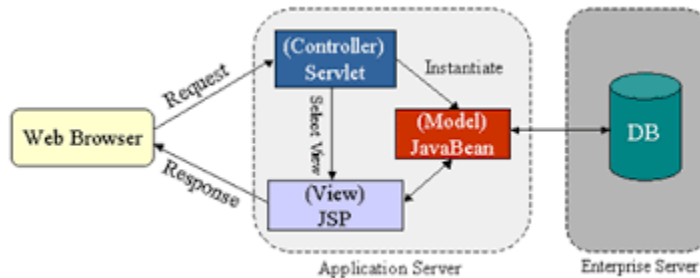


## Unidade 4

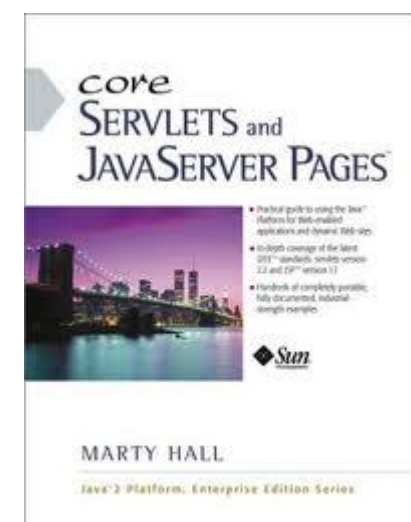
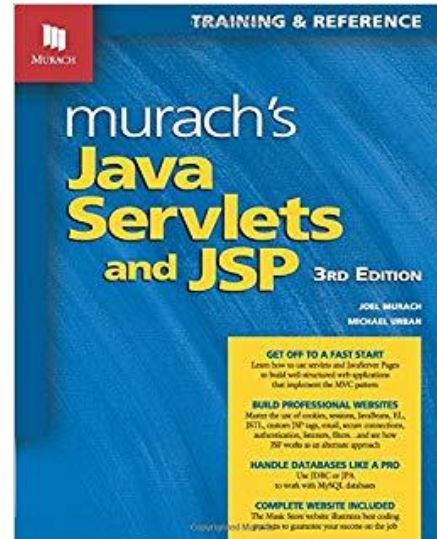
### Tratando dados do Cliente



Prof. Aparecido V. de Freitas  
Doutor em Engenharia  
da Computação pela EPUSP

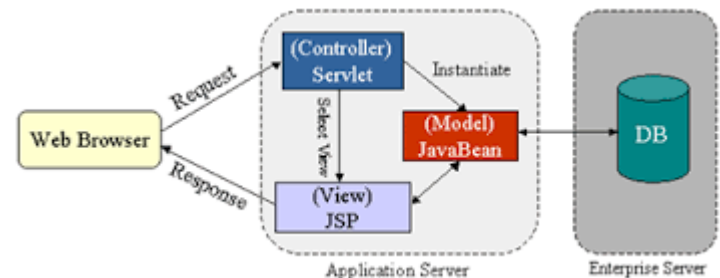
## Referências

- Head First Servlets & JSP – Bryan Basham, Kathy Sierra & Bert Bates
- Core Servlets and Java Server Pages – Marty Hall
- Java Servlets and JSP –Joel Murach – 3rd Edition



## Introdução

- Dados de forms **HTML** podem ser anexados ao final do endereço ao se utilizar o método **GET**  
**http://localhost:8080/app/servlet1?opcao=news&user=x**
- Estes dados também podem ser encaminhados para o servidor (em áreas à parte) quando se utiliza o método **POST**.

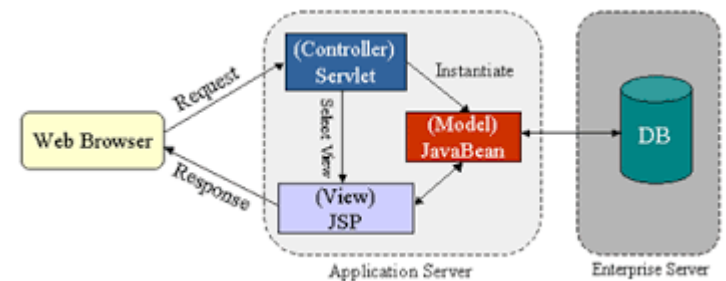


## Processamento de Forms

- ✿ Utilizamos a **TAG FORM** para criar um form **HTML**.
- ✿ O atributo **ACTION** define o endereço do **servlet** ou página **JSP** que irá processar os resultados.

**<FORM ACTION="..."> ... </FORM>**

- ✿ Se **ACTION** form omitido, o dado é submetido à **URL** da página corrente.



## Processamento de Forms

- Use elementos de **INPUT** para coletar dados;
- Textfields são os elementos de entrada mais comuns e criados da seguinte forma:

**<INPUT TYPE="TEXT" NAME = "...">**

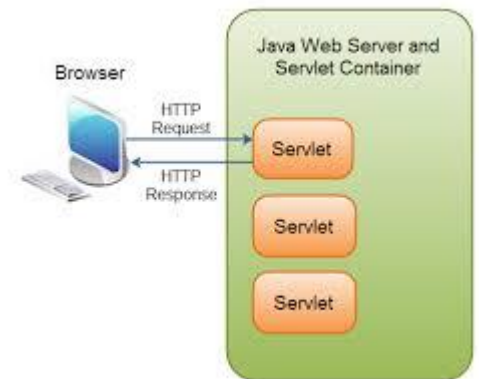


## Processamento de Forms

- Defina um botão de submissão próximo ao final do **FORM**;

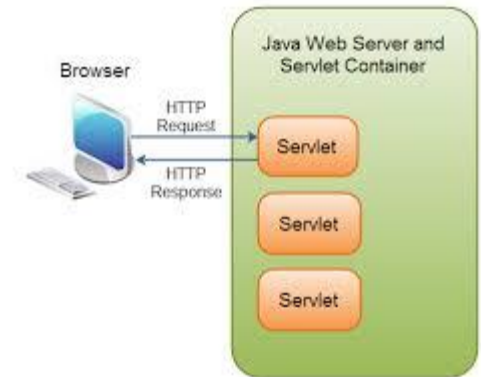
**<INPUT TYPE = "SUBMIT">**

- Quando o botão é pressionado, a **URL** designada em **ACTION** é acionada.



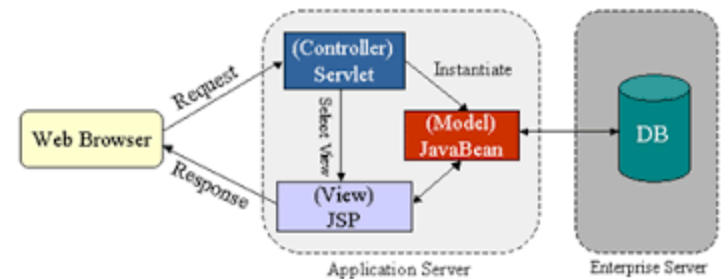
## Submissão de dados

- Com requests **GET**, uma interrogação “?” e pares nome/valor são anexados ao final da URL, separados por “&”.
- Com requests **POST**, o mesmo dado é enviado, mas em linhas de request separadas ao invés de anexados à URL.



# Lendo valores simples

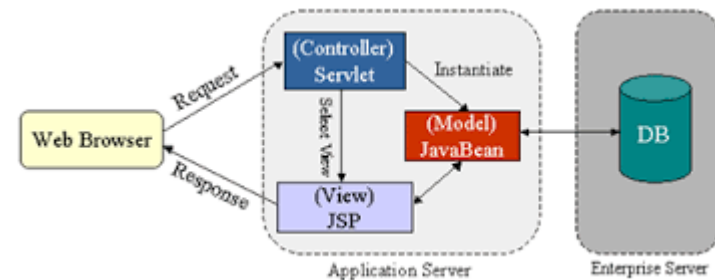
- ⊕ Para ler os dados de um **request (form)**, simplesmente chamamos o método **getParameter** da classe **HttpServletRequest**;
- ⊕ O nome do parâmetro deve ser exatamente igual ao informado no código fonte **HTML**;
- ⊕ O dado retornado é exatamente igual ao informado pelo usuário.





# Método `getParameter()`

- ⊕ Usado da mesma forma quando o dado vem via método **GET** ou **POST**;
- ⊕ Um String **vazio** é retornado se o parâmetro existe mas não tem valor e **null** se parâmetro não existe;
- ⊕ Nomes de parâmetros são case-sensitives.



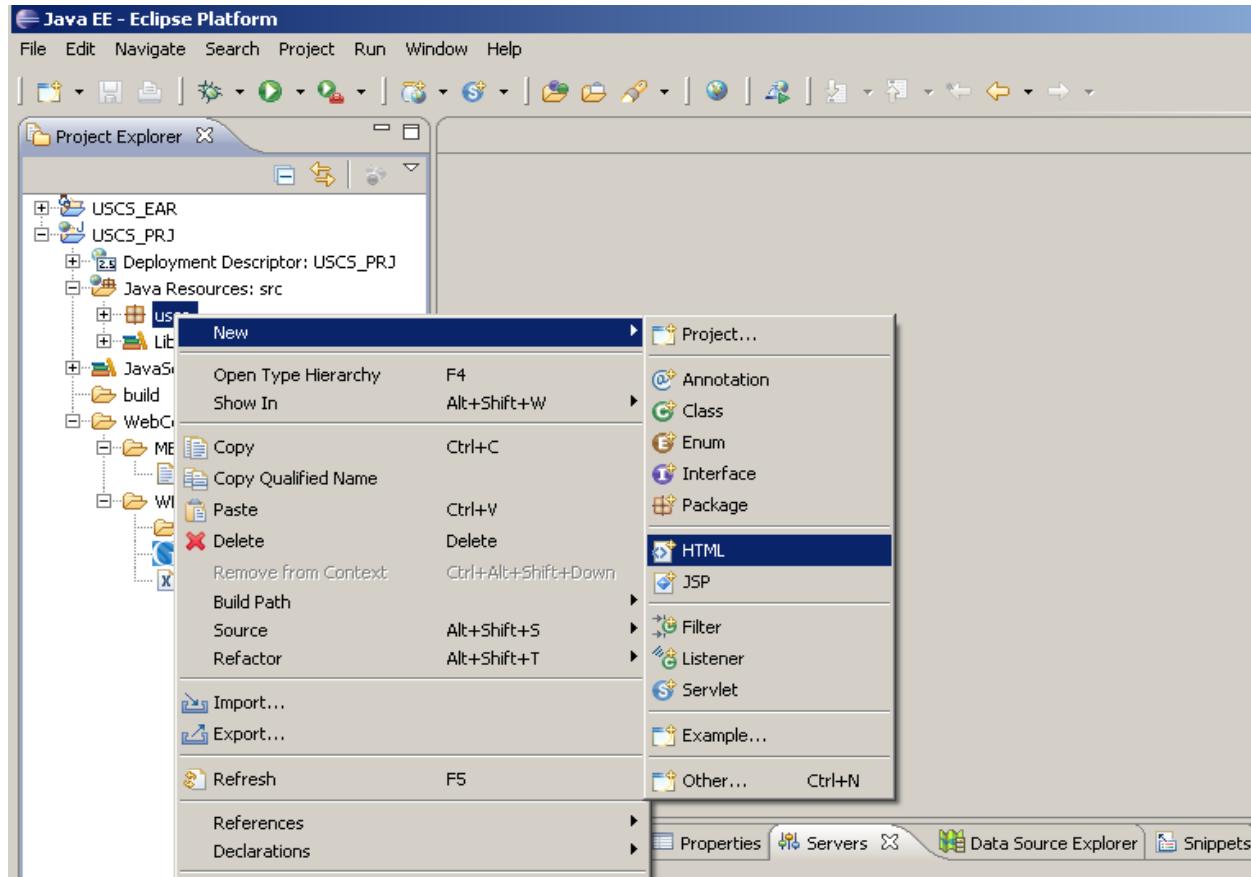
## Atividade – 6

- ✦ Escrever um **servlet** (**Atividade\_06**) que efetua leitura de dois parâmetros (**PARAM1** e **PARAM2**) e os imprime em uma lista **bullet**.
- ✦ Obs. Os parâmetros devem ser informados através de um form **HTML**.

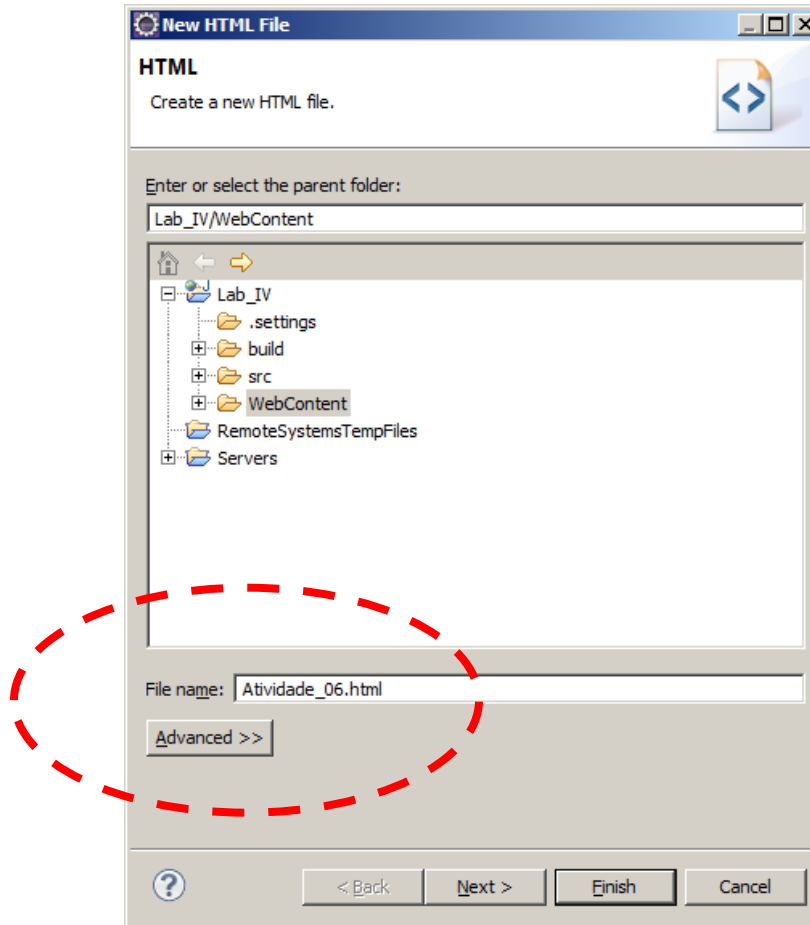
**Obs. Utilizar o ambiente Eclipse / Tomcat**



## Criando HTML sob Eclipse



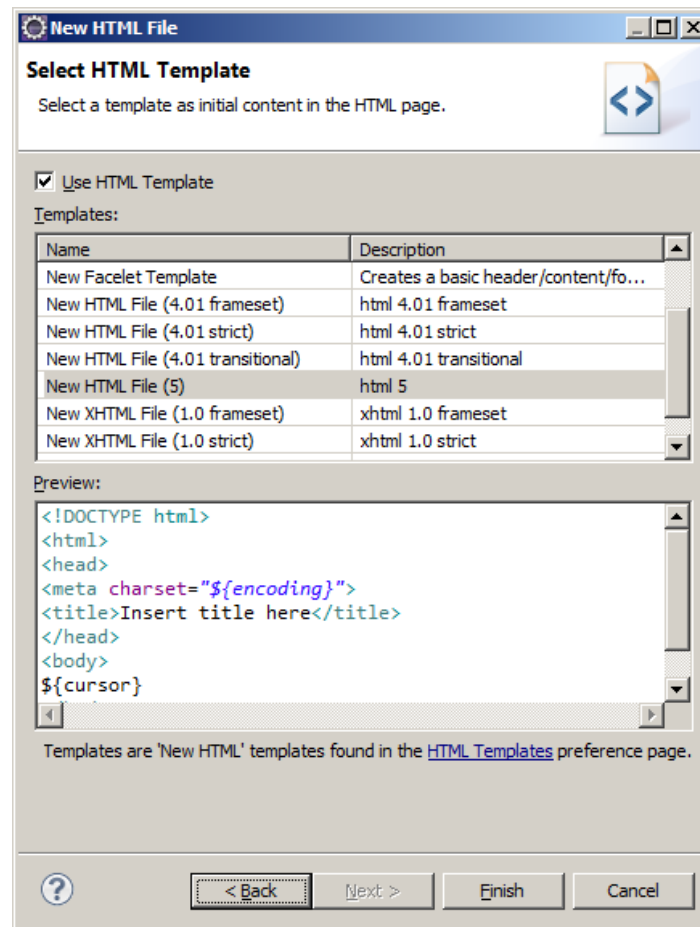
## Informar o nome: **Atividade\_06**



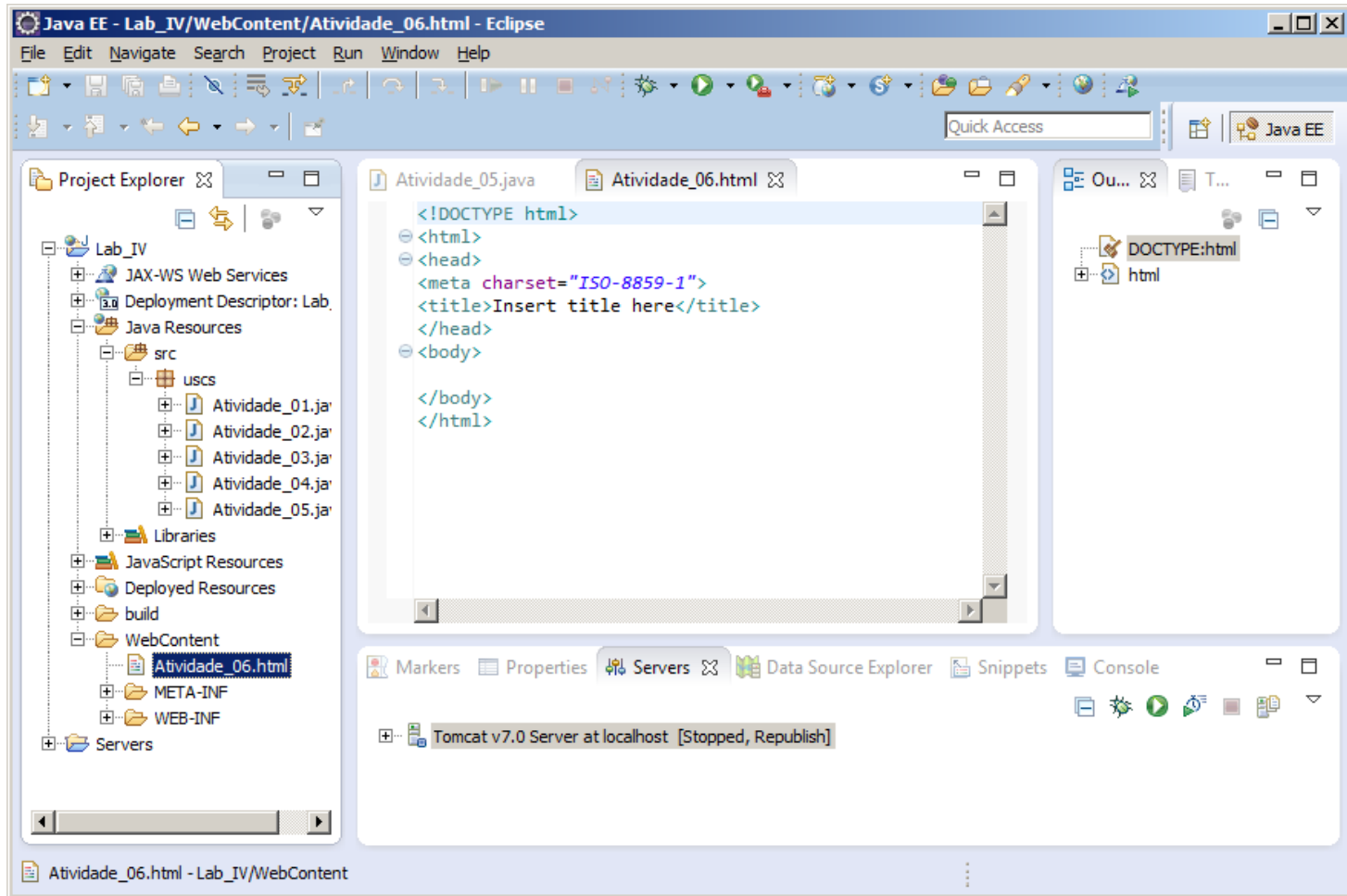
Clicar em **Next**



Clicar em **Finish**



Observe que o arquivo foi criado  
na pasta **WebContent**



# Form HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title> Atividade_06 - Coletando dados de um Form HTML </title>
</head>

<BODY BGCOLOR="#6699FF">

<H1 ALIGN="CENTER">Coletando dados de um FORM HTML</H1>

<FORM ACTION="/Lab_IV/Atividade_06">

    <CENTER> Primeiro Parâmetro:  <INPUT TYPE="TEXT" NAME="param1"><BR> </CENTER>
    <CENTER> Segundo  Parâmetro:  <INPUT TYPE="TEXT" NAME="param2"><BR> </CENTER>

    <HR SIZE=4 WIDTH=50%>

    <CENTER><INPUT TYPE="SUBMIT" VALUE=" ENVIAR "></CENTER>

</FORM>

</BODY>

</html>
```



# Código Servlet

```
package qualit;

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;

/**
 * Servlet implementation class
 */
public class Atividade_06 extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public Atividade_06() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request,
     *                          HttpServletResponse response)
     */
}
```



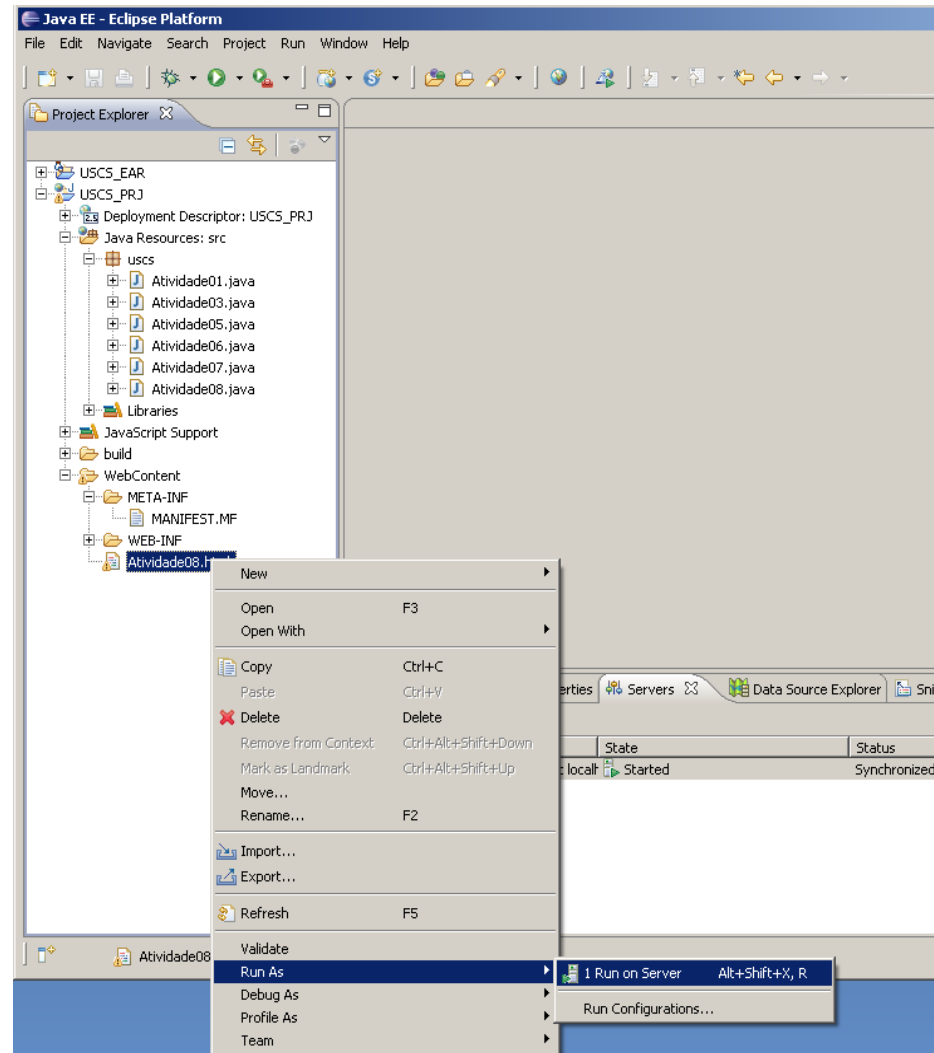


```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
// TODO Auto-generated method stub

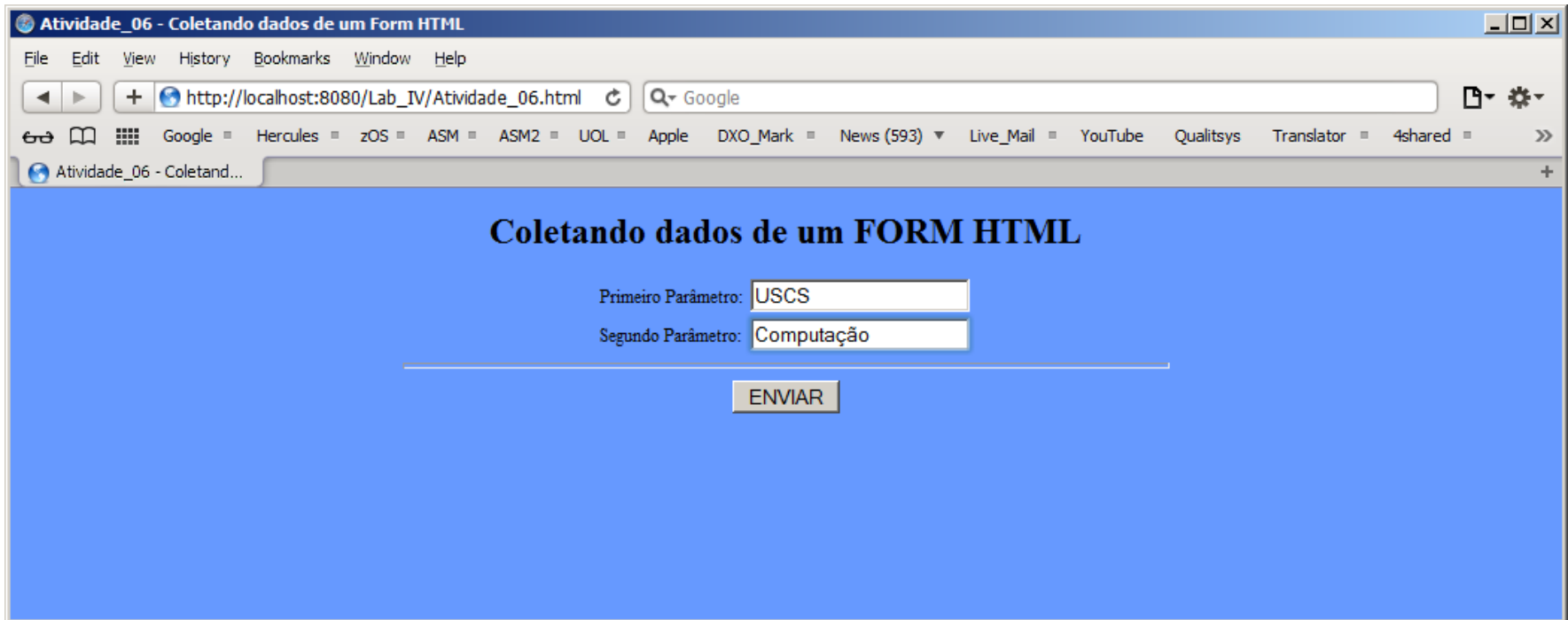
response.setContentType("text/html");
PrintWriter out = response.getWriter();
String Titulo = "Coletando dados de um FORM HTML";
String docType =
    "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 \" +
    \"Transitional//EN\">\n";
out.println(docType +
    "<HTML>\n" +
    "<HEAD><TITLE>" + Titulo +
    "</TITLE></HEAD>\n" +
    "<BODY BGCOLOR=\"#6699FF\">\n" +
    "<H1 ALIGN=\"CENTER\">" + Titulo + "</H1>\n" +
    "<UL>\n" +
    "    <LI><B>param1</B>: "
+ request.getParameter("param1") + "\n" +
    "    <LI><B>param2</B>: "
+ request.getParameter("param2") + "\n" +
    "</UL>\n" +
    "</BODY></HTML>");
}
```



# Processando o FORM



[http://localhost:8080/Lab\\_IV/Atividade\\_06.html](http://localhost:8080/Lab_IV/Atividade_06.html)



The screenshot shows a web browser window with the title "Atividade\_06 - Coletando dados de um Form HTML". The address bar displays the URL "http://localhost:8080/Lab\_IV/Atividade\_06.html". The browser's menu bar includes File, Edit, View, History, Bookmarks, Window, and Help. The toolbar shows navigation buttons, a search bar with "Google", and a list of bookmarks including Google, Hercules, zOS, ASM, ASM2, UOL, Apple, DXO\_Mark, News (593), Live\_Mail, YouTube, Qualitsys, Translator, and 4shared. The main content area has a blue background and contains the following form:

**Coletando dados de um FORM HTML**

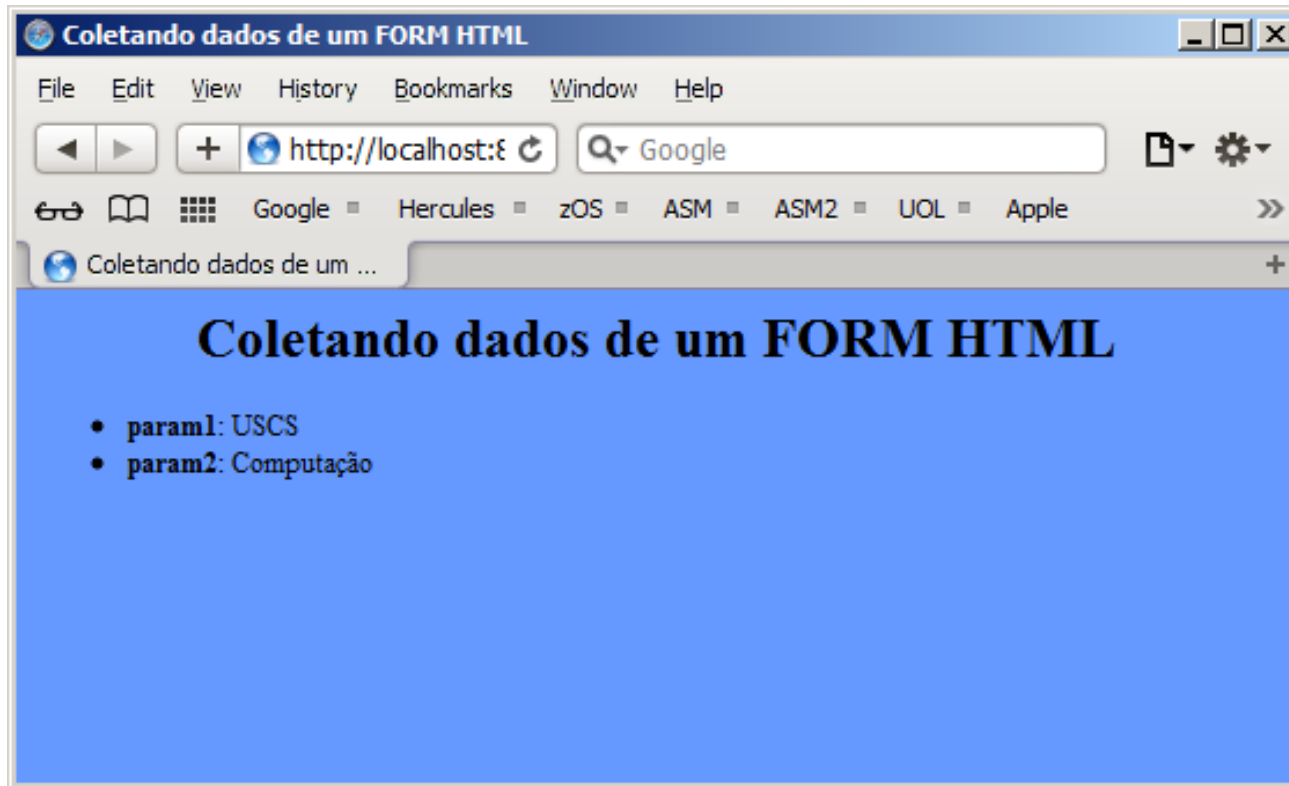
Primeiro Parâmetro:

Segundo Parâmetro:

---

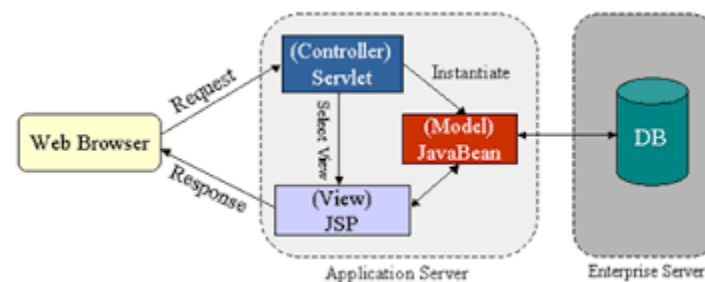


## Response gerada pelo Servlet



## Método getParameterNames()

- ⊕ Método da classe **HttpServletRequest**;
- ⊕ Retorna um objeto do tipo **Enumeration** que contém os nomes dos parâmetros em uma ordem não especificada;
- ⊕ O teste dos elementos presentes no objeto retornado (**Enumeration**) pode ser feito pelo método **hasMoreElements()**;
- ⊕ O próximo elemento a ser retornado na lista pode ser obtido pelo método **nextElement()** que retorna objeto do tipo **Object**.



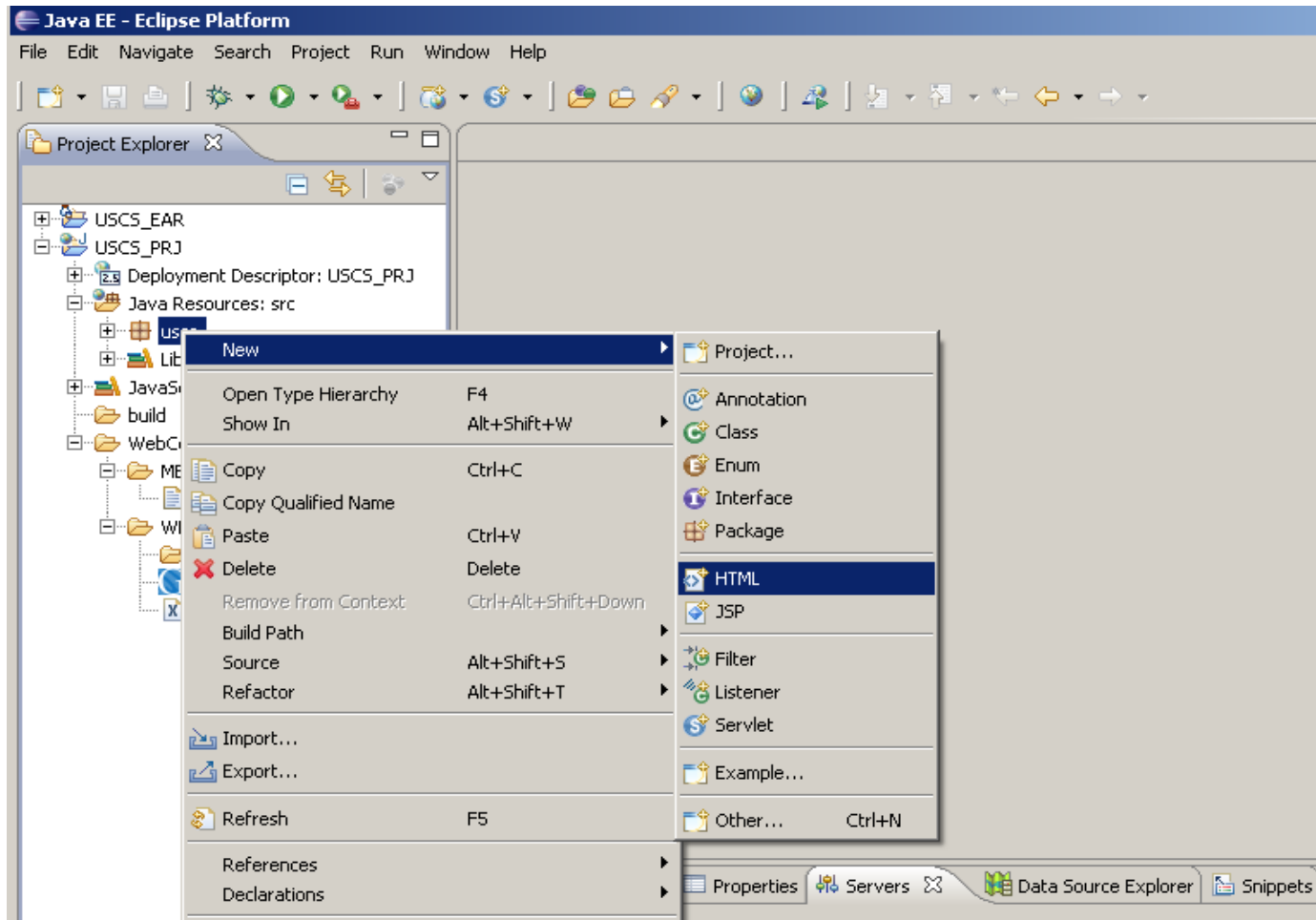
## Atividade – 7

- Escrever um **servlet** (**Atividade\_07**) que efetua leitura de todos os nomes de parâmetros que são enviados e os armazenar em uma tabela;
- Utilizar o método **GET** no formulário **HTML**.

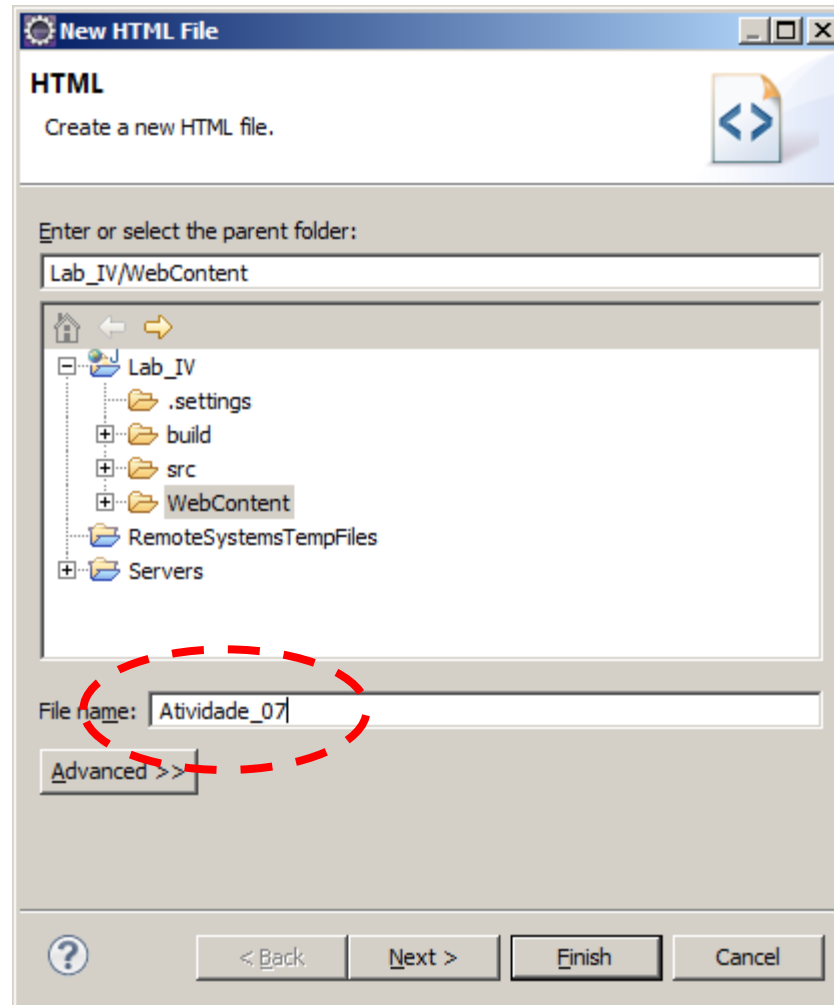
**Obs. Utilizar o ambiente Eclipse/Tomcat**



## Criando HTML sob Eclipse



Informar o nome: **Atividade\_07**

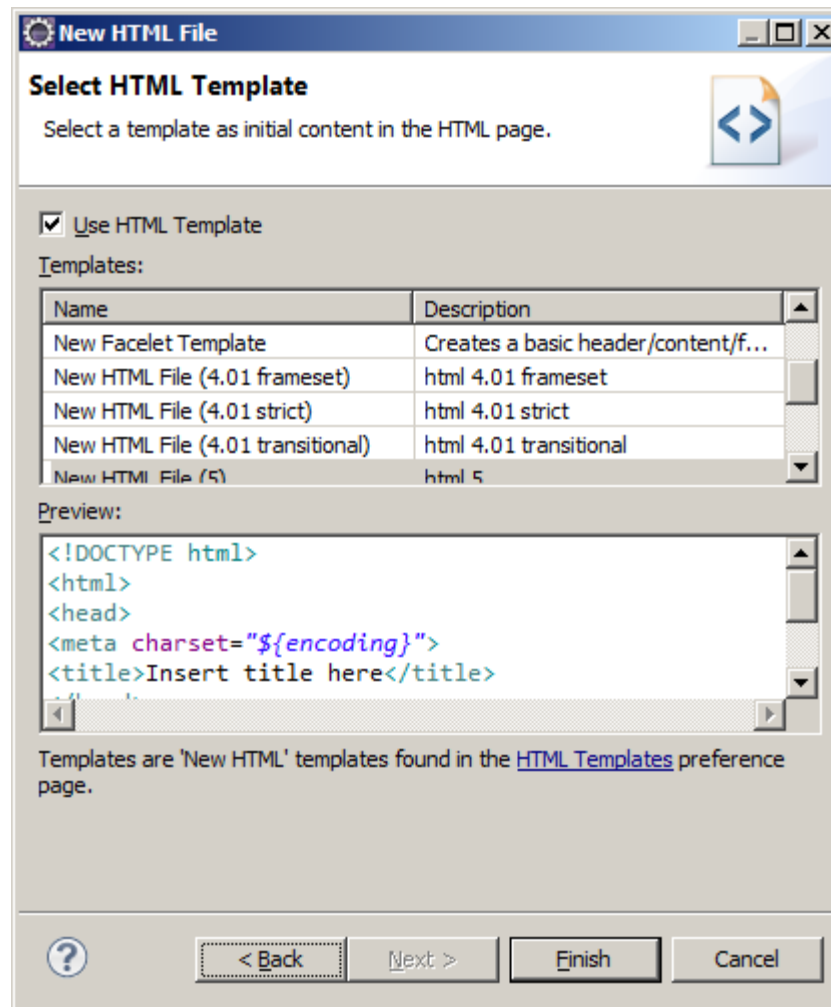


Clicar em **Next**

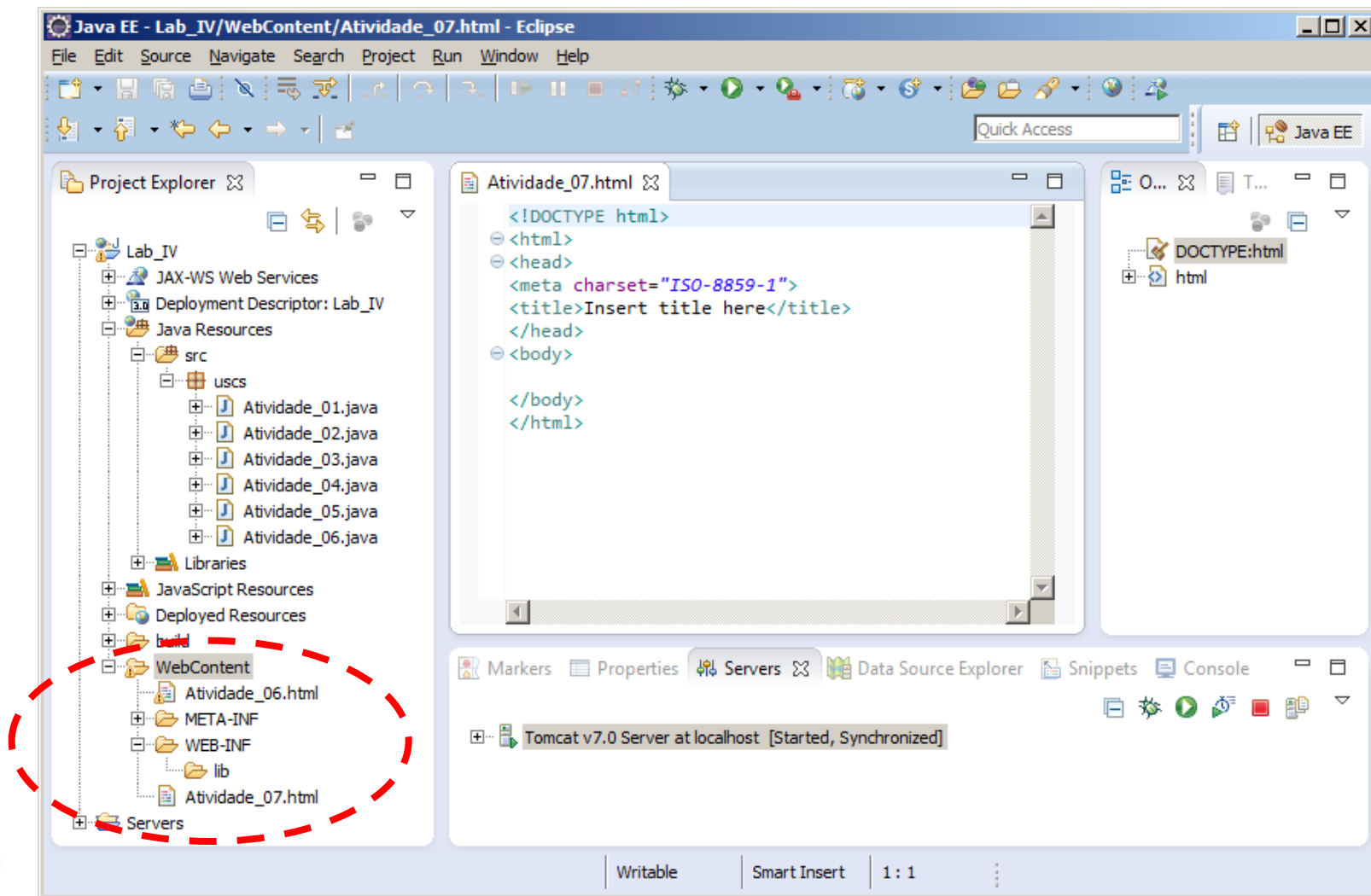




## Clicar em **Finish**



Observe que o arquivo foi criado na pasta **WebContent**



# Form HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<HTML><HEAD><TITLE>Coletando dados de um FORM com Método GET</TITLE></HEAD>
<BODY BGCOLOR="cccccc">
<H1 ALIGN="CENTER">Coletando dados de um FORM com Método GET</H1>

<FORM ACTION="/Lab_IV/Atividade_07">
<HR>
  Código do Produto: <INPUT TYPE="TEXT" NAME="ItemProduto"><BR>
  Descrição: <INPUT TYPE="TEXT" NAME="Descricao"><BR>
  Preço Unitário: <INPUT TYPE="TEXT" NAME="PrecoUnitario"><BR>
<HR>
  Primeiro Nome: <INPUT TYPE="TEXT" NAME="PrimeiroNome"><BR>
  Último Nome: <INPUT TYPE="TEXT" NAME="Sobrenome"><BR>
<HR>
  Endereço de Entrega:
  <TEXTAREA NAME="Endereco" ROWS=3 COLS=40></TEXTAREA><BR>
<HR>
  Cartão de Crédito:<BR>
    <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="Visa">Visa<BR>
    <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="MasterCard">MasterCard<BR>
    <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="Amex">American Express<BR>
  Número do Cartão de Crédito:
    <INPUT TYPE="PASSWORD" NAME="NumeroCartaoCredito"><BR>
    <CENTER><INPUT TYPE="SUBMIT" VALUE="Submissão do Pedido"></CENTER>
</FORM>
</BODY> </HTML>
```



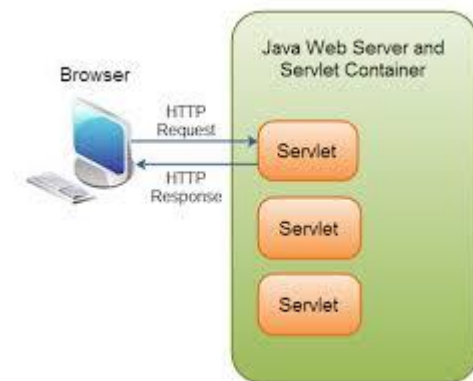
```
package qualit;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.Enumeration;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class Atividade_07
 */
public class Atividade_07 extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public Atividade_07() {
        super();
        // TODO Auto-generated constructor stub
    }
}
```



```
protected void doGet(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {

    // TODO Auto-generated method stub
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String docType = "<!DOCTYPE HTML PUBLIC \":-//W3C//DTD HTML 4.0 " +
    "Transitional//EN\":">\n";
    String Titulo = "Coletando dados de um Form com Método GET";

    out.println(docType +
        "<HTML>\n" +
        "<HEAD><TITLE>" + Titulo + "</TITLE></HEAD>\n" +
        "<BODY BGCOLOR=\"##6699CC\">\n" +
        "<H1 ALIGN=CENTER>" + Titulo + "</H1>\n" +
        "<TABLE BGCOLOR=\"33CCFF\" BORDER=1 ALIGN=CENTER>\n" +
        "<TR BGCOLOR=\"#CC9900\">\n" +
        "<TH>Nome do Parâmetro <TH>Valor do Parâmetro");

    Enumeration ListaNomesParametros = request.getParameterNames();
```



```
while(ListaNomesParametros.hasMoreElements()) {

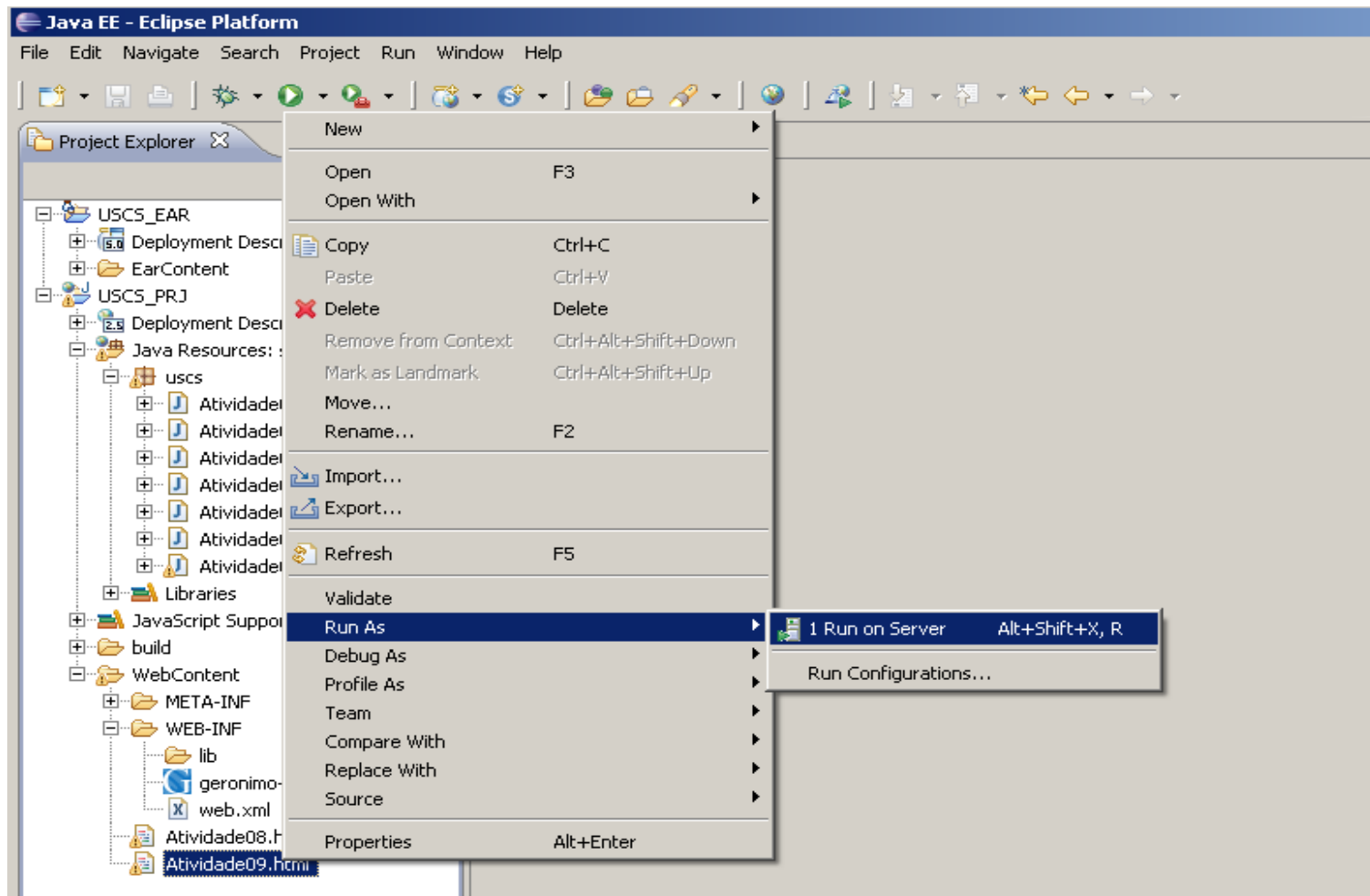
    String NomeParametro = (String)ListaNomesParametros.nextElement();
    out.print("<TR><TD>" + NomeParametro + "\n<TD>");

    String ValorParametro = request.getParameter(NomeParametro);
    if (ValorParametro == "" )
        out.print("<FONT COLOR =\"red\"> Valor não informado...</FONT>");
    else
        out.println(ValorParametro);
    }
    out.println("</TABLE>\n</BODY></HTML>");
}

protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
                        throws ServletException, IOException {
// TODO Auto-generated method stub
}
```



## Processando o FORM



[http://localhost:8080/Lab\\_IV/Atividade\\_07.html](http://localhost:8080/Lab_IV/Atividade_07.html)

Coletando dados de um FORM com Método GET

File Edit View History Bookmarks Window Help

◀ ▶ + [http://localhost:8080/Lab\\_IV/Atividade\\_07.html](http://localhost:8080/Lab_IV/Atividade_07.html) 🔍 Google

🔗 📖 📁 Google ▢ Hercules ▢ zOS ▢ ASM ▢ ASM2 ▢ UOL ▢ Apple ▢ DXO\_Mark ▢ News (593) ▾ Live\_Mail ▢ YouTube ▢ Qualitsys ▢ Translator ▢ 4shared ▢ >>

🌐 Coletando dados de um ...

### Coletando dados de um FORM com Método GET

Código do Produto:

Descrição:

Preço Unitário:

Primeiro Nome:

Último Nome:

Endereço de Entrega:

Cartão de Crédito:

☐ Visa

☐ MasterCard

☐ American Express

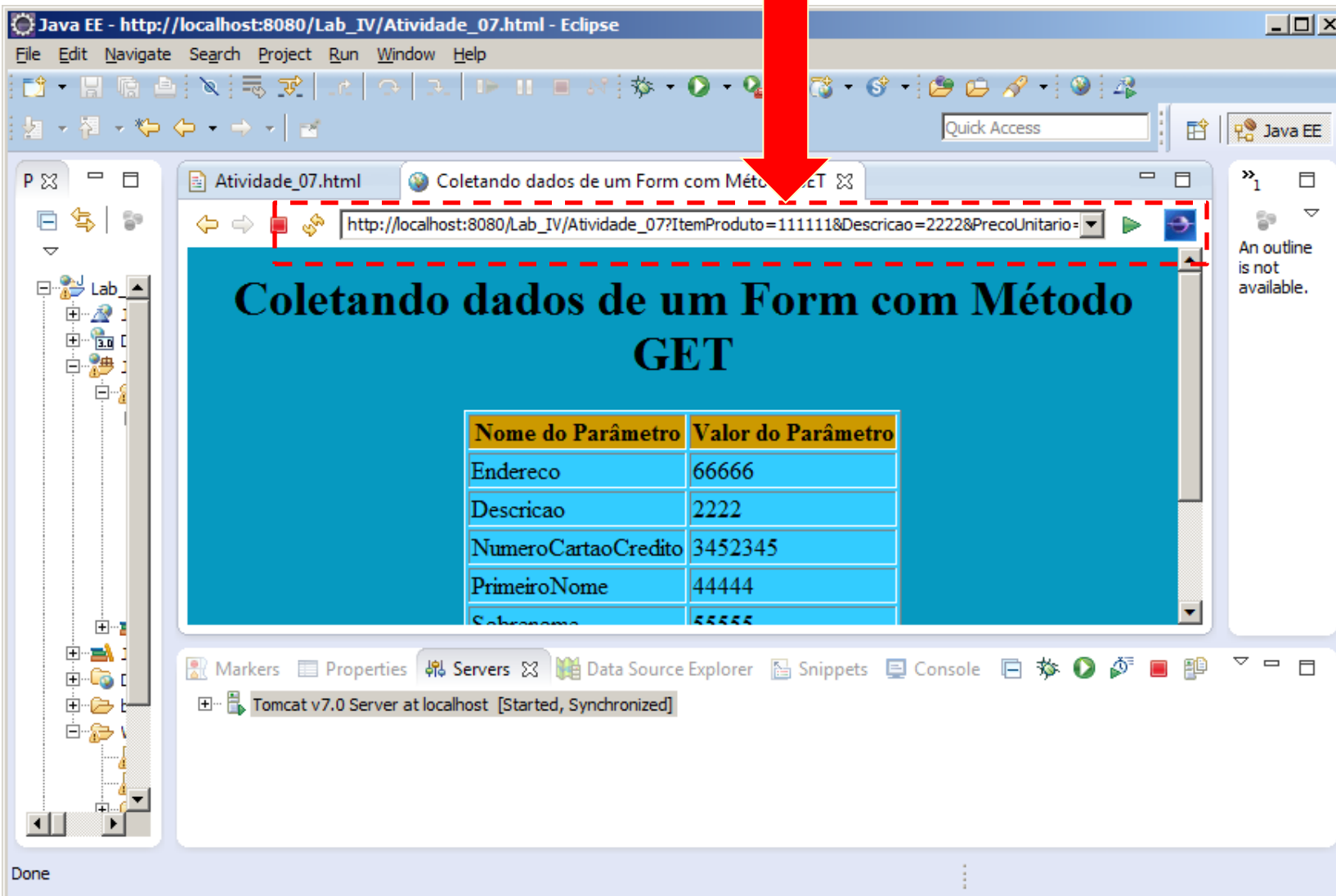
Número do Cartão de Crédito:

Submissão do Pedido





## Response gerada pelo Servlet



The screenshot shows the Eclipse IDE with a web browser window open. The browser address bar displays the URL: `http://localhost:8080/Lab_IV/Atividade_07?ItemProduto=111111&Descricao=2222&PrecoUnitario=`. The main content area of the browser shows the title "Coletando dados de um Form com Método GET" and a table with the following data:

Nome do Parâmetro	Valor do Parâmetro
Endereco	66666
Descricao	2222
NumeroCartaoCredito	3452345
PrimeiroNome	44444
Sobrenome	55555

A red arrow points to the browser window, and a red dashed box highlights the URL and the table. The Eclipse IDE interface includes a project explorer on the left, a console at the bottom showing "Tomcat v7.0 Server at localhost [Started, Synchronized]", and a status bar at the bottom.



# Atividade – 8

- Escrever um servlet (**Atividade\_8**) que efetua leitura de todos os nomes de parâmetros que são enviados e os armazenar em uma tabela.
- Utilizar o método **POST** no formulário HTML.

Obs. Utilizar o ambiente Eclipse/Tomcat



# Form HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<HTML><HEAD><TITLE> Coletando dados de um FORM com Método POST</TITLE></HEAD>
<BODY BGCOLOR="cccccc">
<H1 ALIGN="CENTER">Coletando dados de um FORM com Método POST</H1>

<FORM ACTION="/Lab_IV/Atividade_08" METHOD="POST">
<HR>
  Código do Produto: <INPUT TYPE="TEXT" NAME="ItemProduto"><BR>
  Descrição: <INPUT TYPE="TEXT" NAME="Descricao"><BR>
  Preço Unitário: <INPUT TYPE="TEXT" NAME="PrecoUnitario"><BR>
<HR>
  Primeiro Nome: <INPUT TYPE="TEXT" NAME="PrimeiroNome"><BR>
  Último Nome: <INPUT TYPE="TEXT" NAME="Sobrenome"><BR>
<HR>
  Endereço de Entrega:
  <TEXTAREA NAME="Endereco" ROWS=3 COLS=40></TEXTAREA><BR>
<HR>
  Cartão de Crédito:<BR>
  <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="Visa">Visa<BR>
  <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="MasterCard">MasterCard<BR>
  <INPUT TYPE="RADIO" NAME="TipoCartao" VALUE="Amex">American Express<BR>
  Número do Cartão de Crédito:
  <INPUT TYPE="PASSWORD" NAME="NumeroCartaoCredito"><BR>
<CENTER><INPUT TYPE="SUBMIT" VALUE="Submissão do Pedido"></CENTER>
</FORM>
</BODY></HTML>
```



# Código Servlet

```
package qualit;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.Enumeration;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class Servlet06
 */
public class Atividade_08 extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public Atividade_08() {
        super();
        // TODO Auto-generated constructor stub
    }
```



# Código Servlet

```
protected void doGet(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {

    // TODO Auto-generated method stub
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String docType = "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 \" +
    \"Transitional//EN\">\n";
    String Titulo = "Coletando dados de um Form com Método POST";

    out.println(docType +
        "<HTML>\n" +
        "<HEAD><TITLE>" + Titulo + "</TITLE></HEAD>\n" +
        "<BODY BGCOLOR=\"##6699CC\">\n" +
        "<H1 ALIGN=CENTER>" + Titulo + "</H1>\n" +
        "<TABLE BGCOLOR=\"33CCFF\" BORDER=1 ALIGN=CENTER>\n" +
        "<TR BGCOLOR=\"#CC9900\">\n" +
        "<TH>Nome do Parâmetro <TH>Valor do Parâmetro");

    Enumeration ListaNomesParametros = request.getParameterNames();
```



## Código Servlet

```
while(ListaNomesParametros.hasMoreElements()) {

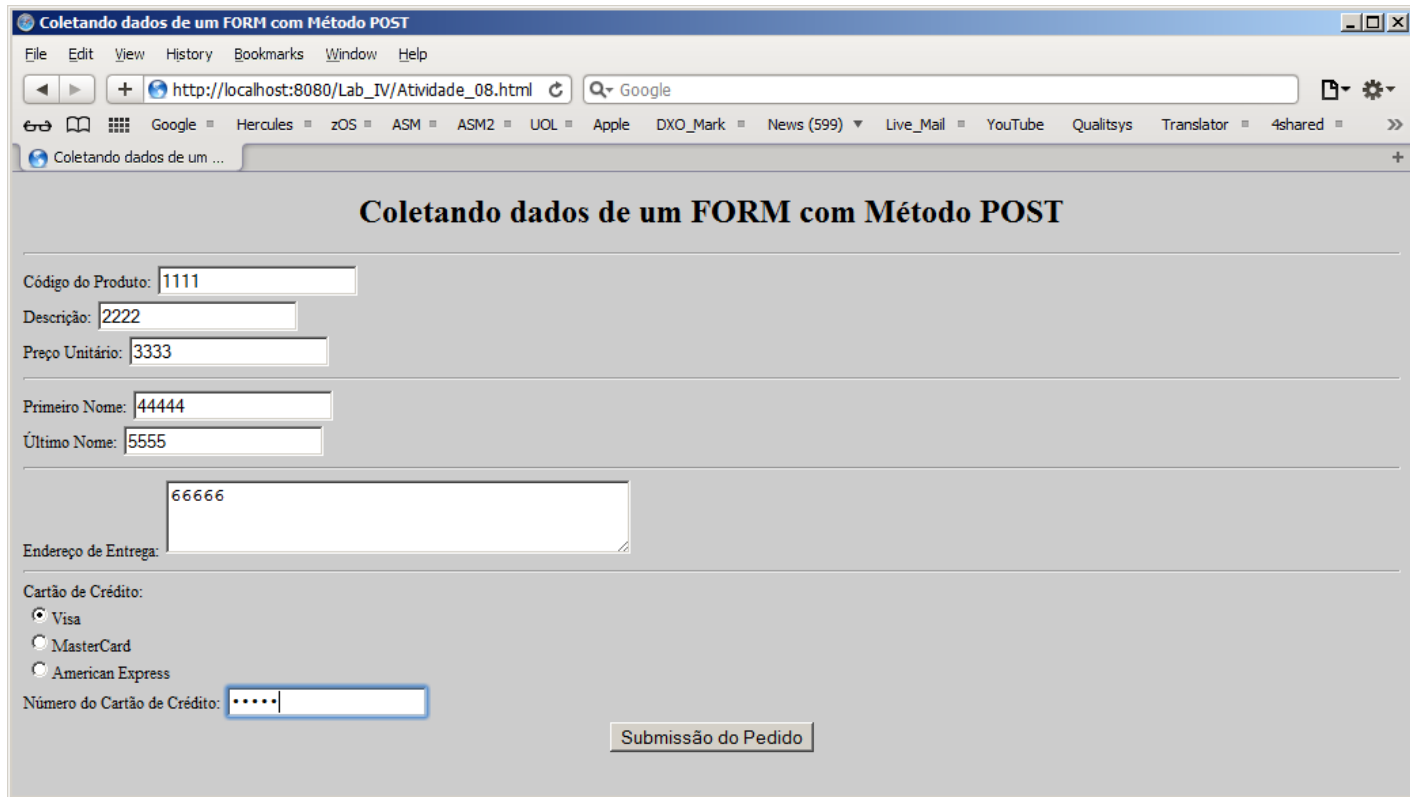
    String NomeParametro = (String)ListaNomesParametros.nextElement();
    out.print("<TR><TD>" + NomeParametro + "\n<TD>");

    String ValorParametro = request.getParameter(NomeParametro);
    if (ValorParametro == "" )
        out.print("<FONT COLOR =\"red\"> Valor não informado...</FONT>");
    else
        out.println(ValorParametro);
    }
    out.println("</TABLE>\n</BODY></HTML>");
}

protected void doPost(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request,response);
}
```



## Submetendo o form

[http://localhost:8080/Lab\\_IV/Atividade\\_08.html](http://localhost:8080/Lab_IV/Atividade_08.html)

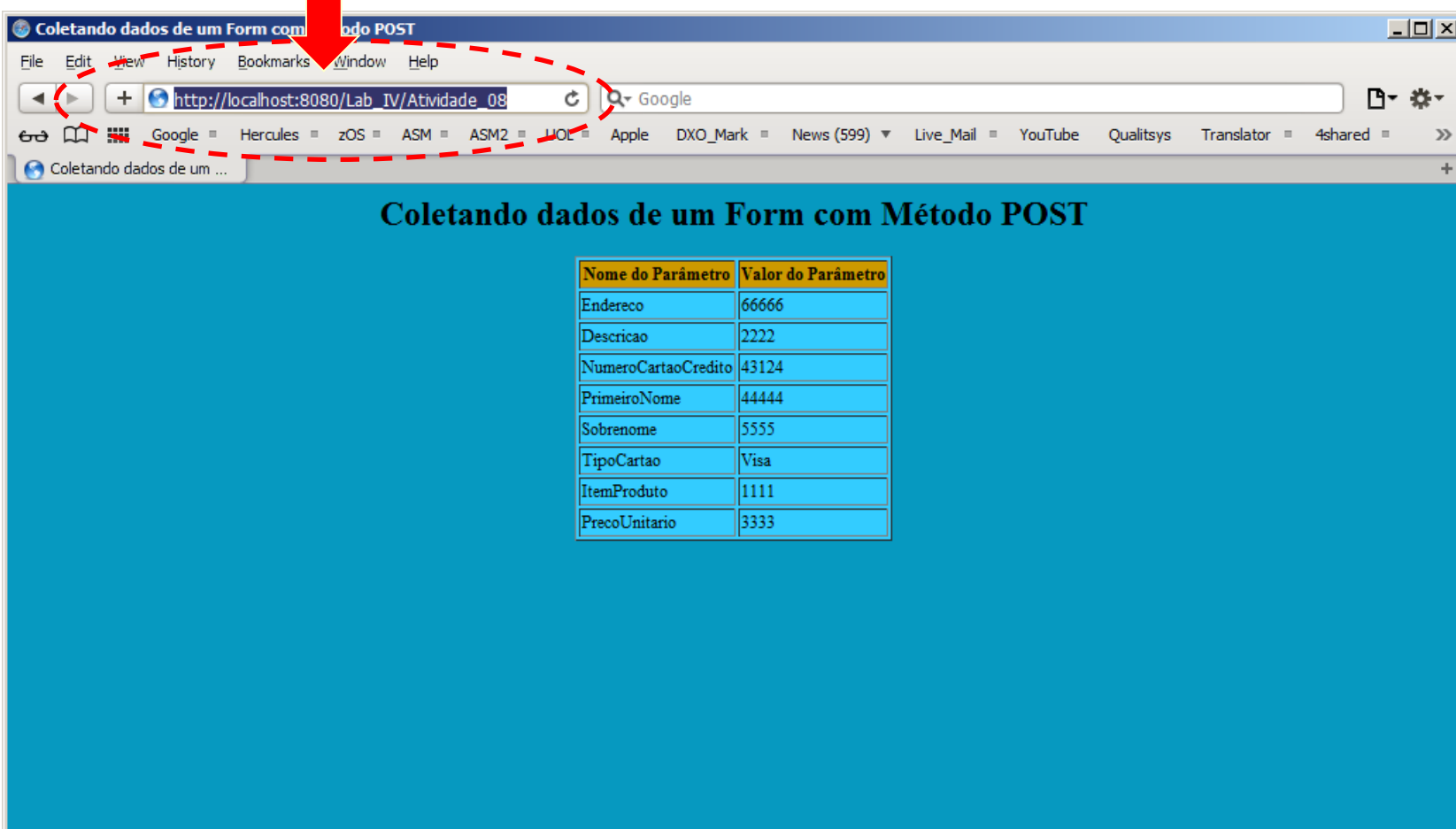
The screenshot shows a web browser window with the title "Coletando dados de um FORM com Método POST". The address bar displays the URL "http://localhost:8080/Lab\_IV/Atividade\_08.html". The browser's menu bar includes File, Edit, View, History, Bookmarks, Window, and Help. The toolbar shows navigation buttons, a search bar with "Google", and a list of bookmarks including Google, Hercules, zOS, ASM, ASM2, UOL, Apple, DXO\_Mark, News (599), Live\_Mail, YouTube, Qualitsys, Tradutor, and 4shared. The page content is titled "Coletando dados de um FORM com Método POST" and contains the following form fields:

- Código do Produto: 1111
- Descrição: 2222
- Preço Unitário: 3333
- Primeiro Nome: 44444
- Último Nome: 5555
- Endereço de Entrega: 66666
- Cartão de Crédito:
  - ☒ Visa
  - ☐ MasterCard
  - ☐ American Express
- Número do Cartão de Crédito: ....

A "Submissão do Pedido" button is located at the bottom right of the form.



## Response gerada pelo Servlet



**Coletando dados de um Form com Método POST**

Nome do Parâmetro	Valor do Parâmetro
Endereco	66666
Descricao	2222
NumeroCartaoCredito	43124
PrimeiroNome	44444
Sobrenome	5555
TipoCartao	Visa
ItemProduto	1111
PrecoUnitario	3333

