


Jakarta EE Web Total


Templates e Tags com Facelets




Tópicos Abordados




- Criando templates e páginas
- Incluindo páginas
- Decorators
- Passando parâmetros para templates
- Tags customizadas



Facelets



- Facelets é a tecnologia de visualização padrão a partir do JSF 2.x
 - Substituiu o JSP, utilizado em versões anteriores do JSF
- Um dos principais usos do facelets é a criação de templates de páginas de uma aplicação JSF



Templates



- Aplicações web são normalmente compostas por várias páginas, que compartilham um modelo comum (template)
- Com facelets, é possível criar estes templates



Criando um Template



- O primeiro passo é criar uma página que vai representar o template

layout.xhtml

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets">
<body>
  <table border="1" width="30%">
    <tr><td align="center">
      <h1><ui:insert name="titulo" /></h1></td>
    </tr>
    <tr><td><ui:insert name="texto" /></td></tr>
  </table>
</body>
</html>
```

Referencia as tags do facelets

Conteúdo que será substituído

Criando uma Página



- Páginas que usam um template devem definir o conteúdo a ser substituído

page.xhtml


```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets">
<body>
  <ui:composition template="/layout.xhtml">
    <ui:define name="titulo">Titulo da Página</ui:define>
    <ui:define name="texto">
      Este é um texto que será utilizado na página
    </ui:define>
  </ui:composition>
</body>
</html>
```

Define a substituição a ser feita nas tags ui:insert do template

Criando uma Página

- Quando uma tag `ui:composition` que referencia um template é encontrada na página, o facelets remove todas as tags externas ao `ui:composition`

```
<ui:composition template="/layout.xhtml">
  <ui:define name="titulo">
    Título da Página
  </ui:define>
  <ui:define name="texto">
    Este é um texto que será utilizado na página
  </ui:define>
</ui:composition>
```




Incluindo Páginas

- Uma página JSF pode incluir outras páginas através da tag `ui:include`

```
<body>
  <h1>
    <ui:include src="/titulo.xhtml" />
  </h1>
</body>
```

Inclui o conteúdo da página `titulo.xhtml` neste local



Incluindo Páginas

```


<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets">

  <body>
    <ui:composition>
      Título da Página
    </ui:composition>
  </body>
</html>

```

Neste caso, não é fornecido um template

Apenas o que está dentro da tag `ui:composition` é incluído



Decorators



- Decorators são utilizados para aplicar um padrão sobre determinado conteúdo

```
<body>  
  <ui:decorate template="/decorator.xhtml">  
    <h:commandButton value="Processar"  
      action="#{bean.processar}" />  
  </ui:decorate>  
</body>
```

Define um template usado para decorar o conteúdo da página



Decorators



```
decorator.xhtml  
  
<body>  
  <ui:composition>  
    <table>  
      <tr>  
        <td bgcolor="#ACACAC"><ui:insert /></td>  
      </tr>  
    </table>  
  </ui:composition>  
</body>
```

Insere o conteúdo a ser decorado



Parâmetros para Templates



- Templates podem receber parâmetros das páginas que os incluem
 - Isto vale para as tags `ui:composition` e `ui:decorate`

```
<ui:decorate template="/decorator.xhtml">  
  <ui:param name="color" value="#CCCCC" />  
  <h:commandButton value="Processar"  
    action="#{bean.processar}" />  
</ui:decorate>
```

O parâmetro `color` é passado para o template



Parâmetros para Templates



```
decorator.xhtml
<h:body>
  <ui:composition>
    <table>
      <tr>
        <td bgcolor="#{color}"><ui:insert /></td>
      </tr>
    </table>
  </ui:composition>
</h:body>
```

Expressão EL para
leitura do parâmetro



Tags Customizadas com Facelets



- O facelets permite a criação de tags customizadas
- Ao utilizar uma tag desse tipo, o conteúdo da página é substituído por outro
- A tag é definida em um arquivo *.xhtml*



Criando o Arquivo da Tag



```
/WEB-INF/tags/foto.xhtml
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:f="http://xmlns.jcp.org/jsf/core"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets">

  <body>
    <ui:composition>
      <center>
        <h:graphicImage library="images" name="#{imagem}" />
        <br />
        <h3>#{legenda}</h3>
      </center>
    </ui:composition>
  </body>
</html>
```



Arquivo de Definição da Tag



- Para que as suas tags sejam reconhecidas pelo JSF, é necessário criar um arquivo de definição de tags
- Este arquivo normalmente fica em */WEB-INF*
- Define um namespace para as tags, bem como nome e local de cada tag criada



Arquivo de Definição da Tag



mytags.taglib.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<facelet-taglib version="2.2"
  xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
    http://xmlns.jcp.org/xml/ns/javaee/web-facelettaglibrary_2_2.xsd">

  <namespace>http://mytags/facelets</namespace>
  <tag>
    <tag-name>foto</tag-name>
    <source>tags/foto.xhtml</source>
  </tag>
</facelet-taglib>
```



Configuração no *web.xml*




- O JSF precisa saber a respeito do arquivo de definição de tags, caso contrário não saberá que ele existe
- A configuração é feita no arquivo *web.xml*

```
<context-param>
  <param-name>javax.faces.FACELETS_LIBRARIES</param-name>
  <param-value>/WEB-INF/mytags.taglib.xml</param-value>
</context-param>
```

Para múltiplos arquivos, a separação é feita com ";"



Utilizando Tags Customizadas




```
<html
  xmlns:my="http://mytags/facelets">
<my:foto
  imagem="duke.png"
  legenda="Duke, o mascote do Java"
/>
```


Referencia a tag pelo namespace e usa o prefixo my

Passa os parâmetros necessários




Duke, o mascote do Java





Softblue



7