

## Criando um template

Nesse exercício vamos criar e aplicar o template utilizado pelas páginas do sistema, igual ao video.

1 - Crie um novo arquivo xhtml com o nome `_template.xhtml` . Ele ficará dentro da pasta `WebContent` do projeto.

2 - O código abaixo é o esqueleto do template que pode ser usado para começar. Cole-o dentro do arquivo

`_template.xhtml` . Repare o novo namespace `xmlns:ui="http://java.sun.com/jsf/facelets"`  
`(http://java.sun.com/jsf/facelets)." :`

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">

  <h:head/>

  <h:body>
    <div id="cabecalho">
      <h:graphicImage library="img" name="logo.png"/>
      <h1></h1>
    </div>

    <div id="conteudo">
    </div>
  </h:body>
</html>
```

3 - Falta indicar as áreas que serão personalizadas nas páginas que utilizarem o template. Utilize a tag `ui:insert` .

Teremos dois `ui:insert` , um dentro da tag `h1` do cabeçalho, pois queremos que o título varie de página para página, e outro dentro da div com id conteúdo. Cada `ui:insert` precisa de um identificador através do atributo `name` . Utilize `titulo` para o primeiro e `conteudo` para o segundo.

4 - Com o template pronto, altere a página `livro.xhtml` para que faça uso dele através da tag `ui:composition` . Não esqueça de importar o namespace do facelets também nesta página.

5 - Redefina o título e o conteúdo do template em `livro.xhtml` . Utilize a tag `ui:define` . Esta tag também possui o atributo `name`, que deve ter o mesmo valor definido no atributo `name` da tag `ui:insert` do template que será sobrescrita.

6 - Teste o resultado.

### Responda

INSERIR CÓDIGO	FORMATAÇÃO
<div>digite seu código aqui</div>	

