

# SERVLETS

PROF. ME. HÉLIO  
ESPERIDIÃO

O que é?

Java Servlet é uma tecnologia Java.

É um modelo simples para construir uma aplicação web completa em Java.

# NetBeans

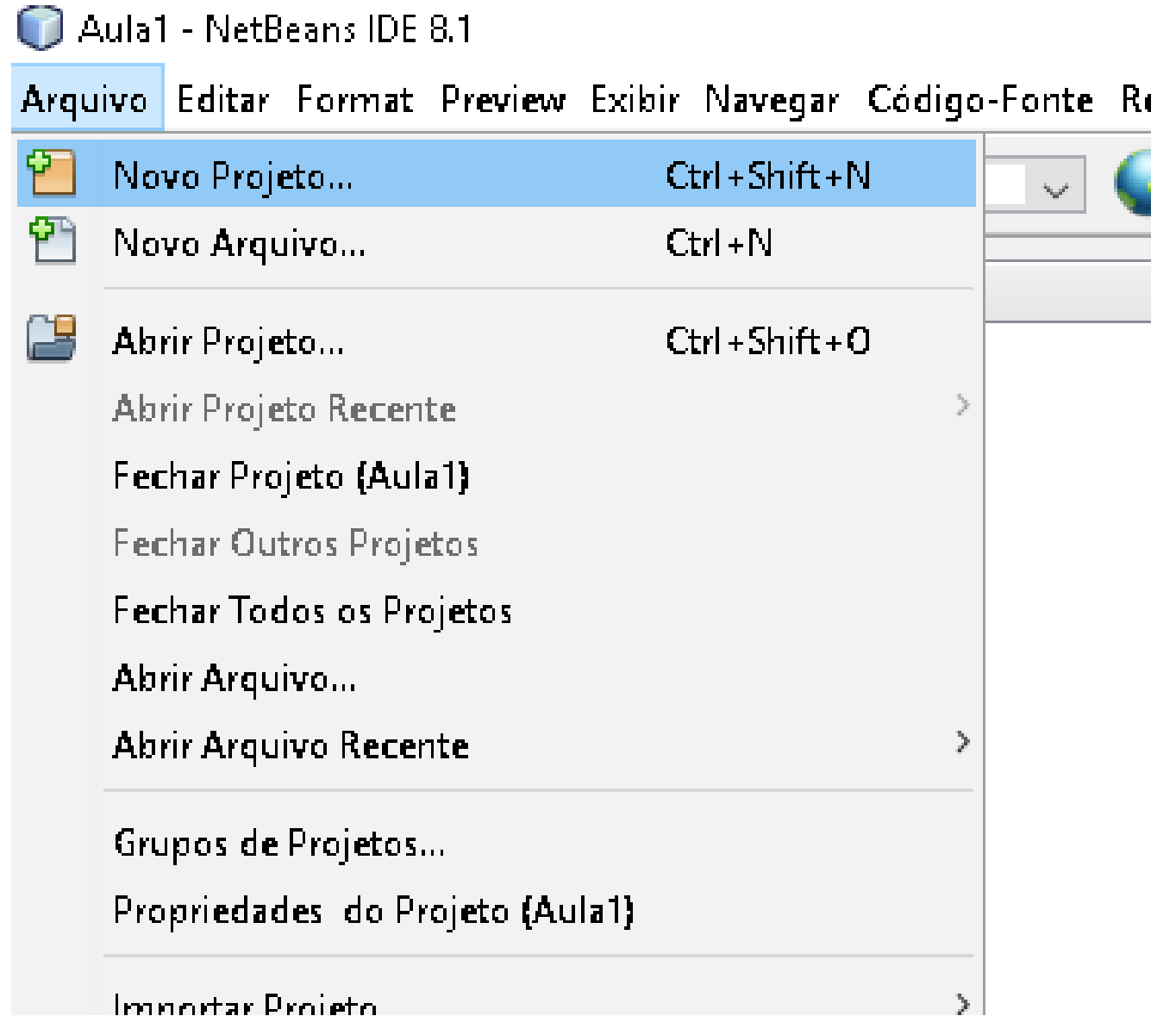
O NetBeans IDE gratuita e de código aberto para desenvolvedores de software nas linguagens Java, C, C++, PHP, Groovy, Ruby, entre outras.

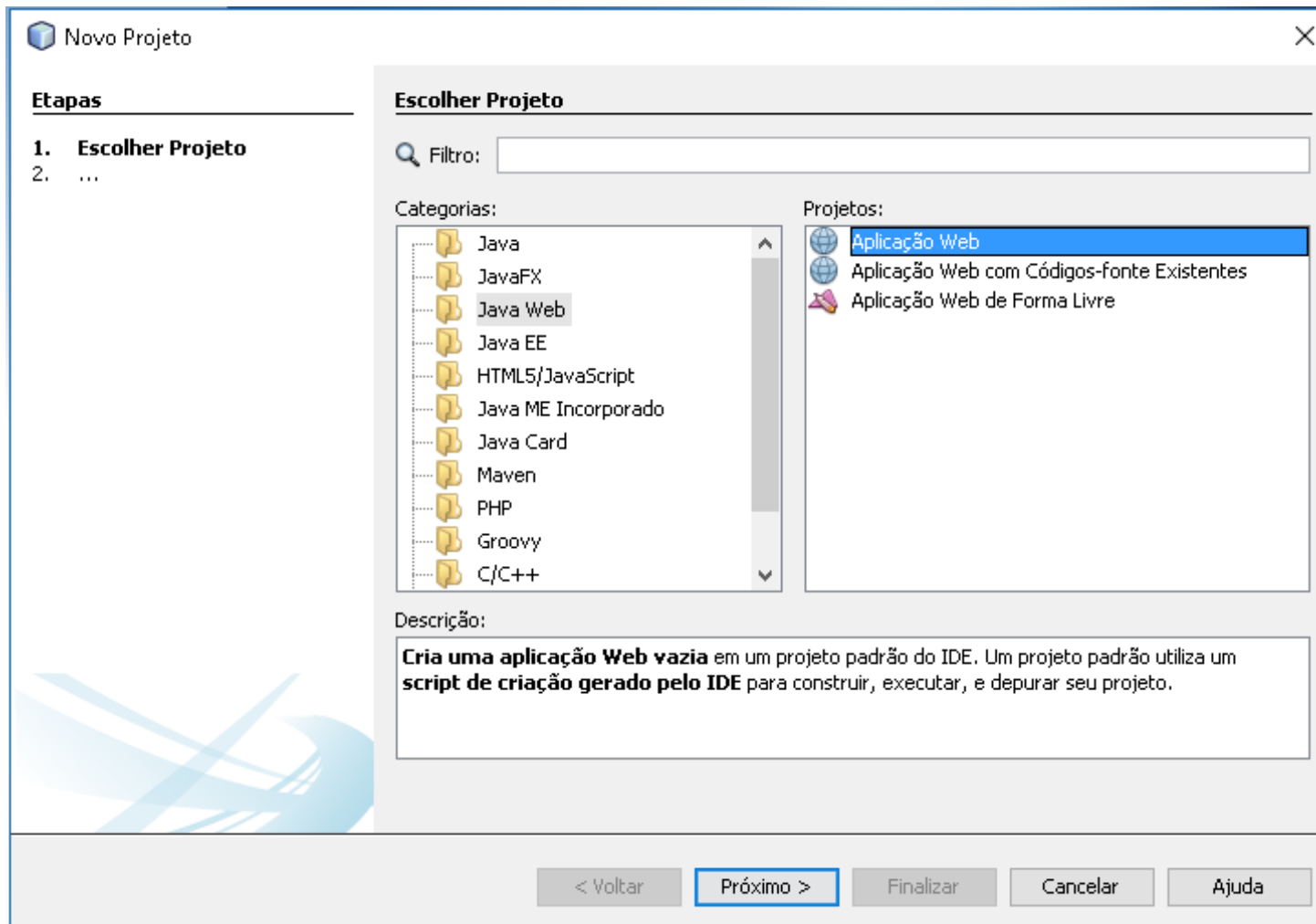
O IDE é executado em muitas plataformas, como Windows, **Linux**, Solaris e MacOS.

O NetBeans IDE oferece aos desenvolvedores ferramentas necessárias para criar aplicativos profissionais de desktop, empresariais, Web e móveis multiplataformas.

# Criar um novo projeto

Crie um novo projeto no NetBeans.





# Crie um projeto de Aplicação WEB

---

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 co  
(consulte a Ajuda para obter detalhes).

< Voltar   Próxio >   Finalizar

# Configurações de arquivos

Determina um nome para o projeto e escolha as pastas onde os arquivos serão salvos.

# Configurações do Servidor

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)>

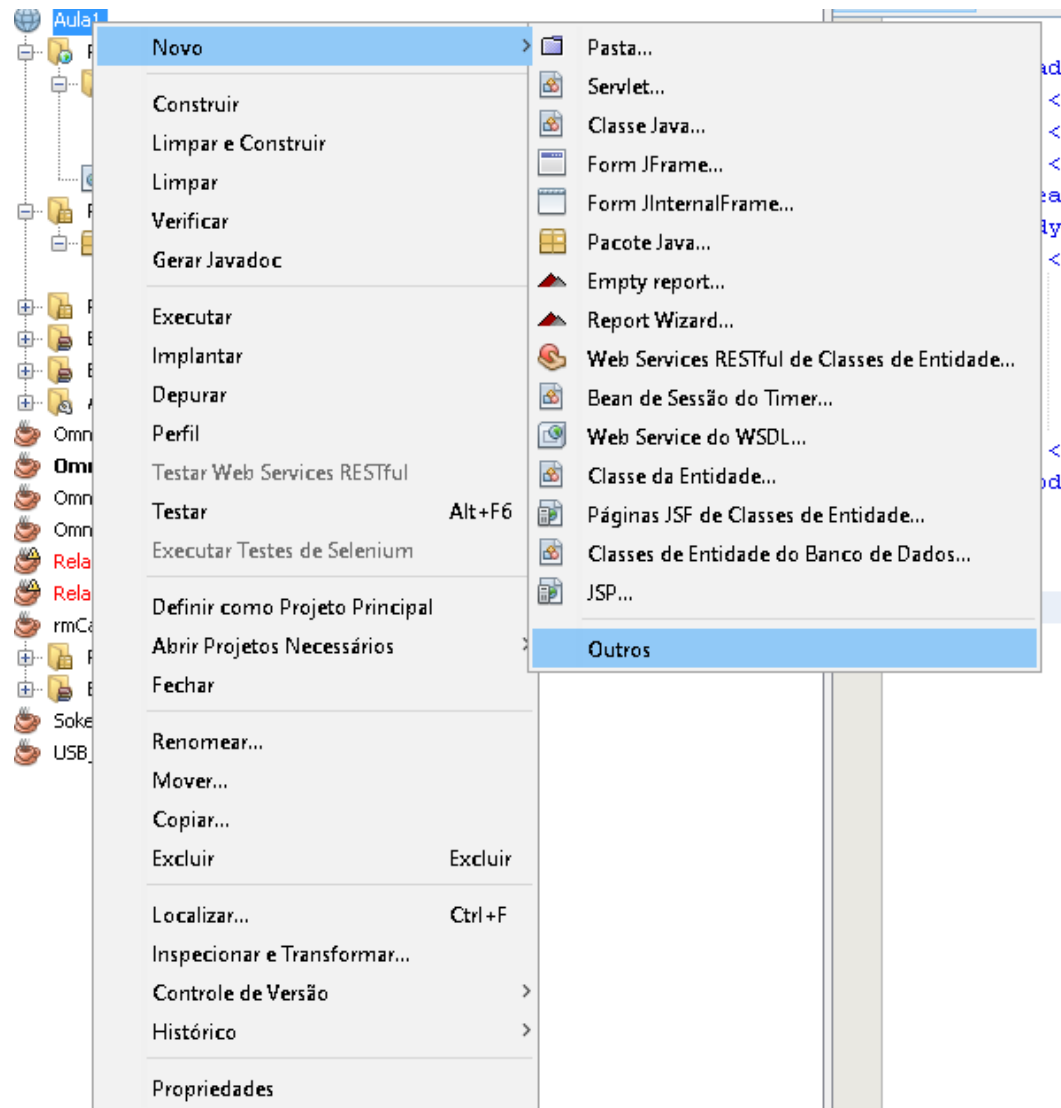
Servidor: GlassFish Server 4.1.1 Adicionar...

Versão do Java EE: Java EE 7 Web

Caminho do Contexto: /Aula1

< Voltar Próximo > Finalizar Cancelar Ajuda

Remova o  
caminho de  
contexto

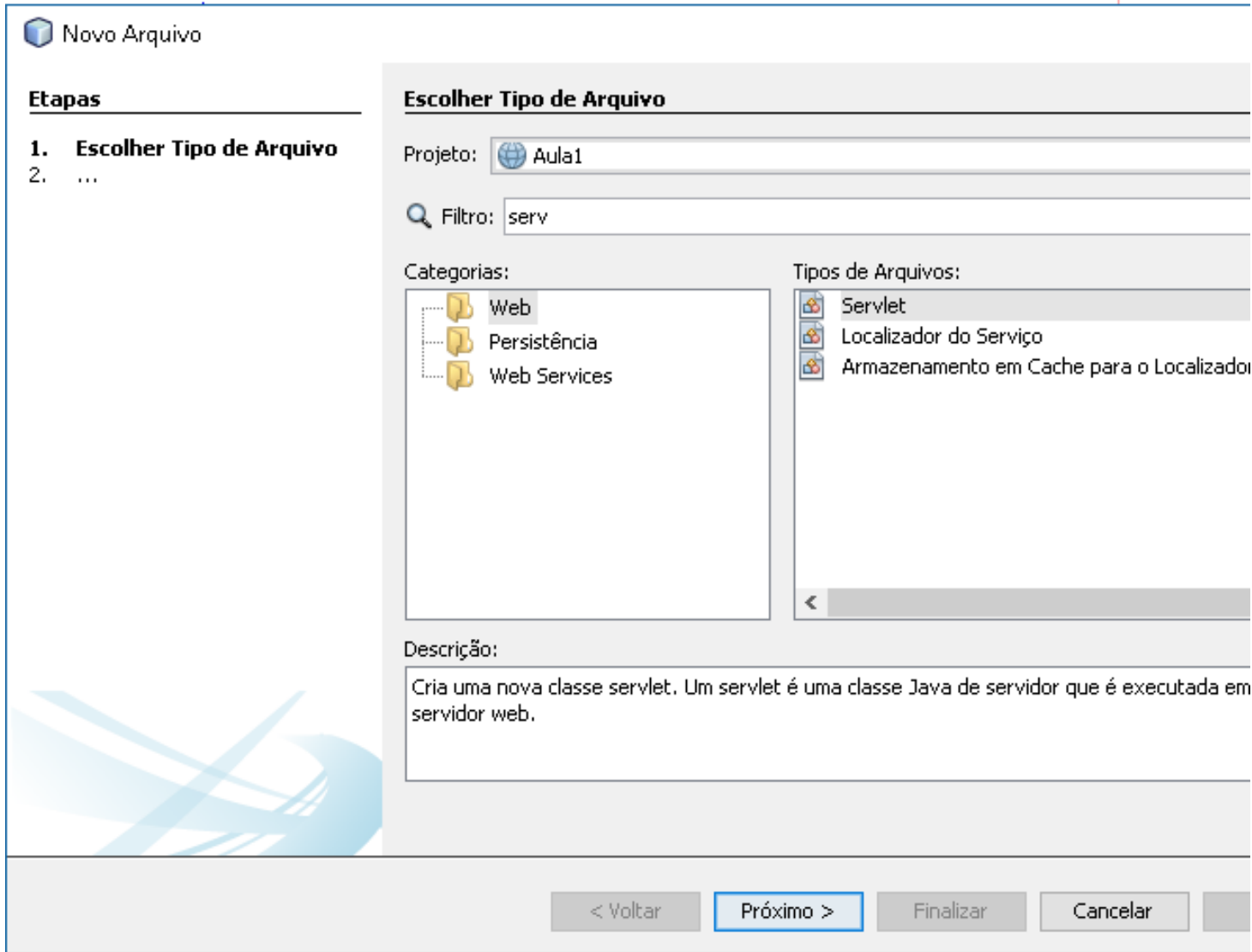


# Criar um servlet

Clique com o botão direito sobre o nome do projeto.

Escolha a opção novo >> outros





# Classe Servlet

Escolha o tipo Servlet

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:

Paquete:

Arquivo Criado:

< Voltar   Próximo >   Finalizar   Cancelar

# Classe Servlet

De um nome para a sua classe.

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:

Nome do Servlet:

Padrão(ões) de URL:

Parâmetros de Inicialização:

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

< Voltar   Próximo >   **Finalizar**   Cancelar   Ajuda

Determine:

Nome do Servlet

Padrão de url

Marque a opção adicionar informações de implantação(web.xml)

# XML de Mapeamento (O arquivo Web.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  <servlet>
    <servlet-name>CadastroClientes</servlet-name>
    <servlet-class>CadastroClientes</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>NovoServlet</servlet-name>
    <servlet-class>NovoServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>CadastroClientes</servlet-name>
    <url-pattern>/CadastroClientes</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>NovoServlet</servlet-name>
    <url-pattern>/NovoServlet</url-pattern>
  </servlet-mapping>
  <session-config>
    <session-timeout>
      30
    </session-timeout>
  </session-config>
</web-app>
```

Nome do Servlet

Nome da classe

Identificação do servlet

Endereço do serviço

Identificação do serviço na rede

# Formulário HTML

---

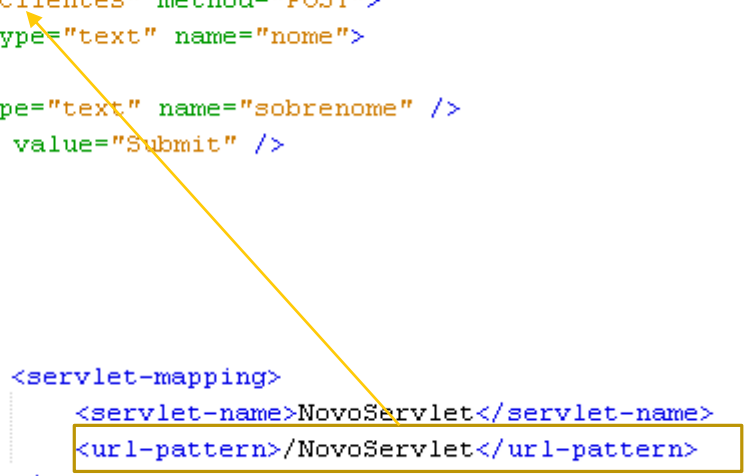
Ao criar o projeto um arquivo `index.html` é criado.

Este arquivo é o primeiro arquivo web da aplicação.

No exemplo ao lado este arquivo foi modificado com a adição de um formulário

```
<html>
  <head>
    <title>TODO supply a title</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <div>teste</div>
    <form action="/CadastroClientes" method="POST">
      First Name: <input type="text" name="nome">
      <br />
      Last Name: <input type="text" name="sobrenome" />
      <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```

```
<servlet-mapping>
  <servlet-name>NovoServlet</servlet-name>
  <url-pattern>/NovoServlet</url-pattern>
</servlet-mapping>
```



# Servlet – CadastroClientes

```
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 CadastroClientes extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    }
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("Dado Recebido: " +request.getParameter("nome") + "<br>");
        out.println("Dado Recebido: " +request.getParameter("sobrenome"));
    }
}
```

Dados recebidos  
por meio do get

Dados recebidos  
por meio do post

Recupera os dados enviados