Programação Servidor em Sistemas Web

Introdução à Tecnologia Servlets

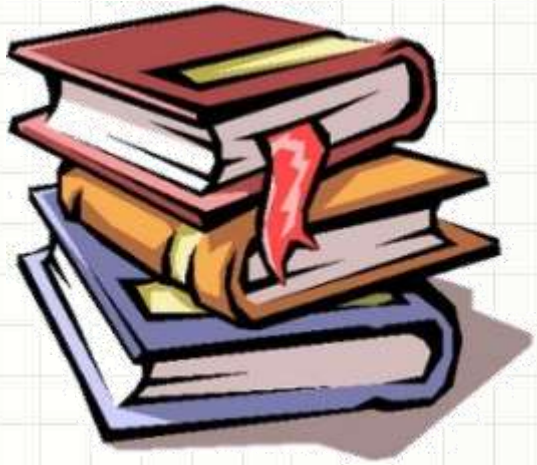
Prof. Dr. Daniel Caetano

2012 - 1

Objetivos

- Apresentar o conceito aplicações orientada a serviços via web
- Apresentar o papel dos contentores Java
- Capacitar para a construção de Aplicações Web simples

Material de Estudo

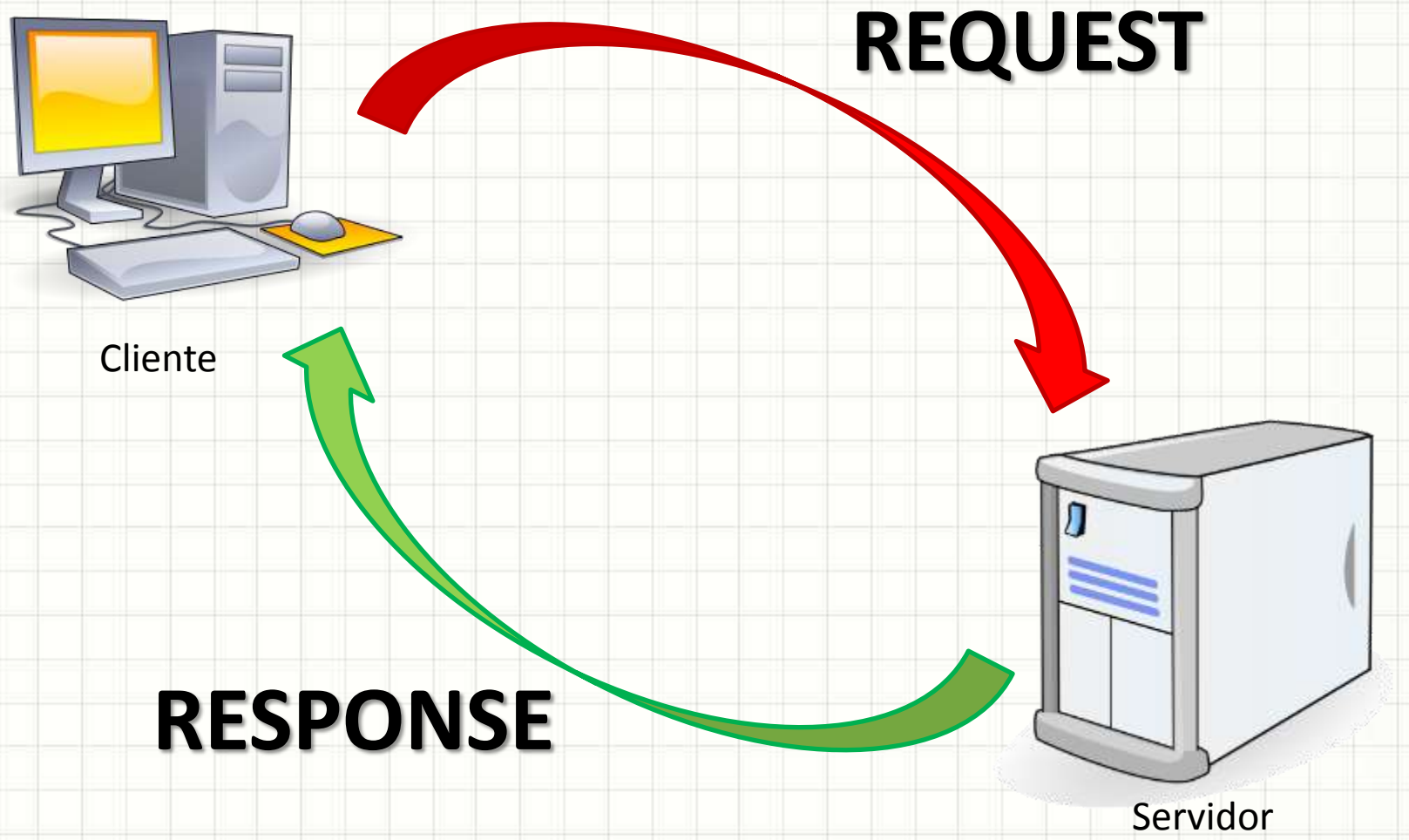


Material	Acesso ao Material
Notas de Aula	http://www.caetano.eng.br/aulas/psw/ (Aula 4)
Apresentação	http://www.caetano.eng.br/aulas/psw/ (Aula 4)
Material Didático	-
Java: Como Programar	(6ª Edição) Páginas 928 a 948

Arquitetura Web e Serviços

Arquitetura Web e Serviços

- Na primeira aula, vimos esse sistema:



Arquitetura Web e Serviços

- Quem é responsável por gerar a **request**?
- O computador **cliente**
  - Navegador
  - Outro programa qualquer
- Quem é responsável por gerar a **response**?
- O computador **servidor**
  - Programa servidor
- O **servidor** presta um **serviço** ao **cliente**

Arquitetura Web e Serviços

- O **cliente: navegador**
- Requisições são geradas quando
  - Digitamos uma nova URL
  - Clicamos em um link
  - Enviamos um formulário
- O **servidor: aplicativo** em computador 24/7
- Ele recebe **request**s e devolve **response**s
  - Apenas quando uma requisição chega ele atua
- É este tipo de aplicativo que iremos desenvolver!

Arquitetura Web e Serviços

- Bem, faz sentido ter um servidor dedicado a um aplicativo, apenas?
- A resposta é: **depende!**
- Depende de quê?
  - Da demanda sobre estes serviços!
- É comum ter **servidores web** e **servidores de banco de dados** dedicados

- Se a demanda de um aplicativo web não é grande, não faz sentido ter toda uma infraestrutura só para ele...

Arquitetura Web e Serviços

- Vários aplicativos web na mesma máquina
  - **Comum**
- No caso do Java, estes pequenos Aplicativos Web são chamados de **servlets**

- Agora... que tal centralizar algumas funcionalidades, como o gerenciamento de conexões?
- Esse é o papel do **contentor Java**

Arquitetura Web e Serviços

- Existem vários contentores Java
  - GlassFish
  - TomCat
  - TomCat + JBoss
  - Dentre outros...

- Neste curso, usaremos o **GlassFish**
  - É o mais completo
  - É o padrão sugerido pela Oracle

O Contentor Java e o Ciclo de Vida

O Contentor Java

- O contentor Java também é conhecido como **servidor de aplicações Java**
- Gerencia a execução de serviços web Java
- Disponibilizar infraestrutura necessária para estes serviços
- Permite a comunicação com outros componentes Java e aplicações

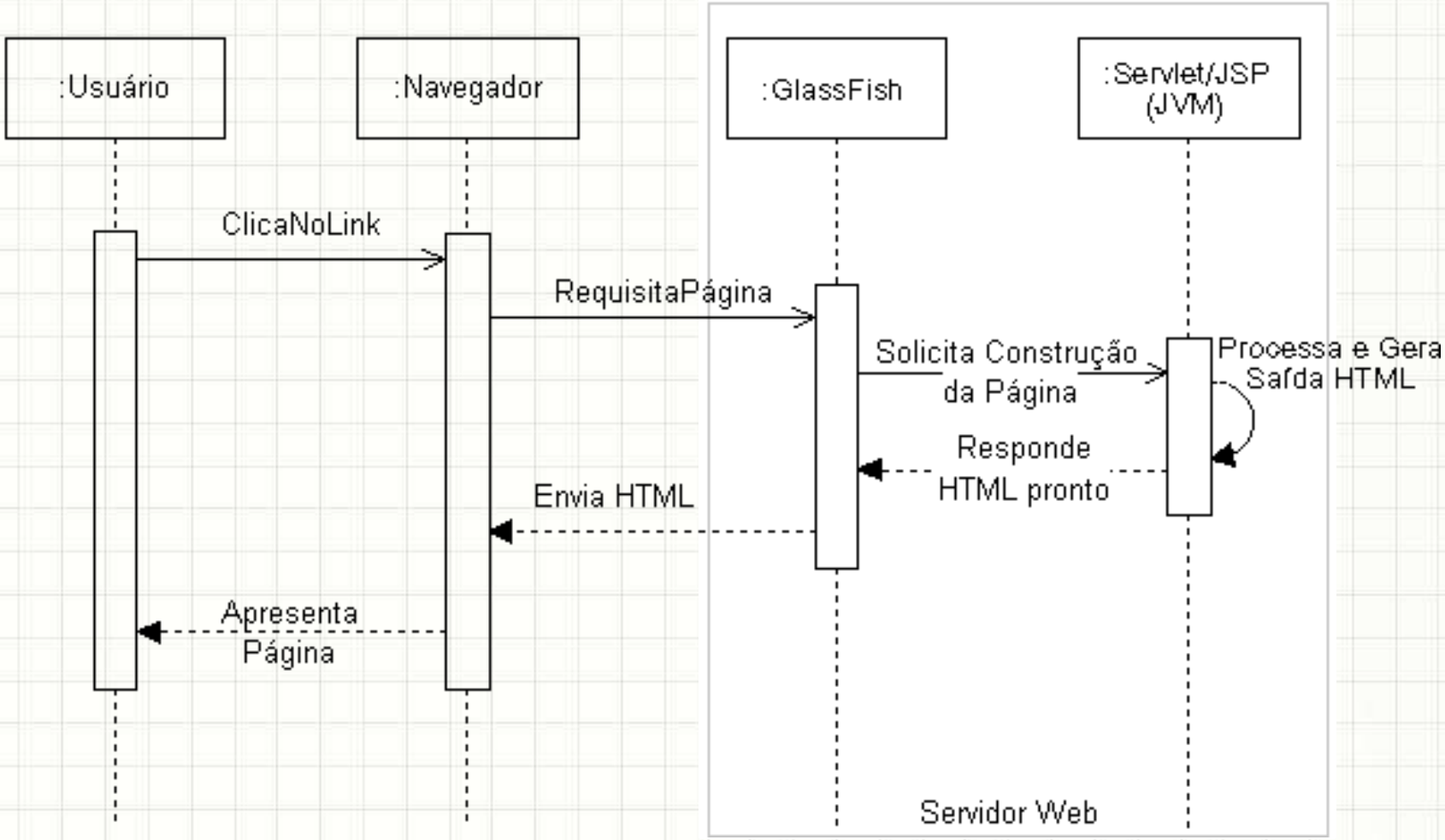
O Contentor Java

- Servidor de Aplicações - Funcionamento básico:
  1. Aguarda pelas requests
  2. Repassa a request para o **servlet** correto
  3. Recebe a response do **servlet**
  4. Devolve a response para o cliente que enviou a request

- Ou seja, ele é responsável por gerenciar as conexões (dentre outras coisas)

O Contentor Java

- Resumidamente...



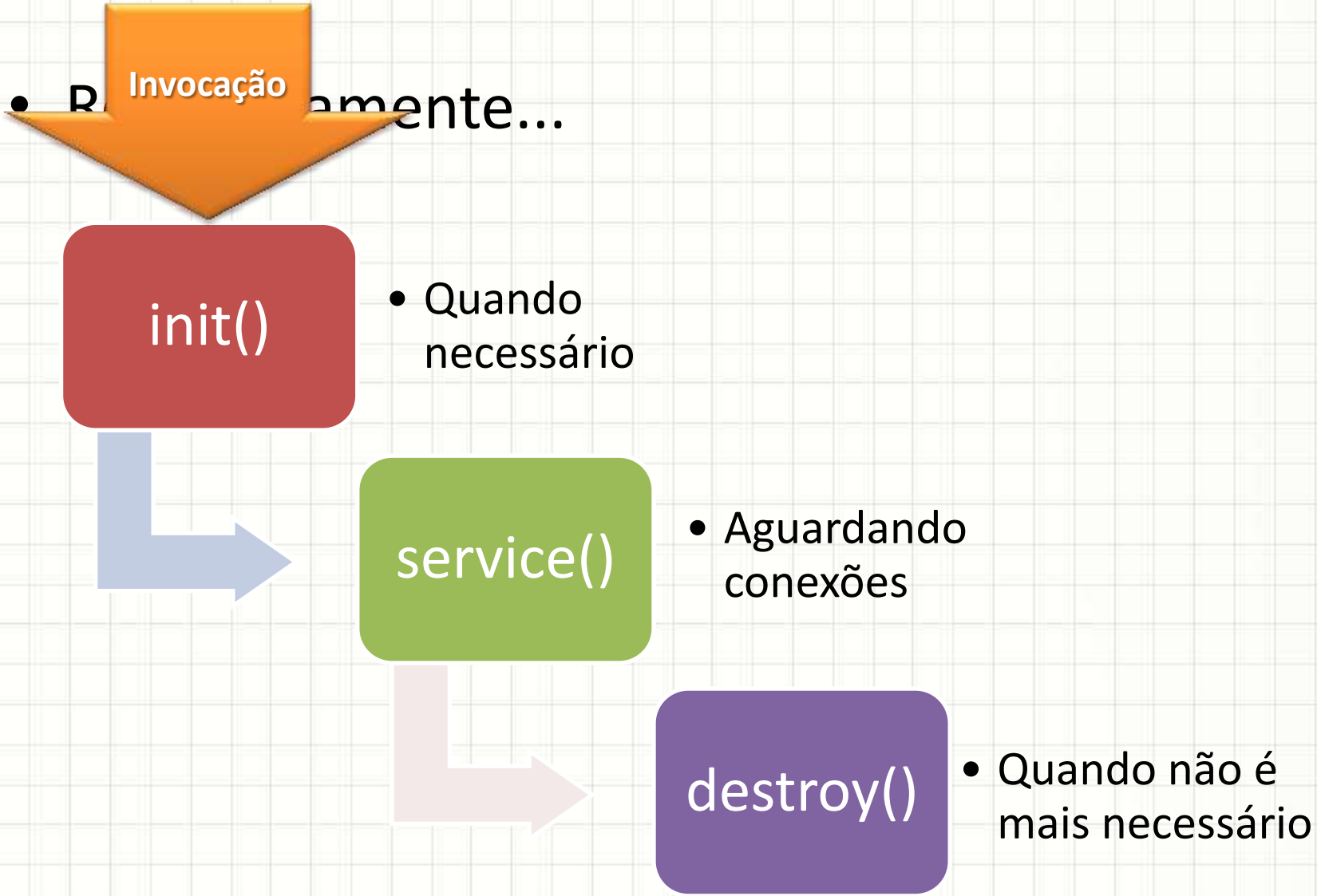
O Contentor Java

- Mas o servidor de aplicativos não faz apenas isso... Dentre outras coisas, ele também cuida do **ciclo de vida do servlet**

1. Quando necessário, inicia o Servlet
2. Coloca este Servlet ativo
3. Finaliza o Servlet quando não é mais necessário

Ciclo de Vida do Servlet

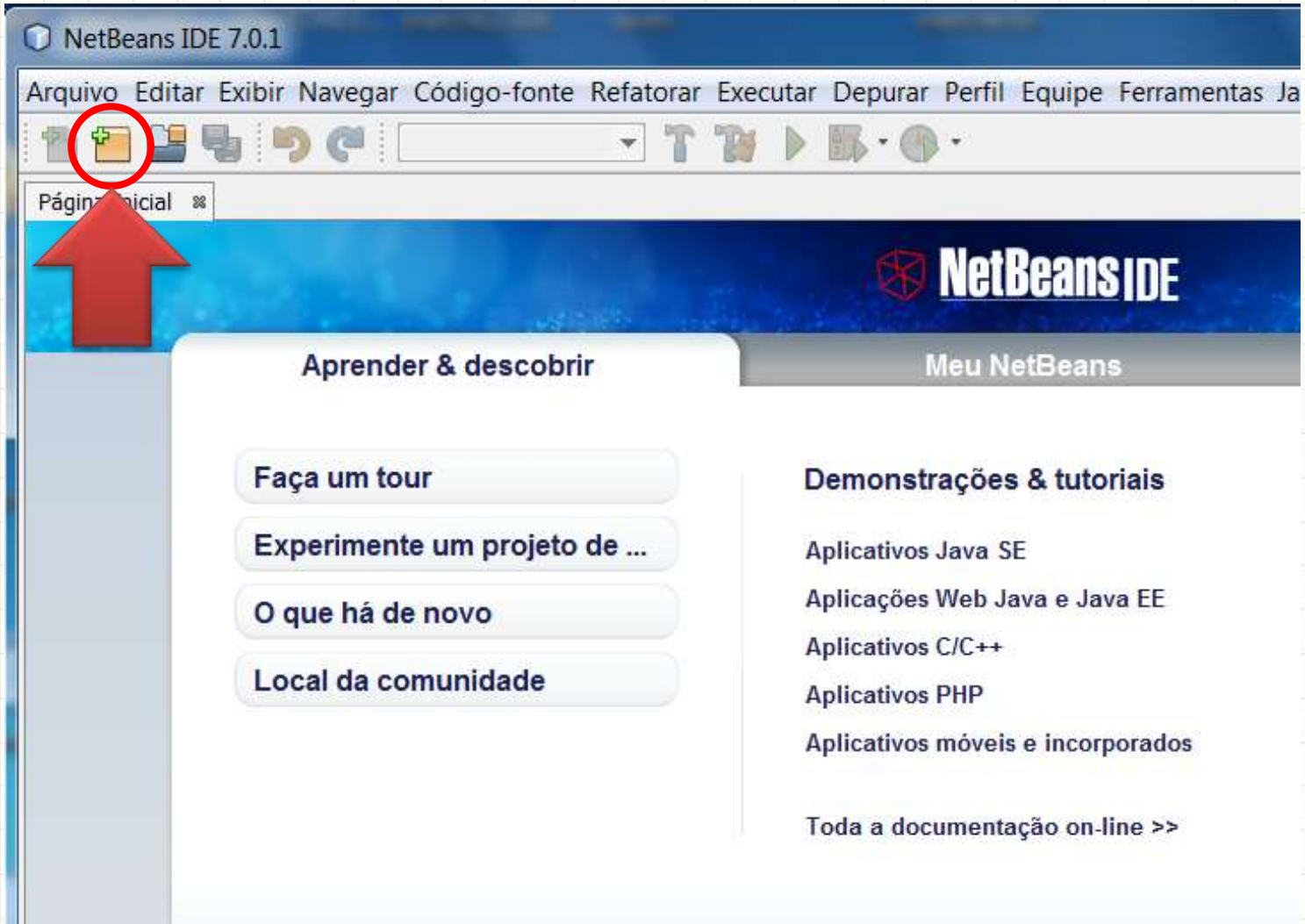
- R[illegible]amente...



Servlets no NetBeans

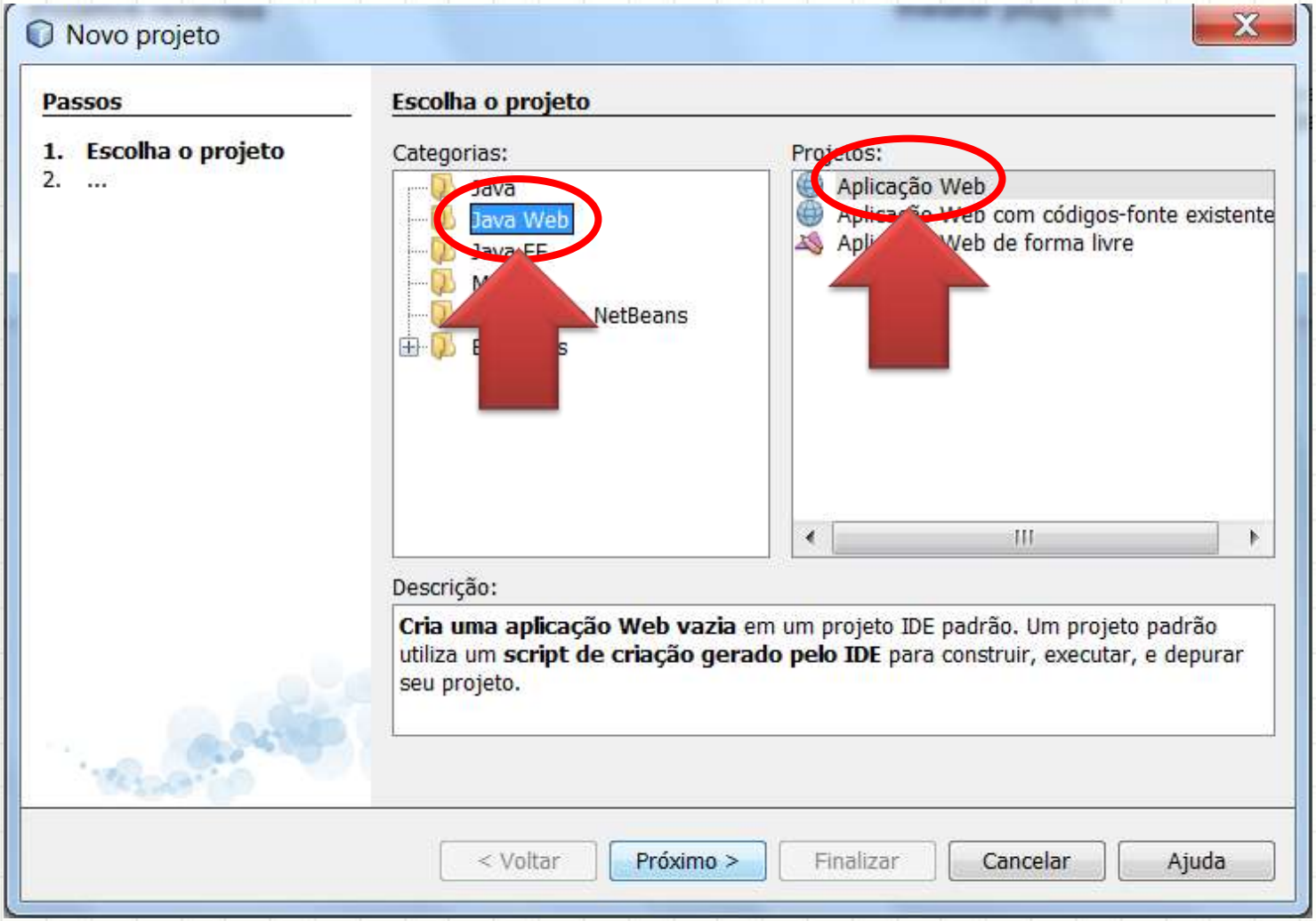
Criando um Servlet

- Iniciaremos criando um projeto



Criando um Servlet

- Escolha o tipo: **Java Web** e **Aplicação Web**

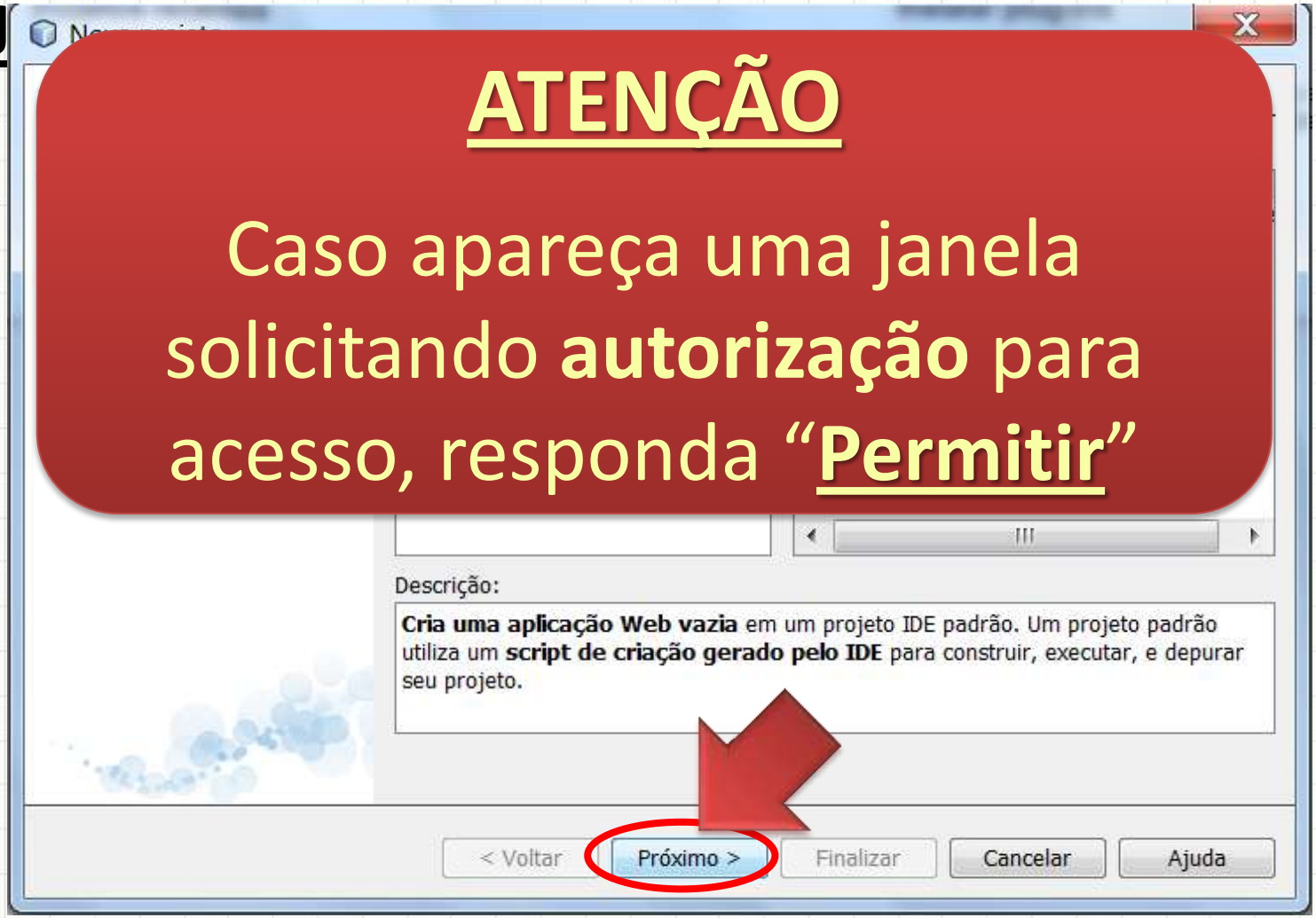


Criando um Servlet

- Escolha o tipo: **Java Web** e **Aplicação Web**

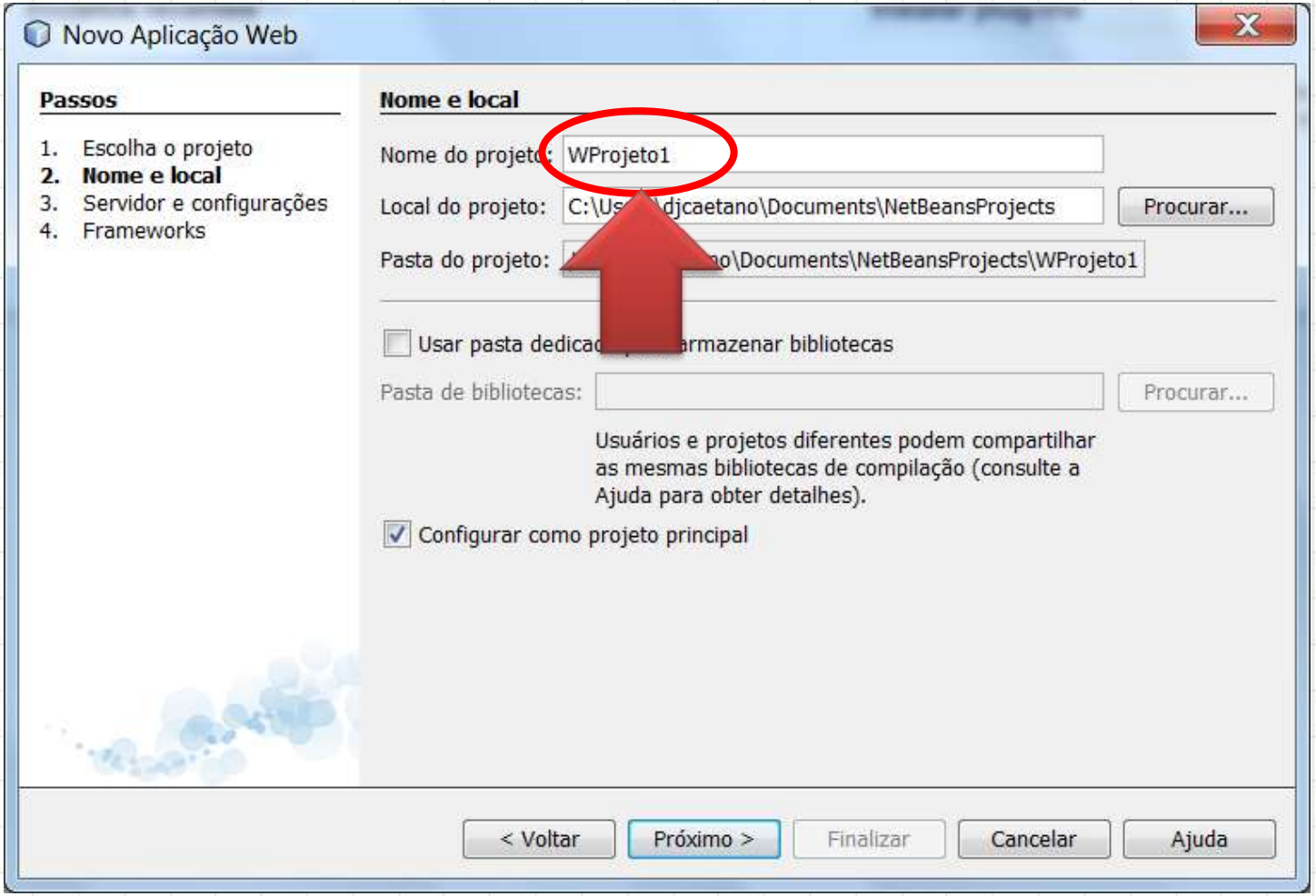
ATENÇÃO

Caso apareça uma janela solicitando **autorização** para acesso, responda "**Permitir**"



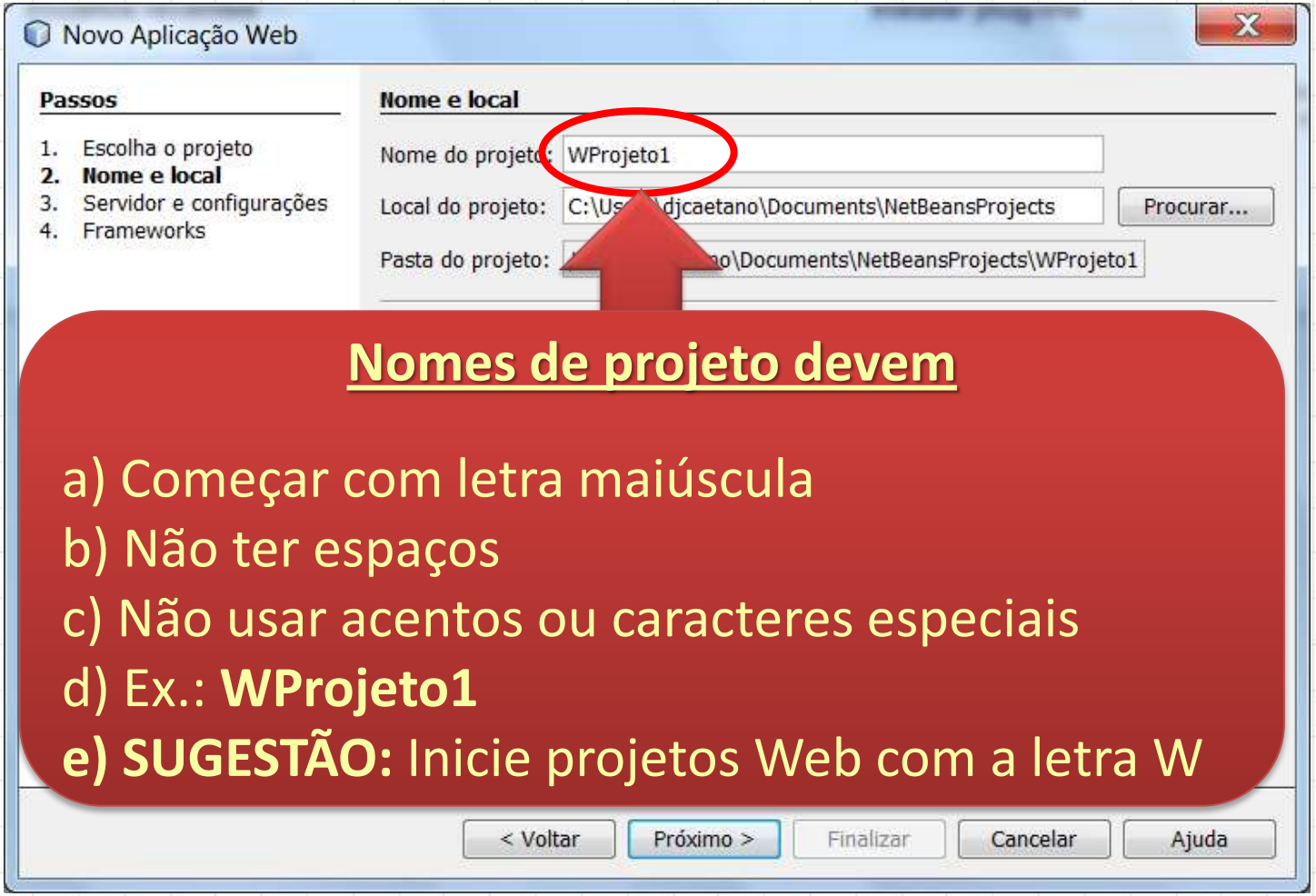
Criando um Servlet

- Dê um nome ao seu projeto: ex.: **WProjeto1**



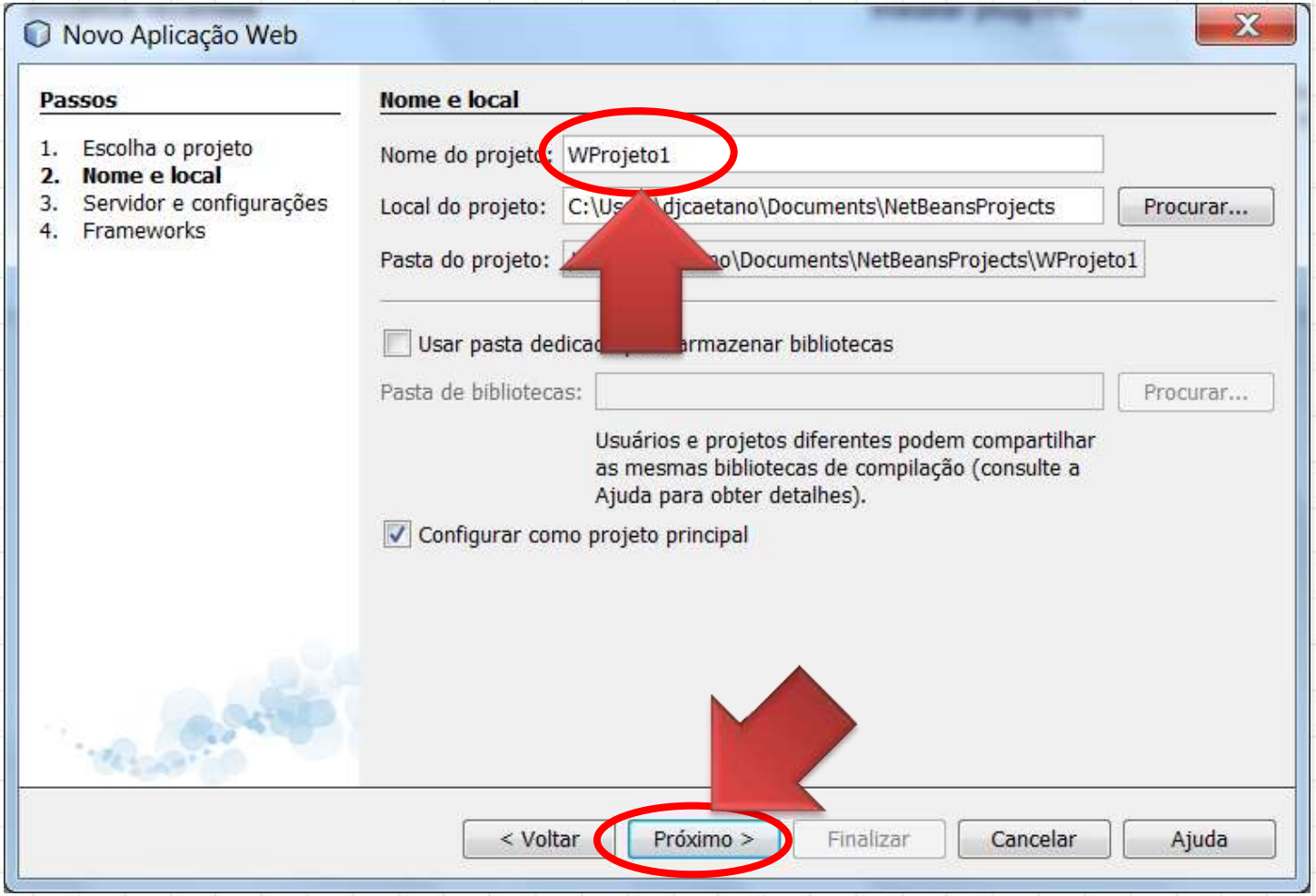
Criando um Servlet

- Dê um nome ao seu projeto: ex.: **WProjeto1**



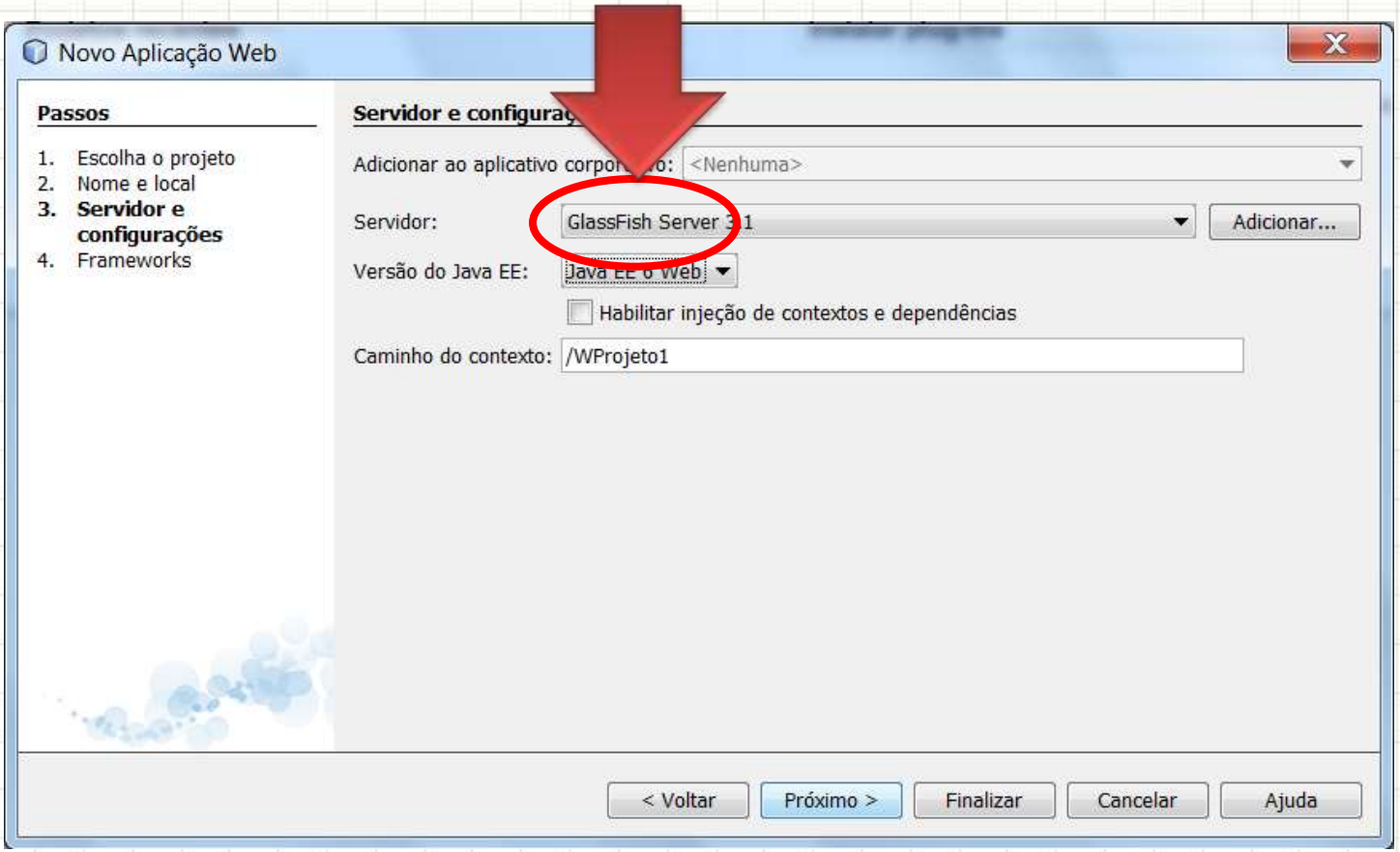
Criando um Servlet

- Dê um nome ao seu projeto: ex.: **WProjeto1**



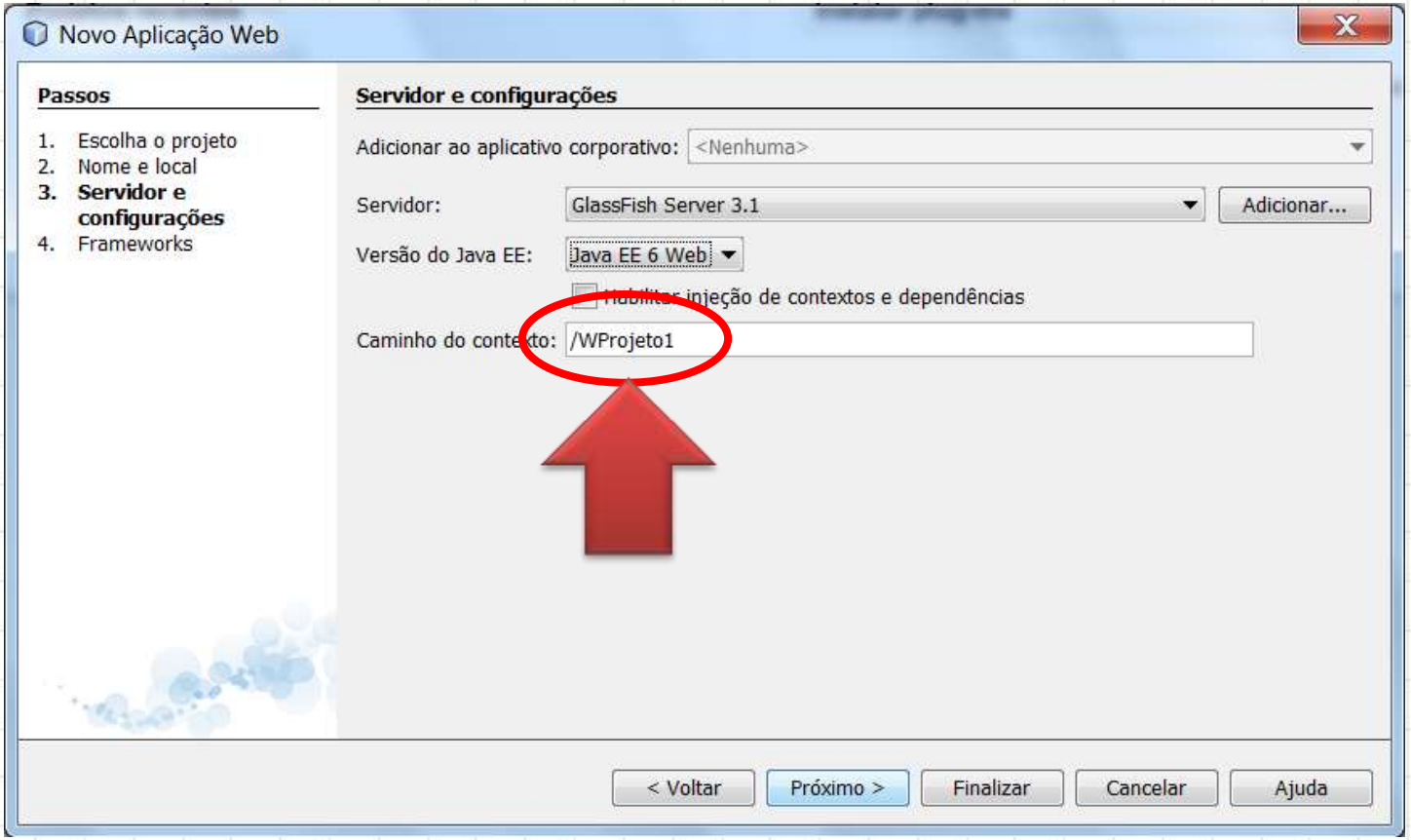
Criando um Servlet

- Selecione o contentor: **GlassFish**



Criando um Servlet

- Este é o endereço da aplicação: **/WProjeto1**



Criando um Servlet

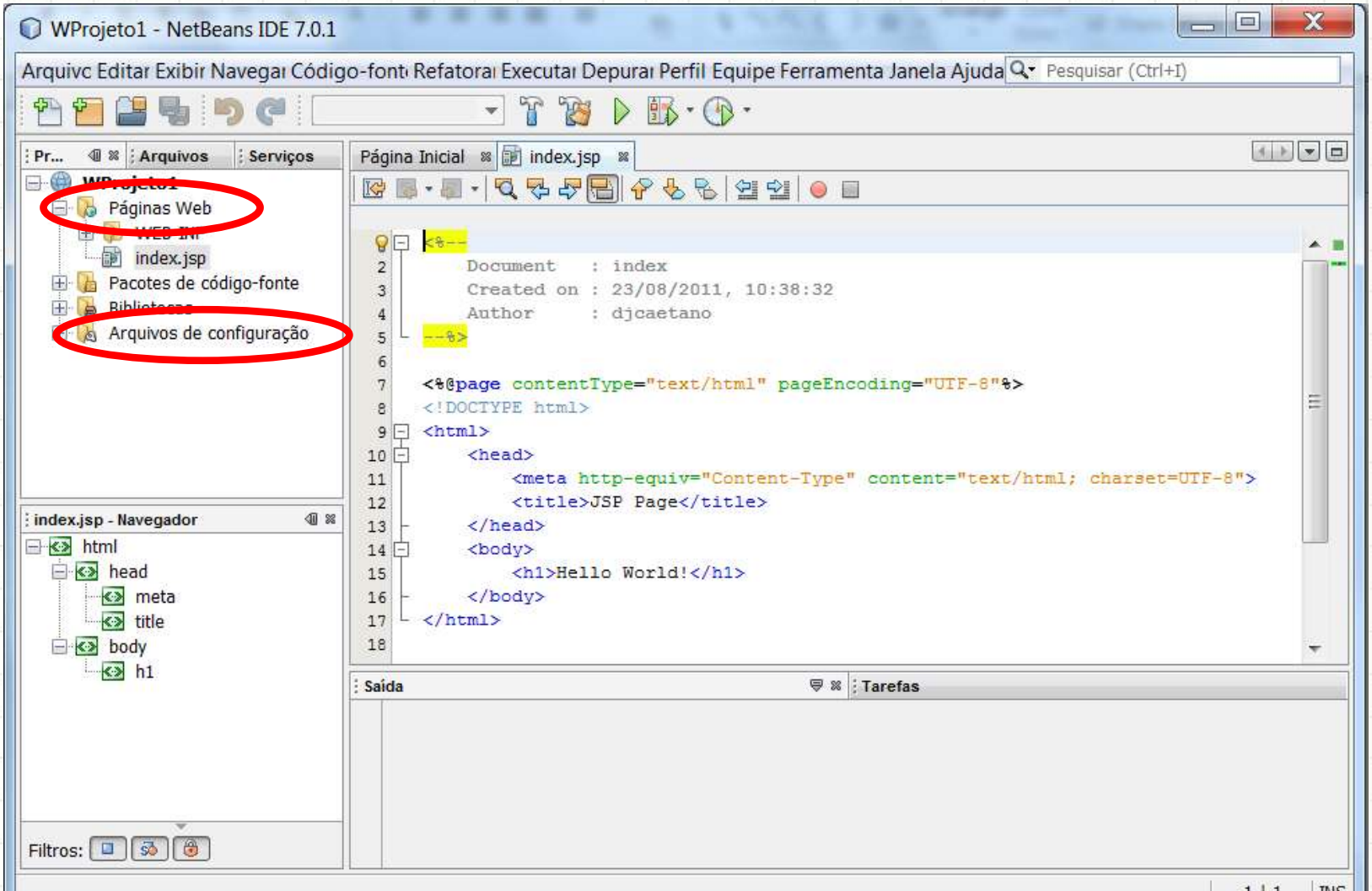
- Clique em **Finalizar**

ATENÇÃO

Caso apareça uma janela solicitando **autorização** para acesso, responda "**Permitir**"

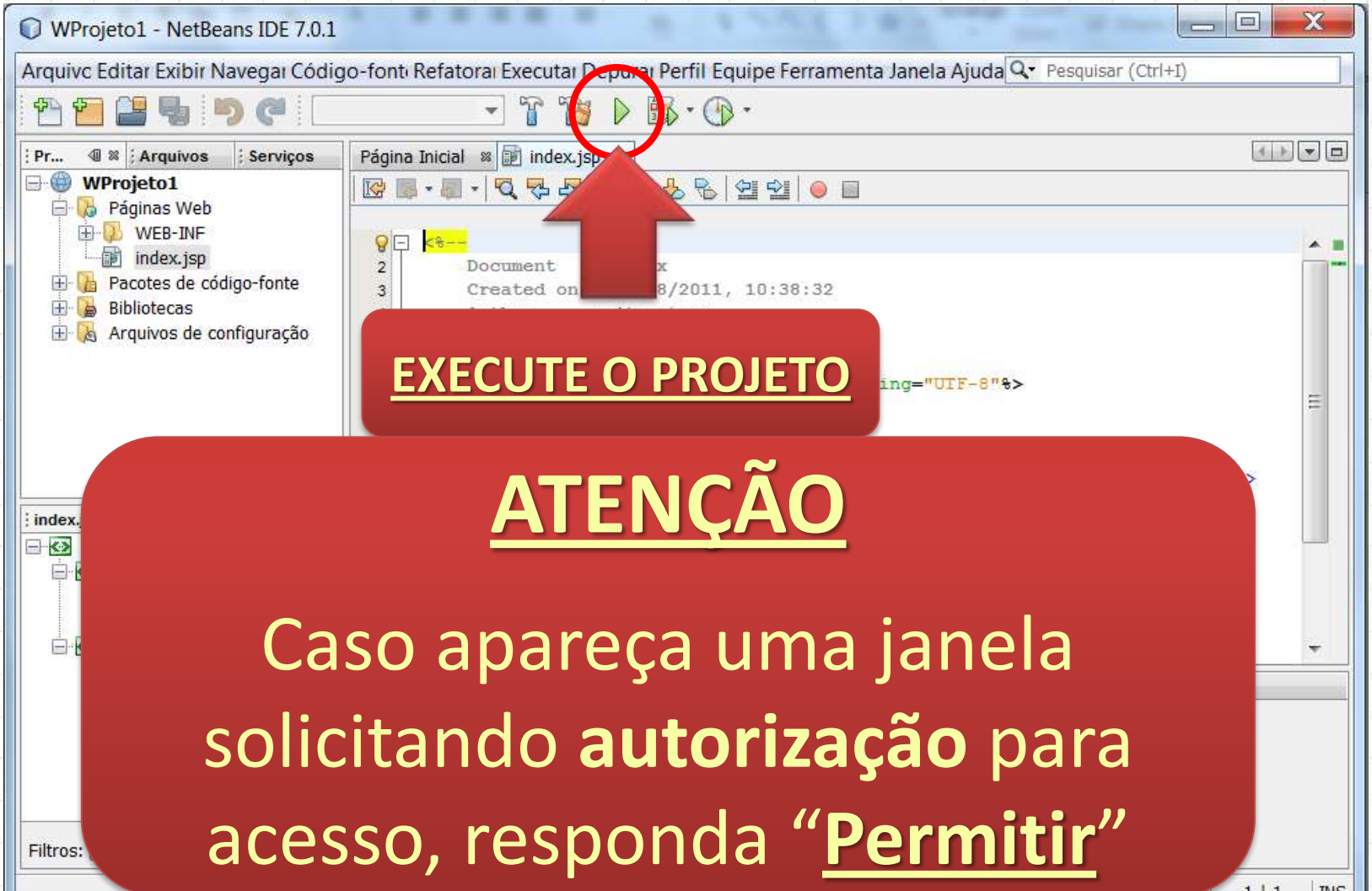
Criando um Servlet

- Esta tela vai aparecer... Há campos novos!



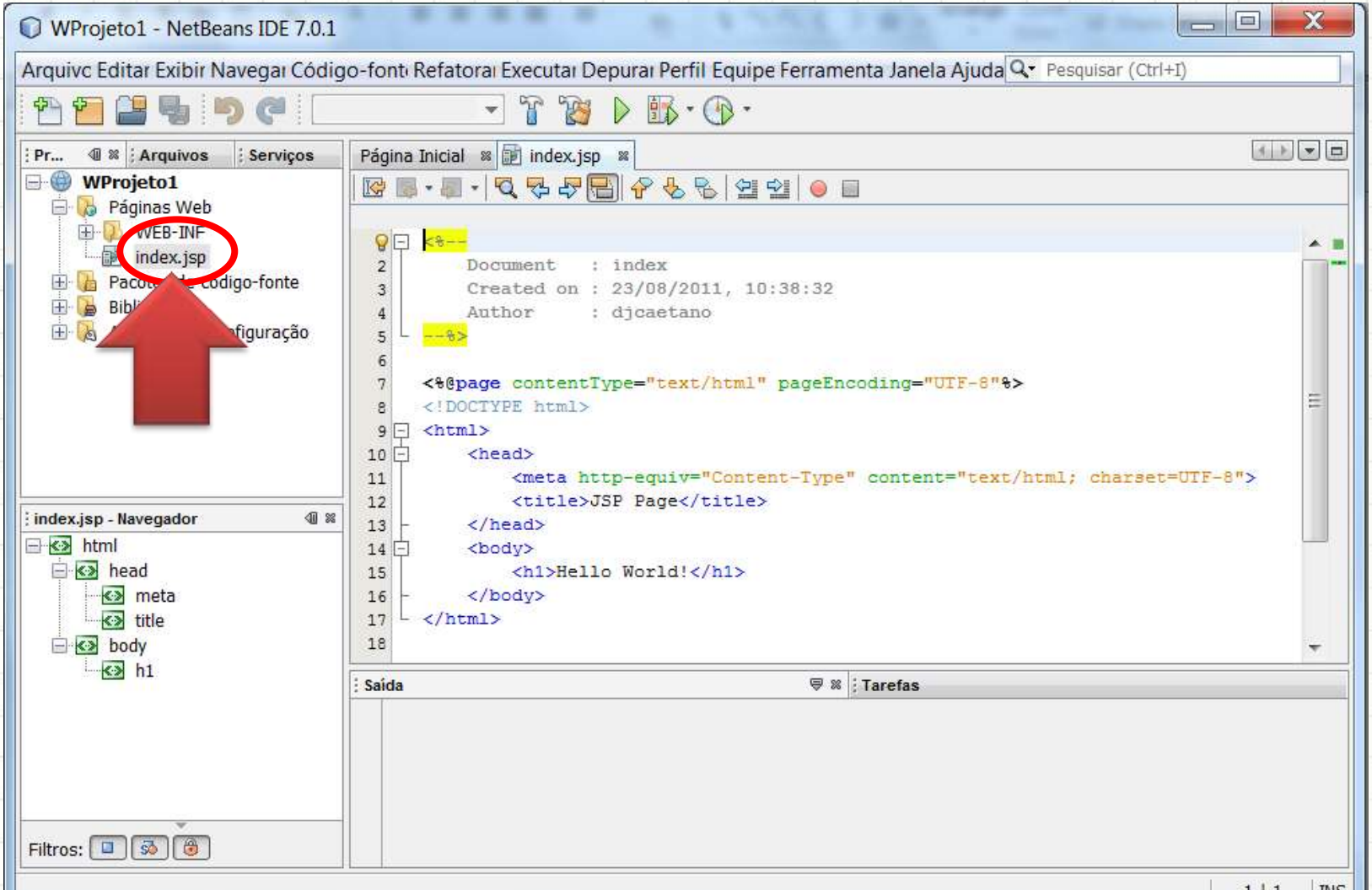
Criando um Servlet

- Execute o projeto e veja o que ocorre!



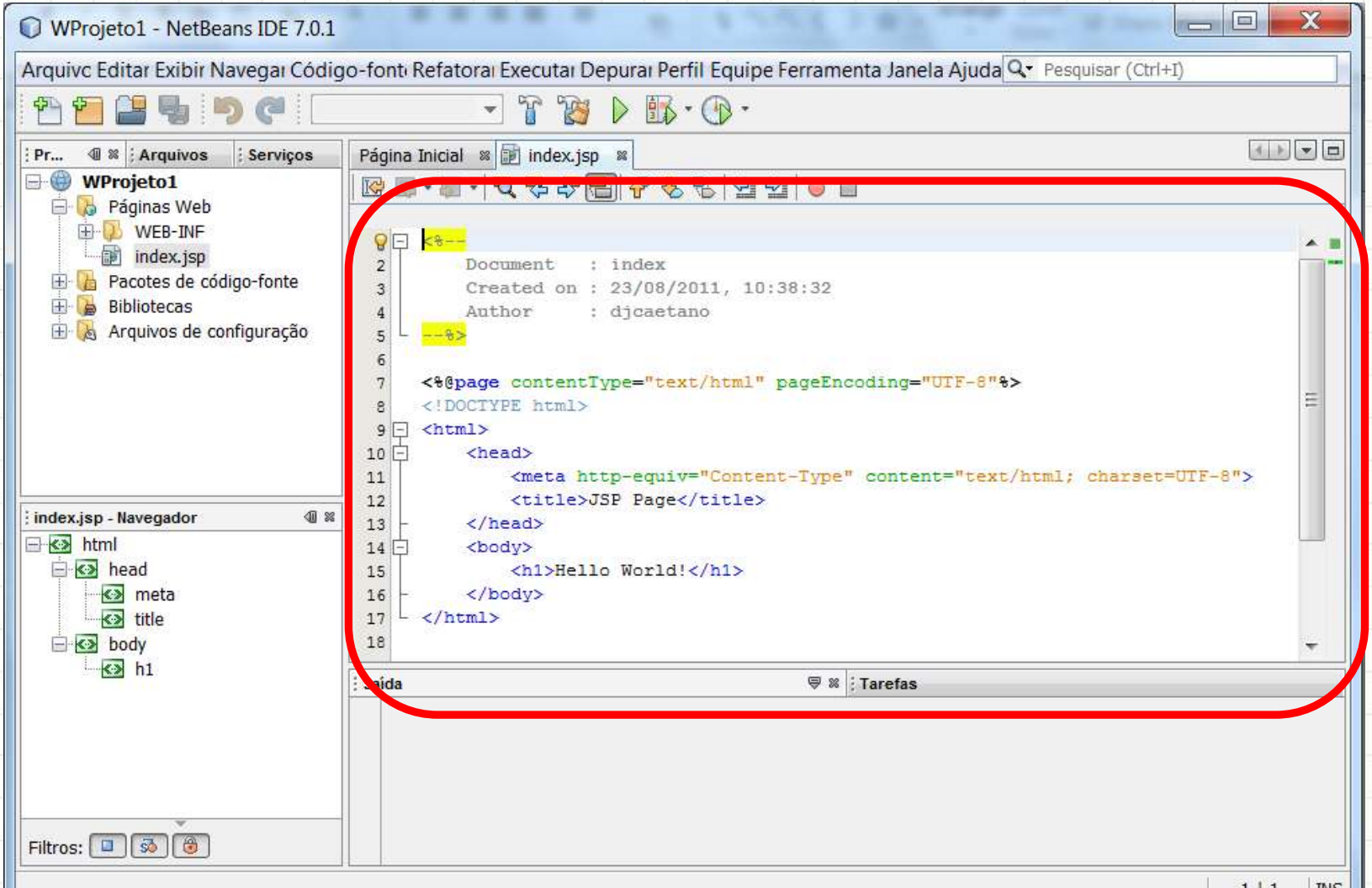
Criando um Servlet

- Por padrão, o **index.jsp** é executado



Criando um Servlet

- Vamos construir um formulário...



Criando um Servlet

- Vamos construir um formulário...

WProjeto1 - NetBeans IDE 7.0.1

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Calculadora de Índice de Massa Corporal</title>
    </head>
    <body>
        <h1>Digite seus dados:</h1>
        <form method="post" action="Imc">
            <p>Peso: <input type="text" name="peso" /></p>
            <p>Altura: <input type="text" name="altura" /></p>
            <p><input type="submit" value="Enviar!" /></p>
        </form>
    </body>
</html>
```

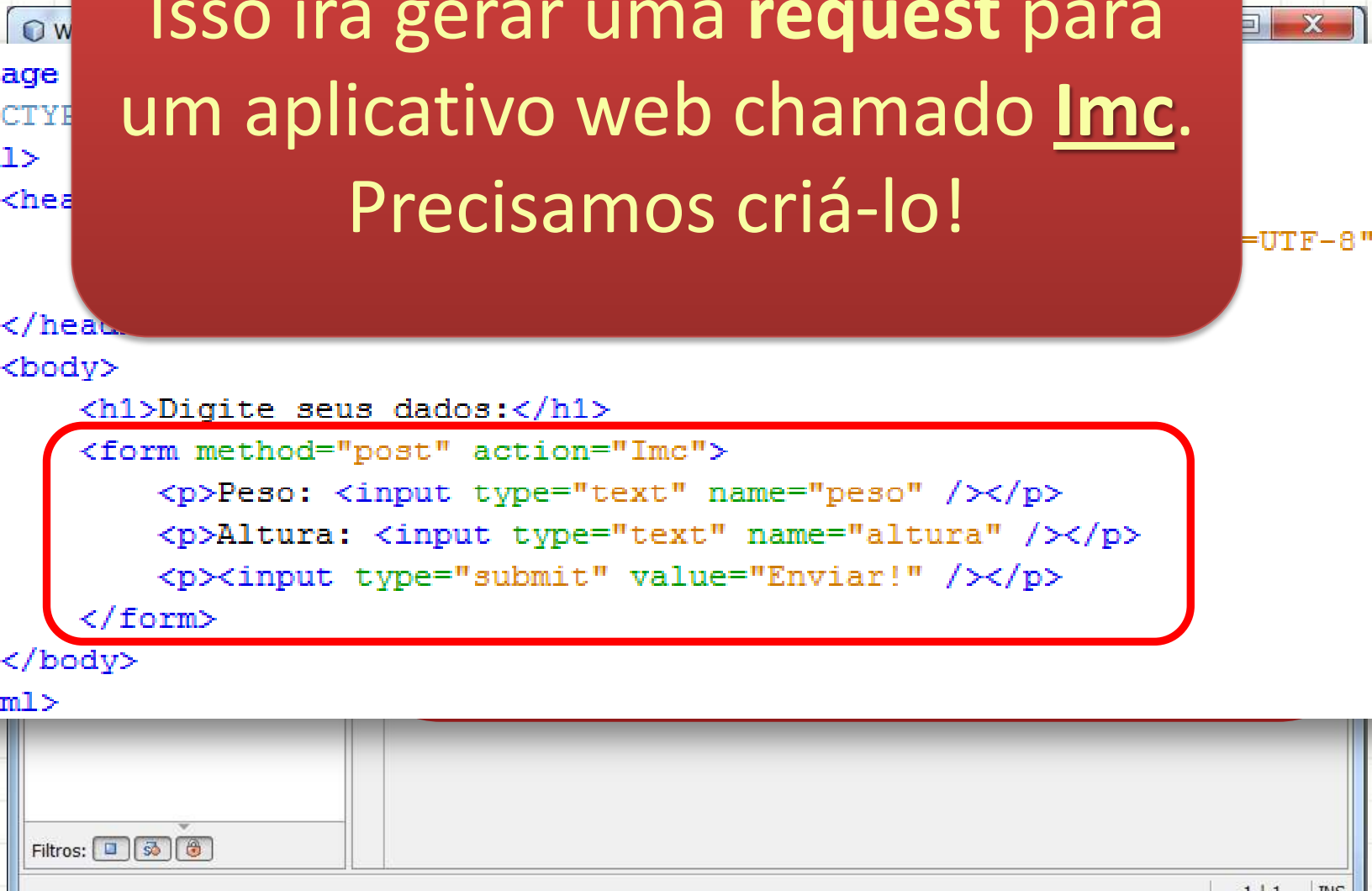
Filtros:

Criando um Servlet

- Va

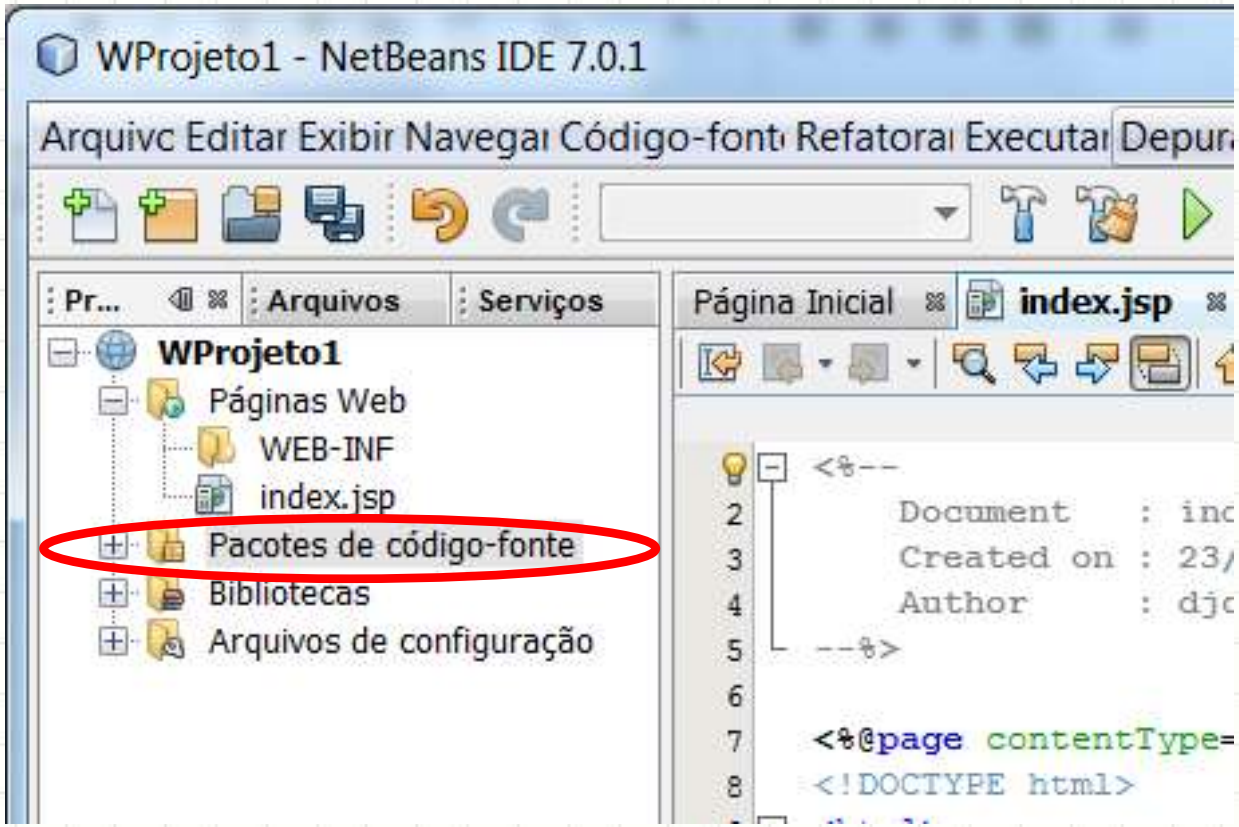
Isso irá gerar uma **request** para um aplicativo web chamado **Imc**. Precisamos criá-lo!

```
<%@page
<!DOCTYP
<html>
    <hea
                                                    =UTF-8">
    </head
    <body>
        <h1>Digite seus dados:</h1>
        <form method="post" action="Imc">
            <p>Peso: <input type="text" name="peso" /></p>
            <p>Altura: <input type="text" name="altura" /></p>
            <p><input type="submit" value="Enviar!" /></p>
        </form>
    </body>
</html>
```



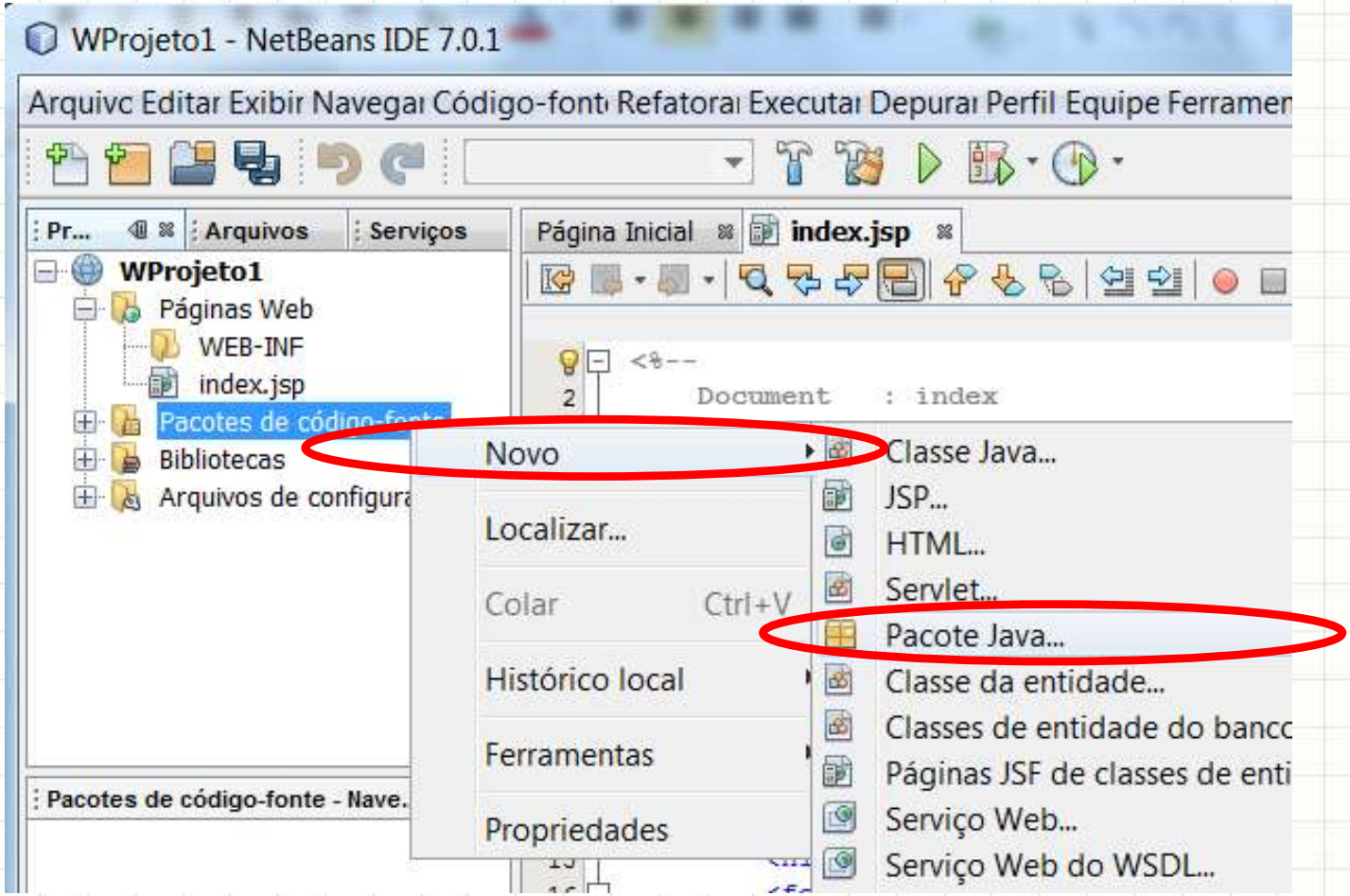
Criando um Servlet

- Clique com o botão direito em “Pacotes de Código Fonte”



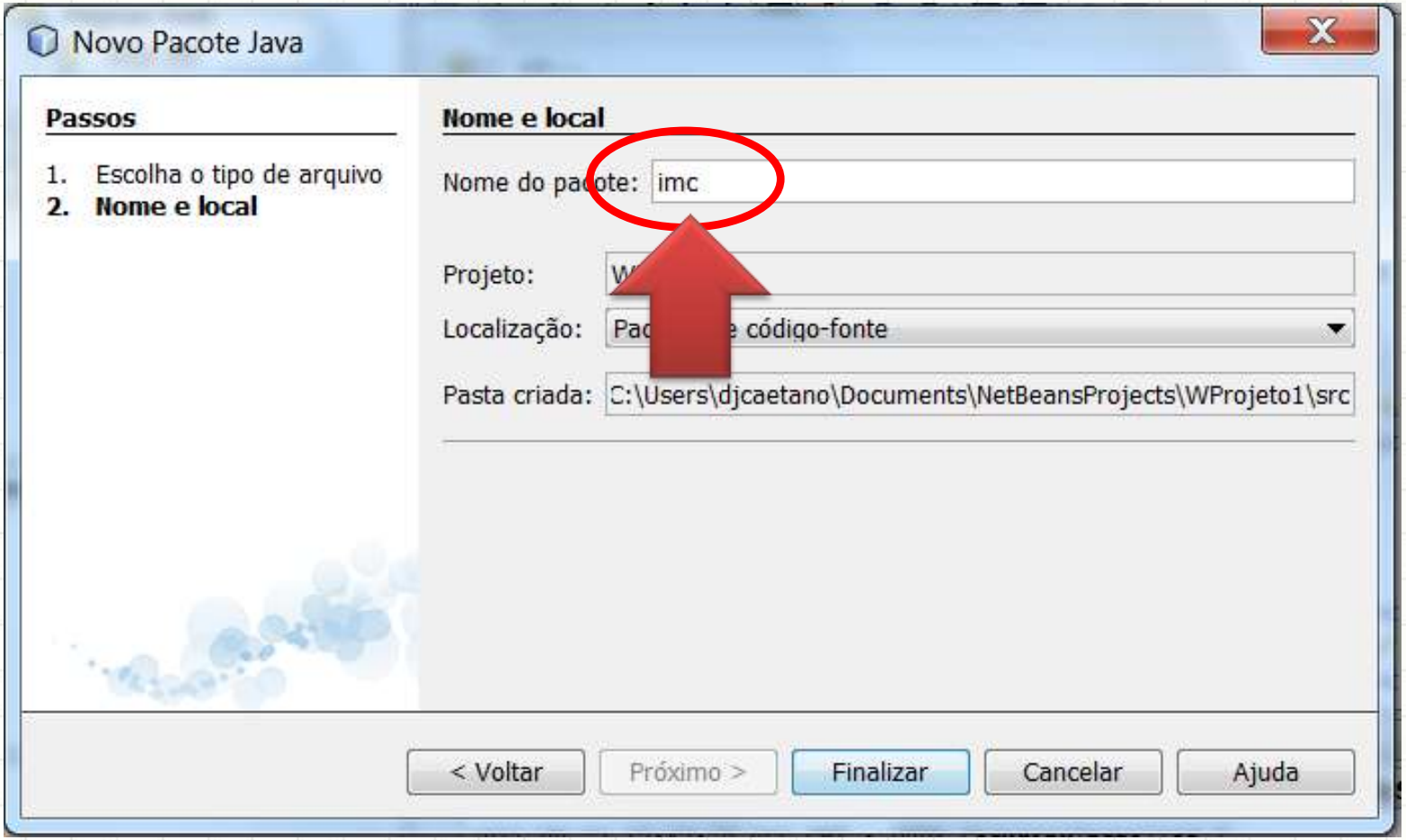
Criando um Servlet

- E selecione **Novo** > **Pacote Java...**



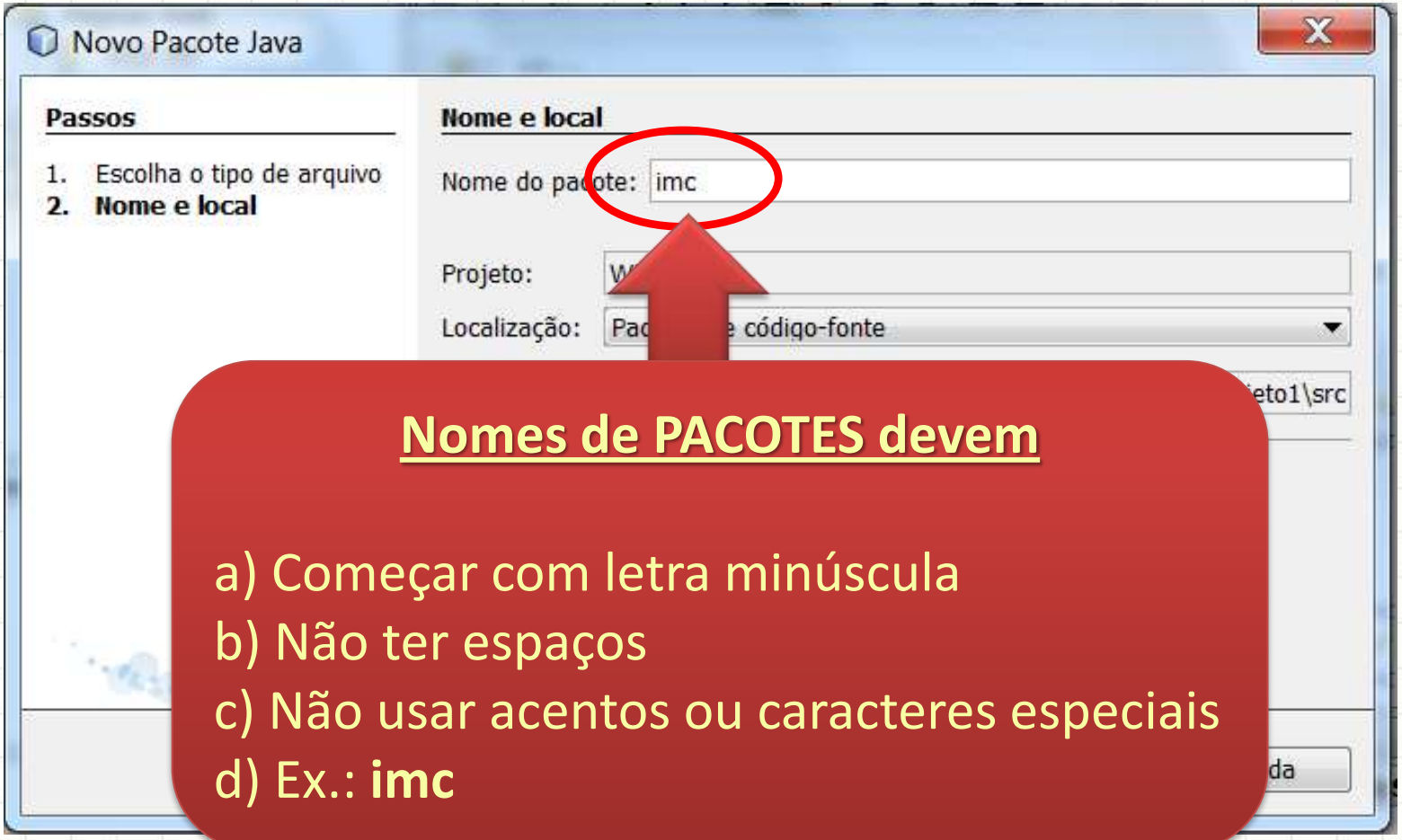
Criando um Servlet

- Agora dê um nome ao pacote: **imc**



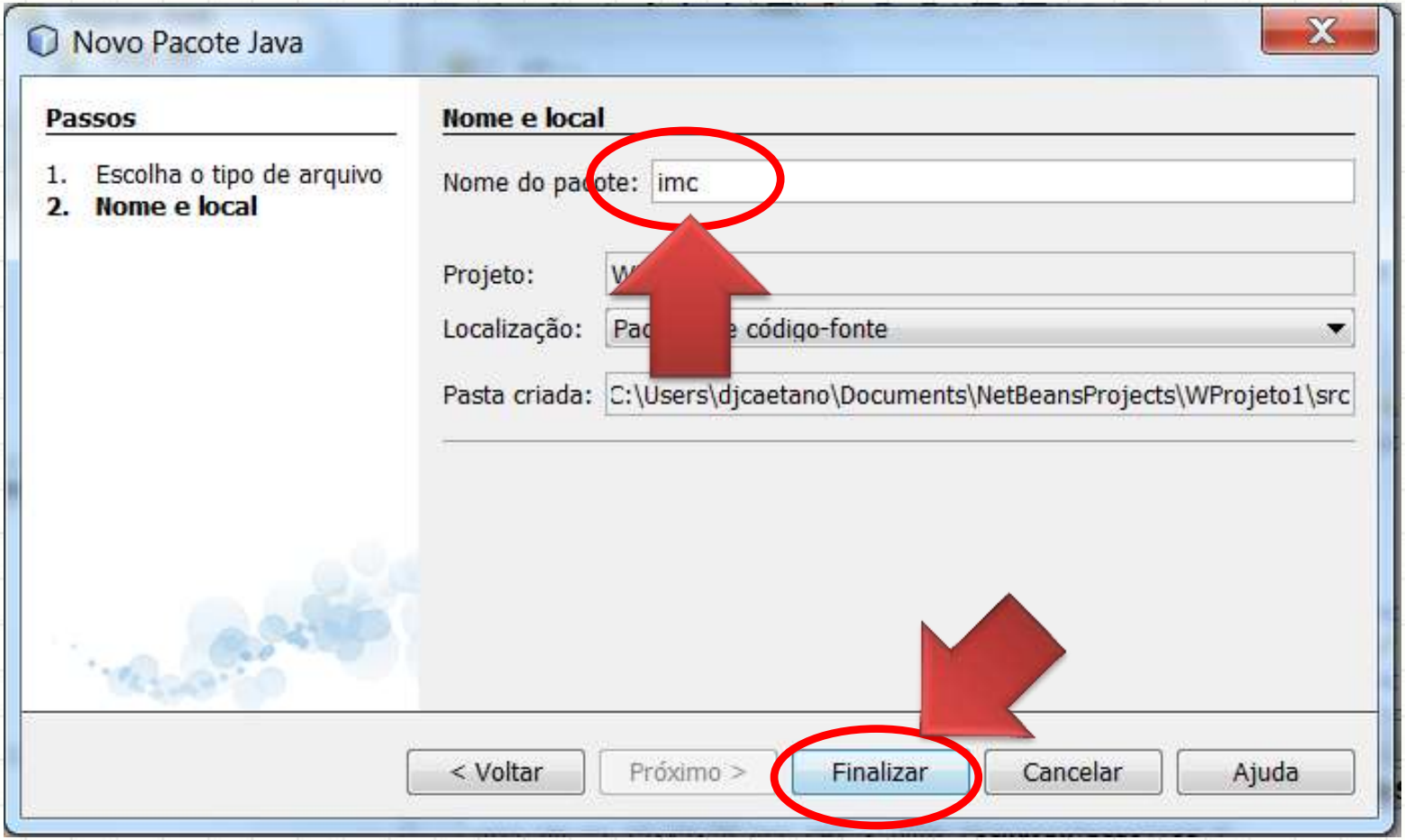
Criando um Servlet

- Agora dê um nome ao pacote: **imc**



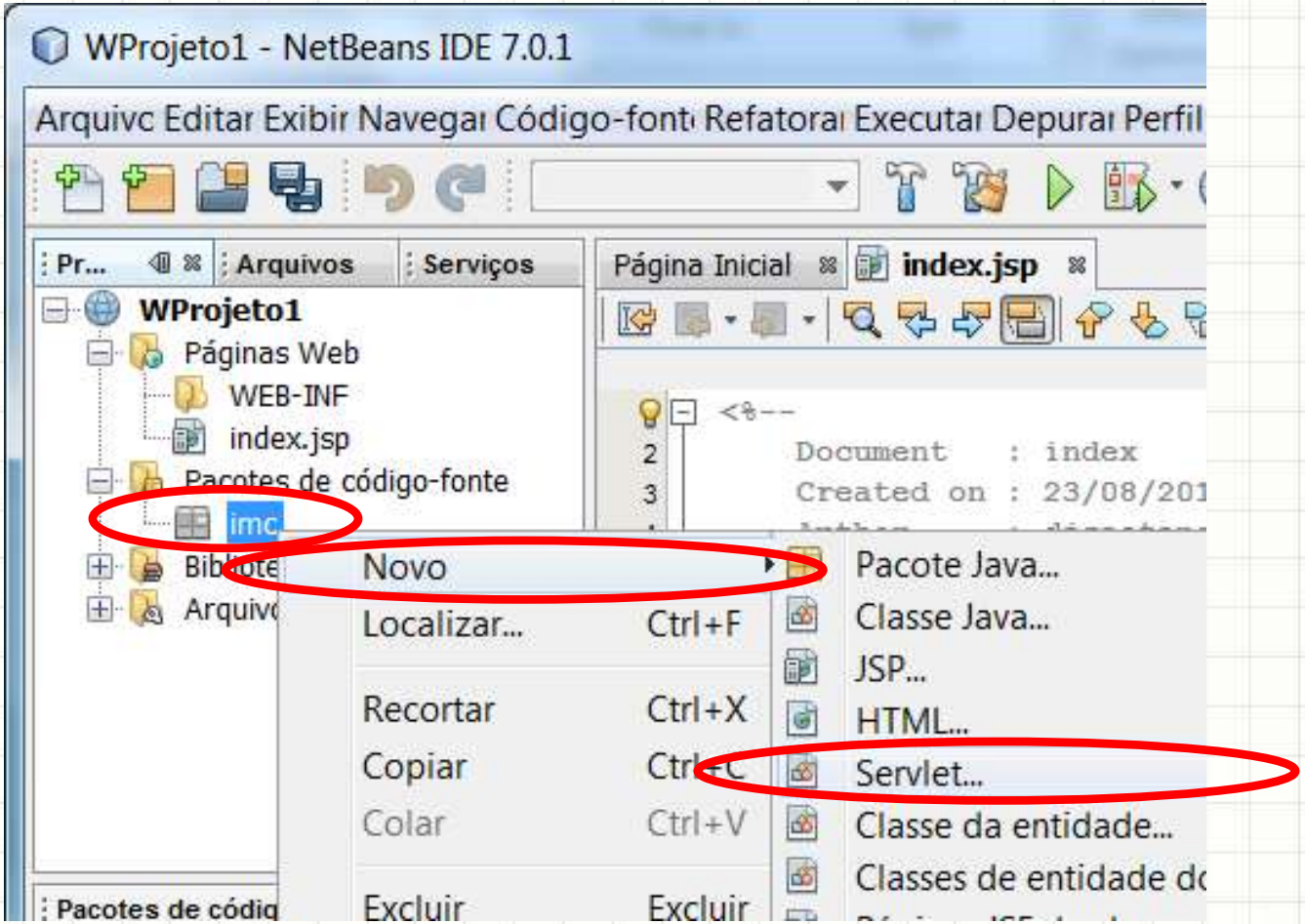
Criando um Servlet

- E clique em **Finalizar**



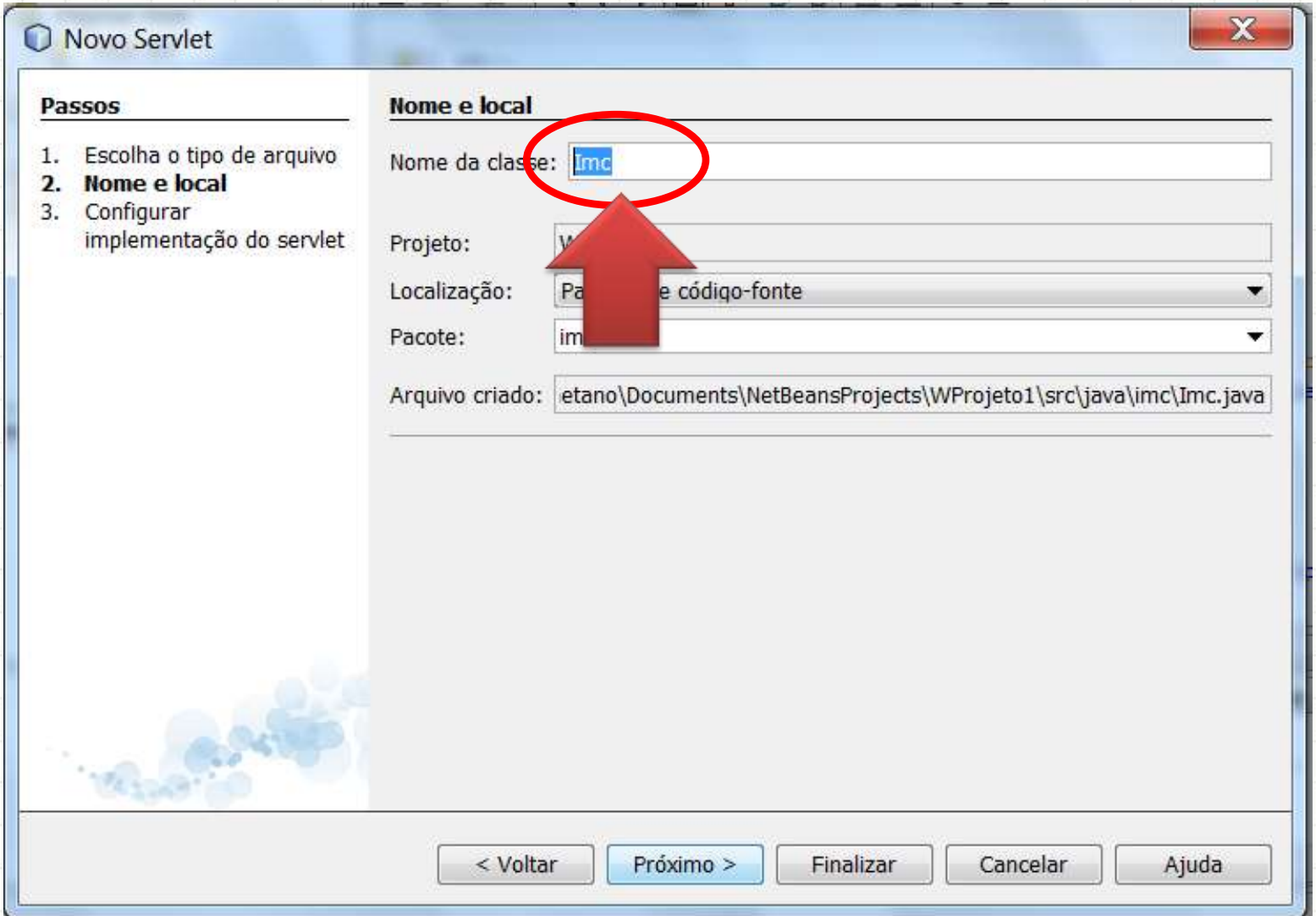
Criando um Servlet

- Agora clique com o botão direito no pacote **imc** e selecione **Novo** > **Servlet...**



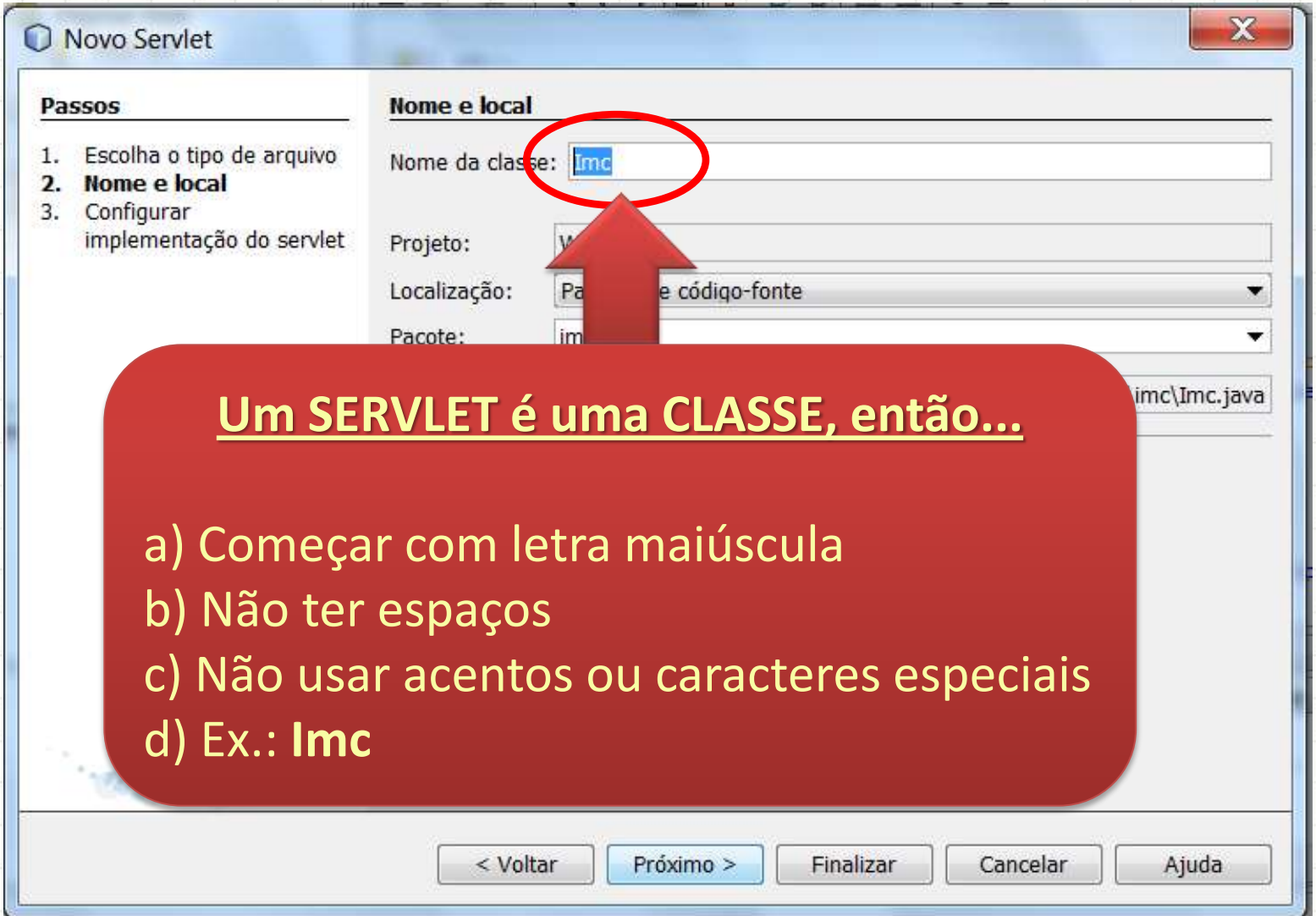
Criando um Servlet

- Agora dê um nome ao servlet: **Imc**



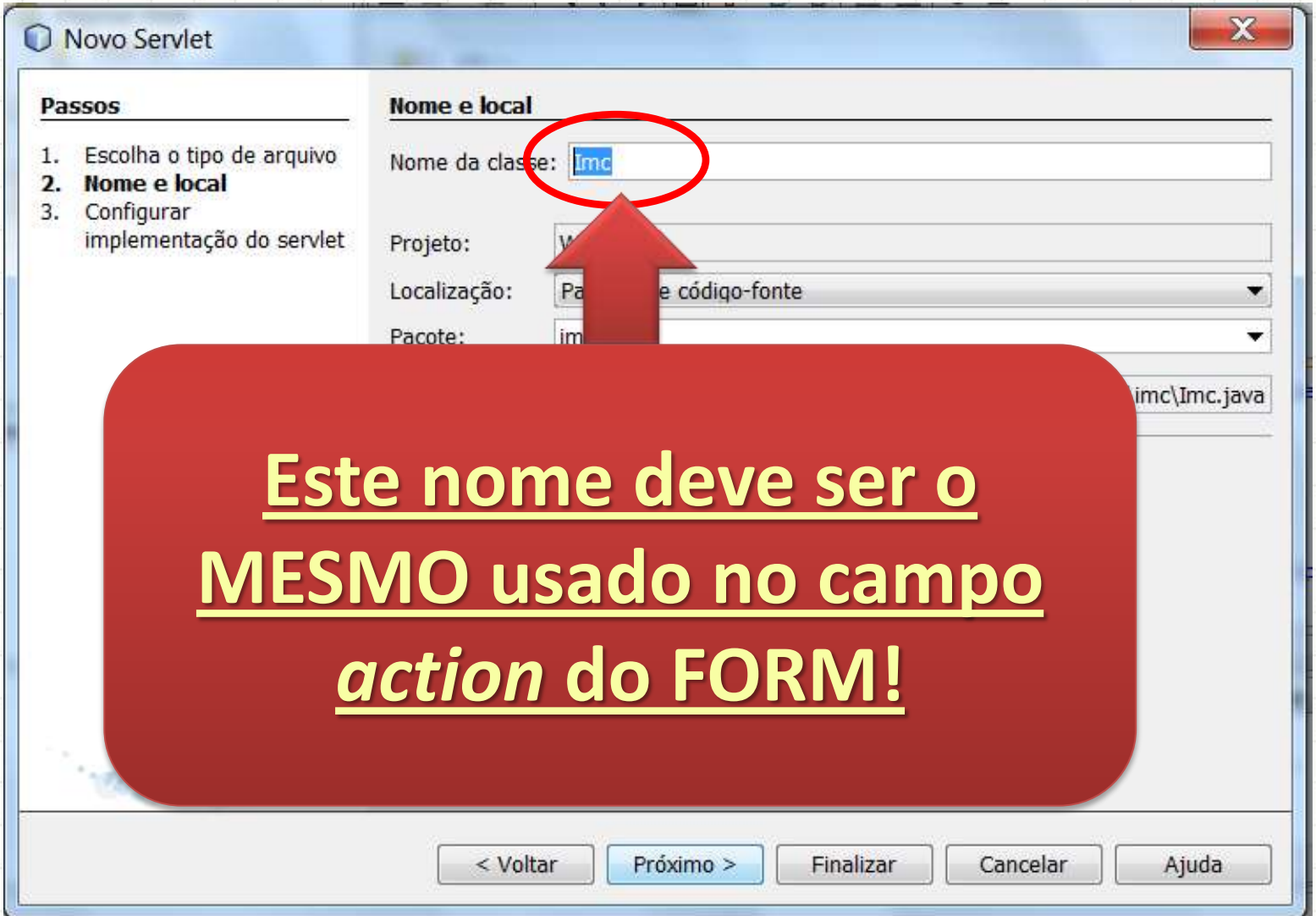
Criando um Servlet

- Agora dê um nome ao servlet: **Imc**



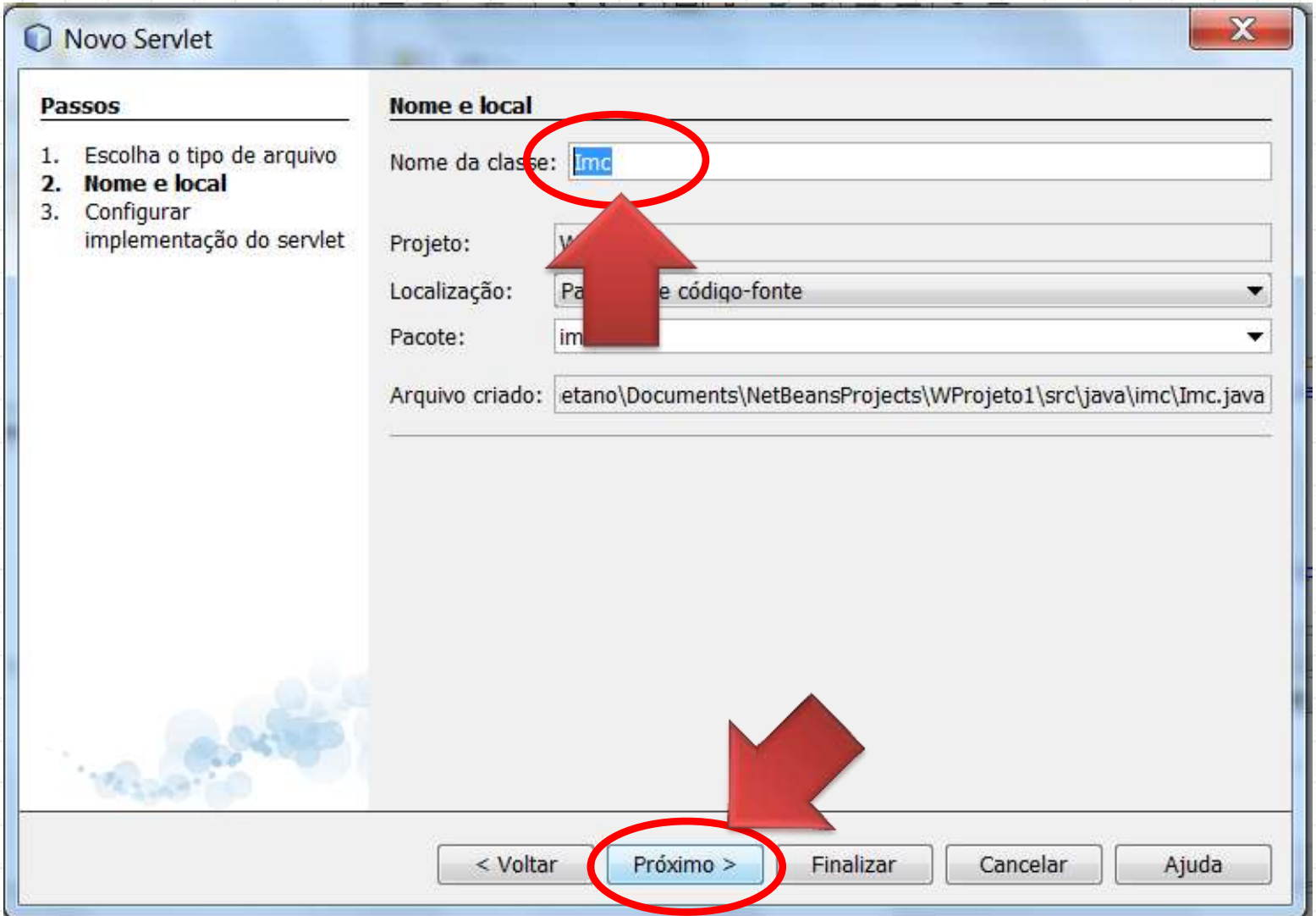
Criando um Servlet

- Agora dê um nome ao servlet: **Imc**



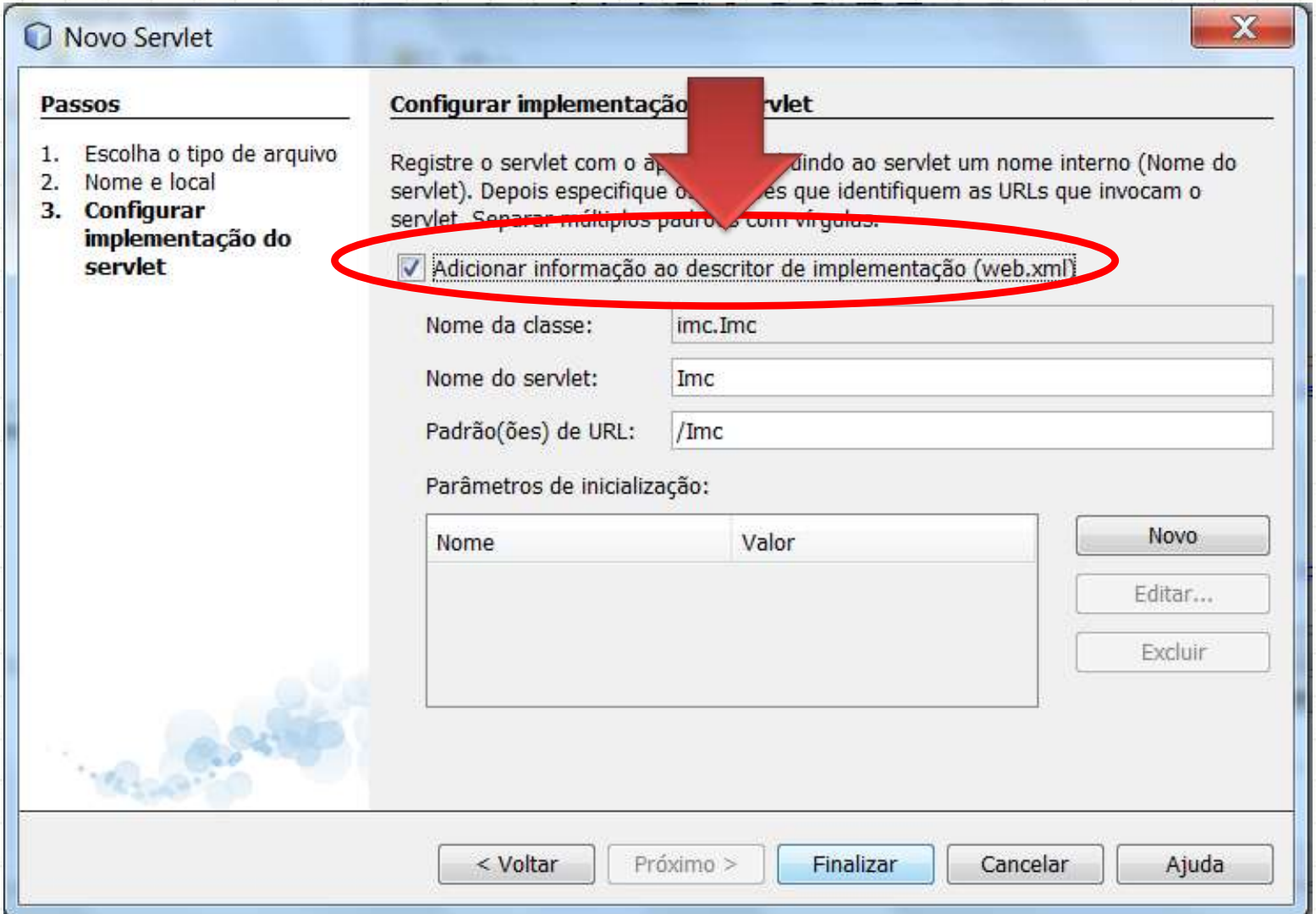
Criando um Servlet

- E clique em **Próximo**



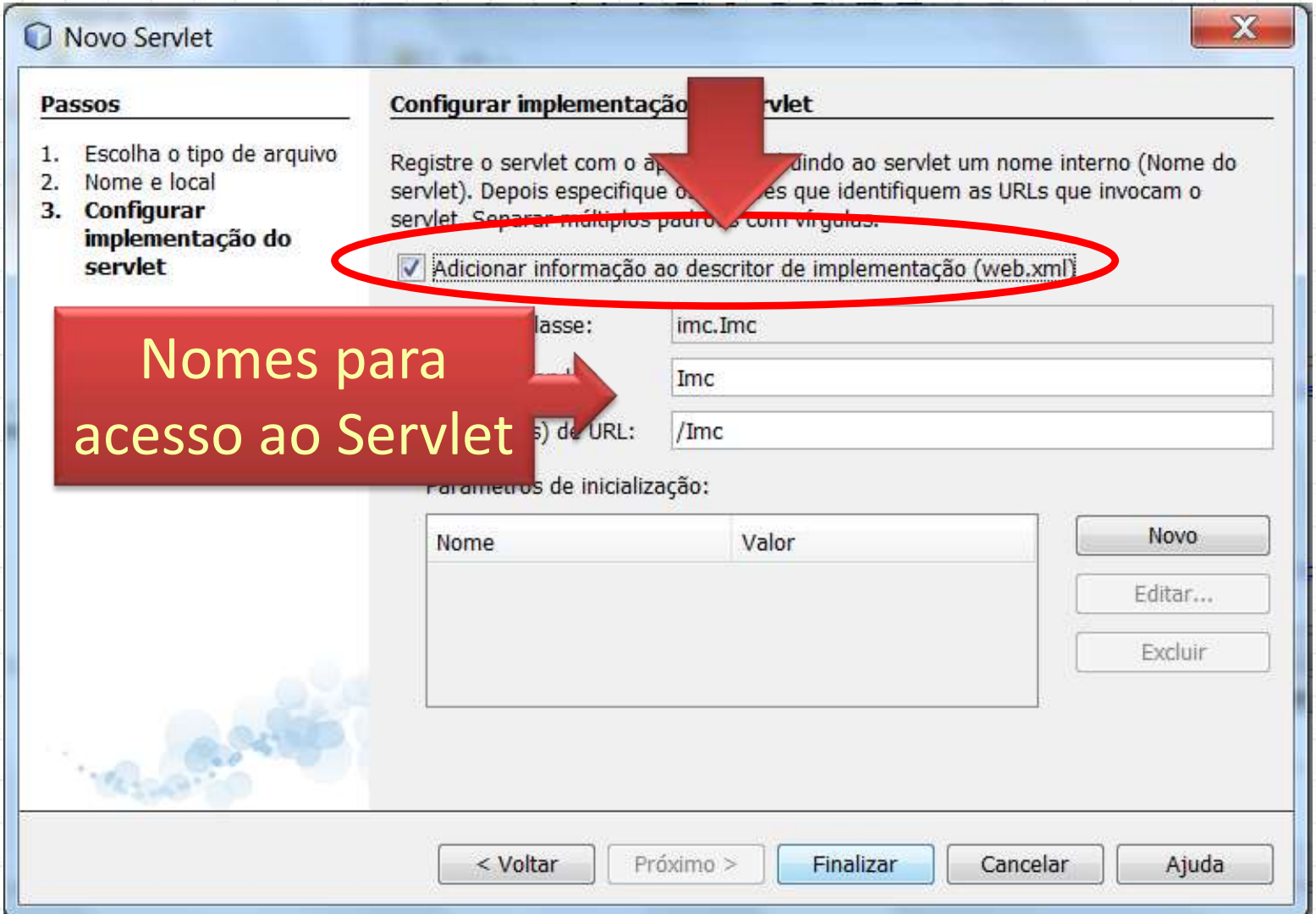
Criando um Servlet

- **MARQUE** a opção **Adiciona informação ao descritor**



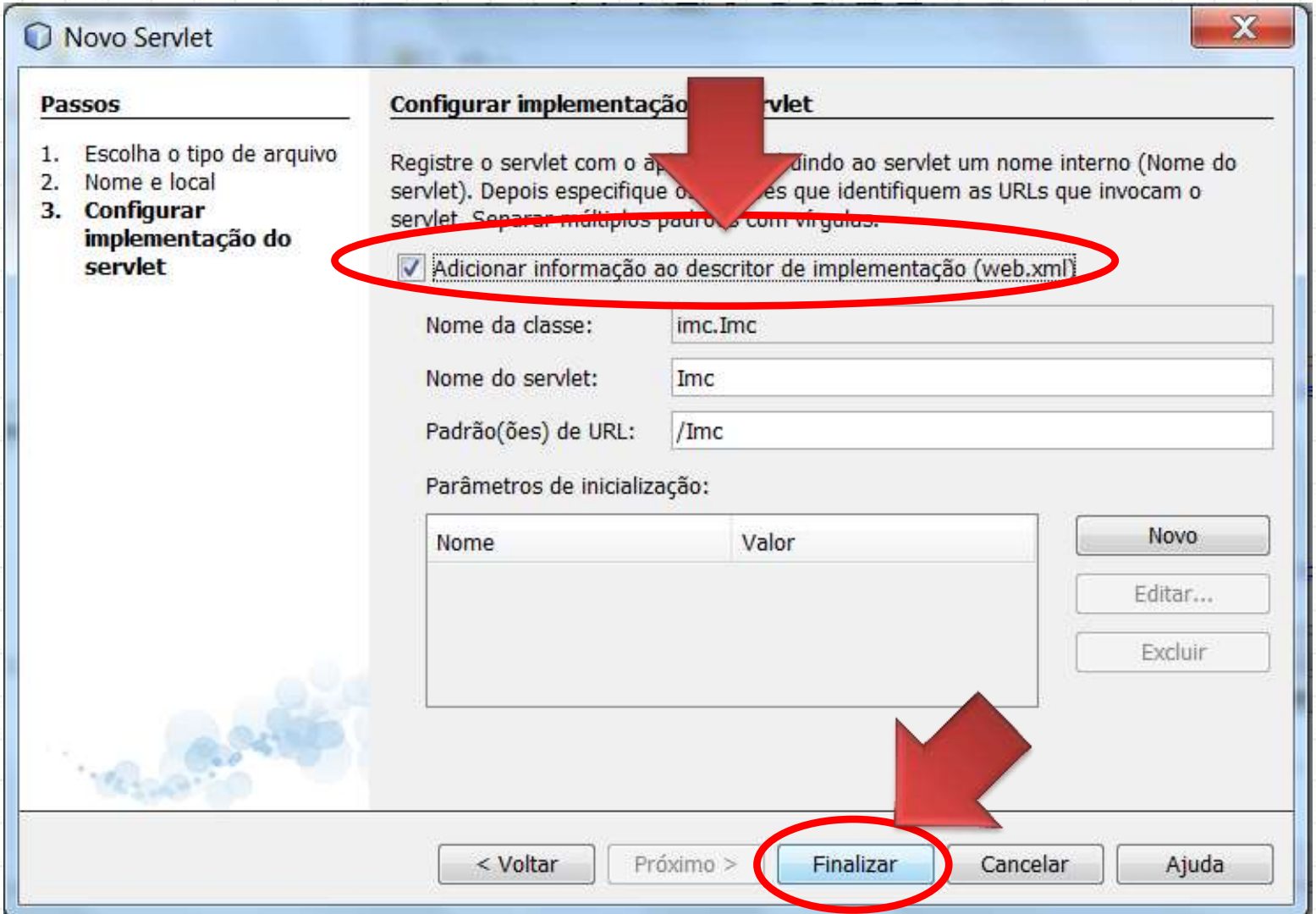
Criando um Servlet

- **MARQUE** a opção **Adiciona informação ao descritor**



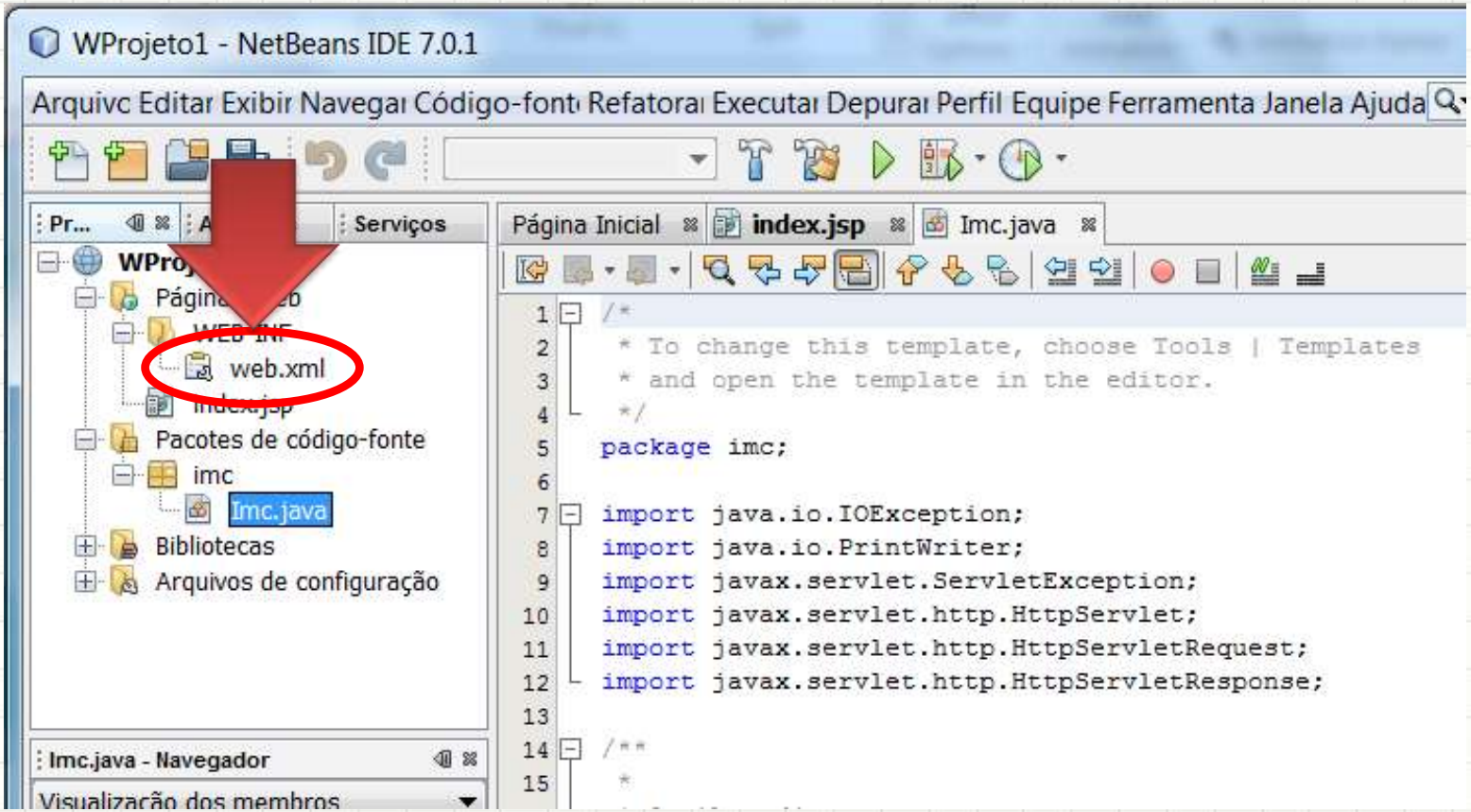
Criando um Servlet

- E clique em **Finalizar**



Criando um Servlet

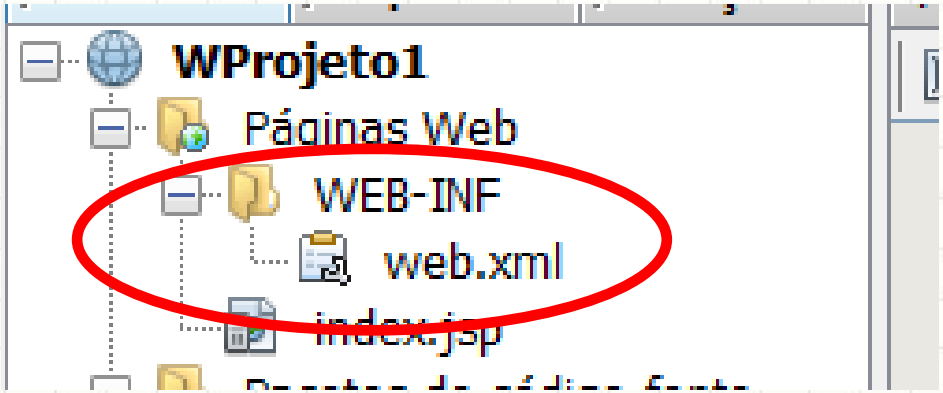
- Isso vai criar uma série de arquivos e abrir o servlet na área de edição... O primeiro é o de configuração:



Configurando o Servlet

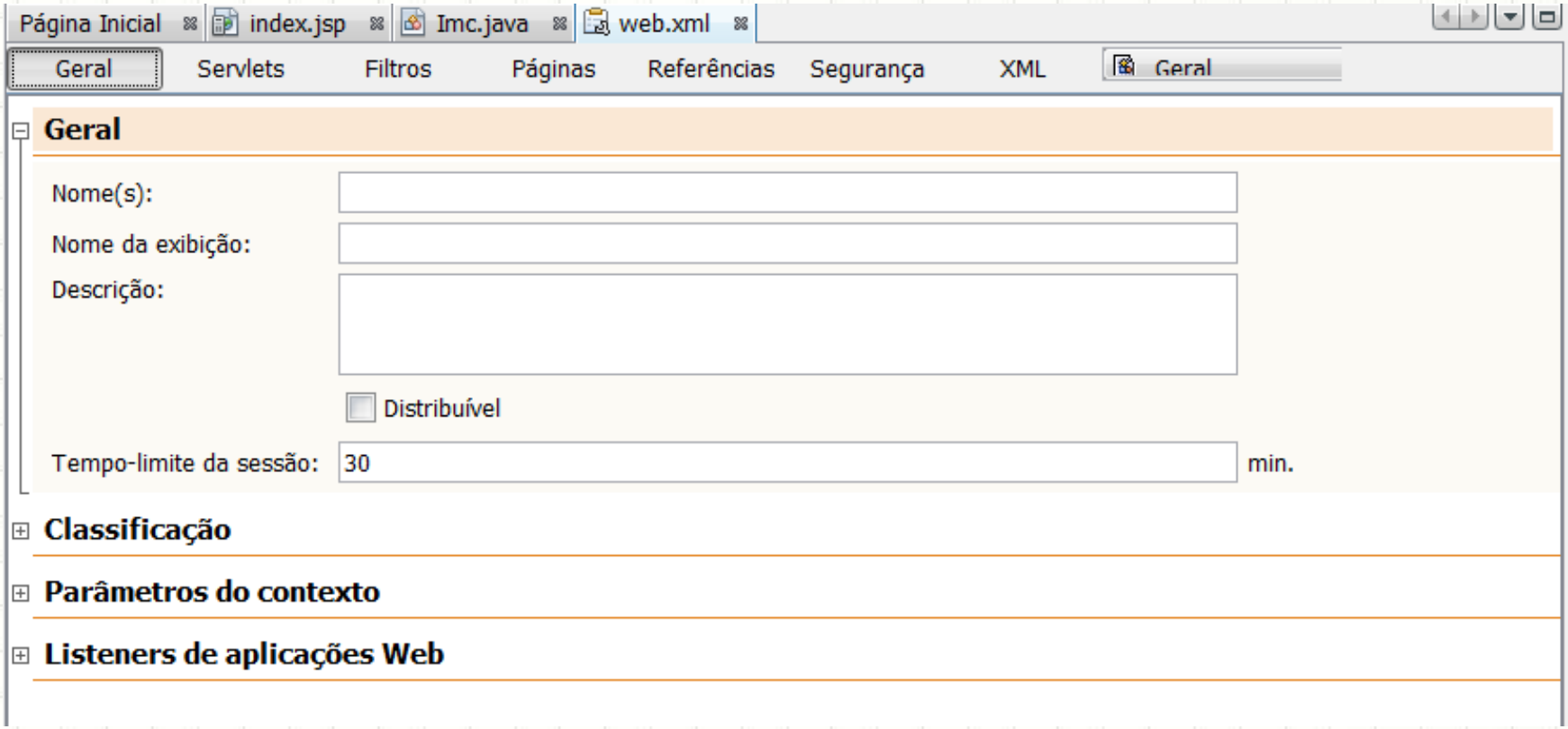
Configurações do Servlet

- O arquivo de configuração é chamado **web.xml...**
- Algumas vezes existem vários deles
- O que nos interessa está na pasta **web-inf**
- Clique duas vezes neste arquivo...



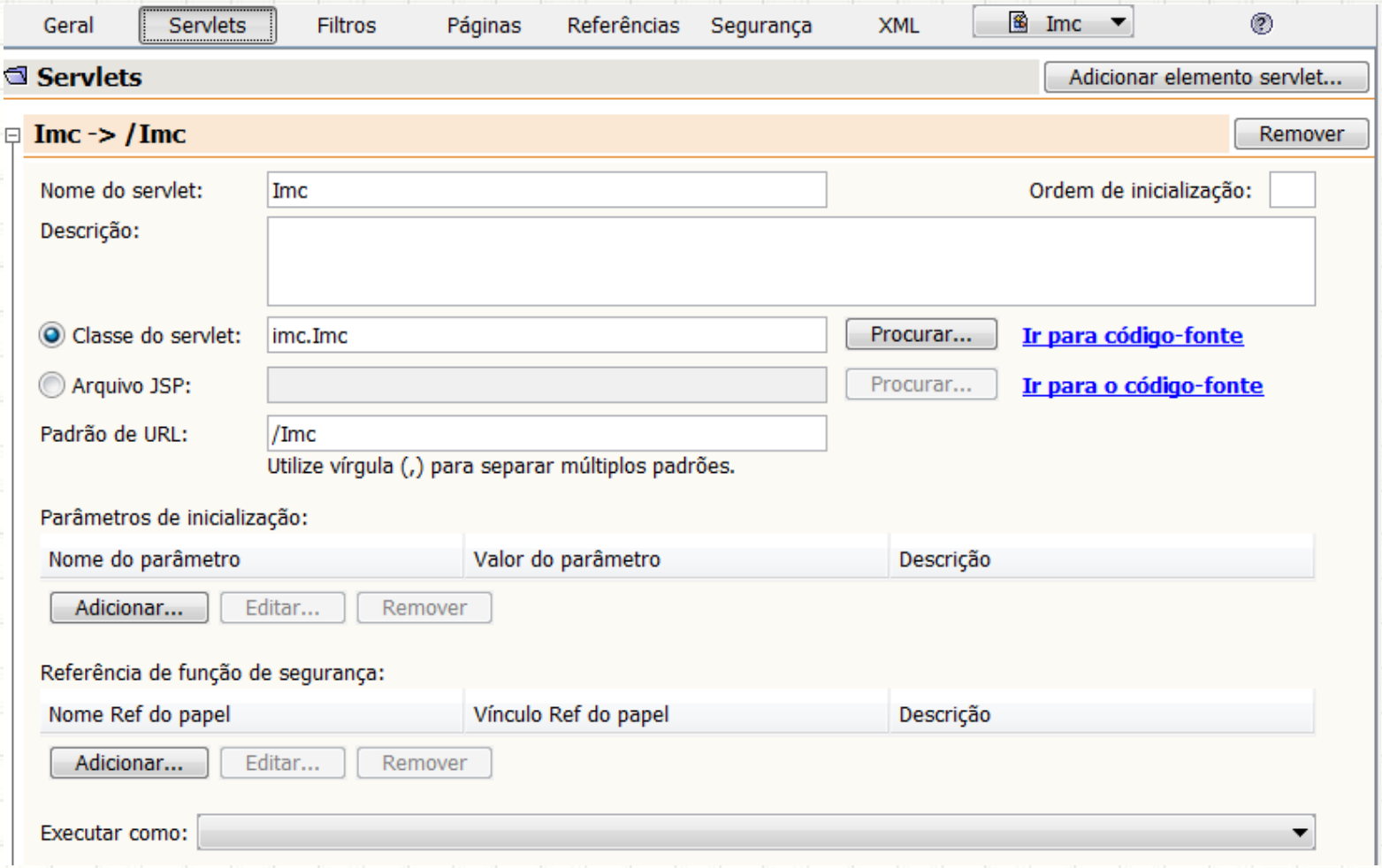
Configurações do Servlet

- A aba **Geral** serve para indicar a descrição do **SISTEMA**, além de indicar o **timeout** (veremos isso depois)



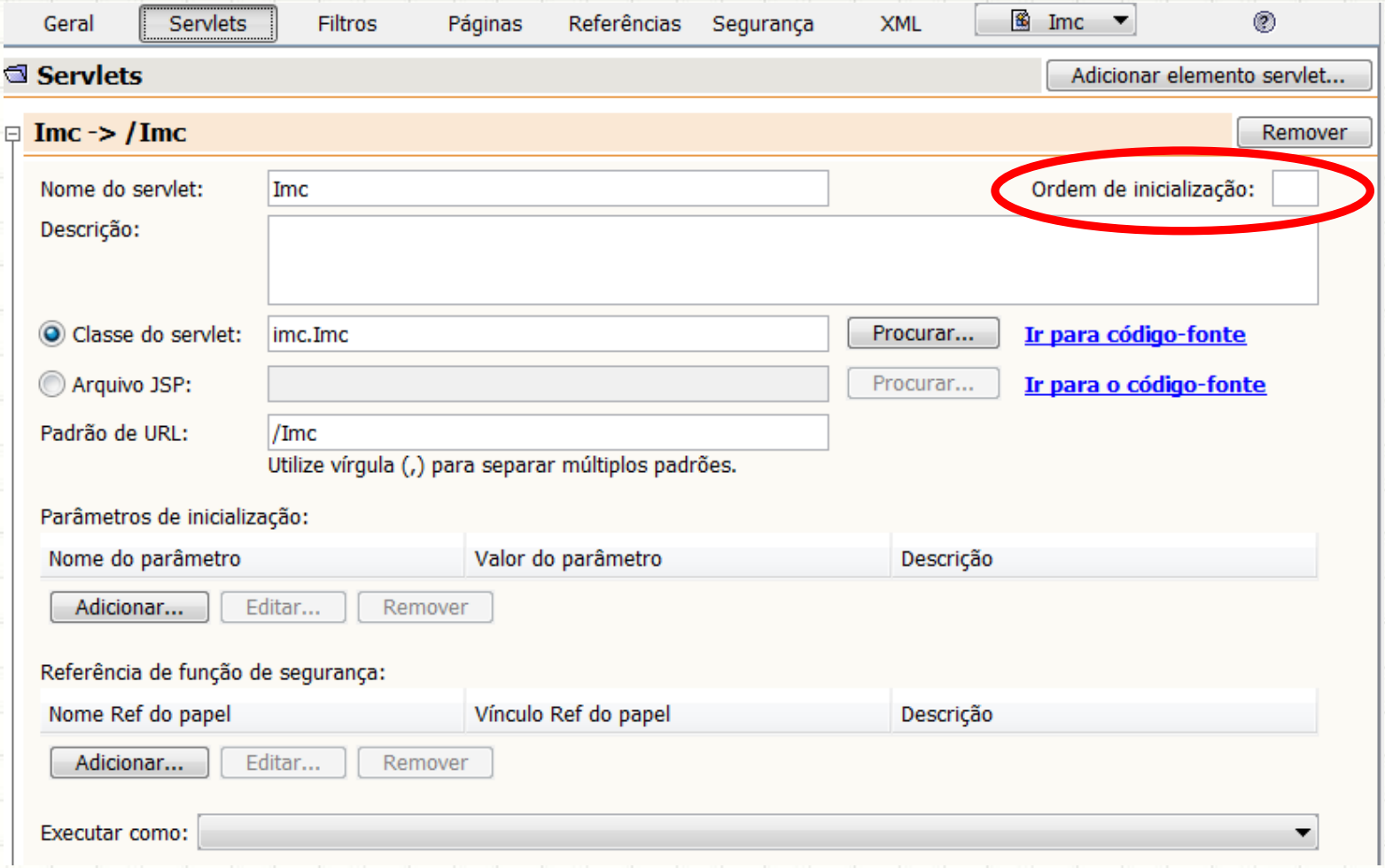
Configurações do Servlet

- A aba **Servlets** configura itens importantes



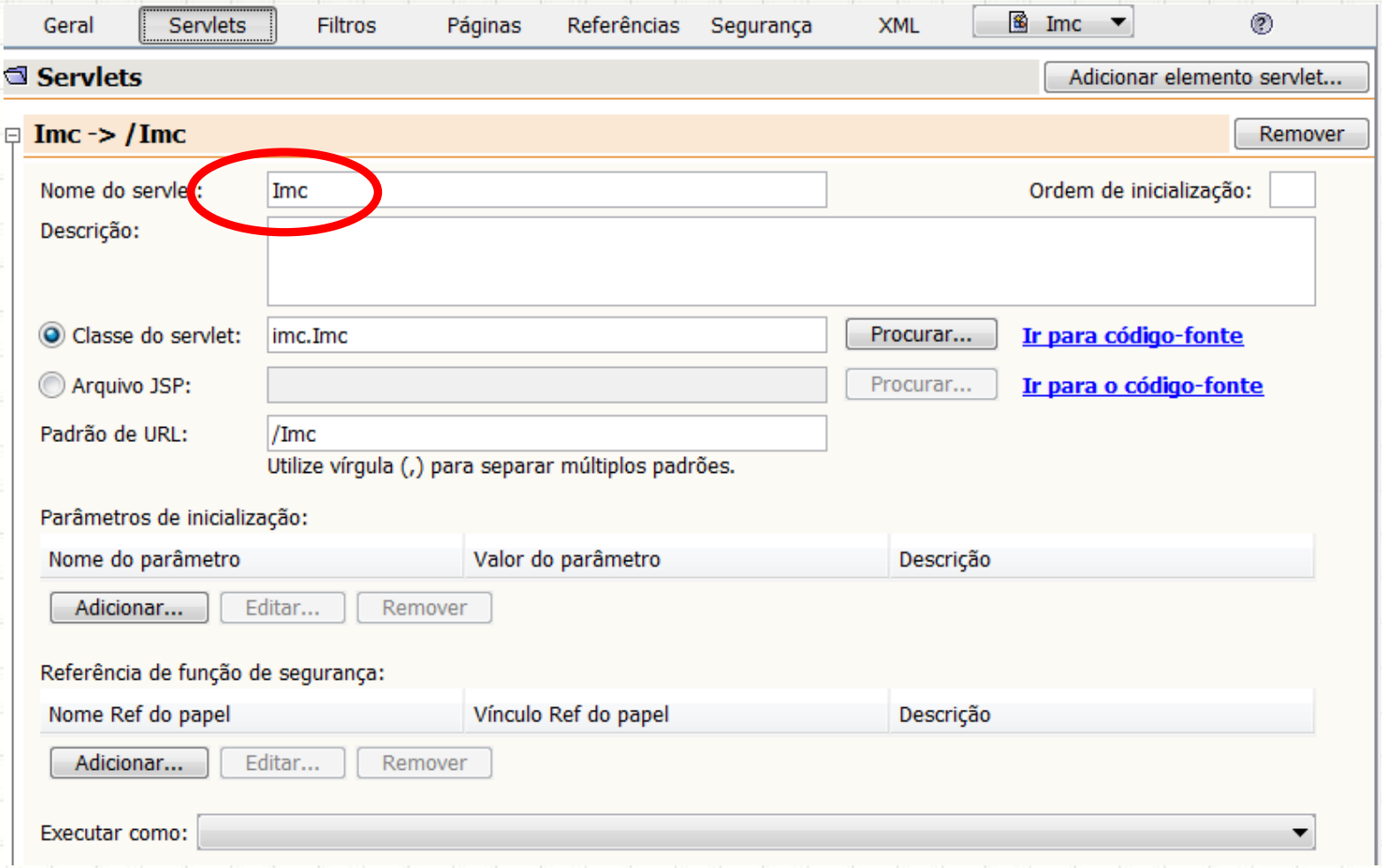
Configurações do Servlet

- A ordem de inicialização



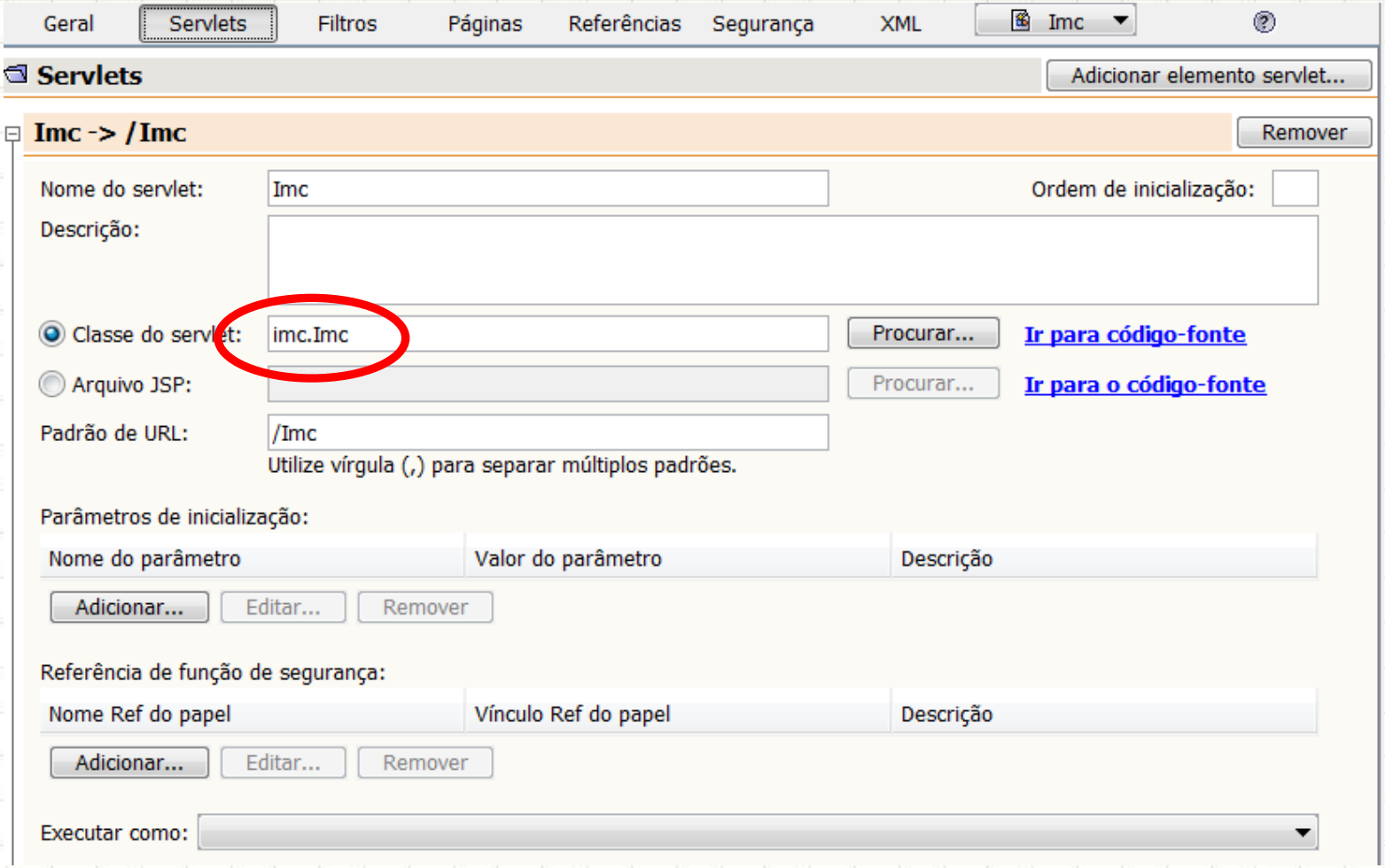
Configurações do Servlet

- O nome do Servlet



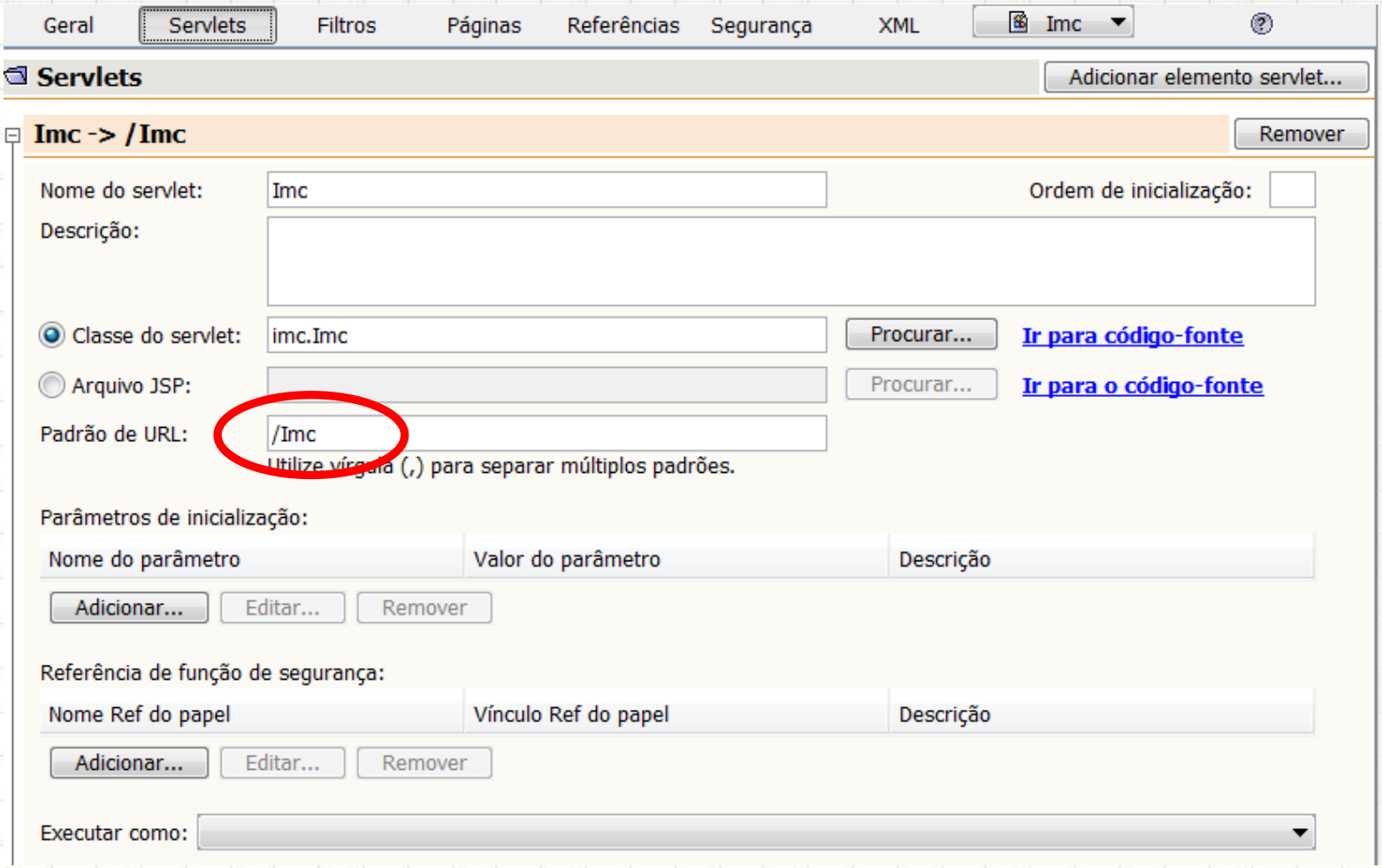
Configurações do Servlet

- A classe do Servlet (pode ser trocada!)



Configurações do Servlet

- E o nome de acesso do Servlet



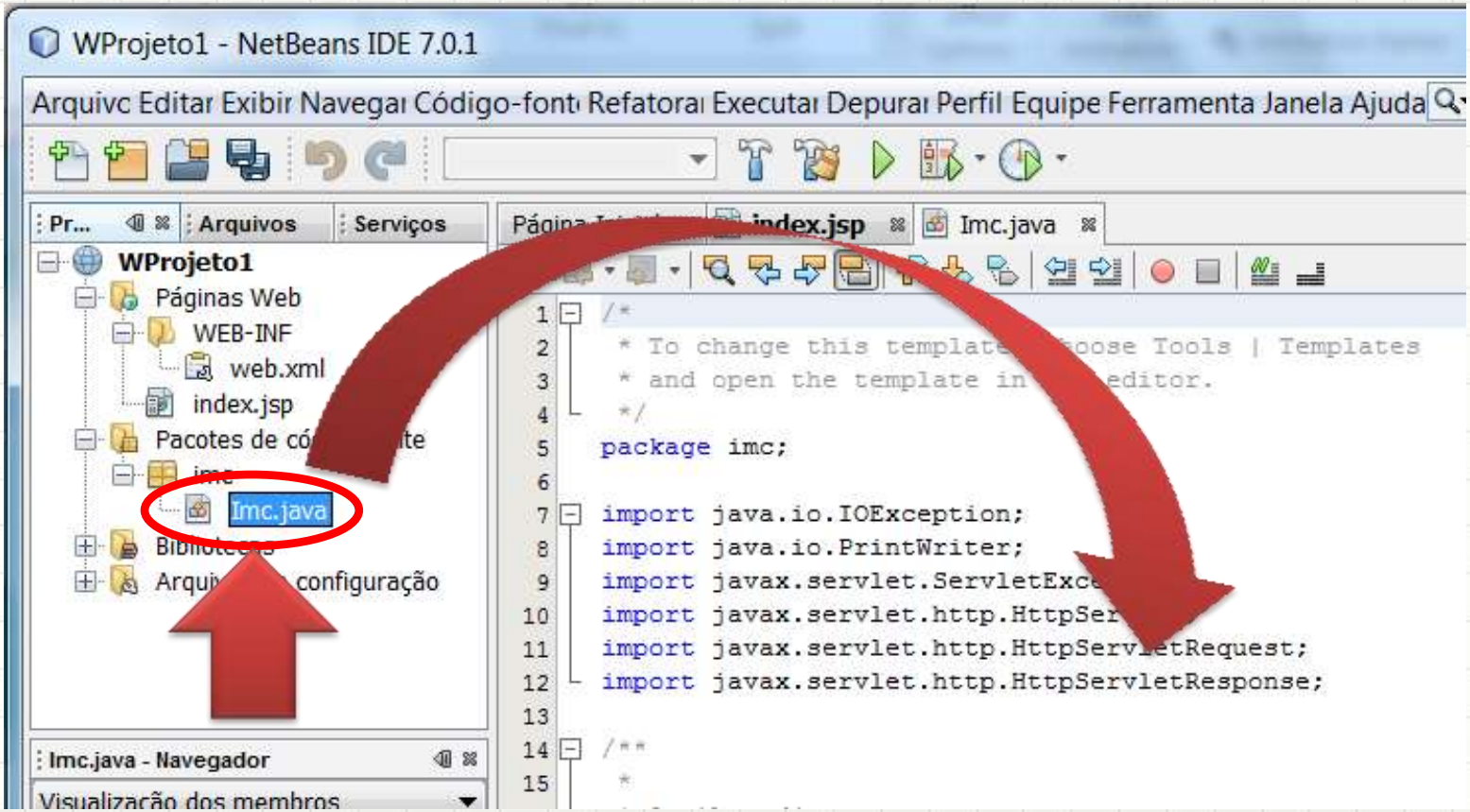
Configurações do Servlet

- Essas e outras configurações podem ser feitas diretamente pela aba XML...
- **Mas pelo NetBeans é mais conveniente!**

Voltando a Editar o Servlet

Criando um Servlet

- O outro arquivo importante é o arquivo do Servlet, neste caso o **Imc.java**.



Limpando a área

- Vamos apagar os comentários do NetBeans

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package imc;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author djcaetano
 */
public class Imc extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code> method
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse res
```

Limpando a área

- Vamos esconder código “desnecessário”

```
package imc;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Imc extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse res
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            /* TODO output your page here
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet Imc</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Servlet Imc at " + request.getContextPath () + "</h1>");
            out.println("</body>");
            out.println("</html>");
             */
        } finally {
            out.close();
```

Entendendo o Servlet

- Observe que o Servlet **extends HttpServlet**

```
package imc;
import ...

public class Imc extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse res
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            /* TODO output your page here
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet Imc</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Servlet Imc at " + request.getContextPath () + "</h1>");
            out.println("</body>");
            out.println("</html>");
             */
        } finally {
            out.close();
        }
    }

    HttpServlet methods. Click on the + sign on the left to edit the code.
}
```

Entend

- Vamos e

Este é o trecho que mais nos interessa!

```
package imc;
import ...

public class Imc extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse res
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            /* TODO output your page here
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet Imc</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Servlet Imc at " + request.getContextPath () + "</h1>");
            out.println("</body>");
            out.println("</html>");
             */
        } finally {
            out.close();
        }
    }

    HttpServlet methods. Click on the + sign on the left to edit the code.
}
```

Entendendo o Servlet

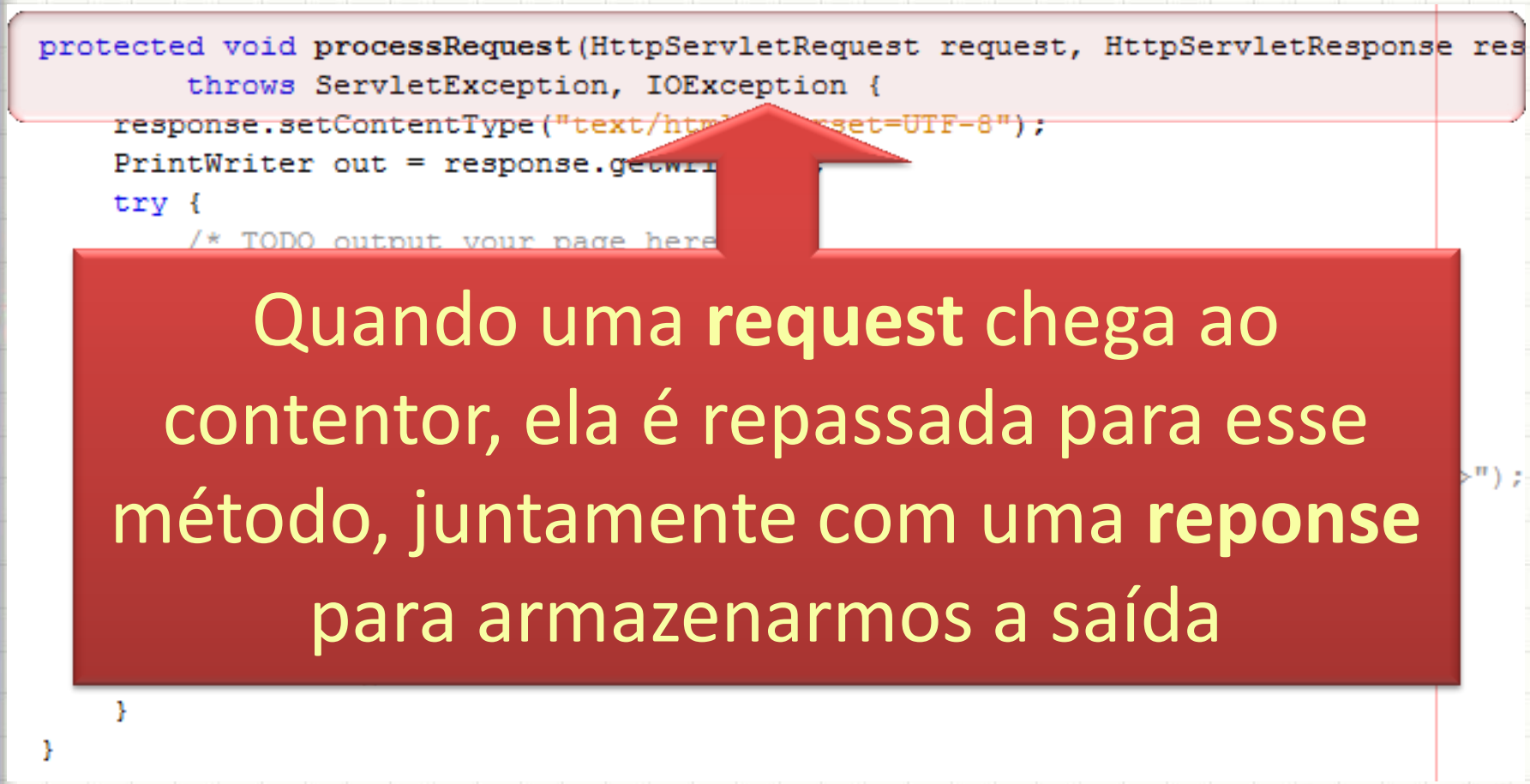
- Vamos entender a estrutura

```
package imc;
import ...

public class Imc extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse res
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            /* TODO output your page here
            out.println("<html>");
            out.println("<head>");

                                                                    + "</h1>");

        } finally {
            out.close();
        }
    }

    HttpServlet methods. Click on the + sign on the left to edit the code.
}
```

Não mexa aqui!

Entendendo o Servlet

- Vamos entender o funcionamento



Entendendo o Servlet

- Vamos entender o funcionamento

```
protected void processRequest(HttpServletRequest request, HttpServletResponse res
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        /* TODO output
        out.println("<html>
        out.println("<head>
```

A primeira coisa feita é a configuração do objeto **out**, que usaremos para imprimir a saída

Programando o Servlet

- Aqui é onde escreveremos o nosso código
- **Temos de imprimir um HTML!**

```
protected void processRequest(HttpServletRequest request, HttpServletResponse res
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        /* TODO output your page here
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Servlet Imc</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Servlet Imc at " + request.getContextPath () + "</h1>");
        out.println("</body>");
        out.println("</html>");
         */
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Acompanhe o professor!

```
protected void processRequest(HttpServletRequest request, HttpServletResponse res
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        /* TODO output your page here
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Servlet Imc</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Servlet Imc at " + request.getContextPath () + "</h1>");
        out.println("</body>");
        out.println("</html>");
         */
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Execute e veja o que acontece!

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Calculadora de Imc: Resultados</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Índice de Massa Corporal: </h1>");
        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Falta fazer o cálculo:
  - IMC = PESO / (ALTURA * ALTURA)

- Mas antes... É preciso pegar os parâmetros na **request**...
- Pegar... Parâmetro... Da request...
  - request.getParameter(“nome”)
- Os nomes são “peso” e “altura” (do form!)
- **NOTA**: as informações da **request** estão sempre no formato **String**!

Programando o Servlet

- Execute e veja o que acontece!

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Calculadora de Imc: Resultados</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Índice de Massa Corporal: </h1>");

        String pesoT = request.getParameter("peso");
        String alturaT = request.getParameter("altura");

        out.println("Peso: " + pesoT);
        out.println("Altura: " + alturaT);

        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Exec

OPA! Não pulou linha!

a) Estamos imprimindo HTML

b) Onde está o e o ???

```
protected                                                    sponse response)
        th
    respon
    PrintW
    try {
        ou
        ou
        ou
        ou
        ou
        out.println(

        String pesoT = request.getParameter("peso");
        String alturaT = request.getParameter("altura");

        out.println("Peso: " + pesoT);
        out.println("Altura: " + alturaT);

        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Execute e veja o que acontece!

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Calculadora de Imc: Resultados</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Índice de Massa Corporal: </h1>");

        String pesoT = request.getParameter("peso");
        String alturaT = request.getParameter("altura");

        out.println("<p>Peso: " + pesoT + "</p>");
        out.println("<p>Altura: " + alturaT + "</p>");

        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Antes de fazer os cálculos, porém...

- Temos que converter estas **Strings** para números com vírgula do tipo **double**
- Existe um método pronto para isso, na classe **Double**... Ele se chama **valueOf**

- **NOTA:** Use PONTO para indicar peso e altura
- Certo: 1.70 Errado: 1,70

Programando o Servlet

- Execute e veja o que acontece!

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Calculadora de Imc: Resultados</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Índice de Massa Corporal: </h1>");

        String pesoT = request.getParameter("peso");
        String alturaT = request.getParameter("altura");
        double peso = Double.valueOf(pesoT);
        double altura = Double.valueOf(alturaT);

        double imc = peso / (altura*altura);
        out.println("<p>IMC: " + imc + "</p>");

        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- Podemos melhorar um pouco mais...

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Calculadora de Imc: Resultados</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Índice de Massa Corporal: </h1>");

        String pesoT = request.getParameter("peso");
        String alturaT = request.getParameter("altura");
        double peso = Double.valueOf(pesoT);
        double altura = Double.valueOf(alturaT);

        double imc = peso / (altura*altura);
        out.printf("<p>IMC: %3.1f</p>", imc);

        out.println("</body>");
        out.println("</html>");
    } finally {
        out.close();
    }
}
```

Programando o Servlet

- O programa tem problemas com “,” nos números? Experimente!
- Corrija substituindo:

```
String pesoT = request.getParameter("peso");
String alturaT = request.getParameter("altura");
double peso = Double.valueOf(pesoT);
double altura = Double.valueOf(alturaT);
```

- Por isso:

```
String pesoT = request.getParameter("peso");
pesoT = pesoT.replaceAll(",",".");
String alturaT = request.getParameter("altura");
alturaT = alturaT.replaceAll(",",".");
double peso = Double.valueOf(pesoT);
double altura = Double.valueOf(alturaT);
```

Programando o Servlet

- Para imprimir mensagens de acordo com o resultado, experimente acrescentar o código abaixo!

```
out.printf("<p>IMC: %3.1f</p>", imc);

if (imc < 18.0) {
    out.println("<p>Cuidado! Abaixo!</p>");
} else if (imc < 25.0) {
    out.println("<p>Parabéns! Peso ideal!</p>");
} else {
    out.println("<p>Cuidado! Acima!</p>");
}

out.println("</body>");
```

Programando o Servlet

- O servlet ainda “capota” quando um “texto” é digitado nos campos numéricos! Experimente!
- Corrija isso acrescentando o “catch” abaixo!

```
            out.println("</body>");
            out.println("</html>");
        } catch(Exception e) {
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Resutado do IMC</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<p>Digite um número, mané!</p>");
            out.println("</body>");
            out.println("</html>");
        } finally {
            out.close();
        }
```

Atividade Estruturada

Orientação Atividades Estruturadas

- Esta disciplina possui Atividades Estruturadas
- Elas serão disponibilizadas no momento oportuno
- A primeira consiste em uma pesquisa (leitura e redação)
- A segunda consiste em compreender e modificar um sistema funcional
- Aguardem maiores informações!

Conclusões

Resumo

- As aplicações Web funcionam como pequenos programas que rodam no servidor
- Estes programas, em Java, são feitos com a tecnologia Servlets
- Um Servlet basicamente recebe uma requisição (**request**) e coloca os resultados em uma resposta (**response**)

- **TAREFA**
  - **Trabalho 2 Online!**

Próxima Aula

- Ainda não estou seguro com esses tais Servlets!
- Na próxima aula, exercitaremos mais...
- E veremos algumas novidades!

PERGUNTAS?

Bom Descanso a Todos!