Choose page language

Apache NetBeans | Plugins | Docs & Support | Community

Search

HOME / Docs & Support

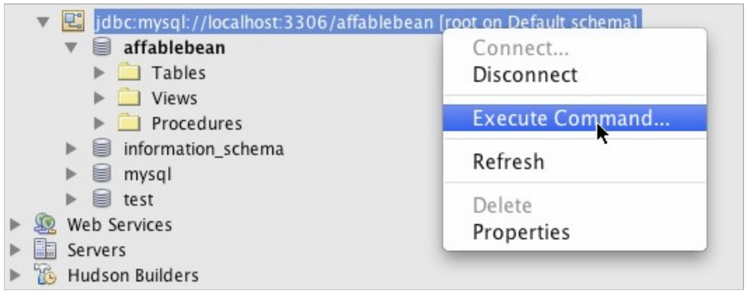
The NetBeans E-commerce Tutorial - Setup Instructions

Tutorial Contents

REQUIRES NetBeans 6.8/6.9

If you want to follow a tutorial unit without having completed previous units, you need to perform some preliminary steps in order to set up your development environment.

1. **Set up your MySQL database server.** Follow the steps outlined in: Communicating with the Database Server.
2. **Create the `affablebean` schema on the database server, and populate the database with sample data:**
   1. Click on affablebean.sql and copy (Ctrl-C; ⌘-C on Mac) the entire contents of the file.
   2. Open the IDE's SQL editor. In the Services window (Ctrl-5; ⌘-5 on Mac), right-click the `affablebean` database connection ( ) node and choose Execute Command.



   The IDE's SQL editor opens.

   3. Paste (Ctrl-V; ⌘-V on Mac) the entire contents of the `affablebean.sql` file into the editor.
   4. Click the Run SQL ( ) button in the editor's toolbar. The script runs on your MySQL server. Tables are generated for the database, and sample data is added to the `product` and `category` tables.
3. **Create a connection pool and JDBC resource on GlassFish.**
   1. In the Services window (Ctrl-5; ⌘-5 on Mac), expand the Servers > GlassFish Server 3 node and choose Properties. In the Servers window that displays, make sure the 'Enable JDBC Driver Deployment' option is selected. If your project requires the MySQL Connector/J driver, this option will ensure that the driver is deployed to GlassFish when your project is deployed. (If the server is already running, you'll need to restart the server.)
   2. In the Services window, right-click the GlassFish Server 3 node and choose Start.
   3. Once the server is running, right-click the GlassFish Server 3 node and choose View Admin Console.
   4. Log into the console (default username/password is: `admin/adminadmin`).
   5. In the Admin Console, in the Tree on the left, expand the Resources > JDBC node, then click the Connection Pools node.
   6. In the Connection Pools interface, click the New button, and enter the following details:
      - **Name:** `AffableBeanPool`
      - **Resource Type:** `javax.sql.ConnectionPoolDataSource`

Download NetBeans IDE

Training

Java Programming Language

Support

Oracle Development Tools Support Offering for NetBeans IDE

Documentation

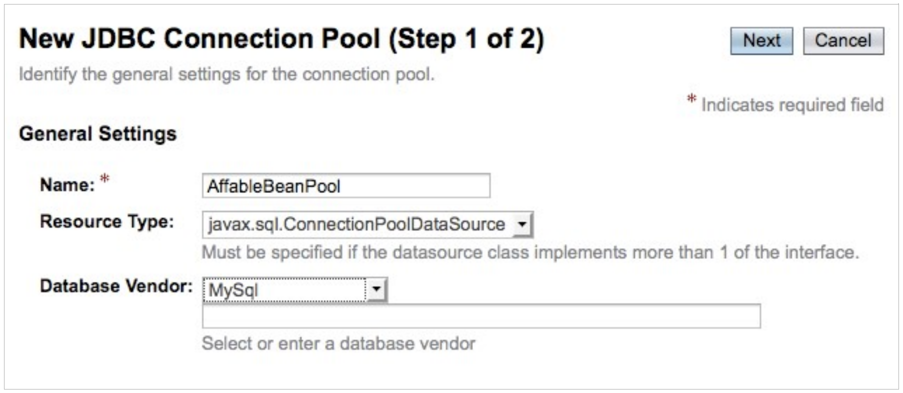
General Java Development
External Tools and Services
Java GUI Applications
Java EE & Java Web Development
Web Services Applications
NetBeans Platform (RCP) and Module Development
PHP and HTML5 Applications
C/C++ Applications
Mobile Applications

Sample Applications
Demos and Screencasts

More

FAQs
Contribute Documentation!
Docs for Earlier Releases

- **Database Vendor:** `MySql`



New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

* Indicates required field

General Settings

Name: * AffableBeanPool

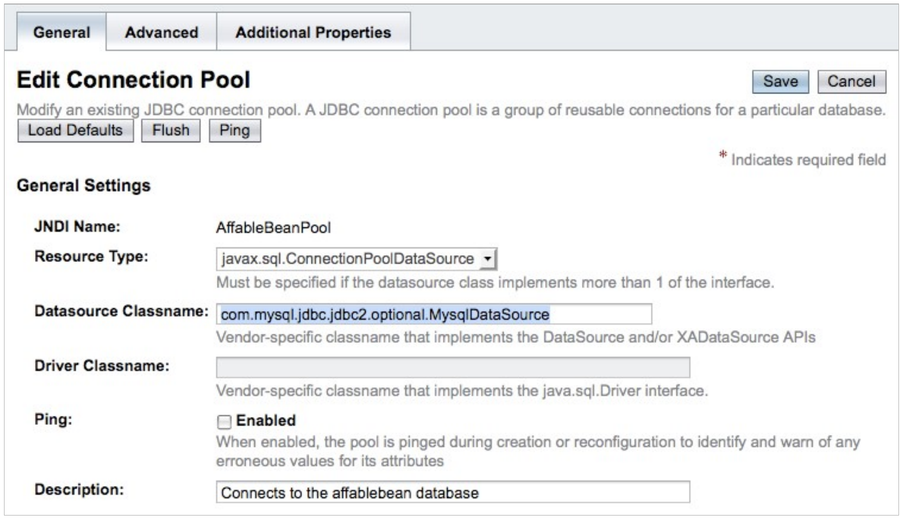
Resource Type: javax.sql.ConnectionPoolDataSource
Must be specified if the datasource class implements more than 1 of the interface.

Database Vendor: MySql
Select or enter a database vendor

7. Click Next. Accept all defaults and click Finish.

8. In the Connection Pools interface, click on your newly created `AffableBeanConnectionPool` to make the following change under the General tab:

   - **Datasource Classname:** `com.mysql.jdbc.jdbc2.optional.MysqlDataSource`



Edit Connection Pool

Modify an existing JDBC connection pool. A JDBC connection pool is a group of reusable connections for a particular database.

* Indicates required field

General Settings

JNDI Name: AffableBeanPool

Resource Type: javax.sql.ConnectionPoolDataSource
Must be specified if the datasource class implements more than 1 of the interface.

Datasource Classname: com.mysql.jdbc.jdbc2.optional.MysqlDataSource
Vendor-specific classname that implements the DataSource and/or XADataSource APIs

Driver Classname:
Vendor-specific classname that implements the java.sql.Driver interface.

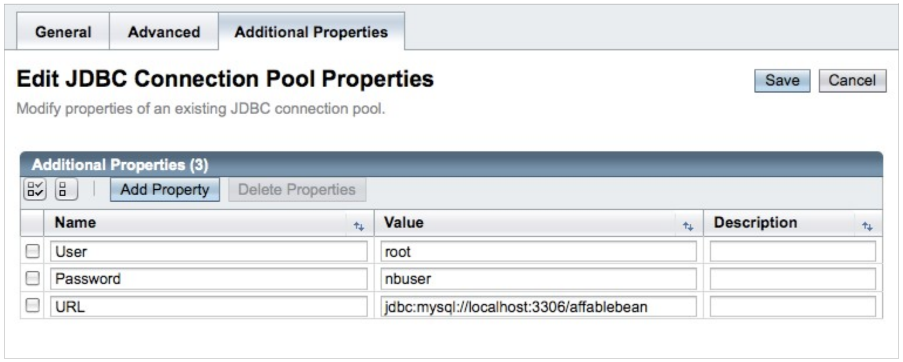
Ping: □ **Enabled**
When enabled, the pool is pinged during creation or reconfiguration to identify and warn of any erroneous values for its attributes

Description: Connects to the affablebean database

9. Click Save.

10. Click the Additional Properties tab and ensure that the following three properties have been set. (There may be other properties listed - these are default settings, however the following three must be set manually.)

    - **User:** `root`
    - **Password:** `nbuser`
    - **URL:** `jdbc:mysql://localhost:3306/affablebean`



Edit JDBC Connection Pool Properties

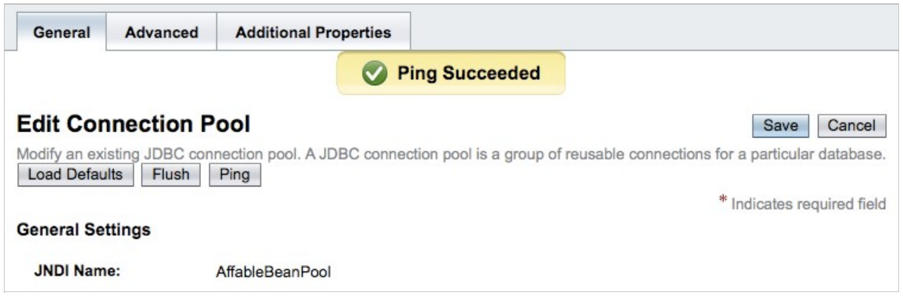
Modify properties of an existing JDBC connection pool.

Additional Properties (3)

	Name	Value	Description
□	User	root	
□	Password	nbuser	
□	URL	jdbc:mysql://localhost:3306/affablebean	

11. Click Save.

12. Click the General tab, then click Ping. You should see a message indicating that the ping succeeded. The `AffableBeanPool` connection pool can now connect to your MySQL database server.



Ping Succeeded

Edit Connection Pool

Modify an existing JDBC connection pool. A JDBC connection pool is a group of reusable connections for a particular database.

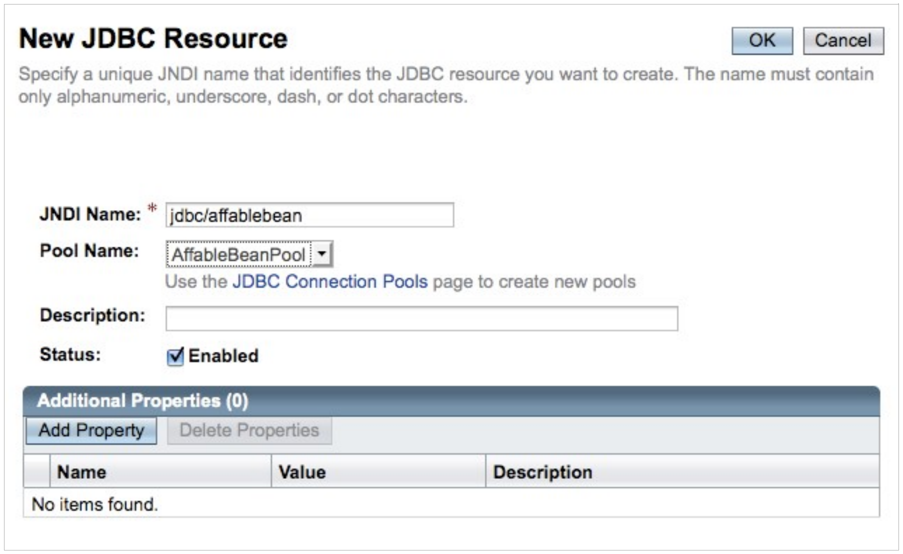
* Indicates required field

General Settings

JNDI Name: AffableBeanPool

13. In the Admin Console's Tree in the left column, click the Resources > JDBC > JDBC Resources node. The JDBC Resources interface opens in the main window.

14. Click the New button to create a new JDBC resource, then enter the following details:
    - **JNDI Name:** `jdbc/affablebean`
    - **Connection Pool:** `AffableBeanPool`



New JDBC Resource

Specify a unique JNDI name that identifies the JDBC resource you want to create. The name must contain only alphanumeric, underscore, dash, or dot characters.

JNDI Name: * jdbc/affablebean

Pool Name: AffableBeanPool

Use the JDBC Connection Pools page to create new pools

Description:

Status: ☑ Enabled

Additional Properties (0)

Name	Value	Description
No items found.		

15. Click OK.

You have set up the MySQL server and can connect to it from the IDE's Services window. You created a database named `affablebean`, and populated the database's `product` and `category` tables with sample data. You then started the GlassFish server, and created a connection pool that enables the server to connect to the `affablebean` database. Finally, you created a JDBC resource which your application can use to reference the server's connection pool.

You can now open and run any of the project snapshots provided for you in the tutorial units.

Send Us Your Feedback

About Us