

Lab 1 - Configurando aplicação Banco usando Maven

Neste laboratório faremos a configuração do Maven, modificando a aplicação Banco que está sendo desenvolvida ao longo do curso. O Maven é uma ferramenta de gerenciamento, construção e implantação de projetos, então iremos utilizá-la para fazer o gerenciamento de dependências.

Exercícios

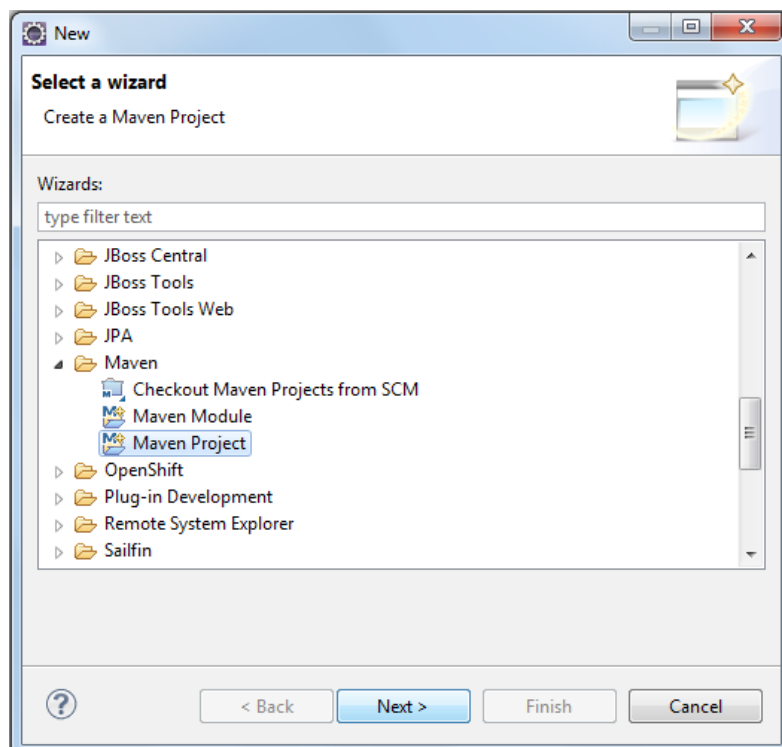
Exercício 1: Configurar Maven modularizado da aplicação banco adicionando as dependências que cada módulo necessite.

Exercício 2: Adicionar todos os fontes feitos até agora nos devidos módulos criados, e resolver dependências.

Exercício 3: Criar um novo módulo Banco-Framework para desenvolver os próximos laboratórios

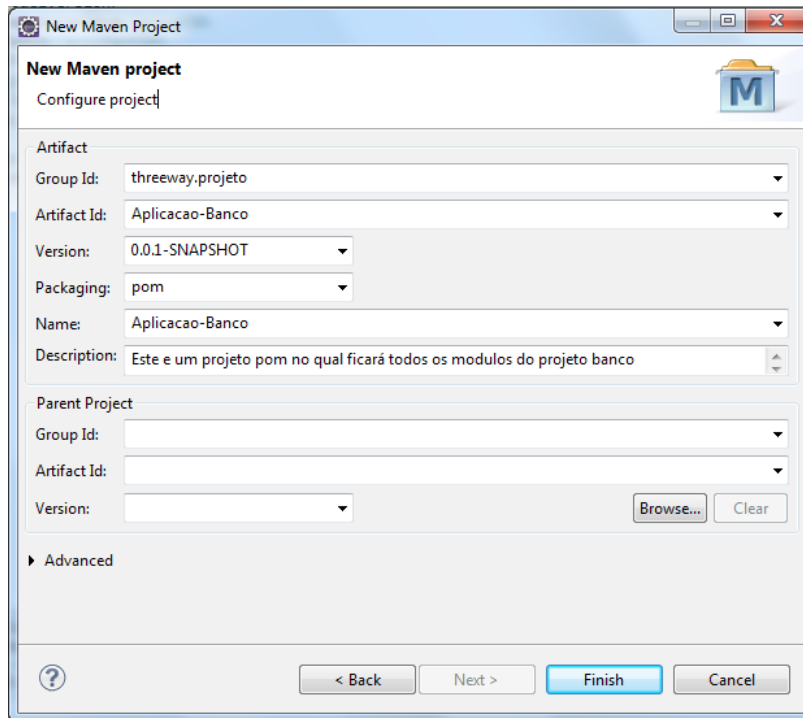
Exercício 1 - Configurar Maven modularizado da aplicação banco adicionando as dependências que cada módulo necessite.

1. - Crie um novo projeto Maven o qual será o empacotador da nossa aplicação. Na barra de ferramentas do Eclipse, vá em **New > Maven Project** ou **New > Other > Maven Project**.

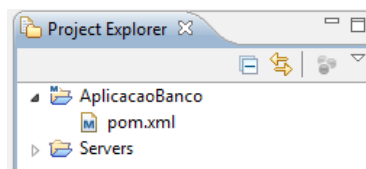


2. Selecione a opção **Create a simple project** e clique em **Next**.

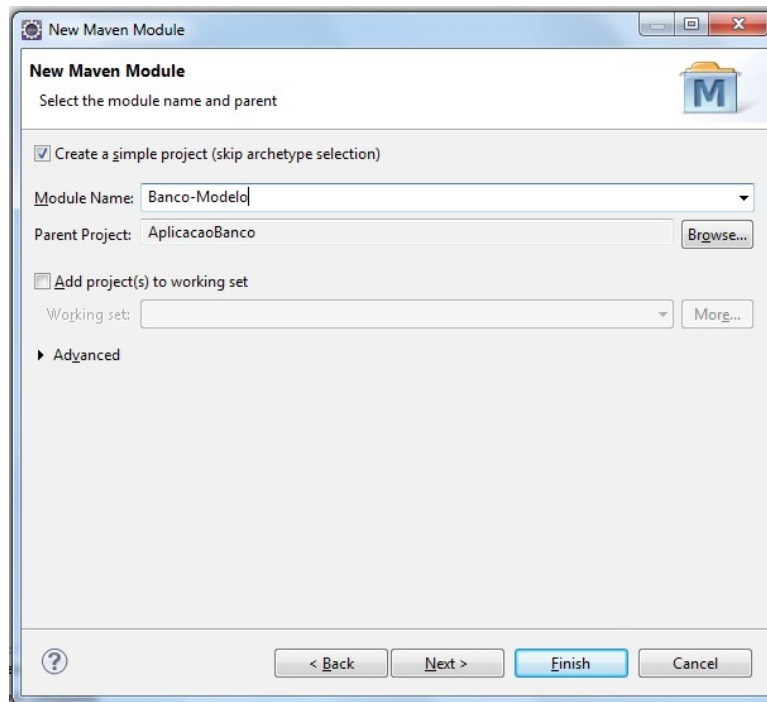
3. Crie o projeto conforme a imagem abaixo. Depois clique em **Finish**.



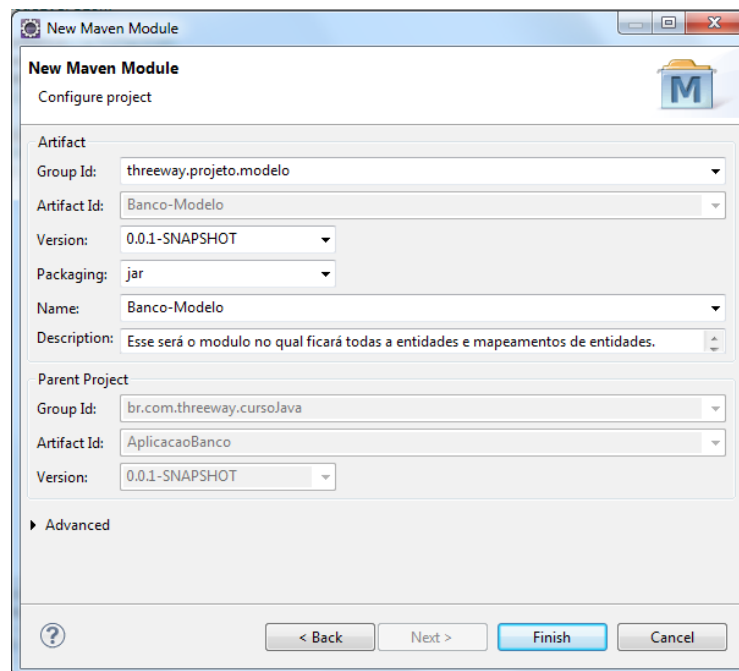
4. Será gerado uma pasta **src**, apague-a e deixe somente o arquivo **pom.xml**, visto que não será adicionado código fonte no projeto empacotador.



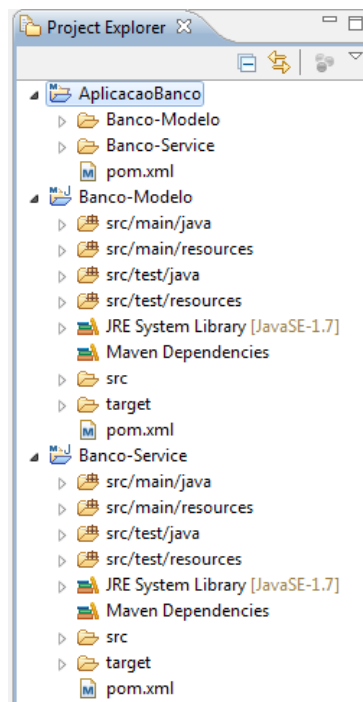
5. Com o projeto empacotador criado, vamos agora criar os módulos.
6. Crie o Modulo *Banco-Modelo*, *Banco-Servico*, *Banco-SwingView*.
- a. Clique com o botão direito no **projeto empacotador (Aplicacao-Banco)** > **New** > **Other** > **Maven** > **Maven Module** e marque a opção **Create a simple project** e depois clique em **Next**.



- b. Certifique que seu modulo está configurado de acordo com a imagem abaixo.

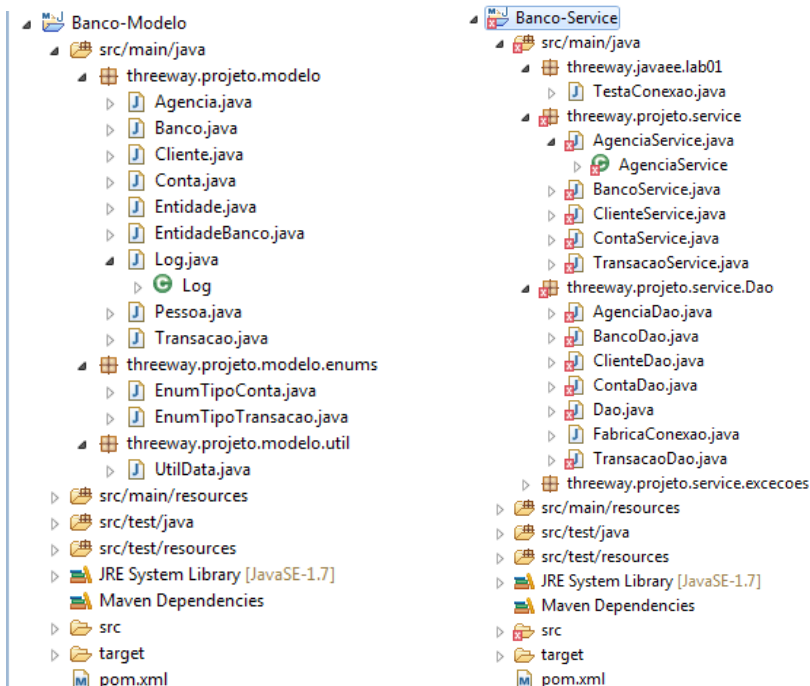


- c. Crie os módulos restantes utilizando essas mesmas configurações, de forma que a estrutura da aplicação fique de acordo com a imagem abaixo:

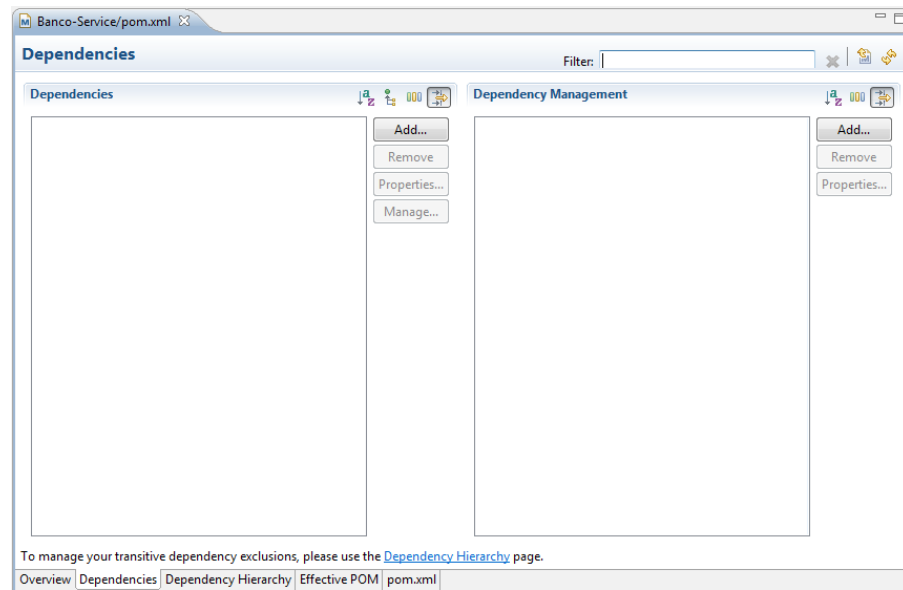


Exercício 2 -Adicionar todos os fontes feitos ate agora nos devidos módulos criados e resolver as dependências.

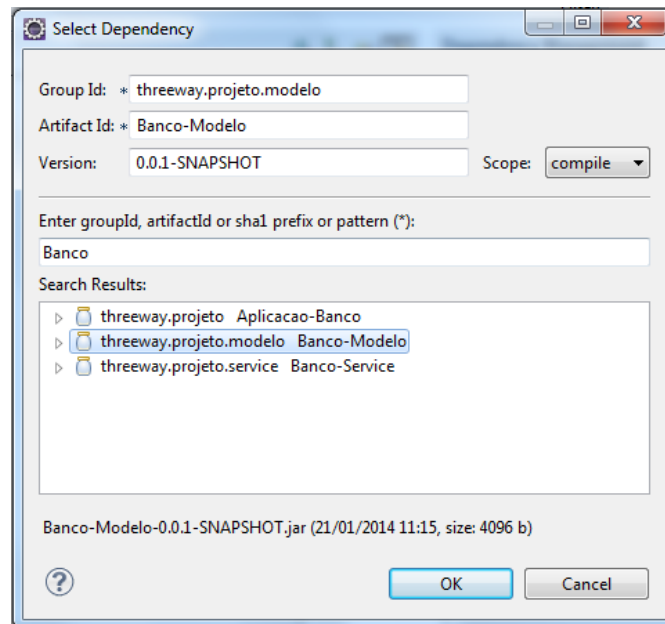
1. Copie todos os fontes de *Banco-Modelo* e *Banco-Service* desenvolvidos até o momento preservando a estrutura de pacotes e cole nos módulos que foram criados.



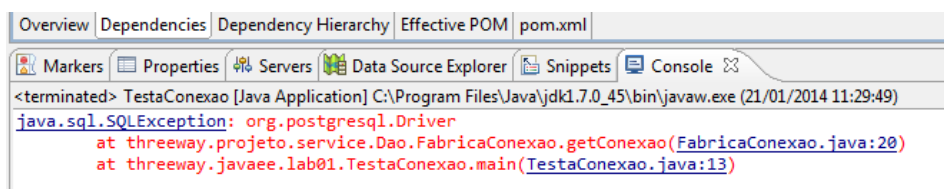
2. Veja que foram gerados vários erros. Isso ocorre pelo fato de haver uma dependência entre o *Banco-Modelo* em *Banco-Service*.
 - a. Abra o *pom.xml* de *Banco-Service* e vá na aba *Dependencies*.



- b. Clique em **Add...** e busque pelo **Artifact Id** como na imagem abaixo:



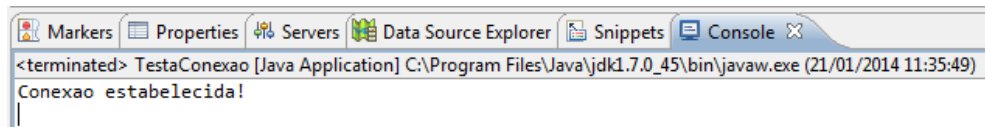
- c. Selecione o Banco-Modelo e clique em **OK**, salve o arquivo para aplicar as mudanças. Os erros já não aparecerão mais.
3. Execute a classe **TestaConexao.java** criada anteriormente. Verá que vai ser lançada uma exceção, pois o Postgres não está nas dependências do módulo *Banco-Service*.



4. Para solucionar esse problema abra o arquivo **pom.xml** de *Banco-Service* em modo de edição e modifique adicionando a dependência do *postgres* conforme a imagem abaixo:

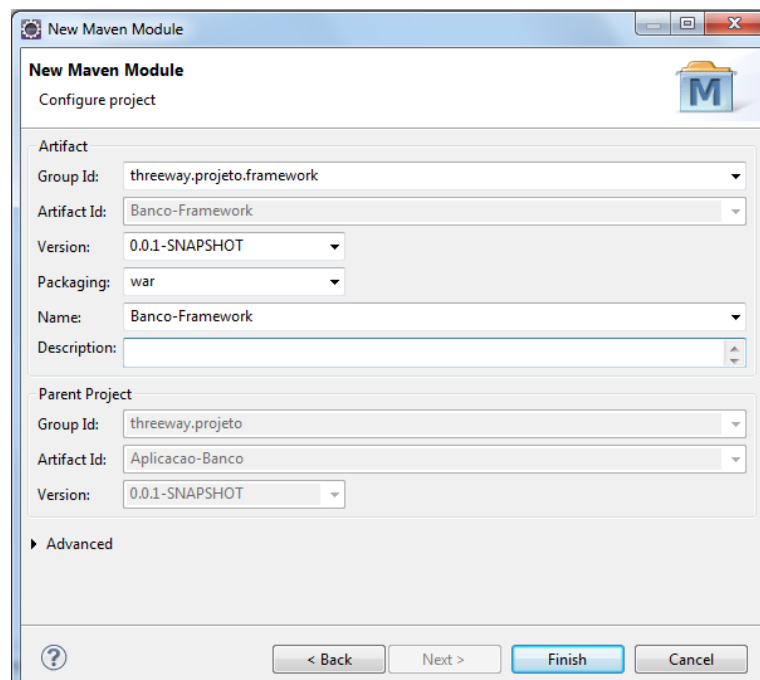
```
<dependency>
  <groupId>postgresql</groupId>
  <artifactId>postgresql</artifactId>
  <version>9.1-901.jdbc4</version>
</dependency>
```

5. Para baixar a dependência e construir a aplicação, clique com o botão direito no empacotador depois em **Run As > Maven install**. Faça isso nos módulos também. Depois execute novamente a classe **TestaConexao.java**.



Exercício 3 - Criar um novo modulo Banco-Framework para desenvolver os próximos laboratórios.

1. Crie um novo módulo chamado *Banco-Framework*, porém ao invés desse modulo ser do tipo JAR ele será do tipo WAR, pois este será executado em um servidor de aplicação JEE.



2. Adicione a dependência de *Banco-Service* em *Banco-Framework*, pois o mesmo utilizará os métodos de negócio da aplicação.

Sugestão de Exercício - Pegue as aplicações de visão Swing e Web desenvolvida ate agora e as converta em módulos maven, lembrando que quando for gerar um modulo que será executado em um servidor de aplicação o mesmo será do tipo WAR. Nesse caso, o módulo WebView será do tipo WAR.

