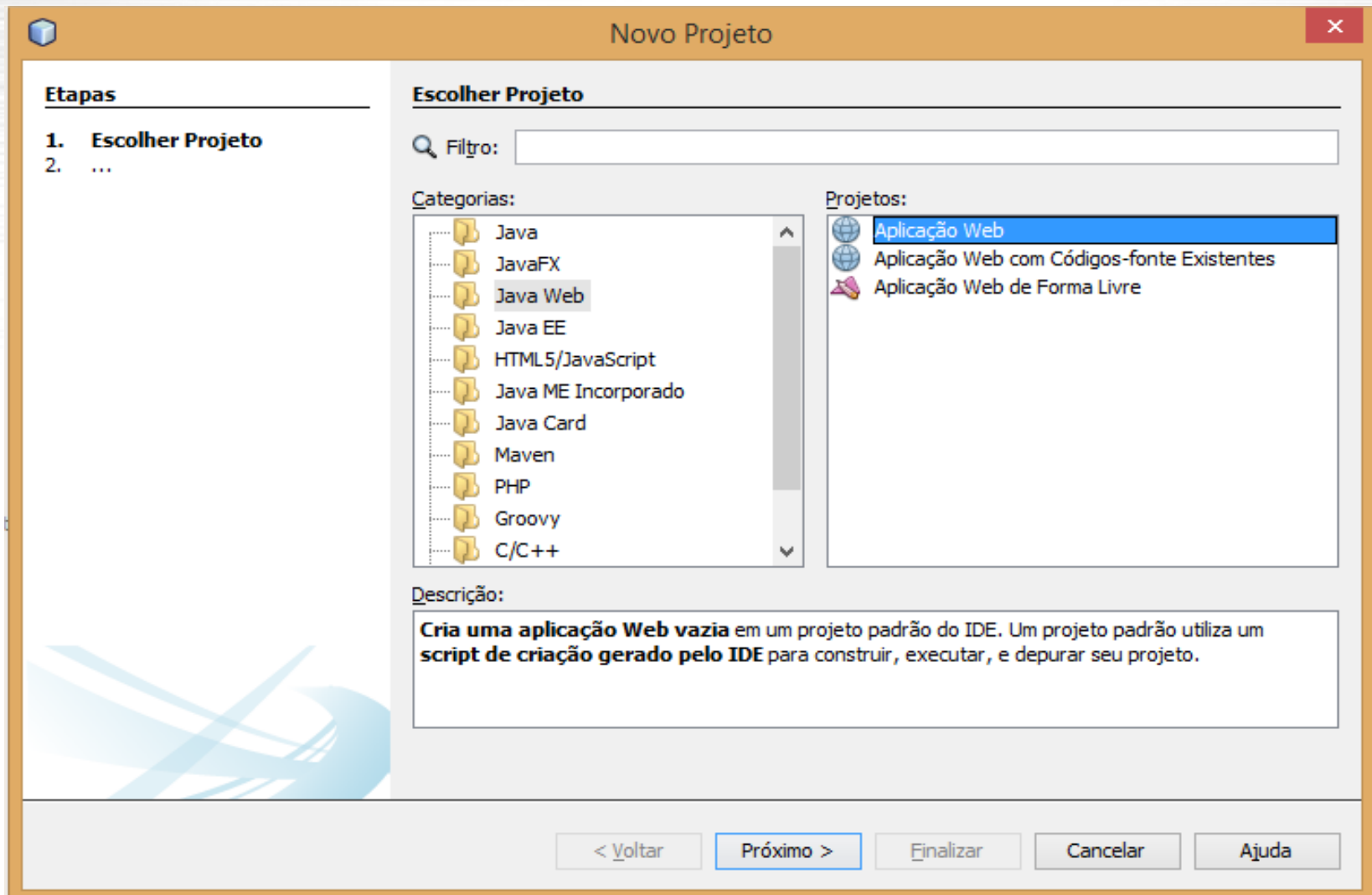
☐ Manter-me logado

**Logar**

Ainda não tem conta? [Cadastre-se](#)

# Exercícios

- Criando o ProjetoLTPWeb:



# Exercícios

- Criando o ProjetoLTPWeb:

**Novo Aplicação Web**

**Etapas**

1. Escolher Projeto
- 2. Nome e Localização**
3. Servidor e Definições
4. Frameworks

**Nome e Localização**

Nome do Projeto:

Localização do Projeto:

Pasta do Projeto:

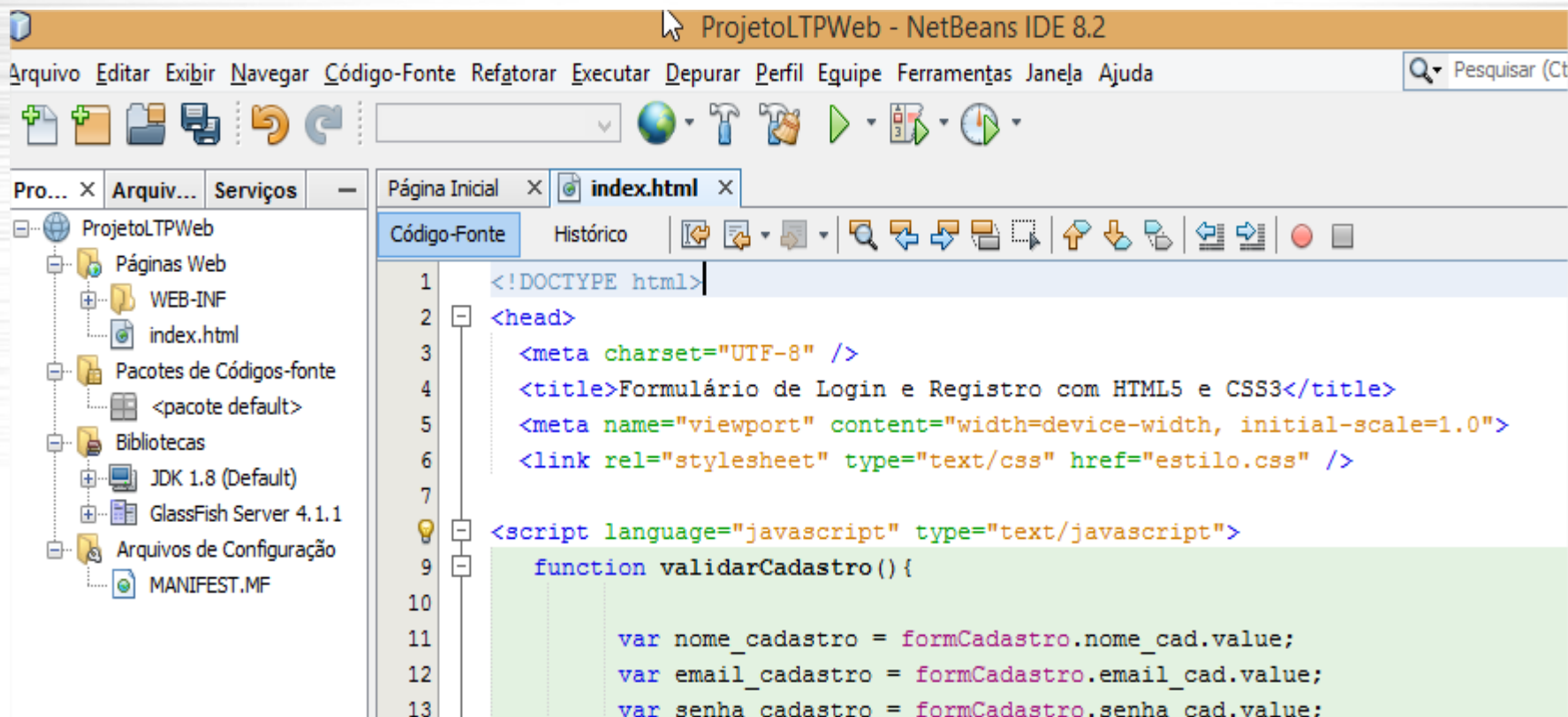
☐ Usar pasta dedicada para armazenar bibliotecas

Pasta Bibliotecas:

Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).

# Exercícios

- Colar o HTML da aplicação:



The screenshot shows the NetBeans IDE 8.2 interface. The title bar reads "ProjetoLTPWeb - NetBeans IDE 8.2". The menu bar includes "Arquivo", "Editar", "Exibir", "Navegar", "Código-Fonte", "Refatorar", "Executar", "Depurar", "Perfil", "Equipe", "Ferramentas", "Janela", and "Ajuda". The toolbar contains various icons for file operations and development. The left sidebar shows the project structure for "ProjetoLTPWeb", including "Páginas Web" (containing "WEB-INF" and "index.html"), "Pacotes de Códigos-fonte", "Bibliotecas" (JDK 1.8 and GlassFish Server 4.1.1), and "Arquivos de Configuração" (MANIFEST.MF). The main editor window displays the source code of "index.html" in the "Código-Fonte" tab. The code is as follows:

```
1 <!DOCTYPE html>
2 <head>
3   <meta charset="UTF-8" />
4   <title>Formulário de Login e Registro com HTML5 e CSS3</title>
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" type="text/css" href="estilo.css" />
7
8   <script language="javascript" type="text/javascript">
9     function validarCadastro() {
10
11       var nome_cadastro = formCadastro.nome_cad.value;
12       var email_cadastro = formCadastro.email_cad.value;
13       var senha_cadastro = formCadastro.senha_cad.value;
```

# Exercícios

- Sem o .css

## Login

Seu e-mail

Sua senha

☐ Manter-me logado

Ainda não tem conta? [Cadastre-se](#)

## Cadastro

Seu nome

Seu e-mail

Sua senha

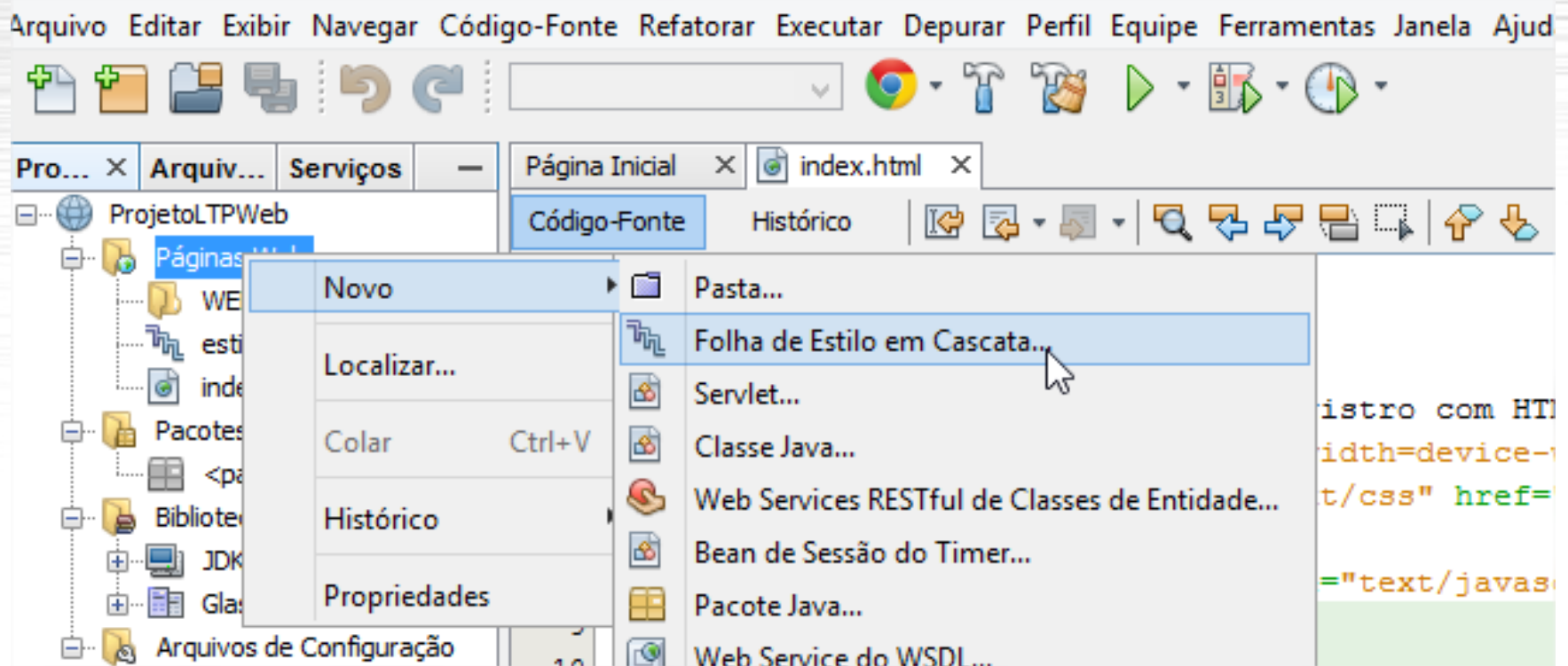
Já tem conta? [Ir para Login](#)

---



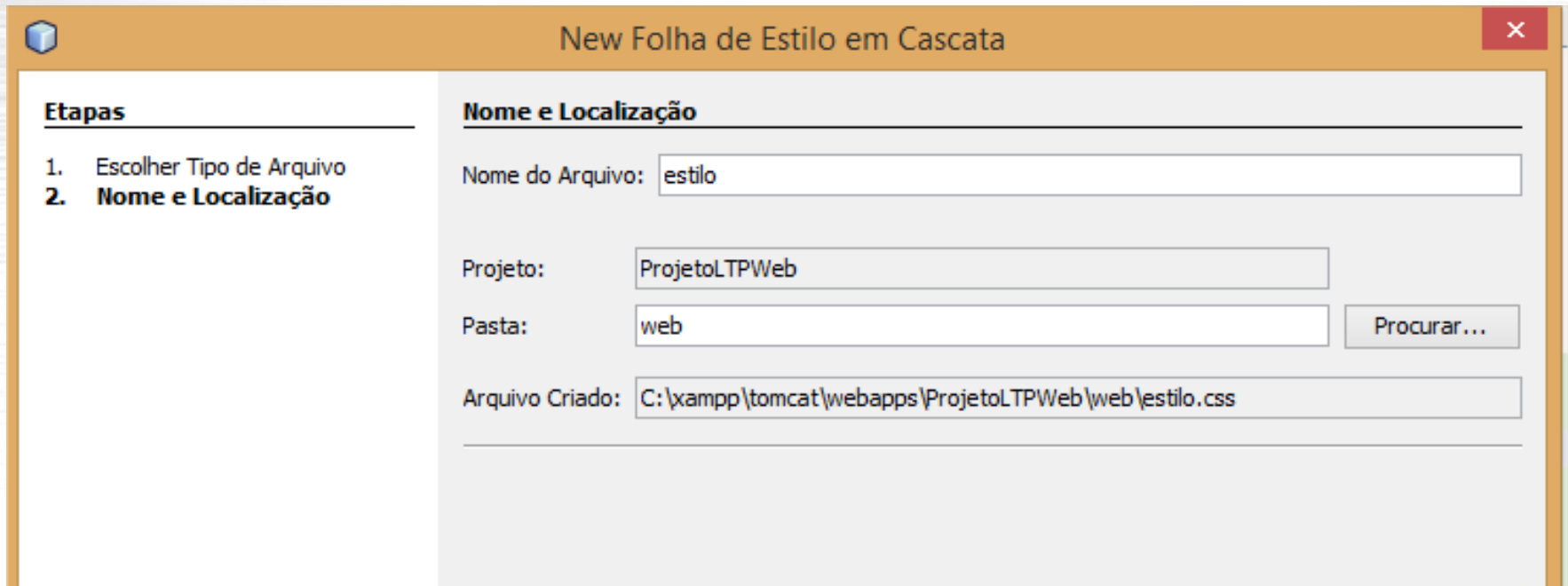
# Exercícios

- Criar o .css no Projeto:



# Exercícios

- Criando o estilo.css



**New Folha de Estilo em Cascata**

**Etapas**

1. Escolher Tipo de Arquivo
2. **Nome e Localização**

**Nome e Localização**

Nome do Arquivo:

Projeto:

Pasta:

Arquivo Criado:

# Exercícios

- Com o .css



The screenshot shows a web browser window with the address bar displaying "localhost:8080/ProjetoLTPWeb/#paracadastro". The page content is a registration form titled "Cadastro" in a large, bold, teal font. Below the title, there are three input fields: "Seu nome" with a placeholder "nome", "Seu e-mail" with a placeholder "contato@projecao.br", and "Sua senha" with a placeholder "ex. 1234". A teal button labeled "Cadastrar" is positioned below the input fields. At the bottom right, there is a link "Ir para Login" in a yellow box, preceded by the text "Já tem conta?".

← → ↻ ⓘ localhost:8080/ProjetoLTPWeb/#paracadastro

## Cadastro

Seu nome

Seu e-mail

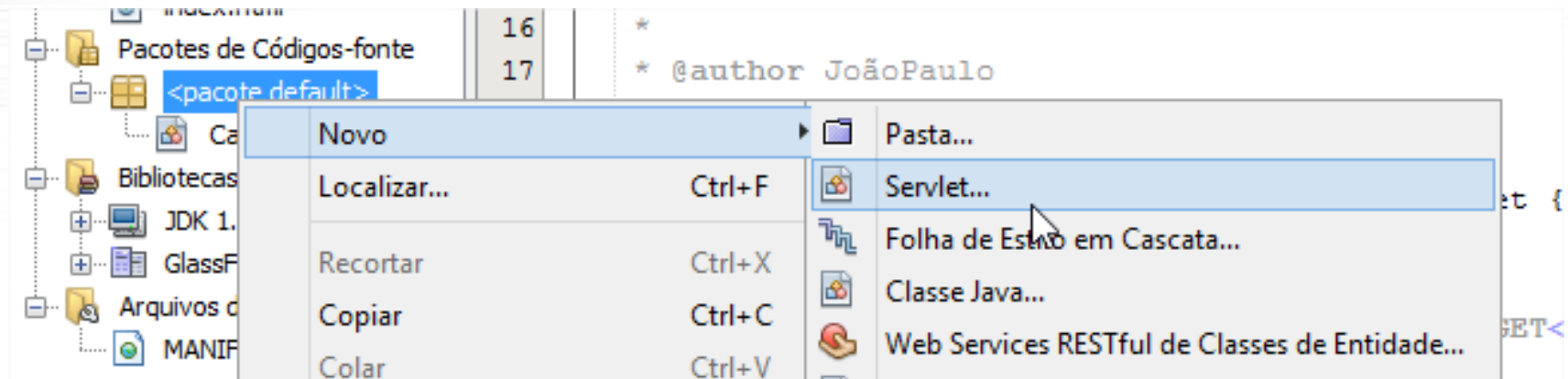
Sua senha

Cadastrar

Já tem conta? [Ir para Login](#)

# Exercícios

- Criando um Servlet para o Cadastro (**CadastroServlet**) e outro para Login (**LoginServlet**):



# Exercícios

- Executando no browser:

The screenshot shows a web browser window with two overlapping panels. The left panel is a registration form titled 'Cadastro' with fields for 'Seu nome' (filled with 'Joao Paulo Pimentel'), 'Seu e-mail' (filled with 'jpapim@gmail.com'), and 'Sua senha' (masked with '.....'). A blue 'Cadastrar' button is at the bottom. The right panel is a confirmation message from 'localhost:8080' stating: 'Usuário cadastrado com sucesso! O e-mail do cadastro jpapim@gmail.com terá que ser validado! Favor confirmar o cadastro no seu e-mail!'. It includes an 'OK' button and the same registration form fields below it. The browser's address bar shows 'localhost:8080/ProjetoLTPWeb/CadastroServlet'.

**Cadastro**

Seu nome  
Joao Paulo Pimentel

Seu e-mail  
jpapim@gmail.com

Sua senha  
.....

Cadastrar

localhost:8080 diz  
Usuário cadastrado com sucesso!  
O e-mail do cadastro jpapim@gmail.com terá que ser validado!  
Favor confirmar o cadastro no seu e-mail!

OK

Seu e-mail  
jpapim@gmail.com

Sua senha  
.....

Cadastrar

localhost:8080/ProjetoLTPWeb/CadastroServlet

## Dados Cadastrados:

**Nome: Joao Paulo Pimentel**

**E-mail: jpapim@gmail.com**

# Exercícios

- Executando no browser:

The image shows a web browser window with two overlapping views. The background view is a login page titled "Login" in teal. It has two input fields: "Seu e-mail" containing "joao.pimentel@projecao.br" and "Sua senha" with masked characters ".....". Below the password field is a checkbox labeled "Manter-me logado". A teal "Logar" button is at the bottom, with a mouse cursor hovering over it. The foreground view is a white modal box with a blue border. It contains the text "localhost:8080 diz" and "Bem vindo ao Sistema joao.pimentel@projecao.br", followed by a blue "OK" button. Below the modal, the browser's address bar shows "localhost:8080/ProjetoLTPWeb/LoginServlet". To the right of the address bar is a yellow "Cadastre-se" button. At the bottom of the browser window, a large white box displays the message "Bem vindo ao Sistema:" in bold black text, followed by "E-mail cadastrado: joao.pimentel@projecao.br" in bold black text, flanked by horizontal red lines.

**Login**

Seu e-mail  
joao.pimentel@projecao.br

Sua senha  
.....

☐ Manter-me logado

Logar

localhost:8080 diz  
Bem vindo ao Sistema joao.pimentel@projecao.br

OK

Seu e-mail  
joao.pimentel@projecao.br

Sua senha  
.....

☐ Manter-me logado

Logar

← → ↻ ⓘ localhost:8080/ProjetoLTPWeb/LoginServlet ?

Cadastre-se

**Bem vindo ao Sistema:**

**E-mail cadastrado: joao.pimentel@projecao.br**



**Por hoje é só!!!**

**Até a próxima aula...**

---