

## Configurando o Datasource

### Transcrição

Uma vez que precisamos de um `datasource`, vamos criá-lo com base no exemplo e então colocaremos algumas configurações adicionais. Ficando nosso código conforme abaixo:

```
<datasource jndi-name="java:jboss/datasources/casadocodigoDS" pool-name="casadocodigoDS">
  <connection-url>jdbc:mysql://localhost:3306/casadocodigo_javaee</connection-url>
  <connection-property name="DatabaseName">
    casadocodigo_javaee
  </connection-property>
  <driver>mysql</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>20</max-pool-size>
  </pool>
  <security>
    <user-name>root</user-name>
  </security>
</datasource>
```

- Começando pela tag `<datasource>`, temos o atributo `jndi-name`, que é a ligação entre o `datasource` que estamos criando e a nossa aplicação;
- Mais abaixo, temos a tag `<connection-url>` que recebe o endereço do nosso banco de dados;
- Em seguida temos a tag `<connection-property>`, que obrigatoriamente precisamos adicionar quando usamos o Wildfly 10. Perceba que estamos informando algo que já está na tag `connection-url`, porém não será levado em consideração lá, apenas aqui na tag `<connection-property>`;
- Na tag `<pool>` nós informamos o número mínimo e máximo de conexões que o `datasource` poderá criar com o banco de dados;
- Por fim, temos a tag `<security>`, onde informamos o nome do usuário do banco de dados na tag `<user-name>`, no nosso caso `root`; Caso você possua uma senha para acessar seu `mysql`, basta adicionar a tag `<password>` abaixo de `user-name`;

Parece que pulamos uma tag, que foi a `<driver>`. Essa tag referencia outra área específica dentro da configuração, que fica logo abaixo de `<datasource>`, é a tag `<drivers>`. Dentro dela temos a tag `<driver>` de exemplo, porém, assim como em `<datasource>`, precisamos criar a nossa configuração de driver. Ficando nosso código, assim:

```
<driver name="mysql" module="com.mysql">
  <datasource-class>com.mysql.jdbc.jdbc2.optional.MysqlDataSource</datasource-class>
</driver>
```

Essa configuração da tag `<driver>`, serve principalmente para informar a classe do `datasource`, o qual fazemos através da tag `<datasource-class>` do código acima. Mas ainda dentro da tag, temos um atributo importante, é o `module`. Todo *driver* **JDBC**, precisa de um **JAR**. Esse **JAR** não fica armazenado no projeto e sim no servidor, para ser

mais específico, no Wildfly os drivers ficam armazenados na pasta: `wildfly/modules/`. Devemos colocar nosso **JAR** dentro desta pasta, com a estrutura a seguir:

```
com/
  mysql/
    main/
      mysql-connector-java-5.1.35.jar
```

Dentro da pasta `main` nós temos que adicionar um arquivo chamado `module.xml` além do **JAR** do driver de conexão. No arquivo `module.xml` temos o caminho para o jar do driver de conexão e uma dependência para a API de persistência do *JDBC* chamada `javax.api`. O `module.xml` ficará assim:

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.3" name="com.mysql">
  <resources>
    <resource-root path="mysql-connector-java-5.1.35.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
  </dependencies>
</module>
```

Agora só falta criar o banco de dados. Abra o seu Terminal e conecte no MySQL com o commando `mysql -u root` (ou utilize uma ferramenta de sua preferência) e crie a base com o comando `create database casadocodigo_javaee;`. Acesse o banco criado com o comando `use casadocodigo_javaee;` e verifique as tabelas existentes usando o `show tables;`. Até aqui, não devemos ter nenhuma tabela na base.

Tudo certo, configuração finalizada. Vamos subir nossa aplicação e depois de subir a aplicação execute o comando `show tables;` novamente. Nesse momento, a tabela `Livro` deve ter sido criada. Compare as colunas usando `desc Livro;` para ver se está de acordo com sua classe. Sua estrutura, deve estar parecida com a seguinte:

```
mysql> desc Livro;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
descricao	longtext	YES		NULL	
numeroPaginas	int(11)	YES		NULL	
preco	decimal(19,2)	YES		NULL	
titulo	varchar(255)	YES		NULL	

```
5 rows in set (0,01 sec)
```