

# Servlets & JSP

Prof. Ramon Chiara

# Java

- JSE – Java Standard Edition
- JME – Java Micro Edition
- JEE – Java Enterprise Edition

# Java

- JSE – Java Standard Edition
- JME – Java Micro Edition
- **JEE – Java Enterprise Edition**

# Web

- Cliente – Servidor
- Requisição – Resposta

# Web

- Cliente – Servidor
  - Cliente – Navegador
  - Servidor – HTTP Server
- Requisição – Resposta

# Web

- Cliente – Servidor
  - Cliente – Navegador
  - Servidor – HTTP Server
    - Endereço
    - Porta – Programas
- Requisição – Resposta

# Web

- Cliente – Servidor
  - Cliente – Navegador
  - Servidor – HTTP Server
    - Endereço
    - Porta – Programas
- Requisição – Resposta
  - HTTP
    - GET, POST, ...
    - 200, 404, 500, ...

# Web (estática)

- Requisição
- Resposta

# Web (estática)

- Requisição
  - Arquivo
- Resposta
  - Conteúdo do Arquivo

# Web (estática)

- Requisição
  - Arquivo
- Resposta
  - Conteúdo do Arquivo

**EXEMPLO**

# Web (dinâmica)

- Requisição
  - Programa (+mapeamento)
- Resposta
  - Conteúdo gerado

# Web (dinâmica)

- Requisição
  - Programa (+mapeamento)
- Resposta
  - Conteúdo gerado

**EXEMPLO**

# Servlets

- Container Web
- Classes

# Servlets

- Container Web
  - Contextos – Projetos
- Objetos
  - Classe que herda (é-um) de HttpServlet
  - `@WebServlet("/mapeamento")`
  - `doGet(HttpServletRequest request, HttpServletResponse response)`
  - `doPost(HttpServletRequest request, HttpServletResponse response)`

# Servlets

- Container Web
  - Contextos – Projetos
- Objetos
  - Classe que herda (é-um) de HttpServlet
  - `@WebServlet("/mapeamento")`
  - `doGet(HttpServletRequest request, HttpServletResponse response)`
  - `doPost(HttpServletRequest request, HttpServletResponse response)`

<http://localhost:8080/contexto/mapeamento>

# Servlets

- doGet / doPost
  - Entrada
    - `String param = request.getParameter("param");`
  - Saída
    - `PrintWriter out = response.getWriter();`
    - `response.setContentType("text/html");`

`http://localhost:8080/contexto/mapeamento?param1=valor1&param2=valor2`

# Servlets

- doGet / doPost
  - Entrada
    - `String param = request.getParameter("param");`
  - Saída
    - `PrintWriter out = response.getWriter();`
    - `response.setContentType("text/html");`

# EXEMPLO

# JSP

- JavaServer Pages

# JSP

- Servlet
  - Programa que tem HTML
- JSP
  - HTML que tem programa

# JSP

- Servlet
  - Programa que tem HTML
- JSP
  - HTML que tem programa
    - `<% %>`
    - `<%@ %>`
    - `<%= %>`
    - `<%! %>`
    - `<jsp:... />`
    - `<%-- --%>`

# JSP

- Servlet
  - Programa que tem HTML
- JSP
  - HTML que tem programa
    - `<% %>` scriptlet
    - `<%@ %>` diretiva
    - `<%= %>` expressão
    - `<%! %>` declaração
    - `<jsp:... />` ações
    - `<%-- --%>` comentário

# JSP

- Servlet
  - Programa que tem HTML
- JSP
  - HTML que tem programa
    - `<% %>`           scriptlet           código
    - `<%@ %>`           diretiva           configuração
    - `<%= %>`           expressão           out.print(...)
    - `<%! %>`           declaração        declarações no servlet
    - `<jsp:... />`       ações            ações específicas
    - `<%-- --%>`       comentário        não executa

# JSP

- Compilados para Servlets
  - request
  - response
  - out
  - ...

# JSP

- Compilados para Servlets
  - request
  - response
  - out
  - ...

# EXEMPLO

# Exercícios

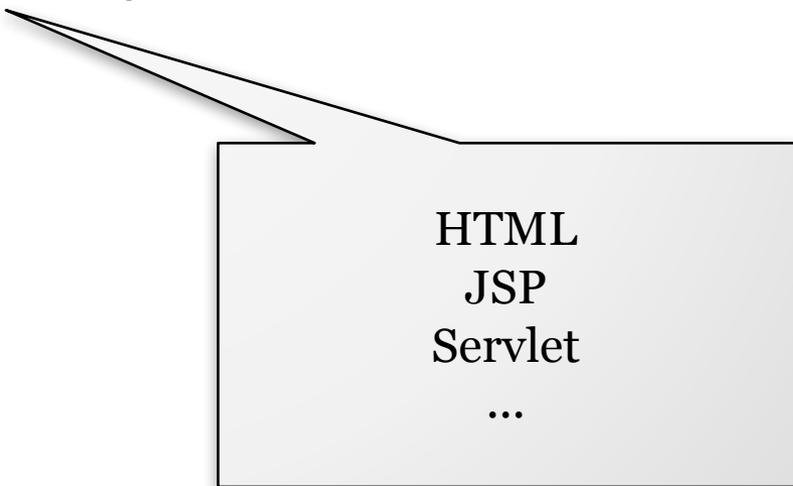
- Faça um servlet que escreva “Olá, mundo”.
- Faça um servlet que recebe um nome e, se ele não estiver vazio, escreva “Olá, nome”. Caso contrário, escreva “Olá, mundo”.
- Faça um servlet que recebe uma temperatura em °C e a converte para °F
- Faça um servlet que monte uma tabela de conversão de temperaturas entre °C e °F (de -20°C a 120°C com passos de 0,5°C)
- Refaça os exercícios anteriores usando JSP

# JSP - Reuso

- `<%@include file="arquivo" %>`
  - Durante a compilação
  - Mais rápido
- `<jsp:include page="pagina" />`
  - Durante a execução
  - Mais flexível (EL)
- **Acesso**
  - `/WEB-INF/`

# JSP - Reuso

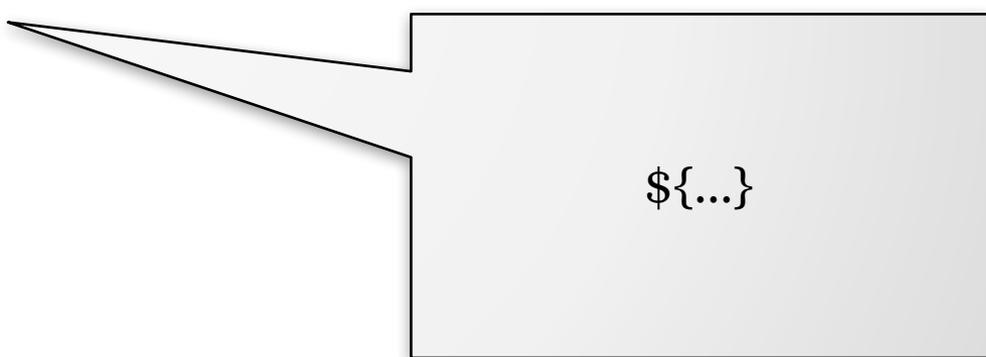
- `<%@include file="arquivo" %>`
  - Durante a compilação
  - Mais rápido
- `<jsp:include page="pagina" />`
  - Durante a execução
  - Mais flexível (EL)
- Acesso
  - `/WEB-INF/`



HTML  
JSP  
Servlet  
...

# JSP - Reuso

- `<%@include file="arquivo" %>`
  - Durante a compilação
  - Mais rápido
- `<jsp:include page="pagina" />`
  - Durante a execução
  - Mais flexível (EL)
- Acesso
  - `/WEB-INF/`



`${...}`

# JSP - Escopos

- application
  - `application.setAttribute("chave", obj); // JSP`
  - `getServletContext().setAttribute("chave", obj); // Servlet`
- session
  - `session.setAttribute("chave", obj); // JSP`
  - `request.getSession().setAttribute("chave", obj); // Servlet`
- request
  - `request.setAttribute("chave", obj); // JSP`
  - `request.setAttribute("chave", obj); // Servlet`
- page
  - `pageContext.setAttribute("chave", obj); // JSP`
  - `// Servlet não tem!`
- Recuperação
  - `escopo.getAttribute("chave");`

# JSP - Redireccionamiento

- Servidor
  - `request.getRequestDispatcher("pagina").forward(request, response);`
- Cliente
  - `response.sendRedirect("pagina");`

# JSP - Expression Language

- $\${var}$ 
  - pageContext
  - request
  - session
  - application
- $\${express\~{a}o}$
- $\${param.nome}$
- $\${pageContext.servletContext.contextPath}$
- $\${obj.propriedade}$
- Operadores
  - empty
  - not
  - ...

# JSP - TagLibs

- JSTL – Java Standard TagLib
  - <https://jstl.java.net/download.html>
  - /WEB-INF/lib

# JSP - TagLibs

- JSTL – Java Standard TagLib
- `<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>`

# JSP - TagLibs

- JSTL – Java Standard TagLib
- `<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>`
- `<c:set var="variavel" value="valor" scope="page"/>`
- `<c:out value="{el}" escapeXml="true"/>`
- `<c:if test="{el}">...</c:if>`
- `<c:choose>`
  - `<c:when test="{el}">...</c:when>`
  - `<c:when test="{el}">...</c:when>`
  - `<c:otherwise>...</c:otherwise>``</c:choose>`
- `<c:forEach items="item1,item2,item3" var="i" varStatus="status">`
  - `{i} / {status.step} / {status.first} / {status.last}``</c:forEach>`

# JSP - TagLibs

- JSTL – Java Standard TagLib
- `<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>`
- `<c:set var="variavel" value="valor" scope="page"/>`
- `<c:out value="{el}" escapeXml="true"/>`
- `<c:if test="{el}">...</c:if>`
- `<c:choose>`
  - `<c:when test="{el}">...</c:when>`
  - `<c:when test="{el}">...</c:when>`
  - `<c:otherwise>...</c:otherwise>``</c:choose>`
- `<c:forEach items="{el}" var="i" varStatus="status">`
  - `{i} / {status.step} / {status.first} / {status.last}``</c:forEach>`

# Exercícios

- Faça um pequeno site com menu para as páginas de resolução dos seguintes exercícios a serem resolvidos usando os tópicos vistos hoje:
  - Faça uma página que recebe o nome, a P1 e a P2 de um aluno e mostra a sua média e se ele passou ou não passou.
  - Faça uma página que recebe uma temperatura em °C e a converte para °F
  - Faça uma página que recebe um número e mostra a tabuada dele.