

Lab 3

In this lab you learn how to integrate a real database in your web application.

In the previous lab you created a desktop class retrieving all persons from a PostgreSQL database.

Now we will use this database in our web application, to manage all users and products.

Refactor

Up until now we used 'inMemory' database classes:

- `PersonDbInMemory`,
- `ProductDbInMemory`.

As those classes are useful while we are developing our web application (no overhead connecting to a real database) we would like to keep them. This means we will have to create extra database classes to work with the SQL database.

Before we start developing those, we will refactor our current model:

1. Create an interface `PersonDb`.
Tip:
 - Right mouse-click
 - Choose *Refactor*
 - Choose *Extract Interface...*
2. `PersonDbInMemory` should implement this interface.
3. Modify `ShopService`, so it has an instance variable of type `PersonDb`.
4. Perform the same steps for `Product`.

Quiz: while we were refactoring our model, we did not have to change anything in the Controller Servlet. Which OO pattern do you recognize?

New Database classes

Create Java database classes to store your products and users in a SQL database:

1. Add the PostgreSQL driver (`postgresql-42.1.4.jar`) in `Webcontent/WEB-INF/lib`. You can find this file on Toledo.
2. In the package "db", create a class that implements the interface `PersonDb`. It will have the same method signature as your in-memory database class, but instead of using a map you will connect to the real database.
3. In "ShopService", use the class you just created instead of the in-memory database class;
4. Run your project. If everything went right, your application is still working, but the person-data are permanently stored in the database.
5. Run the testclass. find a way to delete all testusers.
6. In the same way, create a new database class for your products.

Quiz: now we can easily switch between database classes, using the either the 'InMemory' version or de SQL version. Which OO pattern do you recognize?

Story Persons – High Priority

Implement StoryW10_ErrorPage.