



07

## Importar o projeto e popular o banco

Como visto no video, vamos **usar um banco de dados** para realmente persistir e recuperar os dados. Utilizaremos JPA (*Java Persistence API*) para a persistência. Como JPA não é o foco deste treinamento, já preparamos um projeto que importaremos no Eclipse.

1. Baixe o ZIP pelo link: [Projeto livraria a importar no Eclipse \(http://s3.amazonaws.com/caelum-online-public/JSF/livraria-capitulo3.zip\)](http://s3.amazonaws.com/caelum-online-public/JSF/livraria-capitulo3.zip).
2. Importe o projeto pelo Eclipse. Para isso vá no menu *File->Import*. No item *General*, selecione *Existing project into workspace*. Depois escolha *Select archive file* e *Browse*. Na próxima tela selecione o arquivo **livraria-capitulo3.zip** que você baixou. Após a confirmação, o Eclipse criará um novo projeto importando os arquivos do ZIP.
3. Repare o nome do projeto. Ele se chama **livraria** apenas. O projeto é **igual ao projeto da aula** anterior. Temos os arquivos `livro.xhtml` e `autor.xhtml` e os `ManagedBeans` associados.
4. O projeto está com o JPA configurado e implementado. Abra uma vez a classe `br.com.caelum.livraria.dao.DAO` e o arquivo `persistence.xml` que está na pasta `src/META-INF`. Dentro do XML tem a configuração do banco de dados. Usaremos o MySQL na maquina local com um banco **livrariadb**.
5. É preciso criar o banco antes de subir a aplicação. Usaremos o terminal para se conectar com MySQL (fique a vontade de usar o seu cliente MySQL). Digite o comando `mysql -u root` para se conectar (*use -p para passar uma senha*). No prompt do MySQL criaremos o banco pelo comando: `create database livrariadb`. Deixe o terminal aberto.

ATENÇÃO: se o usuário root do seu MySQL está com senha diferente de branco, você deve alterar o arquivo `src/META-INF/persistence.xml`. Na entrada abaixo com sua senha:

```
<property name="javax.persistence.jdbc.password" value="sua senha" />
```

1. Volte ao Eclipse e rode a classe **PopulaBanco** no pacote `code>br.com.caelum.livraria.dao`. Rode a classe como aplicação Java comun (*sem Tomcat*), use **Run As -> Java Application**. Ao rodar a classe `PopulaBanco` o JPA deve criar automaticamente as tabelas. Verifique o console para achar possíveis problemas.
2. Depois volte ao terminal. Ainda conectado com MySQL digite: `use livrariadb;` para selecionar o banco e depois `show tables;` para mostrar as tabelas. Foram criadas as tabelas **Autor**, **Livro** e o **relacionamento**.

### Responda

INSERIR CÓDIGO	FORMATAÇÃO
Cole aqui o output da console caso tenha tido problemas. Clique aqui para copiar o output da console caso tenha tido problemas.	

