

Criando o conversor para Autor

Considerando o que já fizemos para a *Data de Publicação*, podemos melhorar também a seleção de autores, que hoje é feita salvando os ID's dos autores no `AdminLivrosBean`. Em um sistema O.O. queremos trabalhar com o objeto `Autor`. Vamos criar também um conversor para o `Autor`.

O nome da classe é `AutorConverter` que também deve implementar a interface `Converter`. Para ser encontrada pelo **JSF** usamos a annotation `@FacesConverter("autorConverter")`. Perceba que dessa vez estamos nomeando o conversor ao invés de indicar para qual classe ele irá funcionar, como fizemos no `CalendarConverter`.

Faça o método `getAsObject` retornar um objeto `Autor`, lembrando que a `String` que recebemos como parâmetro terá apenas o `ID` do autor.

Da mesma, forma faça o método `getAsString` retornar o `ID` do `Autor` como texto.

Com isso teremos o nosso converter pronto, mas diferente do `Calendar`, precisamos dizer onde vamos utilizar o conversor. Abra novamente o formulário `livros/form.xhtml` e procure pelo `<h:selectManyListbox ... >`. Nele, adicione o atributo `converter="autorConverter"` e modifique o valor para `value="#{adminLivrosBean.livro.autores}"`. E dentro dele no `<f:selectItems ... >`, podemos alterar o atributo `itemValue` para ficar `itemValue="#{autor}"`, pois o conversor já irá tratar o que precisa ser tratado do `Autor`.

Lembre-se que para o funcionamento correto do `Converter` precisamos ter os métodos `equals` e `hashCode` sobrescritos corretamente na classe `Autor`.

Outra classe que sofrerá mudanças será a `AdminLivrosBean`. Nela, vamos remover o atributo `autoresId`, pois não precisamos mais dele, e dentro do método `salvar`, remova o `foreach` que cria os autores, também não precisamos mais dele. Ficando o método `salvar` assim:

```
@Transactional
public String salvar() {
    dao.salvar(livro);

    context.getExternalContext()
        .getFlash().setKeepMessages(true);
    context
        .addMessage(null, new FacesMessage("Livro cadastrado com sucesso!"));

    return "/livros/lista?faces-redirect=true";
}
```

Veja que ficou muito mais simples e agora podemos usar o `Autor` como sendo objeto em nossa `View` e também nos nossos `Beans`.

Faça um novo *Full Publish* e veja se tudo continua funcionando normalmente.

