Datasource configuration
with GlassFish 4.x

Version : 1.0 - Februari 2015

Rudy De Busscher

Table of Contents

List of Figures

Chapter 1. Introduction

In order to make a connection to a database you just have to copy the jar file to the domain lib folder and make the configuration, through the web console or the command line.

In this tutorial, the MySQL driver will be used, but any other is very similar.

The directory where you installed WildFly will be referred to as `%GLASSFISH%`.

Chapter 2. Copy driver jar file

The driver and datasource class is located in de Connector/J download. Copy the jar file (like mysql-connector-java-5.1.33.jar) to the `%GLASSFISH%/glassfish/lib` directory.

Chapter 3. Configure server

The configuration of the server can be done in two ways. The web console or the command line.

There are 2 parts in making a connection to the database, the **connection pool** with the info of the database, and the **JNDI name** it will be bound in the server.

JDBC Connection Pool

For the connection pool, we need to specify database connection parameters like location of the database and credentials. You can follow these steps to perform it through the web console.

1. Open the administration console in your browser by going to the url **http://localhost:4848**

2. Find the **JDBC Connection pool menu** on the right side (see also image 1), and click on it.

3. Make a new pool, by clicking on the `New...` button.

4. Specify a **pool name** (can be anything, no requirements), select `javax.sql.DataSource` as **Resource type** and `MySql` as **Database vendor**. See also image 2.

5. Click on the `Next...` button.

6. In the section **Pool settings**, you can change connection number and other parameters. The default initial/minimum count of 8 can probably reduced.

7. There are a lot of additional parameters by default. They can be removed or kept. The important thing is the **url**, **user** and **password** parameters. (watch out for the case)

    a. url $\rightarrow$ jdbc:mysql://localhost:3306/<<your-database>>

    b. user $\rightarrow$ the_user

    c. password $\rightarrow$ secret

8. Click on the `Finish` button.

There is an important remark about an empty password further on.
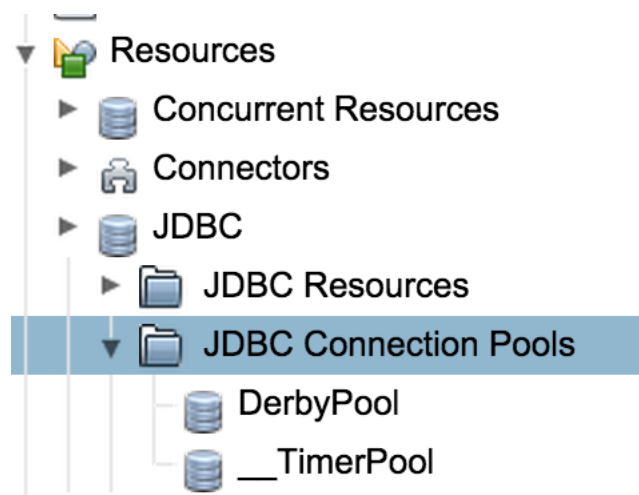
Figure 3.1.

Image 1: JDBC Connection pool menu

Figure 3.2.

Image 2: Pool name and data source information.

From the command line

Start the `asadmin` command line tool, you can find it in the directory `%GLASSFISH%/bin`

1. Make sure the server (or correct term, a domain) is running. If not, you can start one with the command **start-domain**

2. Execute the following command (multi line for easier reading)

create Pool.

```
create-jdbc-connection-pool
    --restype javax.sql.DataSource
    --datasourceclassname com.mysql.jdbc.jdbc2.optional.MysqlDataSource
    --property "user=the_user:password=secret:url=jdbc\\:mysql\\://
localhost\\:3306/rubus" mySqlPool
```

The ∶ must be escaped since it is the delimiter for the different properties. Therefore, it need to be noted as \\∶.

JNDI Name

The last, simple, step we need to do, is to assign a JNDI name to this datasource, so that we can reference it from within a `persistence.xml` file or other ways.

From within the web console, follow these steps.

1. Select the **JDBC Resource menu** on the left (just above the JDBC Connection pool menu)

2. Click on the New... button.

3. Specify the name, must be in the format `jdbc/<<name>>`, where you can choose the name of course.

4. Select the pool, we created in a previous step, see above. (see image 3)

No other input is needed.

Figure 3.3. Create JNDI name for pool

From the command line

With the asadmin tool, you can create also a jdbc resource. I assume you are still in the tool, with a domain running (see above).

You can execute then the following command.

create Resource.

```
create-jdbc-resource --connectionpoolid mySqlPool jdbc/mysql
```

Chapter 4. Varia

Empty password

An empty password can't be specified, not with the web console, not with the command line tool.

Of course, an empty password isn't a best practice. For a local development environment, it can be considered and you can follow the following procedure:

1. Follow the above procedure and put in a wrong value for the password.

2. Open the `%GLASSFISH/glassfish/domains/domain1/domain.xml` file and look for the wrong password you have specified. Make the value empty.

The password value should look like this.

example empty password.

```
----
<property name="password" value=""></property>
----
```

Of course, if your resource isn't defined in `domain1`, look in the corresponding directory.