

Ligação entre XHTML e fonte de dados

Temos a seguinte view:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:h="http://java.sun.com/jsf/html">

  <h:head>
    <title>Argentum Web</title>
  </h:head>
  <h:body>
    <h:outputText value = "Olá JSF!" />
  </h:body>
</html>
```

Aprendemos que o componente `<h:outputText>` é usado para exibir um textos para o usuário. Apesar de burocrático e menos simples do que escrever diretamente no XHTML "Olá JSF!" ele nos permite buscar dados de uma fonte através de uma expression language:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:h="http://java.sun.com/jsf/html">

  <h:head>
    <title>Argentum Web</title>
  </h:head>
  <h:body>
    <h:outputText value = "Olá JSF!" value="#{olaMundoBean.mensagem}" />
  </h:body>
</html>
```

Veja que usamos a expression language (EL) `#{{olaMundoBean.mensagem}}` como valor do atributo `value`. Quando nossa view é acessada através do navegador, o JSF procurará uma instância da classe `OlaMundoBean` e caso ela não existe instanciará uma para nós.

É por isso que o nome utilizado na expressão precisa começar com minúscula e usar o padrão `CamelCase`. É essa a convenção que o JSF usa para saber qual classe instanciar. É dessa instância que ele acessará o atributo `mensagem`. É o valor do atributo que será exibido para o usuário.

Qual das opções abaixo cria a classe `OlaMundoBean` de uma maneira que o atributo `mensagem` seja acessado pela view?

Selecione uma alternativa

A

```
public class OlaMundoBean {  
  
    private mensagem = "Olá";  
  
    public String getMensagem() {  
  
        return this.mensagem;  
    }  
  
    public void setMensagem(String mensagem) {  
        this.mensagem = mensagem;  
    }  
}
```

B

```
@ManagedBean  
public class OlaMundoBean {  
  
    private mensagem = "Olá";  
  
    public String getMensagem() {  
  
        return this.mensagem;  
    }  
  
    public void setMensagem(String mensagem) {  
        this.mensagem = mensagem;  
    }  
}
```

C

```
@ManagedBean  
public class OlaMundoBean {  
  
    private mensagem = "Olá";  
  
    public void setMensagem(String mensagem) {  
        this.mensagem = mensagem;  
    }  
}
```