

# Object Oriented Programming 2

Rolf Haenni & Andres Scheidegger

## Exercises 2

### 1. Storing and Loading Person Data with Text Files

Your JavaFX application from Exercise 1 shall load the person data from a text file at start-up and store back at shut-down. Loading and storing shall be separated from the rest of the application by a *Data Access Object (DAO)*. This is an often-used design pattern, when data has to be persisted in some form. Follow these steps:

1. Introduce an Interface for the DAO, e.g. *PersonDAO*. This interface declares two methods for storing and loading a list of *Person* objects.
2. Create a class, which implements the DAO interface and which stores and loads person data in/from a text file.
3. You may optionally write a JUnit test for your class.
4. Overwrite the *init()* and the *close()* methods of your main class . These methods are inherited from Application and may be used to perform actions at start-up and close-down of a JavaFX application. The *init()* method shall load the person data from a text file and the *close()* method shall store the data to the same text file.

### 2. Storing and Loading Person Data with Serialization

Implement a second DAO class, which uses an *ObjectOutputStream* and an *ObjectInputStream* to store the person data by using object serialization. Again, you may test your DAO class in JUnit test.

### 3. Choosing files in the GUI (optional)

Add a *FileChooser* to your JavaFX application to let the user select the files, where data is loaded from or stored to. Add a *Load* and a *Store* button to your main screen (see screenshot on next page).

