

## Lab 8 - Estruturas de Controle Java

O objetivo deste laboratório é exercitar o conceito de **Simple Tag Handler** e **Tag File**, introduzido como parte de **JSP 2.0**.

**Simple Tag Handler** apresenta um protocolo de invocação mais simples do que um Tag Handler comum. O descritor da *tag library* mapeia as declarações da *tag library* para a sua implementação básica. Um *simple tag handler* é representado por uma classe que implementa a interface *SimpleTag*.

**Tag File** é um arquivo fonte que contém um fragmento de código JSP que é reusável como uma tag customizada, permitindo que você crie tags customizadas usando a sintaxe JSP. Um *tag file* é traduzido em um *tag handler* e depois compilado.

Você vai criar o seu próprio **Simple Tag Handler** e **Tag File** como parte deste laboratório.

### Exercícios

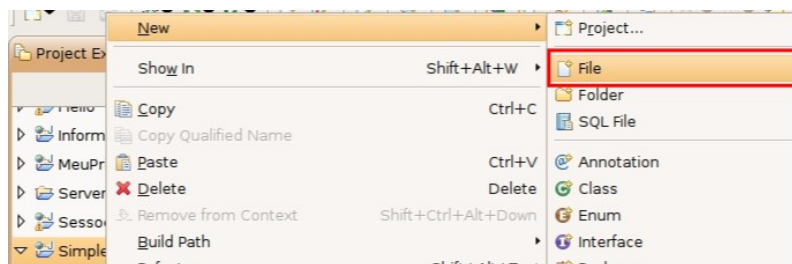
**Exercício 1:** Construir e Executar uma aplicação utilizando **Simple Tag Handler**.

**Exercício 2:** Construir e Executar uma aplicação utilizando **Tag File**.

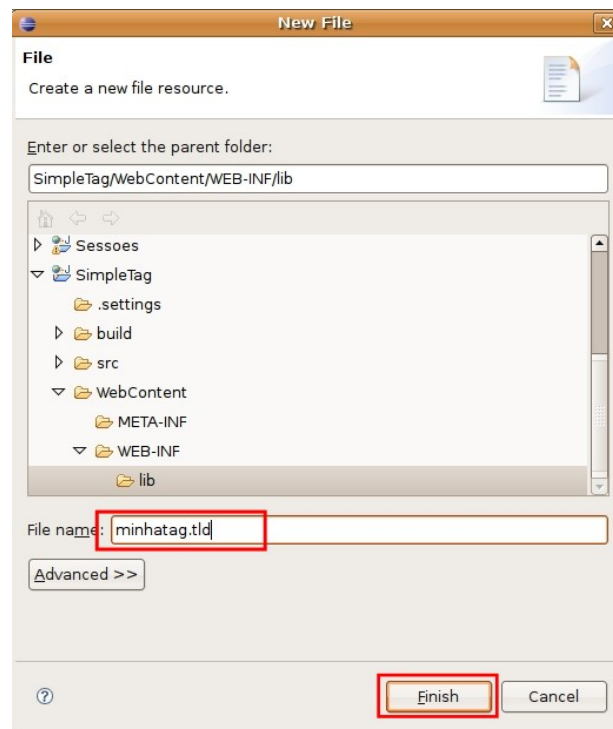
### Exercício 1 - Construir e Executar uma aplicação utilizando Simple Tag Handler

#### 1.1. Criando uma Tag.

1. Crie um novo projeto web chamado **SimpleTag**.
2. Clique com o lado direito do mouse sobre o projeto e selecione **New -> File**.



3. Crie o arquivo chamado **minhatag.tld** no diretório **WEB-INF/lib/** e clique em **Finish**, conforme imagem abaixo.



4. Insira o código abaixo no arquivo.

```
<?xml version="1.0" encoding="UTF-8" ?>
<taglib xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://java.sun.com/xml/ns/j2ee/web-jsptaglibrary_2_0.xsd"
  version="2.0">

  <description>Minha biblioteca SimpleTag</description>
  <tlib-version>1.0</tlib-version>
  <short-name>BibliotecaSimpleTag</short-name>
  <uri>/BibliotecaSimpleTag</uri>
  <tag>
    <description>Exibe Olá Mundo na tela</description>
    <name>olamundo</name>
    <tag-class>simpletag.SimpleTag</tag-class>
    <body-content>empty</body-content>
  </tag>
</taglib>
```

5. Crie a classe **SimpleTag.java** dentro do pacote **simpletag** e insira o código abaixo. Veja que a classe estende de **SimpleTagSupport** que é obrigatório para criar sua própria tag.

```
package simpletag;

import java.io.IOException;

import javax.servlet.jsp.JspException;
import javax.servlet.jsp.tagext.SimpleTagSupport;

public class SimpleTag extends SimpleTagSupport {
  public void doTag() throws JspException, IOException {
    getJspContext().getOut().write( "Olá Mundo!" );
  }
}
```

```
}  
}
```

6. Agora abra o **Deployment Descriptor** com um duplo clique e adicione o código abaixo ao **web.xml**, que é necessário para que a tag que estamos criando seja reconhecida pela aplicação.

```
<jsp-config>  
  <taglib>  
    <taglib-uri>MinhaTag</taglib-uri>  
    <taglib-location>/WEB-INF/lib/minhatag.tld</taglib-location>  
  </taglib>  
</jsp-config>
```

Estamos mapeando a nossa taglib no web.xml

7. Agora crie o arquivo **simpleTagExemplo.jsp** e escreva o código abaixo para utilizar a tag que acabou de ser criada. Analise o código e veja que na primeira linha ele faz a referência ao arquivo da tag através da **taglib**.

```
<%@ taglib prefix="minhatag" uri="/WEB-INF/lib/minhatag.tld" %>  
<html>  
  <head>  
    <title>JSP 2.0 Exemplos - Simple Tag Handler</title>  
  </head>  
  <body>  
    <h1>JSP 2.0 Exemplos - Simple Tag Handler</h1>  
    <hr>  
    <br>  
    <b><u>Resultado:</u></b>  
    <minhatag:olamundo/>  
  </body>  
</html>
```

8. Execute o servidor e acesse o arquivo **simpleTagExemplo.jsp** para visualizar o resultado conforme abaixo.



## 1.2. Criando outra tag

1. Crie uma classe com o nome de **MinhaPropriaSimpleTag.java** dentro do pacote **simpletag** e escreva o código abaixo.

```
package simpletag;  
  
import java.io.IOException;
```

```
import javax.servlet.jsp.JspException;
import javax.servlet.jsp.tagext.SimpleTagSupport;

public class MinhaPropriaSimpleTag extends SimpleTagSupport {
    public void doTag() throws JspException, IOException {
        getJspContext().getOut().write( "Essa é minha nova TAG" );
    }
}
```

2. Adicione sua tag no arquivo **minhatag.tld** conforme mostrado abaixo.

```
<tag>
  <description>Exibe Essa é minha nova Tag</description>
  <name>minhaPropriaTag</name>
  <tag-class>simpletag.MinhaPropriaSimpleTag</tag-class>
  <body-content>empty</body-content>
</tag>
```

3. Modifique o arquivo **simpleTagExemplo.jsp** adicionando o código da nova tag que foi criada para ficar conforme abaixo.

```
<%@ taglib prefix="minhatag" uri="/WEB-INF/lib/minhatag.tld" %>
<html>
  <head>
    <title>JSP 2.0 Exemplos - Simple Tag Handler</title>
  </head>
  <body>
    <h1>JSP 2.0 Exemplos - Simple Tag Handler</h1>
    <hr>
    <br>
    <b><u>Resultado:</u></b>
    <minhatag:olamundo/> <br/>
    <minhatag:minhaPropriaTag/>
  </body>
</html>
```

4. Execute o arquivo **simpleTagExemplo.jsp** e veja o resultado.



## Exercício 2 - Construir e Executar uma aplicação utilizando Tag File

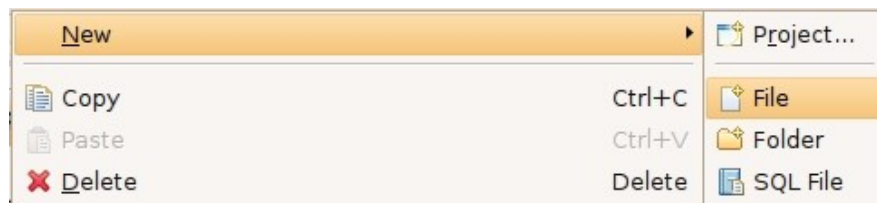
### 2.1. Criando usa Tag File

1. Na pasta **WEB-INF** clique com o lado direito do mouse e selecione **New -> Folder** conforme imagem abaixo.



2. Coloque o nome da pasta de **tags**.

3. Dê um clique com o lado direito do mouse sobre a pasta criada e selecione **New -> File** conforme abaixo.



4. Coloque o nome no arquivo de **ola.tag** e adicione somente o texto abaixo nele.

Ola, sou o arquivo ola.tag.

5. Crie o arquivo **TagFileExemplo.jsp** e adicione o código conforme abaixo, perceba que esta sendo referenciado a **tag** que acabou de ser criado através da **taglib**.

```
<%@ taglib prefix="tags" tagdir="/WEB-INF/tags" %>
<html>
  <head>
    <title>JSP 2.0 Exemplos - Tag File</title>
  </head>
  <body>
    <h1>JSP 2.0 Exemplos - Tag File</h1>
    <hr>
    <br>
    <b><u>Resultado:</u></b>
    <tags:ola/>
    <br>
  </body>
</html>
```

6. Execute o arquivo **TagFileExemplo.jsp** e veja o resultado conforme imagem abaixo.



7. Modifique o arquivo para que saia a mensagem conforme imagem abaixo.



## 2.1. Criando sua Tag File

1. Crie na pasta **tags** o arquivo **painel.tag** e insira o código abaixo. Analise o código e veja que foram declarados alguns atributos no início do arquivo, ao contrário do outro exemplo, e estes atributos estão sendo utilizados no meio do código.

```
<%@ attribute name="cor" %>
<%@ attribute name="corfundo" %>
<%@ attribute name="titulo" %>
<table border="1" bgcolor="${cor}">
  <tr>
    <td><b>${titulo}</b></td>
  </tr>
  <tr>
    <td bgcolor="${corfundo}">
      <jsp:doBody/>
    </td>
  </tr>
</table>
```

2. Crie agora um arquivo **painel.jsp** e escreva o código abaixo para utilizar a tag que acabou de ser criada. Analise o código e veja que no momento em que a **tag painel** está sendo referenciada, os atributos que foram declarados em seu corpo estão sendo passados os valores como propriedades da tag.

```
<%@ taglib prefix="tags" tagdir="/WEB-INF/tags" %>
<html>
  <head>
    <title>Exemplo JSP 2.0 - Painéis usando Tag File</title>
  </head>
```

```
<body>
  <h1>Exemplo JSP 2.0 - Painéis usando Tag File</h1>
  <hr>

  <table border="0">
    <tr valign="top">
      <td>

        <tags:painel cor="#ff8080" corfundo="#ffc0c0" titulo="Painel 01">
          Primeiro Painel<br/>
        </tags:painel>

      </td>
      <td>

        <tags:painel cor="#80ff80" corfundo="#c0ffc0" titulo="Painel 02">
          Segundo Painel.<br/>
          Segundo Painel.<br/>
          Segundo Painel.<br/>
          Segundo Painel.<br/>
        </tags:painel>

      </td>
      <td>

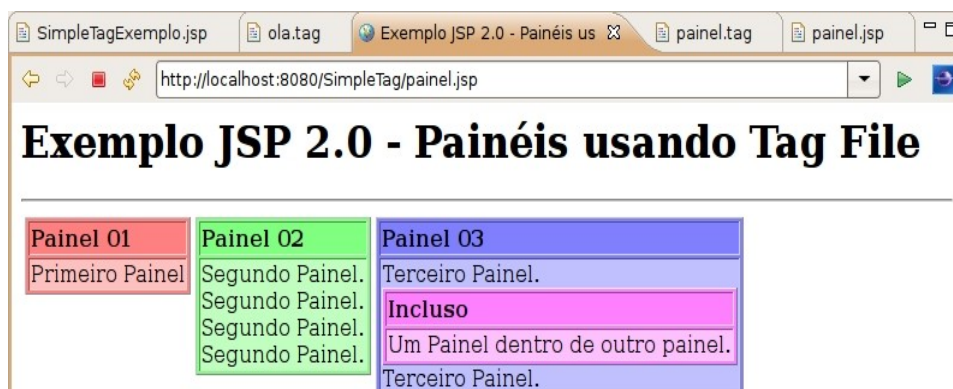
        <tags:painel cor="#8080ff" corfundo="#c0c0ff" titulo="Painel 03">
          Terceiro Painel.<br/>
          <tags:painel cor="#ff80ff" corfundo="#ffc0ff" titulo="Incluso">
            Um Painel dentro de outro painel.
          </tags:painel>
          Terceiro Painel.<br/>
        </tags:painel>

      </td>
    </tr>
  </table>

</body>

</html>
```

3. Execute o arquivo **painel.jsp** e veja o resultado conforme abaixo. Note que quando convocamos a página **painel.jsp** se utiliza **<tags:painel ... >**, passando os valores de 3 parâmetros: *título*, *corfundo* e *cor*. Note que o nome do arquivo deve ser **painel.tag** e deve estar dentro da pasta especificada no atributo **tagdir** de **@taglib**. Note também que **painel.tag** recebe os 3 parâmetros ( *cor*, *corfundo*, *título* ) na chamada da página **painel.jsp**.



4. Modifique o arquivo **painel.jsp** adicionando um quarto painel dentro da tabela, o código deve ficar conforme abaixo.

```
<%@ taglib prefix="tags" tagdir="/WEB-INF/tags" %>
<html>
  <head>
    <title>Exemplo JSP 2.0 - Painéis usando Tag File</title>
  </head>
  <body>
    <h1>Exemplo JSP 2.0 - Painéis usando Tag File</h1>
    <hr>
    <table border="0">
      <tr valign="top">
        <td>
          <tags:painel cor="#ff8080" corfundo="#ffc0c0" titulo="Painel 01">
            Primeiro Painel<br/>
          </tags:painel>
        </td>
        <td>
          <tags:painel cor="#80ff80" corfundo="#c0ffc0" titulo="Painel 02">
            Segundo Painel.<br/>
            Segundo Painel.<br/>
            Segundo Painel.<br/>
            Segundo Painel.<br/>
          </tags:painel>
        </td>
        <td>
          <tags:painel cor="#8080ff" corfundo="#c0c0ff" titulo="Painel 03">
            Terceiro Painel.<br/>
            <tags:painel cor="#ff80ff" corfundo="#ffc0ff" titulo="Incluso">
              Um Painel dentro de outro painel.
            </tags:painel>
            Terceiro Painel.<br/>
          </tags:painel>
        </td>
        <td>
          <tags:painel cor="#8000ff" corfundo="#c0c0ff" titulo="Painel 4">
            Quarto Painel.<br/>
            <tags:painel cor="#ff8000" corfundo="#ffc0ff" titulo="Incluso">
              Um Painel dentro de outro painel.<br/>
              <tags:painel cor="#ff8000" corfundo="#ffc0ff" titulo="incluso no In-
cluso">
                Um Painel dentro de outro painel que está dentro de outro painel.
              </tags:painel>
            </tags:painel>
          </tags:painel>
        </td>
      </tr>
    </table>
  </body>
</html>
```

5. Execute o arquivo **painel.jsp** e veja o resultado conforme abaixo, perceba que foi adicionado mais um elemento na página.





6. Modifique o arquivo **painel.tag** adicionando o atributo **subtítulo** conforme abaixo.

```
<%@ attribute name="cor" %>
<%@ attribute name="corfundo" %>
<%@ attribute name="titulo" %>
<%@ attribute name="subtitulo" %>
<table border="1" bgcolor="{cor}">
  <tr>
    <td><b>${titulo}<u>${subtitulo}</u></b></td>
  </tr>
  <tr>
    <td bgcolor="{corfundo}">
      <jsp:doBody/>
    </td>
  </tr>
</table>
```

7. Modifique agora o arquivo **painel.jsp** colocando na **<tag:painel>** um valor para o atributo **subtítulo** conforme abaixo.

```
<%@ taglib prefix="tags" tagdir="/WEB-INF/tags" %>
<html>
  <head>
    <title>Exemplo JSP 2.0 - Painéis usando Tag File</title>
  </head>
  <body>
    <h1>Exemplo JSP 2.0 - Painéis usando Tag File</h1>
    <hr>
    <table border="0">
      <tr valign="top">
        <td>
          <tags:painel cor="#ff8080" corfundo="#ffc0c0" titulo="Painel 01"
            subtítulo="JSP">
            Primeiro Painel<br/>
          </tags:painel>
        </td>
        <td>
          <tags:painel cor="#80ff80" corfundo="#c0ffc0" titulo="Painel 02"
            subtítulo="2.0">
            Segundo Painel.<br/>
            Segundo Painel.<br/>
            Segundo Painel.<br/>
            Segundo Painel.<br/>
          </tags:painel>
        </td>
      </tr>
    </table>
```

```

</td>
<td>
    <tags:panel cor="#8080ff" corfundo="#c0c0ff" titulo="Painel 03"
        subtitulo="Tag">
        Terceiro Painel.<br/>
        <tags:panel cor="#ff80ff" corfundo="#ffc0ff" titulo="Incluso"
            subtitulo="File">
            Um Painel dentro de outro painel.
        </tags:panel>
        Terceiro Painel.<br/>
    </tags:panel>
</td>
<td>
    <tags:panel cor="#8000ff" corfundo="#c0c0ff" titulo="Panel 4"
        subtitulo="no">
        Quarto Painel.<br/>
        <tags:panel cor="#ff8000" corfundo="#ffc0ff" titulo="Incluso"
            subtitulo="Painel">
            Um Painel dentro de outro painel.<br/>
            <tags:panel cor="#ff8000" corfundo="#ffc0ff" titulo="incluso no Inclusos"
                subtitulo="!!!">
                Um Painel dentro de outro painel que está dentro de outro painel. <br/>
            </tags:panel>
        </tags:panel>
        Quarto Painel.<br/>
    </tags:panel>
</td>
</tr>
</table>
</body>
</html>

```

12. Execute o arquivo **panel.jsp** e veja o resultado conforme imagem abaixo.



## 2.2. Outro Exemplo de Tag File

1. Crie um arquivo **produtos.tag** dentro da pasta **tags** com o código abaixo. Analise o código e veja qual biblioteca será necessário adicionar ao projeto para que execute corretamente os passos descritos no arquivo da tag.

```

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ attribute name="precoNormal" fragment="true" %>
<%@ attribute name="avenda" fragment="true" %>
<%@ variable name="nome" %>
<%@ variable name="preco" %>
<%@ variable name="precoOriginal" %>
<%@ variable name="precoVenda" %>
<table border="1">

```

```
<tr>
  <td>
    <c:set var="nome" value="Hand-held Color PDA"/>
    <c:set var="preco" value="$298.86"/>
    <jsp:invoke fragment="precoNormal"/>
  </td>
  <td>
    <c:set var="nome" value="4-Pack 150 Watt Light Bulbs"/>
    <c:set var="precoOriginal" value="R$12.98"/>
    <c:set var="precoVenda" value="$12.32"/>
    <jsp:invoke fragment="avenda"/>
  </td>
  <td>
    <c:set var="nome" value="Telefone Celular Digital"/>
    <c:set var="preco" value="R$968.74"/>
    <jsp:invoke fragment="precoNormal"/>
  </td>
  <td>
    <c:set var="nome" value="Piano"/>
    <c:set var="preco" value="R$10,800.00"/>
    <jsp:invoke fragment="precoNormal"/>
  </td>
  <td>
    <c:set var="nome" value="Carro Luxuoso"/>
    <c:set var="precoOriginal" value="R$523,980.00"/>
    <c:set var="precoVenda" value="R$421,070.00"/>
    <jsp:invoke fragment="avenda"/>
  </td>
</tr>
</table>
```

2. Crie um arquivo **produtos.jsp** e escreva o código abaixo.

```
<%@ taglib prefix="tags" tagdir="/WEB-INF/tags" %>
<html>
  <head>
    <title>Exemplos JSP 2.0 - Exibe Produtos Tag File</title>
  </head>
  <body>
    <h1>Exemplo JSP 2.0 - Exibe Produtos Tag File</h1>
    <hr>
    <h2>Produtos</h2>
    <tags:produtos>
      <jsp:attribute name="precoNormal">
        Item: ${nome}<br/>
        Preço: ${preco}
      </jsp:attribute>
      <jsp:attribute name="avenda">
        Item: ${nome}<br/>
        <font color="red"><strike>Era: ${precoOriginal}</strike></font><br/>
        <b>Agora: ${precoVenda}</b>
      </jsp:attribute>
    </tags:produtos>
    <tags:produtos>
      <jsp:attribute name="precoNormal">
        <b>${nome}</b><br/> @ ${preco} ea.
      </jsp:attribute>
      <jsp:attribute name="avenda">
        <b>${nome}</b><br/> @ ${precoVenda} ea. (era: ${precoOriginal})
      </jsp:attribute>
    </tags:produtos>
```

```
</tags:produtos>
</body>
</html>
```

3. Execute o arquivo **produtos.jsp** e veja o resultado conforme abaixo, analise o arquivo **JSP** e o arquivo **tag** para entender como esta sendo montado o conteúdo da página.

produtos.jsp

produtos.tag

Exemplos JSP 2.0 - Exibe Produtos Tag File

http://localhost:8080/SimpleTag/produtos.jsp

## Exemplo JSP 2.0 - Exibe Produtos Tag File

### Produtos

Item: Hand-held Color PDA Preco: \$298.86	Item: 4-Pack 150 Watt Light Bulbs <del>Era: R\$12.98</del> Agora: \$12.32	Item: Telefone Celular Digital Preco: R\$968.74	Item: Piano Preco: R\$10,800.00	Item: Carro Luxuoso <del>Era: R\$523,980.00</del> Agora: R\$421,070.00
Hand-held Color PDA @ \$298.86 ea.	4-Pack 150 Watt Light Bulbs @ \$12.32 ea. (era: R\$12.98)	Telefone Celular Digital @ R\$968.74 ea.	Piano @ R\$10,800.00 ea.	Carro Luxuoso @ R\$421,070.00 ea. (era: R\$523,980.00)