

Desenvolvimento de Sistemas Orientado a Objetos

Facelets e Backing Beans

2021/01

Facelets e Backing Beans

Entrada/saída

Ações

Ajax

Exercícios

Referências

Facelet

```
<!DOCTYPE html>
<!-- Bibliotecas de tags -->
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:f="http://xmlns.jcp.org/jsf/core"
      xmlns:h="http://xmlns.jcp.org/jsf/html">

  <h:head>
    <title>Facelets / Backing Beans</title>
  </h:head>

  <h:body>
    <h3>Facelets e Backing Beans</h3>
  </h:body>
</html>
```

Backing Bean

```
import javax.enterprise.context.RequestScoped;  
import javax.inject.Named;  
  
@Named  
@RequestScoped  
public class IndexView {  
  
}
```

Entrada/saída

- ▶ *outputText*: É um componente para saída de dados.
- ▶ Por meio do atributo “value”, é vinculado a um método/variável do backing bean.

Entrada/saída

```
@Named
@RequestScoped
public class IndexView {

    private String texto = "Hello facelets!";

    public String getTexto() {
        return texto;
    }

}
```

Entrada/saída

```
...  
<h:body>  
    <h3>Facelets e Backing Beans</h3>  
  
    <h:outputText value="#{indexView.texto}" />  
</h:body>  
...
```

Entrada/saída

- ▶ *inputText*: é um componente para entrada de texto.
- ▶ Por meio do atributo “value”, é vinculado a um método/variável do backing bean.

Entrada/saída

```
@Named
@RequestScoped
public class IndexView {
    ...
    private String mensagem;

    public String getMensagem() {
        return mensagem;
    }

    public void setMensagem(String mensagem) {
        this.mensagem = mensagem;
    }
    ...
}
```

Entrada/saída

```
<h:body>  
  <h3>Facelets e Backing Beans</h3>  
  ...  
  
  Mensagem: <h:inputText value="#{indexView.  
    mensagem}" />  
</h:body>
```

Ações

- ▶ *commandButton*: são componentes utilizados para realizar ações (disparar requisições).
- ▶ A propriedade “action” define o método do backing bean a ser executado.
- ▶ São incluídos em formulários.

Ações

```
@Named
@RequestScoped
public class IndexView {
    ...
    public void atualizaMensagem() {
        texto = String.format("Você digitou: %s",
                               mensagem);
    }
    ...
}
```

Ações

```
<h:body>
...
  <h:form>
    ...

    <h:commandButton value="Enviar"
      action="#{indexView.atualizaMensagem}" />
  </h:form>
</h:body>
```

Ajax

- ▶ É possível configurar requisições Ajax em determinados componentes, por exemplo, os `commandButtons`.
- ▶ A propriedade “execute” indica os elementos que serão processados na execução.
- ▶ A propriedade “update” indica os elementos que serão atualizados após a requisição.

Ajax

```
...  
<h:outputText id="saida"  
    value="#{indexView.texto}" />  
...  
  
<h:commandButton value="Enviar"  
    action="#{indexView.atualizaMensagem}">  
  
    <f:ajax execute="@form" render="saida" />  
</h:commandButton>  
...
```

Exercícios

1. Utilizando os componentes estudados, construa uma página que receba 2 números inteiros e apresente a soma dos números ao usuário. A soma deverá ser realizada em um método do backing bean.

Exercícios

2. Ajuste o exemplo anterior, para que usuário possa escolher uma das quatro operações fundamentais: soma, subtração, multiplicação e divisão.

Exercícios

3. Construa uma página que receba uma string e apresente para o usuário o número de caracteres na string informada.

Exercícios

4. Construa uma página que receba uma data de nascimento no formato dd/mm/aaaa e calcule a idade.

Exercícios

5. Utilizando uma das expressões regulares apresentadas em `https://howtodoinjava.com/java/regex/java-regex-validate-email-address/`, construa uma página que receba um email e informe se o mesmo é válido ou não.

Referências

- ▶ Bond, M., et al. *Aprenda J2EE em 21 dias*. São Paulo, Pearson Education do Brasil, 2003.

- ▶ Deitel, P.; Deitel, H. *Java: como programar*, 8ª edição. São Paulo, Pearson Prentice Hall, 2010.

- ▶ Scholtz, B.; Tijms, A. *The Definitive Guide to JSF in Java EE 8: Building Web Applications with JavaServer Faces*. Berkeley - CA - USA, Apress, 2018.