|  |
| --- |
|  |

# flowRoot2832.jpg

# Java para Web: Java Server Faces

**Exercícios Propostos**

**Templates e Tags com Facelets**

**Carlos Tosin**

# Histórico

|  |  |
| --- | --- |
| **Data** | **Observação** |
| 29/12/2011 | Criação do documento |

# Sobre a Softblue

Fundada em 2003 na cidade de Curitiba-PR por André Milani e Carlos Tosin, inicialmente para dedicar-se à construção de soluções para web, a **Softblue** rapidamente expandiu seus negócios para a realização de cursos e treinamentos, devido ao fato de seus sócios serem altamente especializados em determinadas áreas de TI, certificados nas tecnologias em que atuam e autores de livros de informática sobre os mesmos temas.

Atualmente, a **Softblue** disponibiliza cursos e treinamentos on-line, permitindo que pessoas do Brasil inteiro possam realizá-los, estudando e aprendendo novas tecnologias, aprimorando seus conhecimentos para o mercado de trabalho, de acordo com sua disponibilidade de horários.

# Exercícios Propostos

## Exercício 1

A partir de uma página XHTML única, separe o layout em partes. Primeiro crie um arquivo para o cabeçalho e outro para o rodapé, e depois faça a importação dos mesmos na página do layout principal através da tag *ui:include*. Utilize também as tags *ui:define*, *ui:insert* e *ui:composition* para proporcionar a divisão do layout.

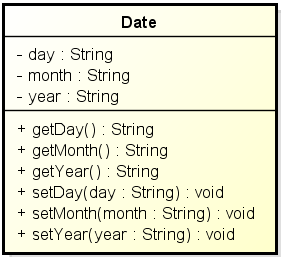
O objetivo deste exercício é exercitar as facilidades de montagem de layout de páginas de aplicações proporcionadas pelo facelets. O layout pode ser de criação própria ou pode ser obtido a partir de sites como o [http://www.freecsstemplates.org](http://www.freecsstemplates.org/). Na resolução deste exercício, o layout usado como base foi obtido deste site e é chamado *blackwhite*. Caso você deseje trabalhar sobre o mesmo layout, ele pode ser obtido diretamente no site mencionado anteriormente ou então juntamente com o código de resolução deste exercício (*blackwhite.zip*), no arquivo ZIP dos exercícios do módulo.

## Exercício 2

Utilizando os recursos do facelets, crie uma tag customizada chamada *data*. Ao ser utilizada, esta tag mostra três caixas de texto, para que o usuário digite o dia, o mês e o ano que compõem esta data. A figura abaixo mostra o resultado visual do uso da tag.



Esta tag deve receber como atributo uma value expression (*dateObj*) que referencia um objeto do tipo *Date*, cuja classe você deverá criar de acordo com o diagrama a seguir.



Por exemplo, suponha que você declare a tag na página XHTML desta forma:

*<my:date dateObj="#{dateBean.date}" />*

Ao fazer isto, você estará instruindo a tag a colocar os valores digitados nos três campos (dia, mês a ano) nos atributos do objeto referenciado por *dateBean.date* (*day*, *month* e *year*, respectivamente).