

Taglibs

```
1 <%@page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
2
3 <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
4
5 <%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
6
7 <%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
8
9 <%@taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml"%>
10
11 <%@taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions"%>
```

Core Tag Library

- Contém as tags essenciais, padrão para a execução, por exemplos loops, expressões, etc...Seria o básico do tratamento de entrada e saída, **input and output**.

```
3 <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
```

XML Tag Library

- Usada para acessar os elementos XML. Como o XML é usada em muitos aplicativos da Web, processamento de XML é uma característica importante do JSTL.

```
9 <%@taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml"%>
```

Formatting Tag Library

- Usada para dados , formatação e análise. Algumas dessas marcas vai analisar dados, tais como datas, de forma diferente dependendo da localidade atual.

```
5 <%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
```

Formatting Tag Library

- Usada para dados , formatação e análise. Algumas dessas marcas vai analisar dados, tais como datas, de forma diferente dependendo da localidade atual.

```
5 <%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
```

Functions Tag Library

- Usada para funções, como upperCase entre outras.

```
11 <%@taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions"%>
```

DataBase Tag Library

- São tags que pode ser usadas para acessar banco de dados, usando comandos SQL.

```
7 <%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
```

Exercício apache_jstl

- Criar um projeto web chamado Exercicio18.
- Vamos ver vários exemplos com algumas tag libs.

Exemplo Tag Core

- **Set** cria uma variável chamada context, setando o valor do contexto da aplicação, podendo ser setado em um dos escopos.
- **Out** imprime na tela o conteúdo que foi passado para a variável contexto.

```
17 <c:set var="context" scope="request" value="${pageContext.request.contextPath}"/>
18
19 <c:out value="${context}"/>
```

Exemplo 2 Tag Core

- **Remove**, retira a variável do contexto que ela estiver.
- **Choose** condição que valida se o valor é real.

```
25 <!-- <c:remove var="context"/> -->
26 <br>
27 <c:choose>
28   <c:when test="{context != null}">
29     Contexto disponível.
30   </c:when>
31
32   <c:otherwise>
33     Contexto não disponível.
34   </c:otherwise>
35 </c:choose>
36 <c:remove var="context"/>
```

Exemplo 3 Tag Core

- **forEach**, percorre uma lista passando o valor de cada iteração para a variável **item**.
- **out**, imprime o valor da variável **item**.

```
40 <%  
41 »   String[] letras = {"A", "B", "C", "D", "E", "F", "G", "H"};  
42 »   request.setAttribute("lista", letras);  
43 %>  
44 <c:forEach var="item" items="${lista}">  
45 »   <c:out value="${item}" /><br>  
46 </c:forEach>
```

Exemplo Tag XML

- **set**, cria uma variável criando um valor em formato xml.

```
64 <c:set var="xmltext">
65 <usuarios>
66 <usuario>
67 <nome>João</nome>
68 <senha>1234</senha>
69 </usuario>
70 <usuario>
71 <nome>Maria</nome>
72 <senha>4321</senha>
73 </usuario>
74 </usuarios>
75 </c:set>
```

Exemplo Tag XML

- **parse**, transforma a variável xmltext em uma variavel **com o nome de output**.
- **ForEach**, percorre todo conteúdo da variável selecionada e setada na variavel **item**.

```
77 <x:parse xml="{xmltext}" var="output"/>
78 <ul class="list">
79 <x:forEach select="$output/usuarios/usuario/nome" var="item">
80   <li>Nome Usuario: <x:out select="$item"/></li>
81 </forEach>
82 </ul>
```

Exemplo Tag XML

- **import**, seta o conteúdo do arquivo xml na variavel user.
- **parse**, seta o conteúdo do xml na variável saída.
- **out**, primeiro imprime o conteúdo da senha do usuário na posição 2 do xml. Segundo imprime o nome do usuário na posição 1 do arquivo xml.

```
88 <c:import var="user" url="http://localhost:8080/Exemplo18/usuarios.xml"/>
89 
90 <x:parse xml="{user}" var="saida"/>
91 <b>A senha do segundo usuario</b>:
92 <x:out select="$saida/usuarios/usuario[2]/senha" />
93 <br>
94 <b>O nome do primeiro usuario</b>:
95 <x:out select="$saida/usuarios/usuario[1]/nome" />
```

Exemplo Tag Function

- **replace**, seta na variavel **string2** substituindo o texto **primeira** pela palavra **segunda**.

```
99 <c:set var="string1" value="Esta é a primeira String."/>
100 <c:set var="string2" value="{fn:replace(string1,'primeira','segunda')}" />
101 
102 ${string1}<br>
103 ${string2}
```

Exemplo2 Tag Function

- **containsIgnoreCase**, testa com uma condição booleana se existe o conteúdo teste na variável mensagem, ignorando se está em maiúscula ou minúscula.

```
109 <c:set var="mensagem" value="Mensagem de teste."/>
110 
111 <c:if test="{fn:containsIgnoreCase(mensagem, 'teste')}">
112   <p>palavra teste encontrada.</p>
113 </c:if>
```