



07

Exercício 6



7%

Dado o seguinte cenário:

```
@Stateless
public class LivroDao {
    // implementação
    omitida
}
```

[COPIAR CÓDIGO](#)

```
@Stateless
public class AutorDao {
    // implementação
    omitida
}
```

[COPIAR CÓDIGO](#)

O que está errado na implementação da classe abaixo na arquitetura Java EE:

```
public class LivroBean {
    private AutorDao
    autorDao = new AutorDao();
    // criação do DAO (1)
    private LivroDao
    livroDao = new LivroDao();
    // criação do DAO (2)
```

ATIVIDADES
7 DE 11FÓRUM DO
CURSOVOLTAR
PARA
DASHBOARDMODOS
NOTURNOABRIR
CADERNO

113.6k xp





7%

ATIVIDADES
7 DE 11FÓRUM DO
CURSOVOLTAR
PARA
DASHBOARD

113.6k xp



```
// outros métodos e  
atributos omitidos  
}
```

COPIAR CÓDIGO

Opinião do instrutor

Ao usar EJB, não podemos mais instanciar o `AutorDao` e o `LivroDao` na mão. Estamos assumindo o controle ao criar o DAO naquelas linhas (1) e (2). Nesse caso não estamos usando o `AutorDao` e o `LivroDao` como EJBs.

O DAO está sendo administrado pelo *EJB Container*. Portanto, quem cria o DAO é o *EJB Container* e não a minha classe. Consequentemente precisamos pedir ao *EJB Container* para passar aquela instância que ele está administrando. Felizmente, isso é fácil de fazer, basta usar a anotação `@Inject` :

```
public class LivroBean {
```



7%

ATIVIDADES
7 DE 11FÓRUM DO
CURSOVOLTAR
PARA
DASHBOARD

113.6k xp



```
@Inject
private AutorDao
autorDao; // sem new (1)
```

```
@Inject
private LivroDao
livroDao; // sem new (2)
```

```
// outros métodos e
atributos omitidos
}
```

COPIAR CÓDIGO