

04

## (Obrigatório) Preparando o ambiente

Para realizar esse treinamento é necessário possuir algumas ferramentas instaladas:

- Eclipse (<http://www.eclipse.org/downloads/>)
- Mysql (<http://dev.mysql.com/downloads/mysql/>)
- Tomcat 8: (<http://tomcat.apache.org/download-80.cgi>)

1) **Tomcat 8:** O primeiro passo é configurar o Tomcat 8 como servidor dentro de Eclipse. Baixe o [Tomcat](http://tomcat.apache.org/download-80.cgi) (<http://tomcat.apache.org/download-80.cgi>) e extraia o ZIP. Depois configure o Tomcat como servidor no Eclipse. Isto é, na view "Servers > New Server". Escolha o Tomcat na versão 8 e aponte para a pasta da instalação do Tomcat.

2) **Mysql:** Com o MySQL instalado, crie um banco com o nome **projeto\_jpa**. No terminal:

```
mysql -u root -p  
create database projeto_jpa;  
use projeto_jpa;  
show tables;
```

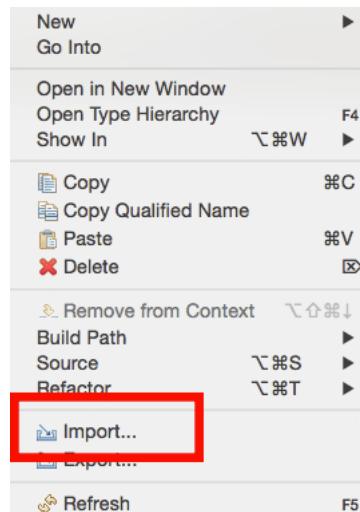
Não temos ainda nenhuma tabela no banco. Ao subir o nosso projeto automaticamente serão criadas as tabelas.

3) **Download do projeto:** Usaremos um projeto web já com algumas classes prontas que poderá ser baixado [aqui](https://s3.amazonaws.com/caelum-online-public/jpa-avancado/stages/projeto-jpa2-inicial.zip) (<https://s3.amazonaws.com/caelum-online-public/jpa-avancado/stages/projeto-jpa2-inicial.zip>). Após o download, extraia o ZIP.

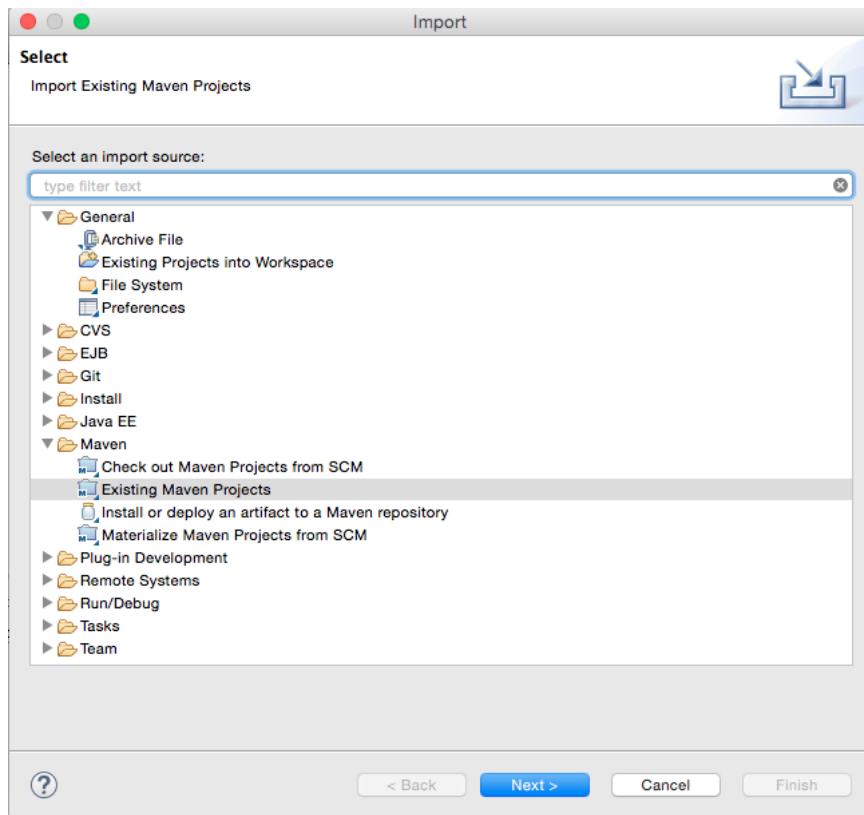
Renomeie a pasta do projeto para `projeto-jpa-2`.

4) **Importando o projeto no Eclipse:**

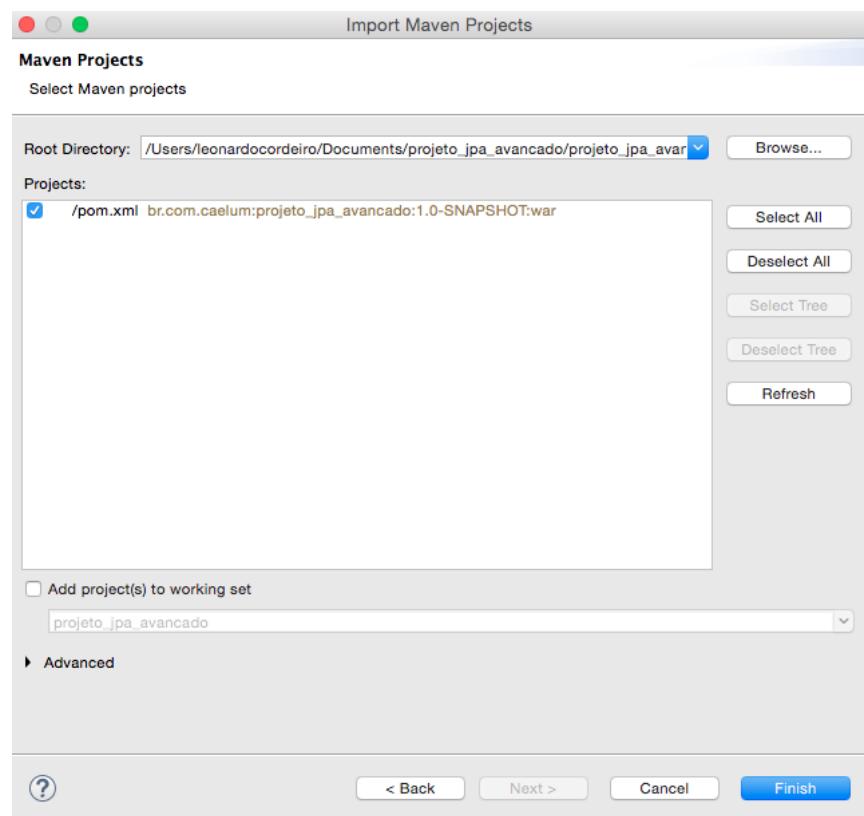
Na lista de projetos (Package Explorer) clique com o botão direito > import.



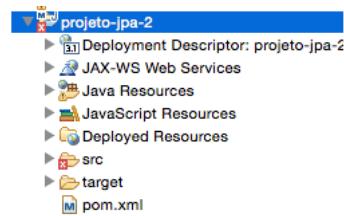
Procure pela pasta **Maven** e clique em **Existing Maven Projects** (necessário Eclipse Kepler ou mais novo).



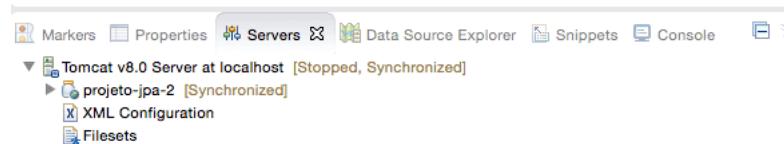
Selecione o projeto em **Browser** e em seguida clique em **Finish**.



**Observação:** A importação pode demorar alguns minutos pois o Maven vai baixar os JARs necessários. O projeto vai acusar alguns erros no JSP que você pode ignorar tranquilamente, no entanto todas as classes devem compilar!



5) No Eclipse, na view *Servers*, associe o projeto com o Tomcat (basta arrastar o projeto-jpa-2 no Tomcat).



6) Vá até o método `getDataSource` da classe `JPAConfigurator` e veja se os dados de acesso ao banco de dados estão de acordo com as configurações da sua máquina:

```
@Bean
public DataSource getDataSource() {
    DriverManagerDataSource dataSource = new DriverManagerDataSource();

    dataSource.setDriverClassName("com.mysql.jdbc.Driver");
    dataSource.setUrl("jdbc:mysql://localhost/projeto_jpa");
    dataSource.setUsername("root");
    dataSource.setPassword("");

    return dataSource;
}
```

7) Reinicie o Tomcat e fique de olho no console! Se der certo, tente acessar: <http://localhost:8080/projeto-jpa-2/> (<http://localhost:8080/projeto-jpa-2/>)