

Iniciando no mundo dos Servlets



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 (Internet Explorer® e Mozilla Firefox®).

Ao receber uma requisição, um Servlet pode capturar os parâmetros desta requisição, efetuar qualquer processamento inerente a uma classe Java, e devolver uma página HTML.



Entendendo o HelloWorld

```
import java.io.*; import javax.servlet.*;
import javax.servlet.http.*;
public class HelloWorld extends HttpServlet
{
    public void doGet ( HttpServletRequest request,
                        HttpServletResponse response )
        throws ServletException, IOException
    {
        PrintWriter out;
        response.setContentType("text/html");
        out = response.getWriter();
        out.println( "Hello World!" );
        out.close();
    }
}
```



Entendendo o Hello World

Classes do pacote java.io
utilizadas para produzir
a saída do servlet

Pacote com implementação das
interfaces e classes de Servlet

Implementação de Servlet
para funcionar com HTTP

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

```
public class HelloWorld extends HttpServlet
{
    ....
}
```

Servlets HTTP devem herdar
dessa classe



Entendendo o Hello World

Método chamado quando um usuário faz uma requisição

```
public void doGet ( HttpServletRequest request,  
                  HttpServletResponse response )  
    throws ServletException, IOException  
{  
    ...  
}
```

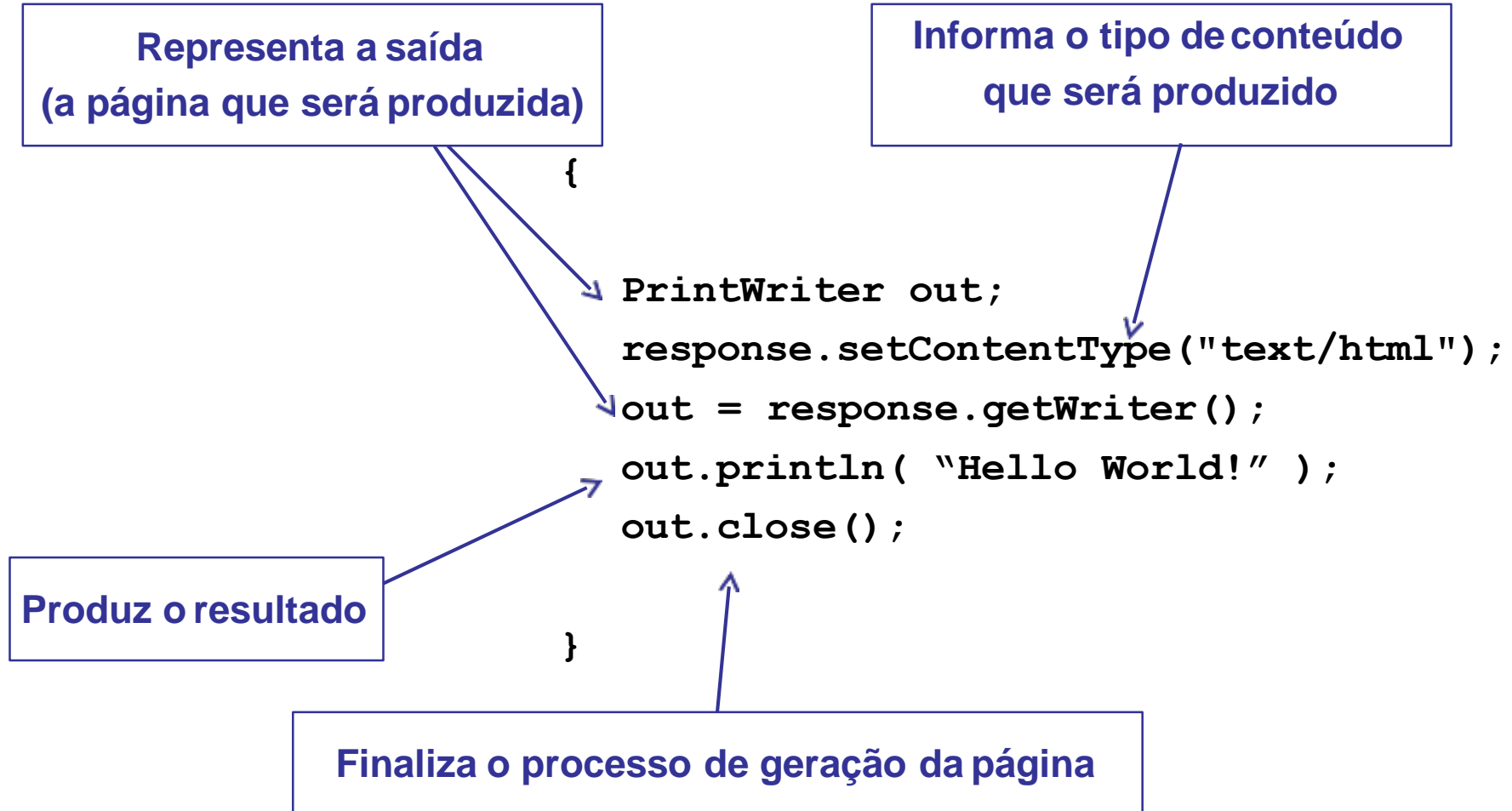
Exceção geral disparada quando o servlet tem algum problema

Exceção disparada quando problemas na produção da página acontecem

Encapsula os dados relativos a requisição (request) e o resultado que será produzido (response)



Entendendo o Hello World



Vantagens dos Servlets

- **Portabilidade:** A aplicação desenvolvida pode ser implementada em diversas plataformas, como por exemplo: Windows, Unix, Linux e MAC.
- **Facilidade de programação:** A programação é orientada a objetos.
- **Flexibilidade.**



Como rodar Servlets?

- Para rodar Servlets é necessário instalar um servidor web que implemente a especificação de Servlets;
- Pelo fato de Servlet ser uma especificação, e não uma implementação, o código escrito em um servidor de um fornecedor deverá funcionar sem alterações em outro servidor de outro fornecedor.



Como rodar Servlets?

- A especificação da API de Servlets está atualmente na versão 2.4;
- Existem dezenas de fornecedores e projetos open source que disponibilizam servidores compatíveis com a especificação de Servlets;
- O software servidor é conhecido como container.



Containers

Alguns containers disponíveis:

	SUN Microsystems Atualmente conhecido como SUN One
	A solução IBM é conhecida como WebSphere
	BEA WebLogic Líder do mercado em soluções pagas
	JRUN – Solução da Macromedia Boa integração com seus produtos
	Borland Application Server





Ciclo de vida de um Servlet



Conversor Monetário

```
1 <form action="/appConversorMoeda/ConversorMoedaServlet" method="post">
2   <h3>
3     Converter o valor de
4     <input type="text" name="txtValor" value="" />
5   </h3>
6   <h3> Da moeda
7     <select name="lstDe">
8       <option value="1" selected>Real</option>
9       <option value="2">Dólar</option>
10      <option value="3">Euro</option>
11    </select>
12  </h3>
13  <h3> Para a moeda
14    <select name="lstPara">
15      <option value="1">Real</option>
16      <option value="2" selected>Dólar</option>
17      <option value="3">Euro</option>
18    </select>
19  </h3>
20  <input type="submit" value="converter" name="btnConverter" />
21 </form>
```



```
1  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
2                                  throws ServletException, IOException{
3
4      String strValor = request.getParameter("txtValor");
5
6      String strDe = request.getParameter("lstDe");
7      String strPara = request.getParameter("lstPara");
8
9      String resultado = Converter(strValor, strDe, strPara);
10
11     response.setContentType("text/html;charset=iso-8859-1");
12     PrintWriter out = response.getWriter();
13     try {
14         out.println("<html>");
15         out.println("<head><title>Conversor Moeda </title> </head>");
16         out.println("<body>");
17         out.println("<h1> Resultado da Conversão </h1>");
18         out.println(resultado);
19         out.println("<br/>");
20         out.println("<a href='index.jsp'>Voltar ao conversor </a>");
21         out.println("</body>");
22         out.println("</html>");
23     } finally {
24         out.close();
25     }
```



```
1  protected String Converter(String strValor, String strDe, String strPara){
2
3      double vlDolar = 2.31;  // cotacao do dolar
4      double vlEuro  = 3.14;  // cotacao do euro
5
6      DecimalFormat df = new DecimalFormat("0.00");
7      String saida = "";
8      strValor = strValor.replace(",", ".");
9      //conversao dos dados
10     double valor = Double.parseDouble(strValor);
11     int moedaDe = Integer.parseInt(strDe);
12     int moedaPara = Integer.parseInt(strPara);
13
14     switch(moedaDe){
15         case 1:
16             if ( moedaPara == 1)
17                 saida = "Escolha moedas diferentes na conversão.";
18             else if (moedaPara == 2){
19                 saida = df.format(valor) + " Real = ";
20                 saida += df.format((valor / vlDolar)) + " Dólar ";
21             }
22             else if (moedaPara == 3){
23                 saida = df.format(valor) + " Real = ";
```



```
24         saida += df.format((valor / vlEuro)) + " Euro ";
25     }
26     break;
27 case 2:
28     if ( moedaPara == 1){
29         saida = df.format(valor) + " Dólar = ";
30         saida += df.format((valor * vlDolar)) + " Real ";
31     }
32     else if (moedaPara == 2){
33         saida = "Escolha moedas diferentes na conversão.";
34     }
35     else if (moedaPara == 3){
36         saida = df.format(valor) + " Dólar = ";
37         saida += df.format(((valor*vlDolar)/vlEuro)) + " Euro ";
38     }
39     break;
```

O programa testará todos os cases
E depois retornará a saída

