Politecnico di Milano

Software Engineering 2 Project
a.y. 2016-17

Java Enteprise Edition

Toolkit installation and configuration manual

Version 1.2

Valero Castelli
November 17, 2016

Contents

Chapter 1

Introduction

1.1 Purpose of this document

This document contains a short guide to install the pieces of software that we are going to use withint the laboratory on Java Enterprise Edition of the Software Engineering II course.
Specifically, the following software is needed:

- Java Development Kit (JDK), Standard Edition, version 8. This is a prerequisite for NetBeans to run correctly and to be able to compile Java Applications. If you have already installed this component from the Software Engineering I course, you can skip the part of the guide that deals with its installation.
- GlassFish Server: this is the JEE engine that is going to run our application logic, which is going to be implemented in terms of Enterprise Java Beans. As mentioned during the lectures, this could be replaced by another engine compatible with the JEE specification.
  GlassFish is stable, well-known and open source. These are the main reasons for us to adopt it.
- Apache Tomcat: contrary to GlassFish, which provides a full implementation of the JEE framework, Tomcat is specifically designed to run servlets. For this reason, it can be used to run the server-side of applications, and it's a good idea to install it along with GlassFish.
- NetBeans: this is the IDE we are going to use for the development. You may also use Eclipse, but NetBeans is better integrated with GlassFish and easier to configure and to work with.
- MySQL: this will be our DBMS. It can be replaced by any other relational DBMS. We have chosen this as it is well known, open source and sufficiently robust to handle a small database as the one we will develop.

1.2 Notes

In the following sections we will provide details on how to install these components on three operating systems: Linux, Windows and macOS.
Please notice that instructions for Linux have been thought to be as distribution-independent as possible. However, there are a couple of points where the installation process involves interacting with the package manager of the distribution. In these cases, which have been highlighted in the guide, we will make reference to Ubuntu/Debian and the APT package manager as an example.

Chapter 2

Linux

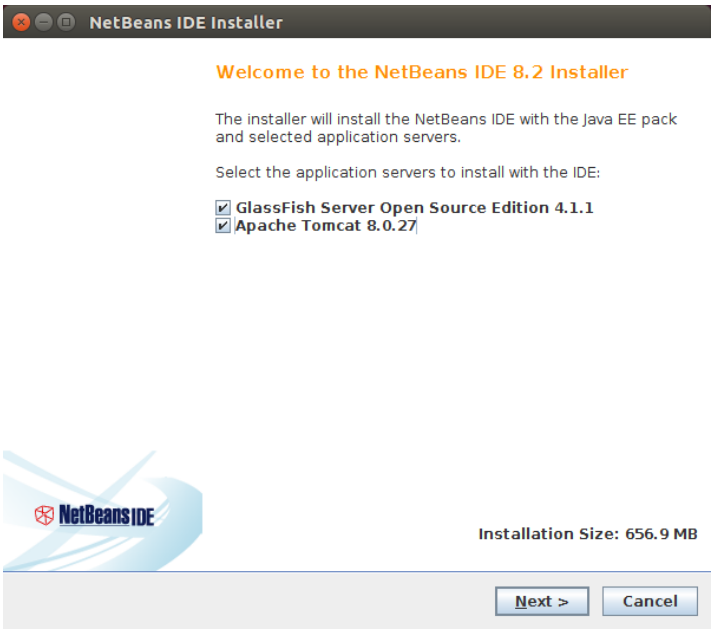
2.1 JDK Configuration

- Open the browser and head to **https://goo.gl/jXOIDJ**
- Accept the license agreement (suggested version: Java SE Development Kit 8u112)
- Click on the .tar.gz file corresponding to your platform (Linux x86 / Linux x64)
- When asked to open or download, select download
- Open the terminal and, if you are not in your home folder, change directory to reach it
- Navigate to the download folder with **cd Downloads**
- Type the following command: **tar zxvf jdk-8u112-linux-x64.tar.gz -C /usr/src**

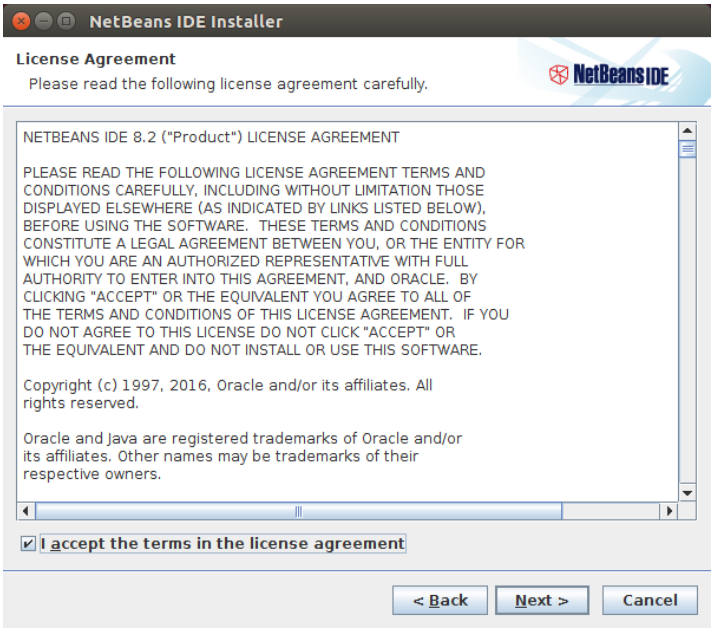
2.2 Install NetBeans Enterprise Edition

- Open the browser and head to **https://goo.gl/T0IdR0**
- Select the Linux x86/64 platform
- Click on the download button at the bottom of the Java EE column (save the .sh file)
- Open the terminal and, if you are not in your home folder, change directory to reach it
- Navigate to the download folder with **cd Downloads**

- Type the following command: **sh netbeans-8.2-javaee-linux.sh –javahome /usr/src/jdk1.8.0_112/**
- The window of the installer wizard will open
- Select both GlassFish Server (default) and Apache Tomcat, then click Next

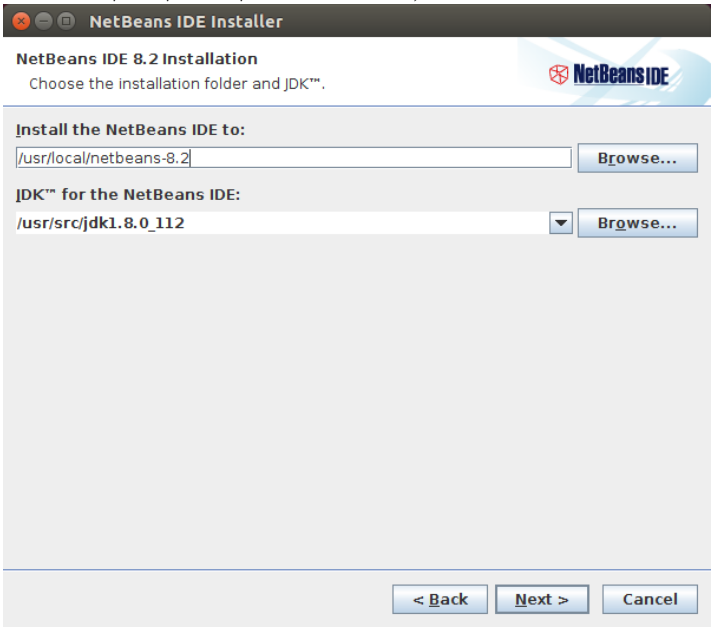


- Accept agreement, then click Next

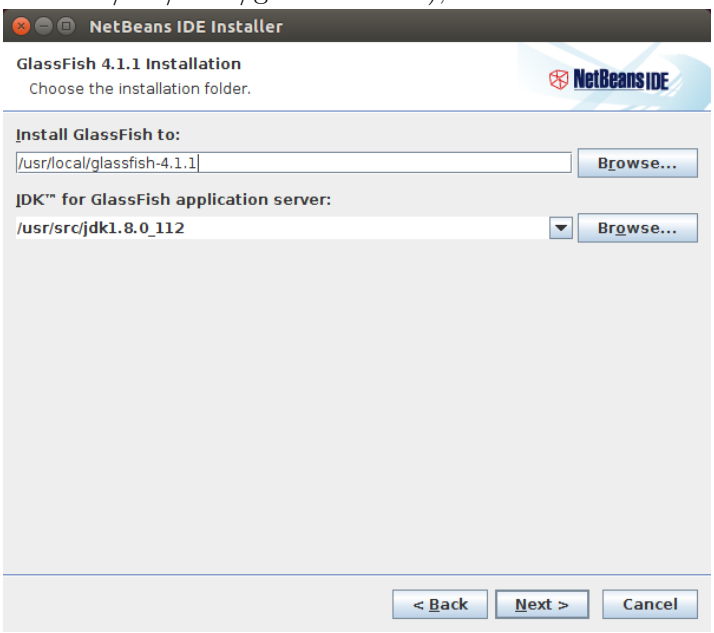


- Take note of the location where NetBeans will be installed at (by

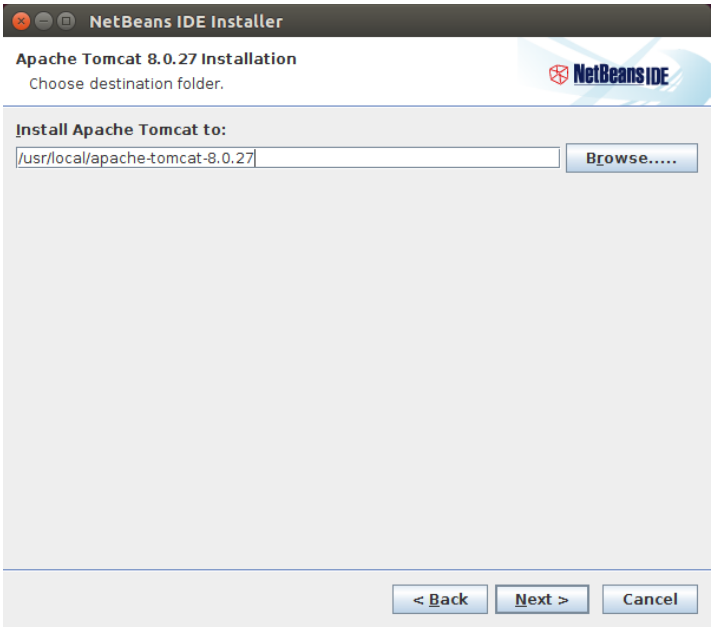
default: /usr/local/netbeans-8.2), then click Next



- Take note of the location where GlassFish will be installed at (by default: /usr/local/glassfish-4.1.1), then click Next



- Take note of the location where Tomcat will be installed at (by default: /usr/local/apache-tomcat-8.0.27), then click Next



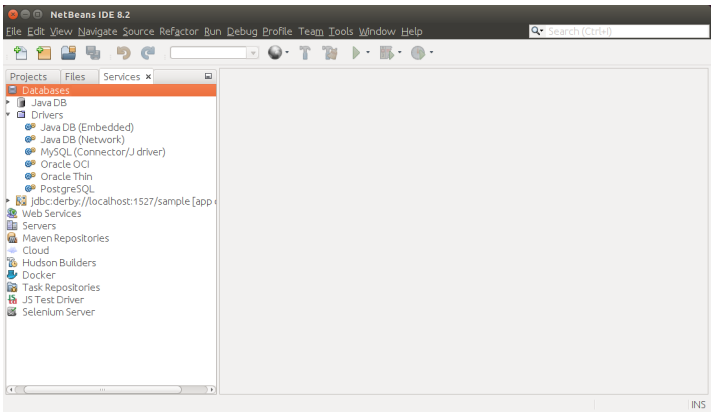
2.3 Install MySQL

- Open the browser and head to **https://goo.gl/x14GMC**
- From the dropdown menu, select Linux Generic
- Download Linux - Generic (glibc 2.5), Compressed TAR archive (.tar.gz extension), for your architecture (32 bit/64 bit)
- Click on "No thanks, just start my download"
- When asked to open or download, select download
- Open the terminal and, if you are not in your home folder, change directory to reach it
- Navigate to the download folder with **cd Downloads**
- Type the following command: **sudo mkdir /usr/local/mysql**
- Open the browser and head to **https://goo.gl/NLtUUb**
- Download the platform independent tar.gz archive
- Click on "No thanks, just start my download"
- When asked to open or download, select download
- Open the terminal and, if you are not in your home folder, change directory to reach it

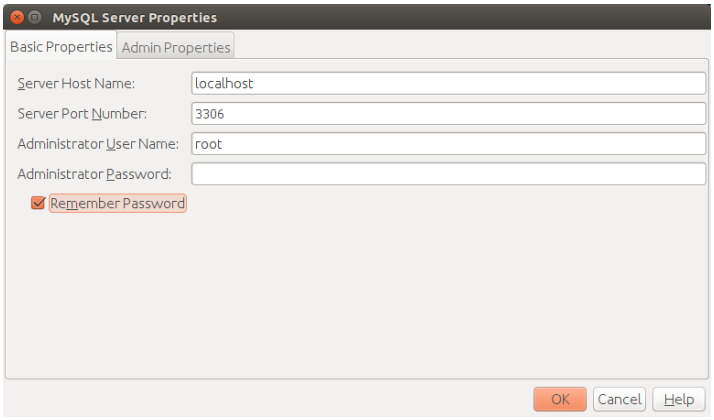
- Navigate to the download folder with **cd Downloads**
- Type the following command: **sudo chmod -R 777 /usr/local/mysql**
- Type the following command: **tar zxvf mysql-connector-java-5.1.40.tar.gz -C /usr/local/mysql**
- NOTE: in the following steps, before strip-components there are actually **two dashes** (not one)!
- Type the following command: **tar zxvf mysql-5.7.16-linux-glibc2.5-x86_64.tar.gz -C /usr/local/mysql –strip-components=1**
- NOTE: at this point of the installation process, it is necessary to install the libaio1 library. Unfortunately, there is no universal binary, so you will have to install it by means of the package manager of your linux distribution.
  As an example, for Ubuntu/Debian, type the following command: **sudo apt-get install libaio1**
- Type the following command: **cd /usr/local/mysql/bin**
- NOTE: in the following step, before initialize-insecure and user there are actually **two dashes** (not one)!
- Type the following command: **./mysqld –initialize-insecure –user=root**
- Close the terminal.

2.4 Configure MySQL integration with NetBeans

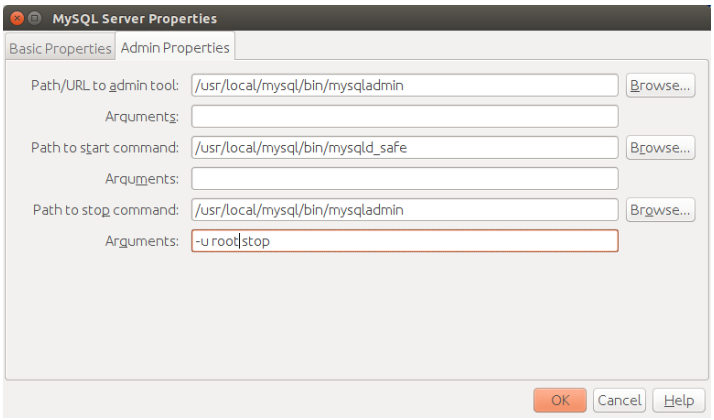
- Open NetBeans
- Select the Services Panel
- Right click on Databases and select Register MySQL Server



- In the window, verify that the Administrator Password field is empty; leave everything else unchanged
- Tick the save password checkbox



- Click on the Admin properties tab
- Path to admin tool: /usr/local/mysql/bin/mysqladmin
- Path to start command: /usr/local/mysql/bin/mysqld_safe
- Path to stop command: /usr/local/mysql/bin/mysqladmin
- Stop command arguments: -u root stop



- Click ok

2.5 Manage MySQL from within NetBeans

Here you can find the basic instructions to create a database and start and stop the MySQL Server.
The same instructions could also be followed to manage a different DBMS Server, if properly configured.
For further reference: **https://netbeans.org/kb/docs/ide/mysql.html**

2.5.1 Start MySQL Server

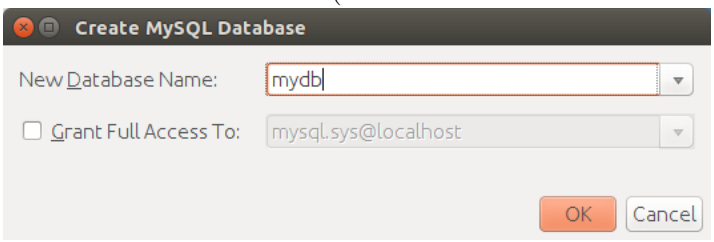
- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: start server

2.5.2 Stop MySQL Server

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: stop server

2.5.3 Create a database

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: Create Database (for this to work the server must be running)

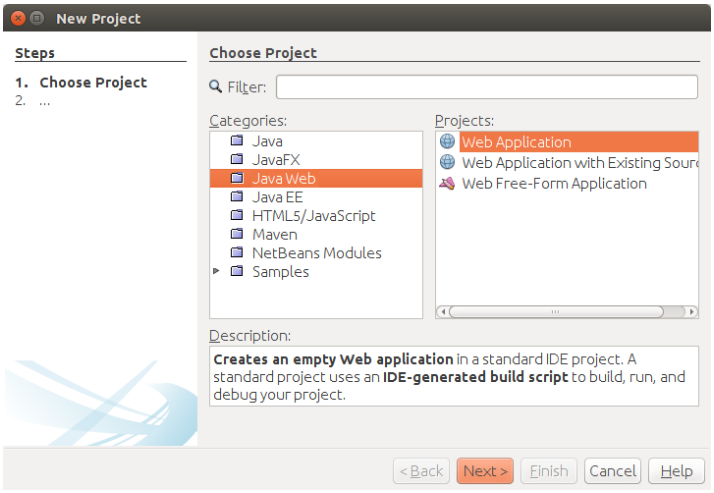


- This will also automatically create a JDBC connection

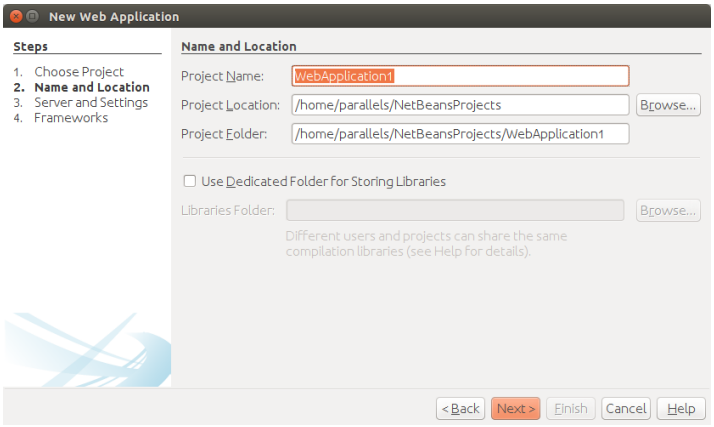
2.6 Create a new Web Application Project

- Click on the File menu and select New Project.

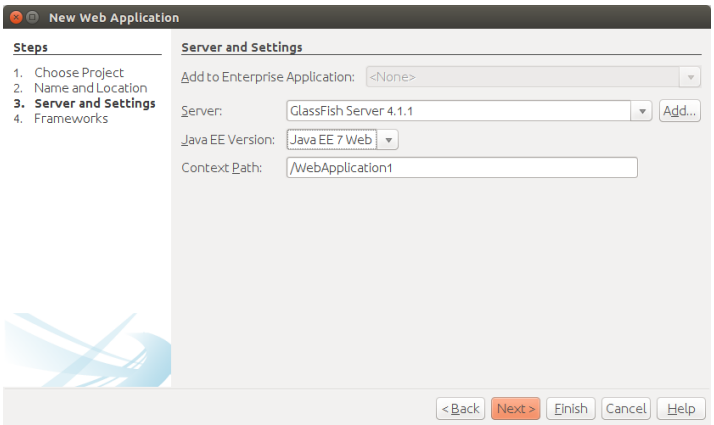
- Select Java Web, Web Application.



- Choose a name for your Web Application.



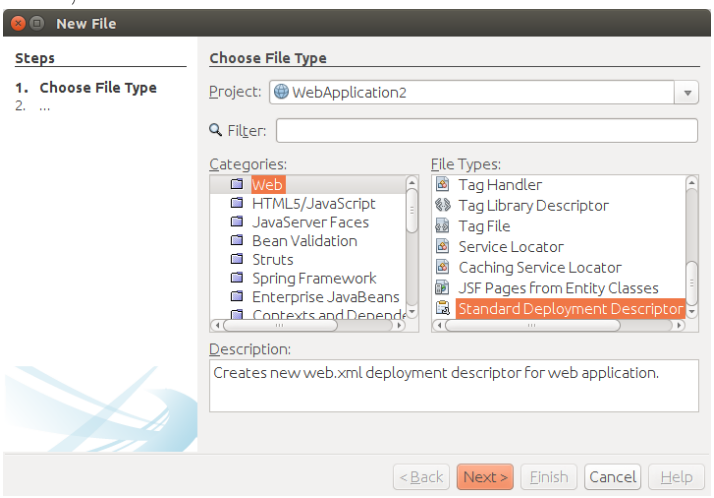
- Check that the project is using Java EE 7 and GlassFish 4.



- Click Finish.
- Right click on your newly created project in the project view and select

New − > Other.

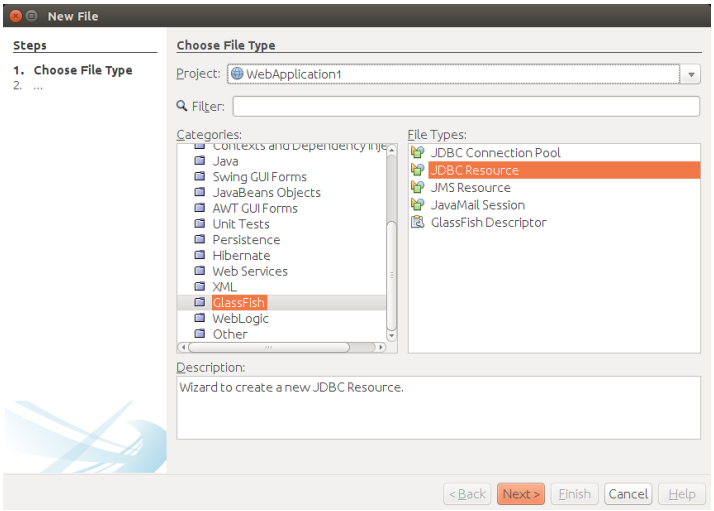
- In the window that opens, select the Web category and then, from the list on the right, choose Standard Deployment Descriptor (web.xml). Then, click Next.



- Click Finish.

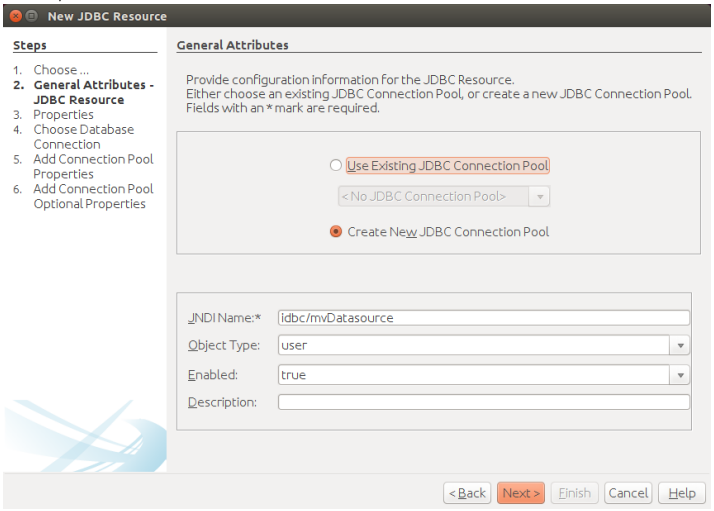
2.7 Create a connection pool for the project

- Open the New File wizard.
- Select the GlassFish server category, then JDBC Resource and click Next.

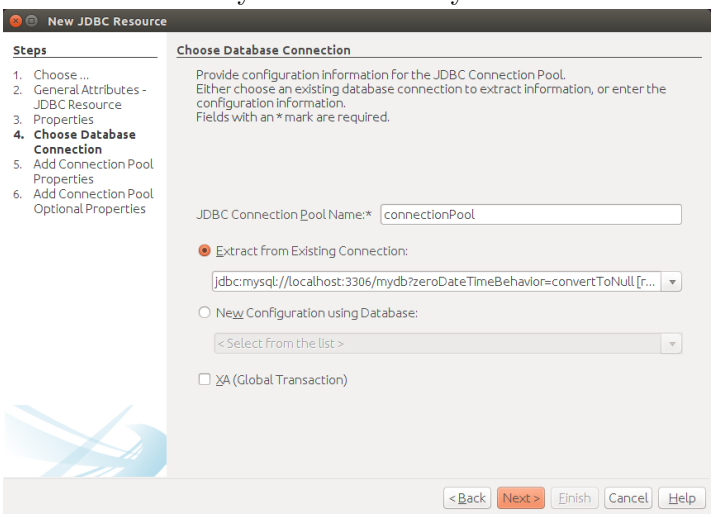


- Select "Create New JDBC connection pool".

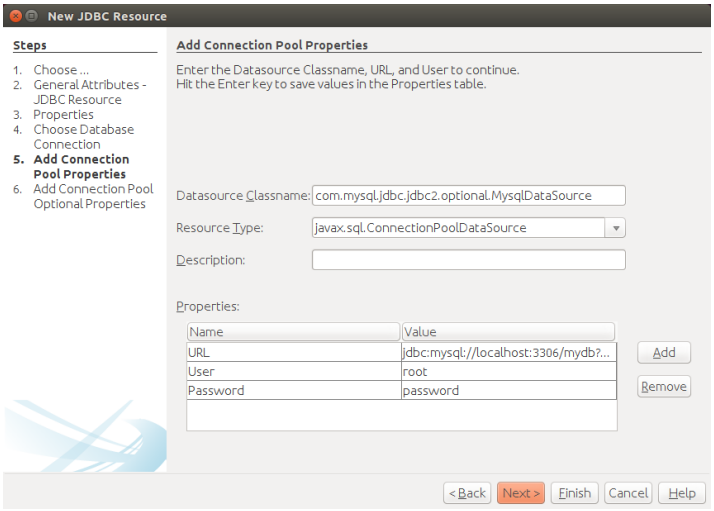
- Choose a JDNI resource name. In this example, Im using the default jdbc/myDatasource. Click Next.



- Click Next again.
- Choose a name for your connection Pool.
- Select "Extract from existing connection"; then, from the dropdown menu select the MyDB connection you created before.



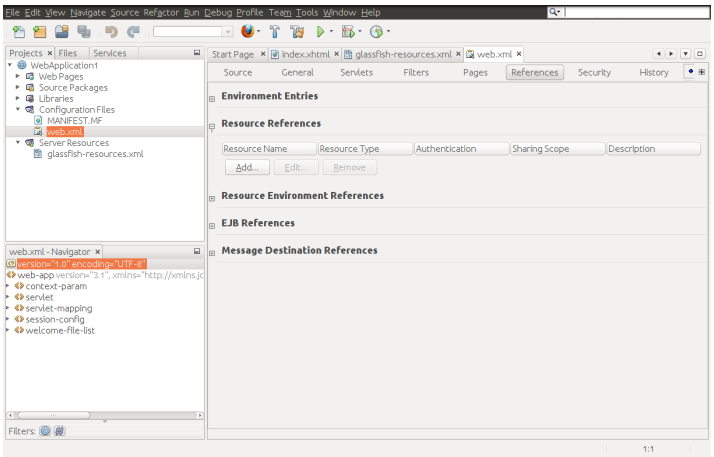
- In the Resource Type menu, choose javax.sql.ConnectionPoolDataSource. Click Next.



- Click Finish.
- In project view, move glassfish-resource.xml from Configuration Files to Server Resources.

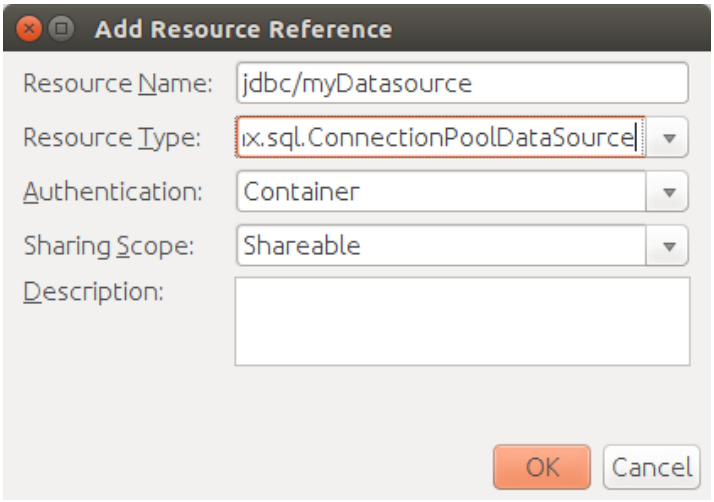
2.8 How to reference the data source from the application

- In the projects window, expand Configuration Files and double click web.xml.
- Click the references tab.
- Expand the resource references heading and click Add.



- As the Resource Name, use the one defined before (in this example, jdbc/myDatasource)

- Type javax.sql.ConnectionPoolDataSource in the Resource type field. Click OK.

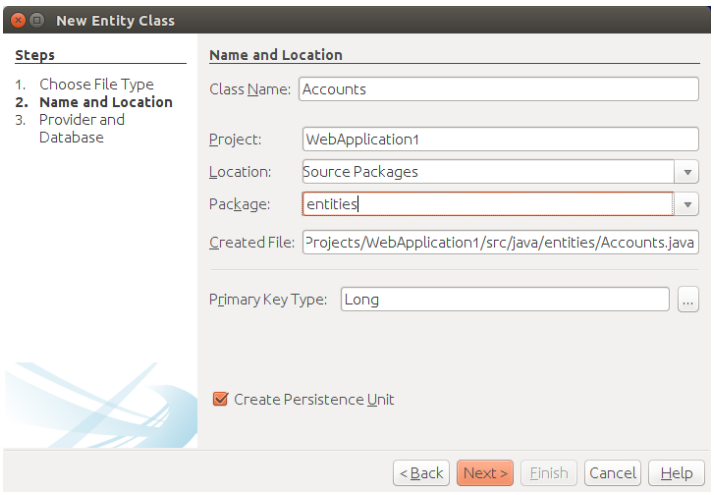


2.9 Create a new package

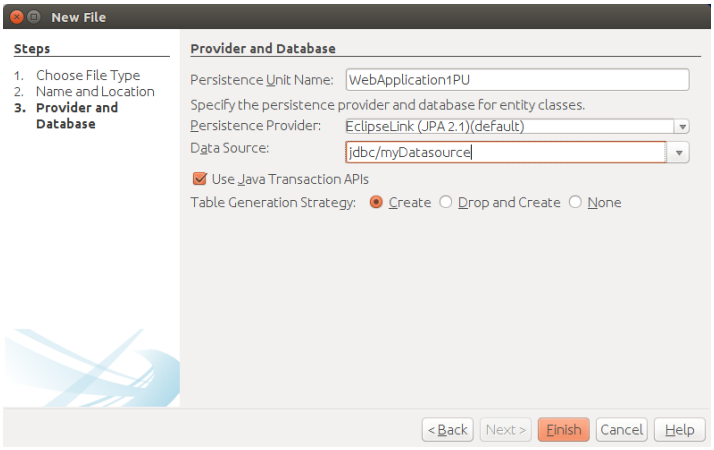
- Right click on your newly created project in the project view and select New − > Java Package.
- In the window that opens, type the name of the package that you want to create.
- Click Finish.

2.10 Create a new Entity Class

- Right click on your newly created project in the project view and select New − > Entity Class.
- In the window that opens, type the name of your Entity. Remember, each Entity Class implicitly corresponds to a table in your database.
- Under the Package dropdown menu, select the name of the package that you previously created for your Entity classes.
- If this is the **first time** you create an Entity Class in your project, make sure that the checkbox "Create Persistence Unit" is selected and click Next. Otherwise, click Finish.



- Choose a name for the Java Persistence Unit.
- From the dropdown menu, select the connection to your database.
- Make sure that the checkbox "Use Java Transaction APIs" is ticked and that "Create" is selected for "Table Generation Strategy". This will create a table in your database that corresponds to your new Entity Class.



- Click Finish.

2.11 How to deploy an application

- In the project panel, right click on the application
- Click Deploy

2.12 How to run an application

- In the project panel, right click on the application
- Click Run
- The web browser will automatically open on the main page of your application.

Chapter 3

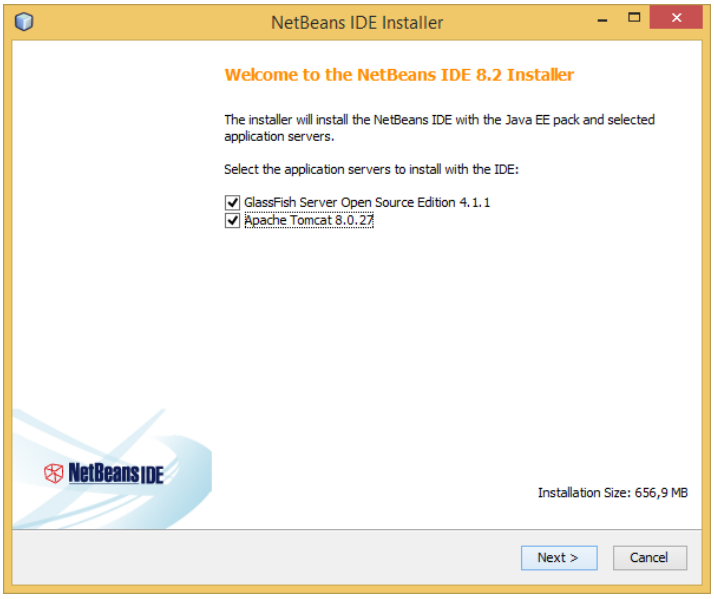
Windows

3.1 JDK Configuration

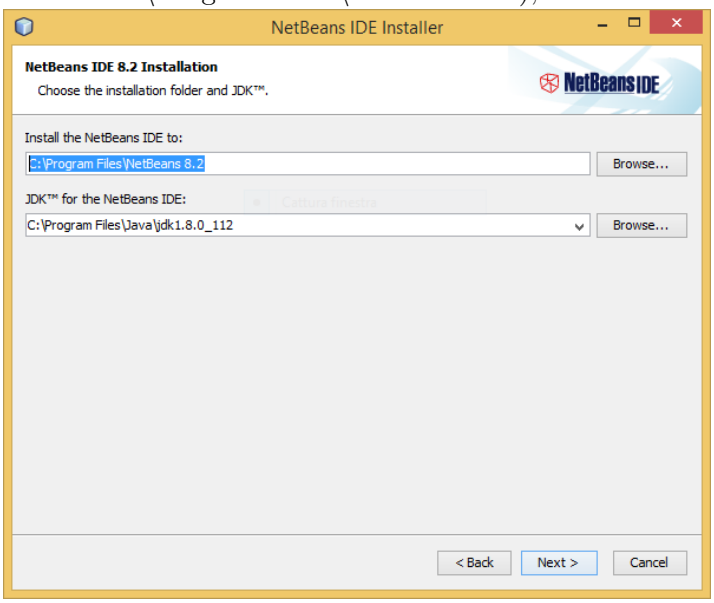
- Open the browser and head to **https://goo.gl/jXOIDJ**
- Accept the license agreement (suggested version: Java SE Development Kit 8u112)
- Click on the Windows file corresponding to your platform (Windows x86 / Windows x64)
- When asked to open or download, select download
- Open the installer and follow the instructions

3.2 Install NetBeans Enterprise Edition

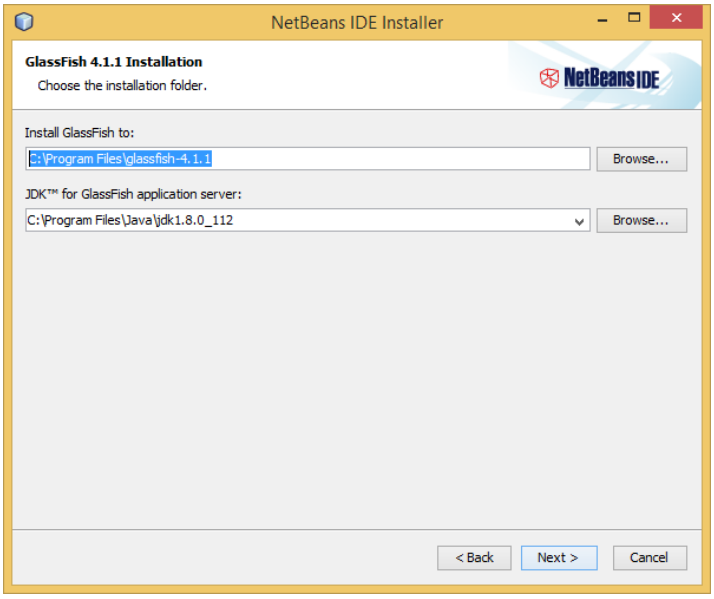
- Open the browser and head to **https://goo.gl/T0IdR0**
- Select the Windows platform
- Click on the download button at the bottom of the Java EE column (save the .sh file)
- Open the installer
- The window of the installer wizard will open
- Select both GlassFish Server (default) and Apache Tomcat, then click Next



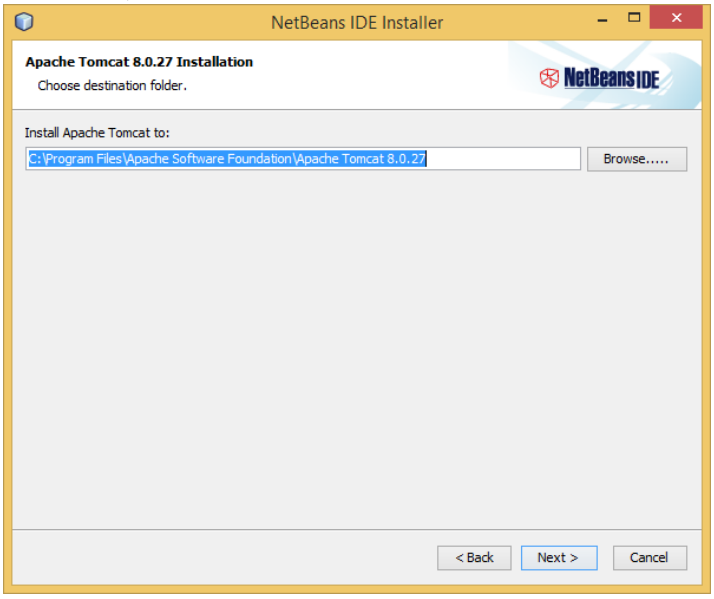
- Accept agreement, then click Next
- Take note of the location where NetBeans will be installed at (by default: C:\Program Files\NetBeans 8.2), then click next



- Take note of the location where GlassFish will be installed at (by default: C:\Program Files\glassfish-4.1.1), then click next



- Take note of the location where Tomcat will be installed at (by default: C:\Program Files\Apache Software Foundation\Apache Tomcat 8.0.27), then click next

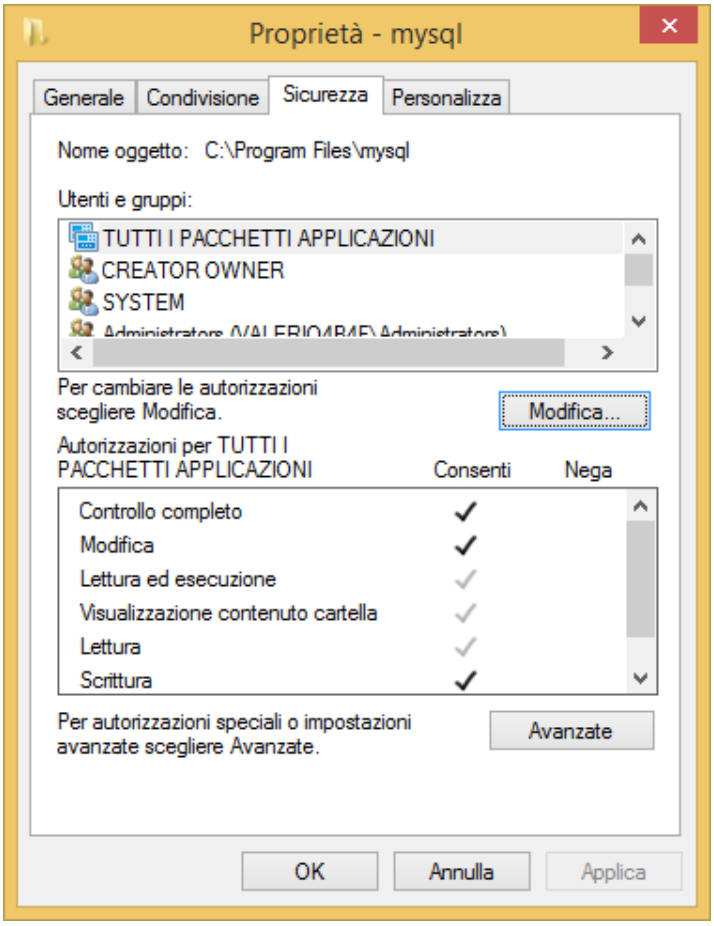


3.3 Install MySQL

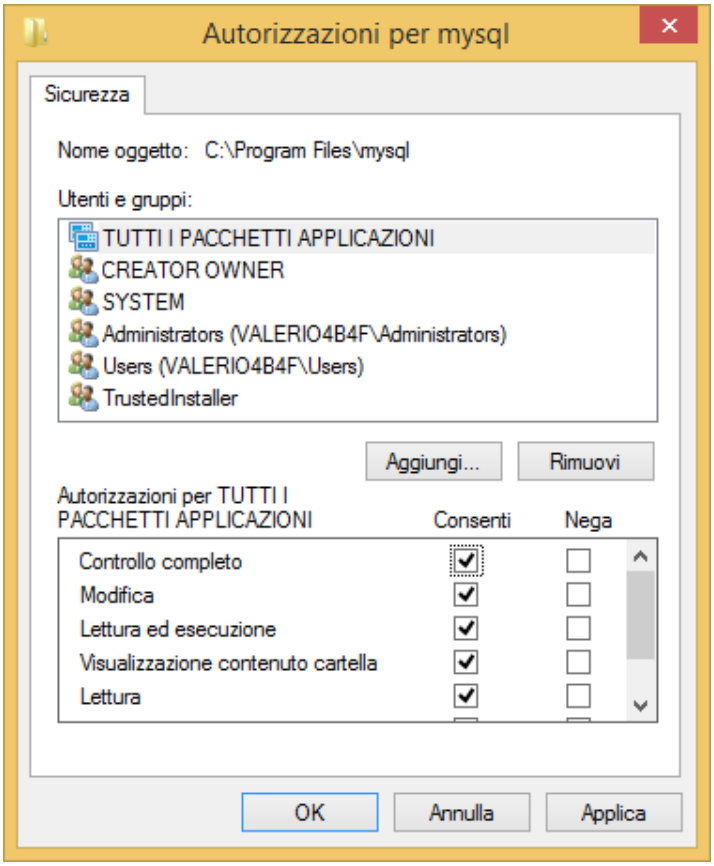
- Open the browser and head to **https://goo.gl/x14GMC**
- From the dropdown menu, select Windows
- Download the Windows ZIP archive for your architecture (32 bit/64

bit)

- Click on "No thanks, just start my download"
- When asked to open or download, select download
- Extract the folder and copy it into C:\Program Files\mysql
- Navigate to C:\Program Files
- Right click on mysql folder
- Select Properties
- In the General tab, untick read only and propagate the change to all the files within the folder



- In the Security tab, for "All applications packages", for your own user and for CREATOR OWNER, click on Modify and tick "Full control"



- NOTE: due to an unidentified bug in how Windows handles MySQL, the Security tab must be reconfigured at each reboot of the OS before using NetBeans, otherwise MySQL will refuse to open a connection.
- Open command prompt and navigate to C:\Program Files\mysql\bin with **cd C:\Program Files\mysql\bin**
- NOTE: in the following step, before initialize-insecure and user there are actually **two dashes** (not one)!
- Type the following command: **mysqld –initialize-insecure –user=root**
- Close the command prompt.

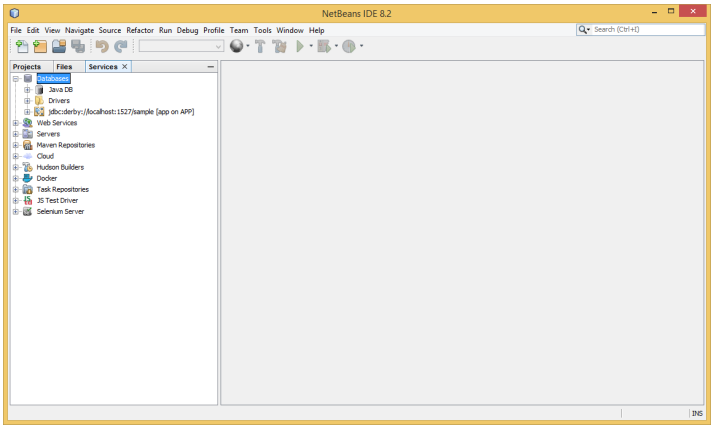
3.4 Install MySQL connectors

- Open the browser and head to **https://goo.gl/NLtUUb**
- Download the platform independent zip archive
- Click on "No thanks, just start my download"

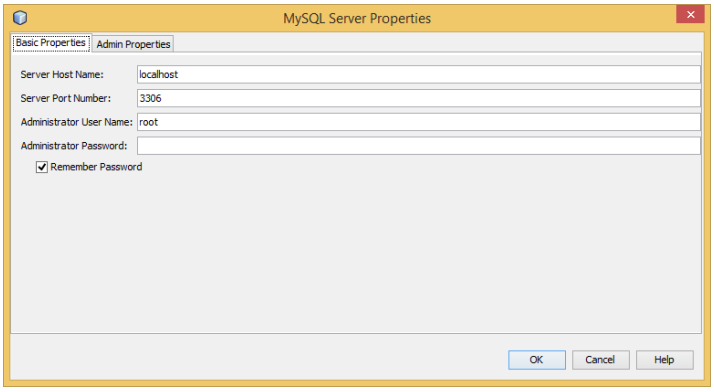
- When asked to open or download, select download
- Extract in C:\Program Files\mysql

3.5 Configure MySQL integration with NetBeans

- Open NetBeans
- Select the Services Panel
- Right click on Databases and select Register MySQL Server

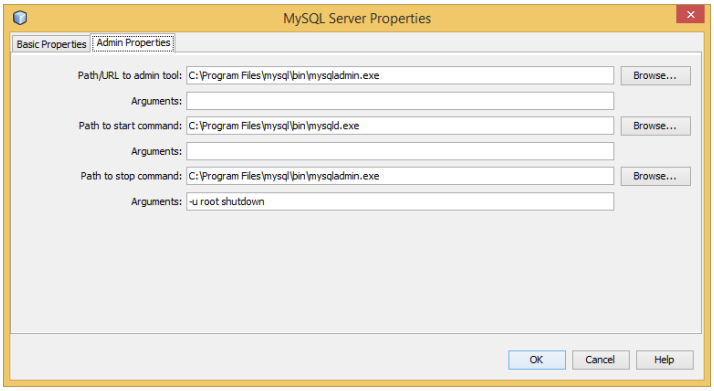


- In the window, verify that the Administrator Password field is empty; leave everything else unchanged
- Tick the save password checkbox



- Click on the Admin properties tab
- Path to admin tool: C:\Program Files\mysql\bin\mysqladmin.exe
- Path to start command: C:\Program Files\mysql\bin\mysqld.exe
- Path to stop command: C:\Program Files\mysql\bin\mysqladmin.exe

- Stop command arguments: -u root shutdown



- Click ok

3.6 Manage the SQL Server

Here you can find the basic instructions to create a database and start and stop the SQL Server.
For further reference: **https://netbeans.org/kb/docs/ide/mysql.html**

3.6.1 Start the SQL Server

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: start server

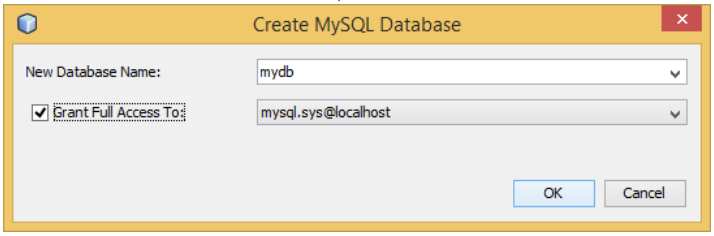
3.6.2 Stop the SQL Server

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: stop server

3.6.3 Create a database

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server

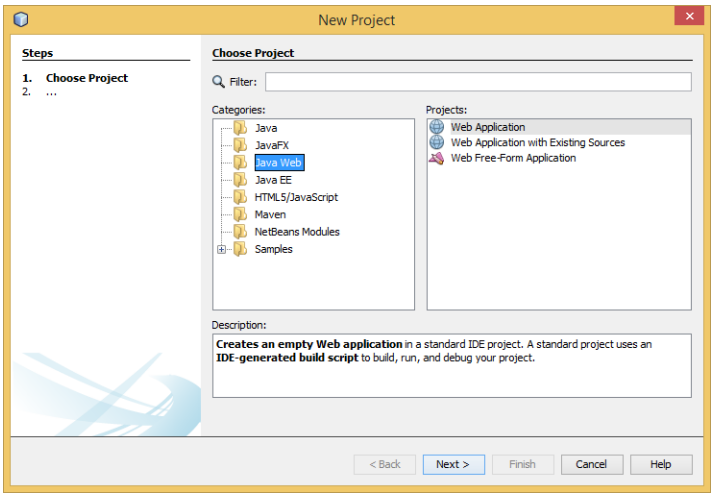
- Choose: Create Database (for this to work the server must be running)



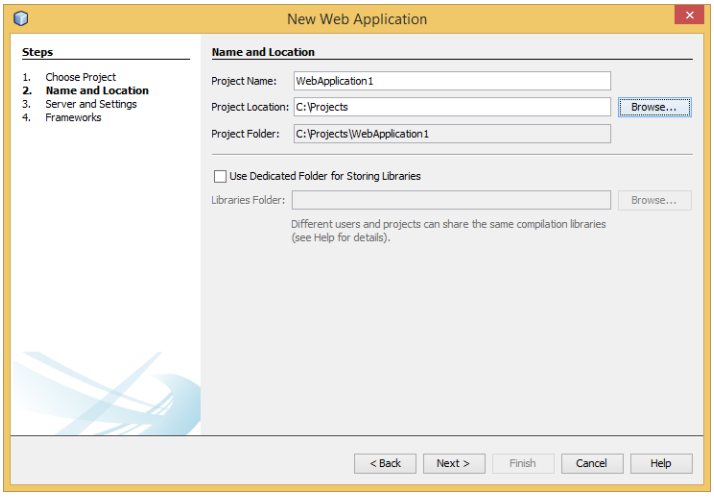
- This will also automatically create a JDBC connection

3.7 Create a new Web Application Project

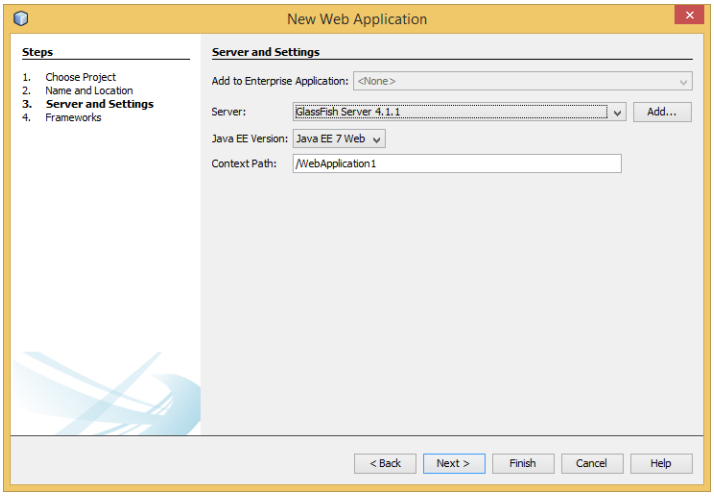
- Click on the File menu and select New Project.
- Select Java Web, Web Application.



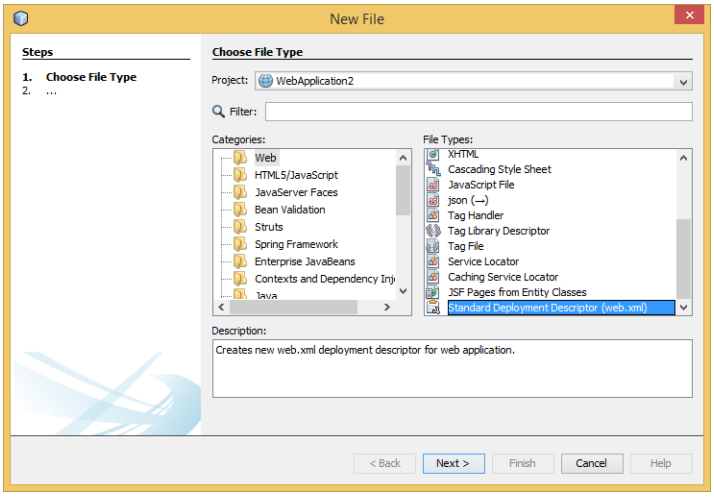
- Choose a name for your Web Application.



- Check that the project is using Java EE 7 and GlassFish 4.



- Click Finish.
- Right click on your newly created project in the project view and select New $->$ Other.
- In the window that opens, select the Web category and then, from the list on the right, choose Standard Deployment Descriptor (web.xml). Then, click Next.

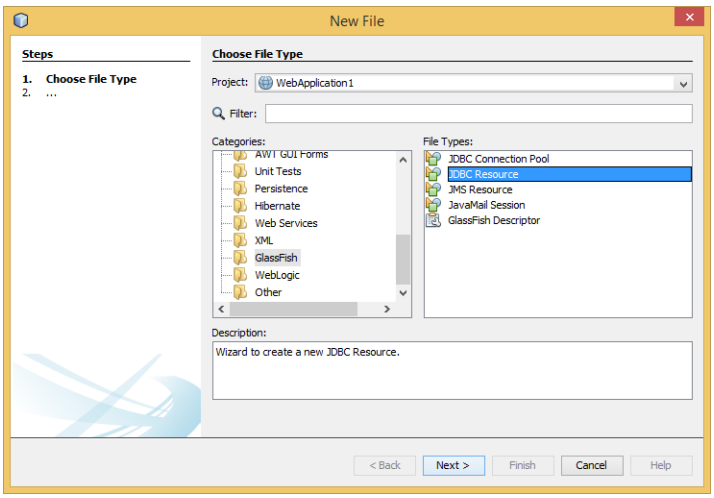


- Click Finish.

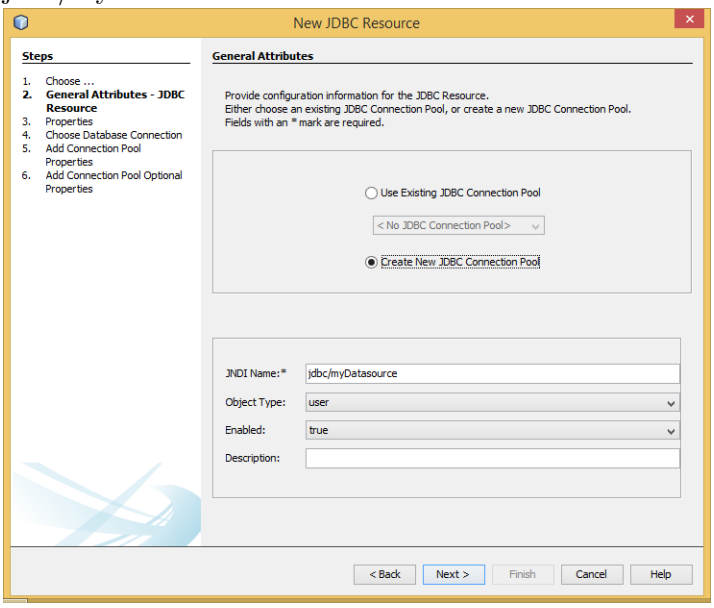
3.8 Create a connection pool for the project

- Open the New File wizard.

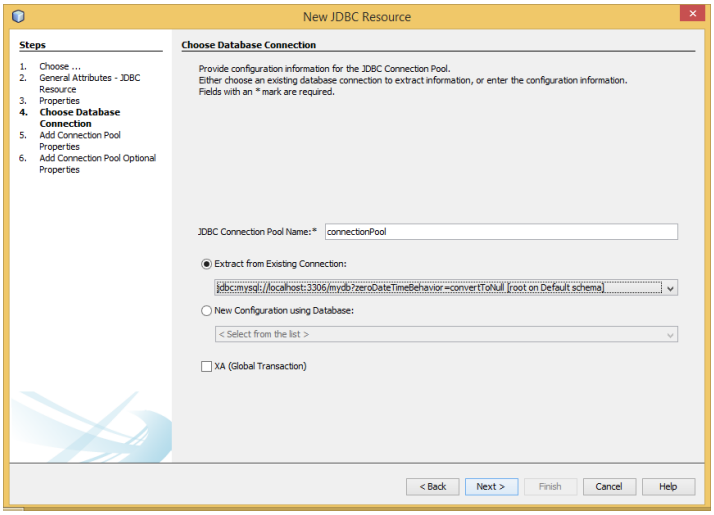
- Select the GlassFish server category, then JDBC Resource and click Next.



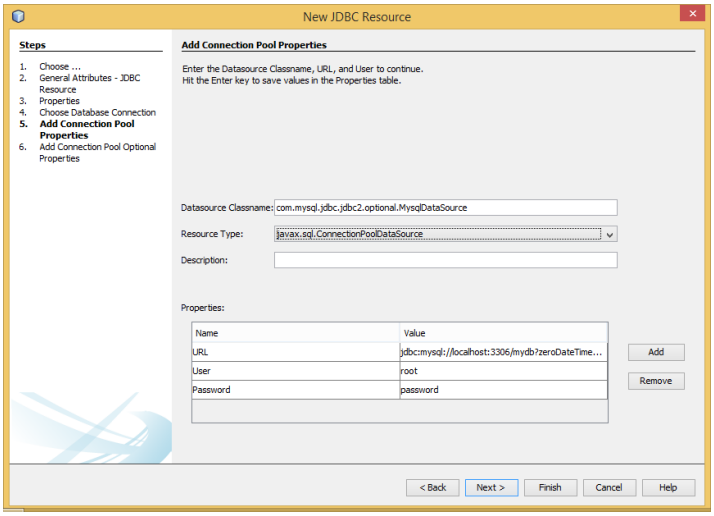
- Select "Create New JDBC connection pool".
- Choose a JDNI resource name. In this example, Im using the default jdbc/myDatasource. Click Next.



- Click Next again.
- Choose a name for your connection Pool.
- Select "Extract from existing connection"; then, from the dropdown menu select the MyDB connection you created before.



- In the Resource Type menu, choose javax.sql.ConnectionPoolDataSource. Click Next.

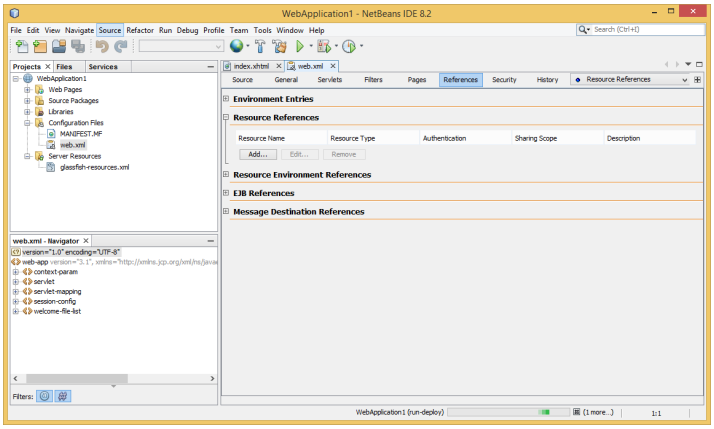


- Click Finish.
- In project view, move glassfish-resource.xml from Configuration Files to Server Resources.

3.9 How to reference the data source from the application

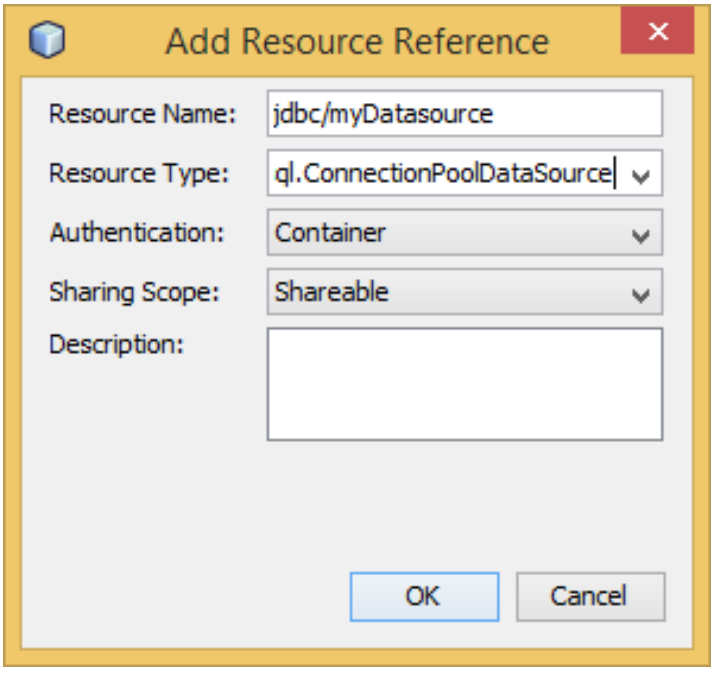
- In the projects window, expand Configuration Files and double click web.xml.
- Click the references tab.

- Expand the resource references heading and click Add.



- As the Resource Name, use the one defined before (in this example, jdbc/myDatasource)

- Type javax.sql.ConnectionPoolDataSource in the Resource type field. Click OK.



3.10 Create a new package

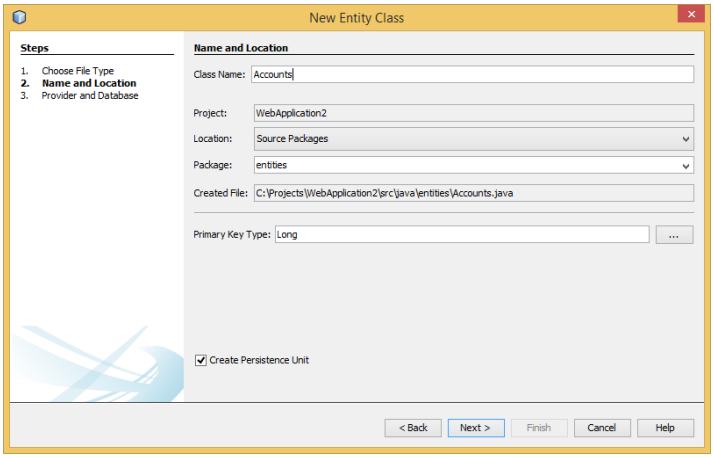
- Right click on your newly created project in the project view and select New $->$ Java Package.

- In the window that opens, type the name of the package that you want to create.

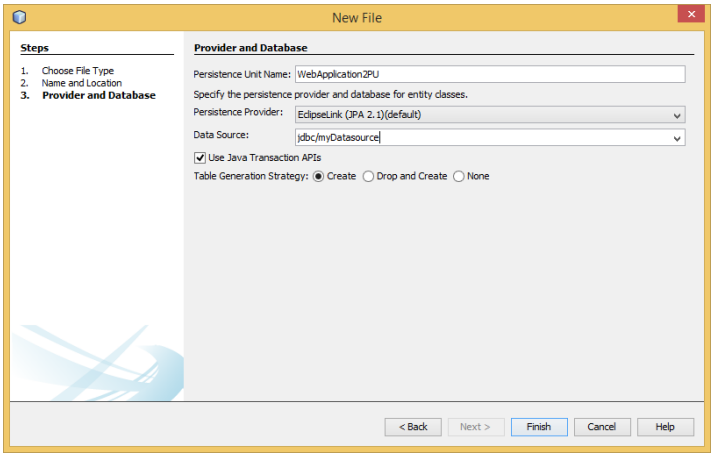
- Click Finish.

3.11 Create a new Entity Class

- Right click on your newly created project in the project view and select New $->$ Entity Class.
- In the window that opens, type the name of your Entity. Remember, each Entity Class implicitly corresponds to a table in your database.
- Under the Package dropdown menu, select the name of the package that you previously created for your Entity classes.
- If this is the **first time** you create an Entity Class in your project, make sure that the checkbox "Create Persistence Unit" is selected and click Next. Otherwise, click Finish.



- Choose a name for the Java Persistence Unit.
- From the dropdown menu, select the connection to your database.
- Make sure that the checkbox "Use Java Transaction APIs" is ticked and that "Create" is selected for "Table Generation Strategy". This will create a table in your database that corresponds to your new Entity Class.



- Click Finish.

3.12 How to deploy an application

- In the project panel, right click on the application
- Click Deploy

3.13 How to run an application

- In the project panel, right click on the application
- Click Run
- The web browser will automatically open on the main page of your application.

Chapter 4

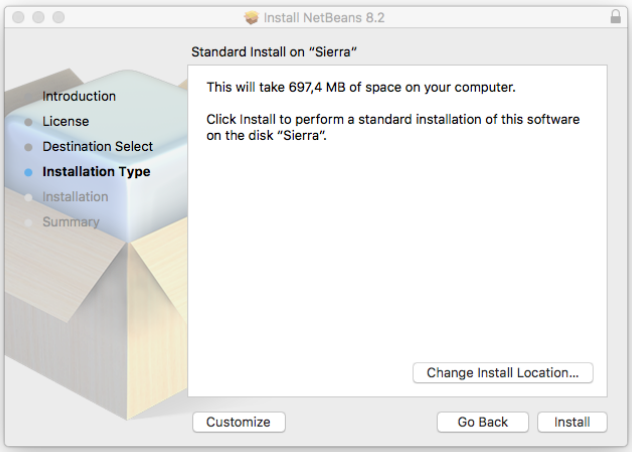
macOS

4.1 JDK Configuration

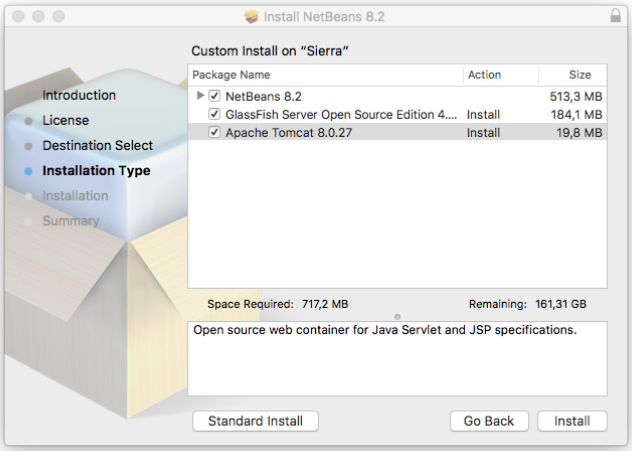
- Open the browser and head to **https://goo.gl/jXOIDJ**
- Accept the license agreement (suggested version: Java SE Development Kit 8u112)
- Click on the Mac OS X file
- When asked to open or download, select download
- Open the installer and follow the instructions

4.2 Install NetBeans Enterprise Edition

- Open the browser and head to **https://goo.gl/T0IdR0**
- Select the Mac OS X platform
- Click on the download button at the bottom of the Java EE column (save the .sh file)
- Mount the DMG image
- Open the installer package
- Accept agreement, then click Next
- When presented with the possibility to customise the install, select customize



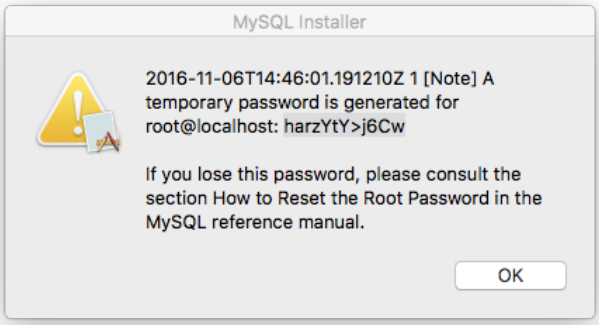
- Select both GlassFish Server (default) and Apache Tomcat, then click Install



4.3 Install MySQL and MySQL connectors

- Open the browser and head to **https://goo.gl/x14GMC**
- From the dropdown menu, select Mac OS X
- Download Mac OS X DMG archive
- Click on "No thanks, just start my download"
- If asked to open or download, select download
- Mount the dmg image

- Open the installer and follow the instructions
- Write down the password for root@localhost which is generated at the end of the install process



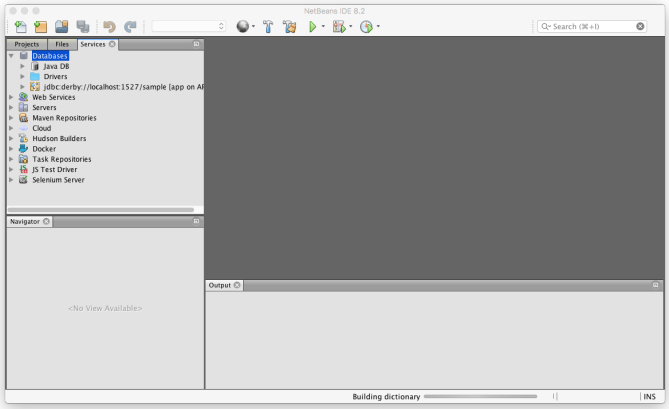
- Open the browser and head to **https://goo.gl/NLtUUb**
- Download the platform independent zip archive
- Download the platform independent zip archive
- Click on "No thanks, just start my download"
- If asked to open or download, select download
- In Finder, select the Go menu
- Click on Go to Folder
- Enter the following path: **/usr/local/mysql-5.7.16-osx10.11-x86_64**
- Move the extracted MySQL connector folder in **/usr/local/mysql-5.7.16-osx10.11-x86_64**(requires authentication, type in your password)
- Open terminal
- Type the following command: **sudo chown -R _mysql:admin /usr/local/mysql-5.7.16-osx10.11-x86_64**
- You will be prompted to enter a password. This is the password of your account, **NOT** the temporary password of MySQL.
- Type the following command: **sudo chmod -R u+rwX,g+rwX,o-rwx /usr/local/mysql-5.7.16-osx10.11-x86_64**
- Type the following command: **sudo chmod -R g+rwx /usr/local/mysql-5.7.16-osx10.11-x86_64**

- Type the following command: **cd /usr/local/mysql-5.7.16-osx10.11-x86_64/bin**
- NOTE: in the following steps, before initialize-insecure, user and skip-password there are actually **two dashes** (not one)! Also, please note that the final ampersand (&) symbol is actually part of the command; if you forget to put it, you will not be able to return to the command line.
- Type the following command: **sudo killall mysqld**
- NOTE: this command must be repeated until the terminal prints "No matching processes were found".
- Type the following command: **sudo ./mysqld –user=mysql &**
- NOTE: if the command fails with the error **InnoDB: Check that you do not already have another mysqld process using the same InnoDB data or log files**, repeat the above step until you are sure that no copy of mysqld is running anymore.
- Type again enter to return to the command line.
- Type the following command: **sudo ./mysql -u root -h 127.0.0.1 -p**
- You will be prompted to enter a password. This is the temporary password that you took note of at the end of the installation process.
- NOTE: in the following command, write **exactly** the word **password** and don't forget to terminate the line with a semicolon!
- Type the following command: **ALTER USER 'root'@'localhost' IDENTIFIED BY 'password';**
- Type **exit**
- Type the following command: **sudo killall mysqld**
- NOTE: this command must be repeated until the terminal prints "No matching processes were found".
- Close the terminal.

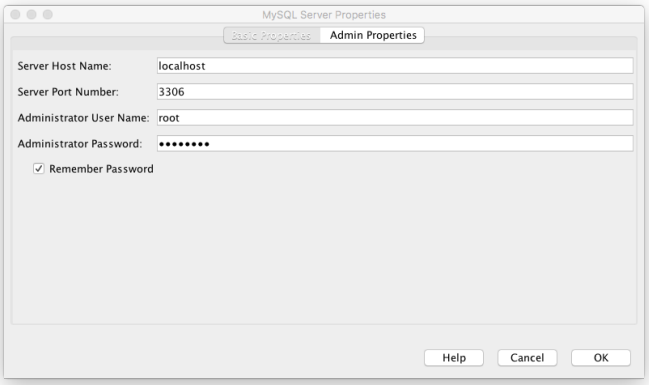
4.4 Configure MySQL integration with NetBeans

- Open NetBeans
- Select the Services Panel

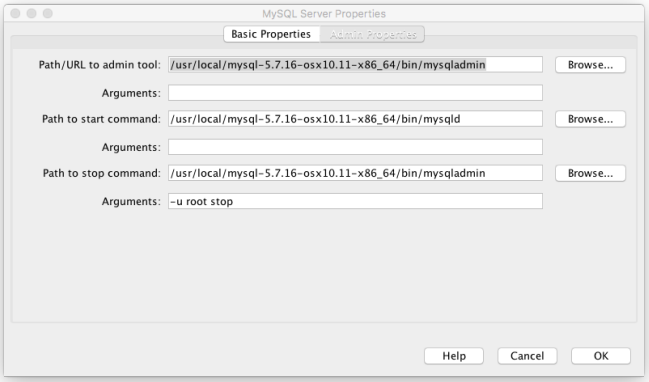
- Right click on Databases and select Register MySQL Server



- In the window, set the Administrator Password field to **password**; leave everything else unchanged
- Tick the save password checkbox



- Click on the Admin properties tab
- Path to admin tool: /usr/local/mysql-5.7.16-osx10.11-x86_64/bin/mysqladmin
- Path to start command: /usr/local/mysql-5.7.16-osx10.11-x86_64/bin/mysqld
- Path to stop command: /usr/local/mysql-5.7.16-osx10.11-x86_64/bin/mysqladmin
- Stop command arguments: -u root stop



- Click ok

4.5 Manage the SQL Server

Here you can find the basic instructions to create a database and start and stop the SQL Server.
For further reference: **https://netbeans.org/kb/docs/ide/mysql.html**

4.5.1 Start the SQL Server

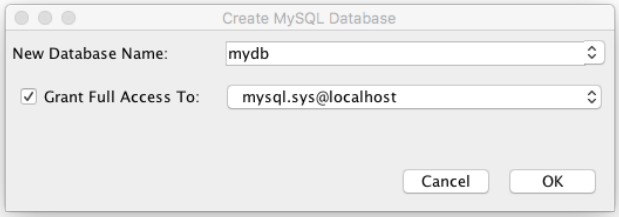
- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: start server

4.5.2 Stop the SQL Server

- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: disconnect
- NOTE: DO NOT choose stop! This is a known bug in the macOS version of the IDE and there is no solution so far.

4.5.3 Create a database

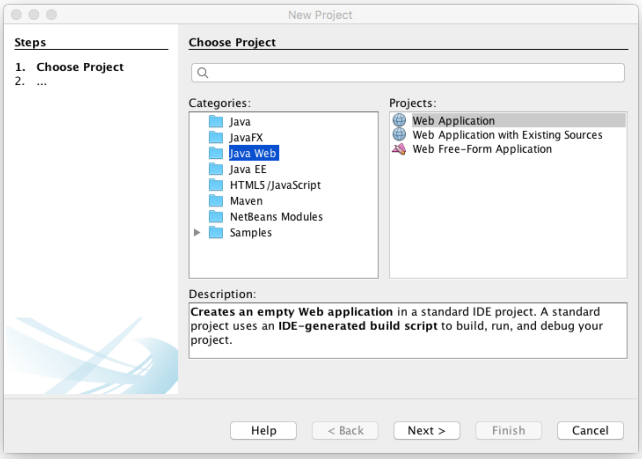
- Select the Service Panel
- Expand the Databases menu
- Right click on MySQL Server
- Choose: Create Database (for this to work the server must be running)



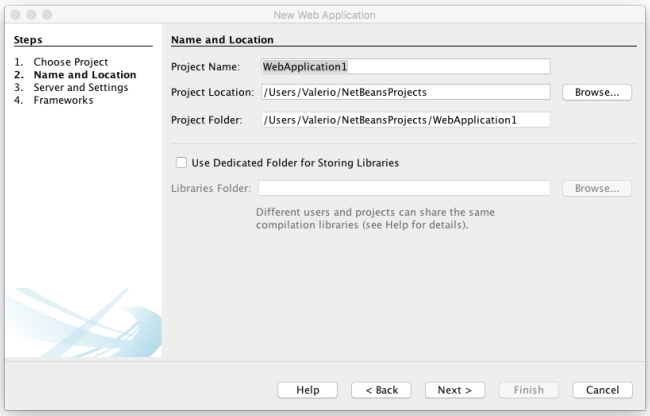
- This will also automatically create a JDBC connection

4.6 Create a new Web Application Project

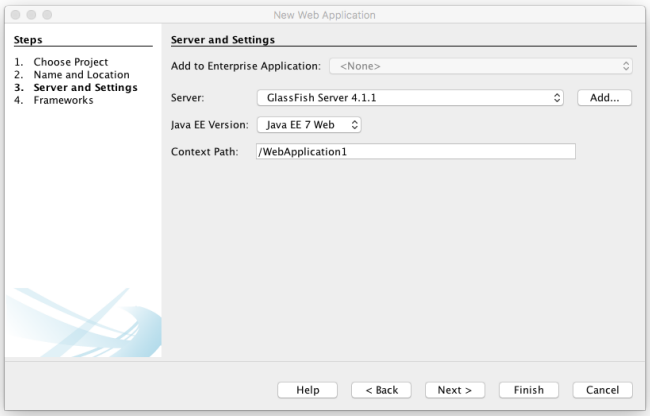
- Click on the File menu and select New Project.
- Select Java Web, Web Application.



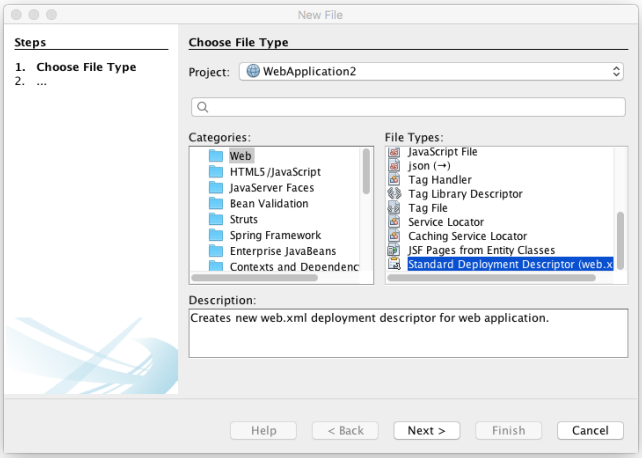
- Choose a name for your Web Application.



- Check that the project is using Java EE 7 and GlassFish 4.



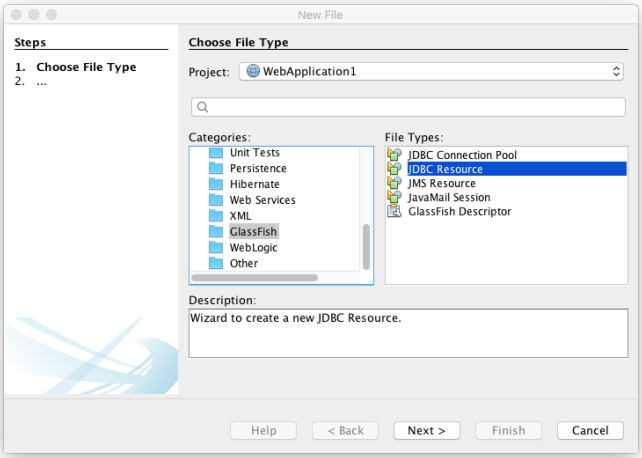
- Click Finish.
- Right click on your newly created project in the project view and select New $->$ Other.
- In the window that opens, select the Web category and then, from the list on the right, choose Standard Deployment Descriptor (web.xml). Then, click Next.



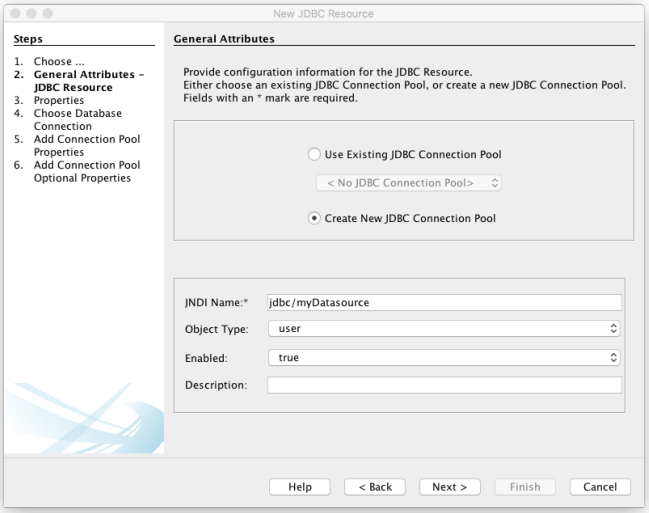
- Click Finish.

4.7 Create a connection pool for the project

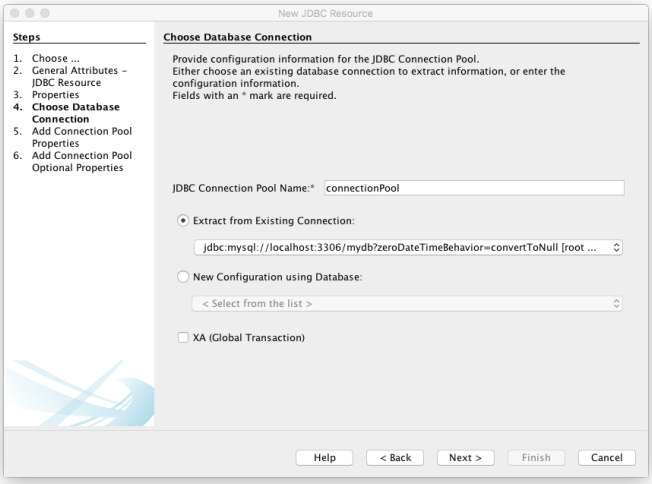
- Open the New File wizard.
- Select the GlassFish server category, then JDBC Resource and click Next.



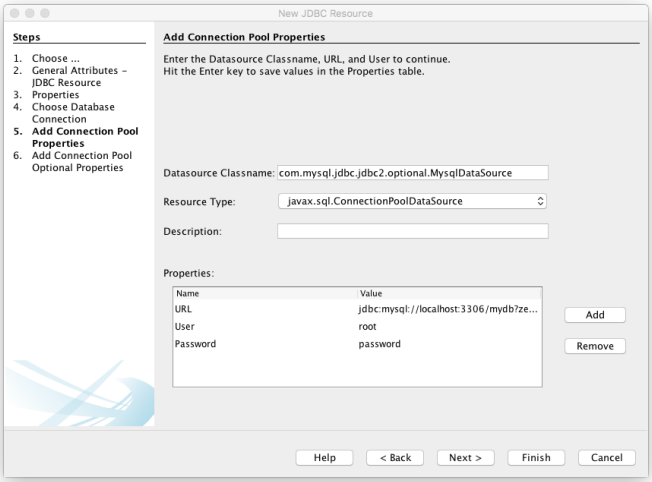
- Select "Create New JDBC connection pool".
- Choose a JDNI resource name. In this example, Im using the default jdbc/myDatasource. Click Next.



- Click Next again.
- Choose a name for your connection Pool.
- Select "Extract from existing connection"; then, from the dropdown menu select the MyDB connection you created before.



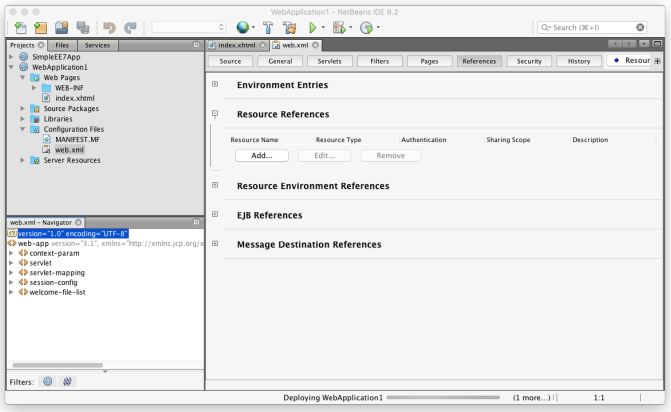
- In the Resource Type menu, choose javax.sql.ConnectionPoolDataSource. Click Next.



- Click Finish.
- In project view, move glassfish-resource.xml from Configuration Files to Server Resources.

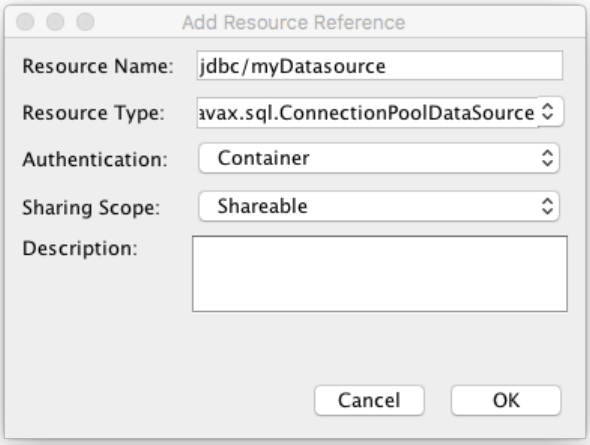
4.8 How to reference the data source from the application

- In the projects window, expand Configuration Files and double click web.xml.
- Click the references tab.
- Expand the resource references heading and click Add.



- As the Resource Name, use the one defined before (in this example, jdbc/myDatasource)

- Type javax.sql.ConnectionPoolDataSource in the Resource type field. Click OK.

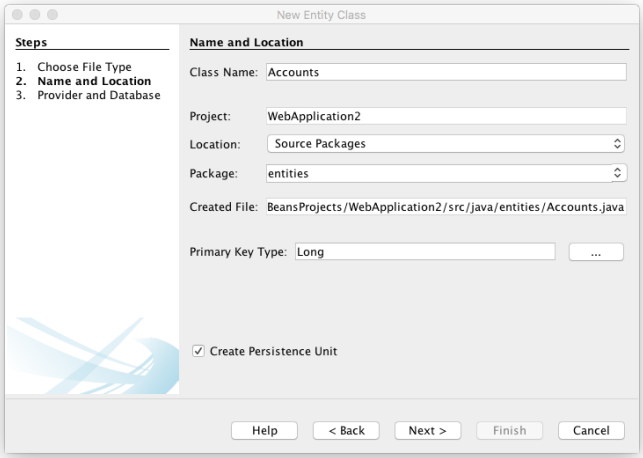


4.9 Create a new package

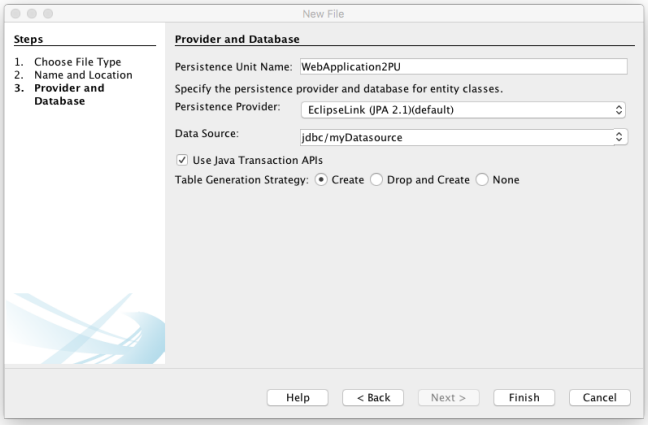
- Right click on your newly created project in the project view and select New $->$ Java Package.
- In the window that opens, type the name of the package that you want to create.
- Click Finish.

4.10 Create a new Entity Class

- Right click on your newly created project in the project view and select New $->$ Entity Class.
- In the window that opens, type the name of your Entity. Remember, each Entity Class implicitly corresponds to a table in your database.
- Under the Package dropdown menu, select the name of the package that you previously created for your Entity classes.
- If this is the **first time** you create an Entity Class in your project, make sure that the checkbox "Create Persistence Unit" is selected and click Next. Otherwise, click Finish.



- Choose a name for the Java Persistence Unit.
- From the dropdown menu, select the connection to your database.
- Make sure that the checkbox "Use Java Transaction APIs" is ticked and that "Create" is selected for "Table Generation Strategy". This will create a table in your database that corresponds to your new Entity Class.



- Click Finish.

4.11 How to deploy an application

- In the project panel, right click on the application
- Click Deploy

4.12 How to run an application

- In the project panel, right click on the application
- Click Run
- The web browser will automatically open on the main page of your application.