

# Assignment 1b

COMP 2526 Object-Oriented Programming with Java

Due on Sunday January 24th at 11:59 PM

## 1 Purpose

Continue to practice basic Java skills, and use a better design to complete an address book program. You will explore the WHY and HOW in object oriented programming.

## 2 Description

For your second assignment, you will begin with the provided collection of six source files, and add the necessary features to complete an address book program. You may only add code to the `ConsoleUserInterface` class and the `AddressBook` class. You may not edit the other classes.

The zip folder `COMP_2526_A1b_Framework` contains an Eclipse project with the starter classes. To use it:

1. Unzip the folder inside your Eclipse workspace
2. In Eclipse, choose File >New >Java Project
3. For Project Name, use the name of the zip folder `COMP_2526_A1b_Framework`
4. Eclipse will detect the existing project in the workspace and you can click Finish.

You may choose to start by examining the compiler error in the `Main` class. Something is missing from the start of the `ConsoleUserInterface` class declaration.

## 3 Requirements

**You must implement the following nine (9) methods in the `ConsoleUserInterface` class:**

1. `readChoice()` reads the user's choice from the console using a `Scanner`
2. `readPerson()` reads in the `Person` data from the console using a `Scanner` and returns the new `Person`
3. `readName()` reads in the name of a `Person` from the console using a `Scanner`
4. `display()` displays a single `Person`'s data
5. `displayAll()` displays all people in the database
6. `run()` performs the address book functions
7. `displayMenu()` displays the available menu
8. `displayErrorMsg()` displays the `String` message passed on to the user interface
9. `displayMsg()` displays the `String` message passed on to the user interface

Hint: nearly all the necessary code can be found in `Main.java` from `assignment1a`.

**You must also implement the following five (5) methods in the `AddressBook` class:**

1. `addPerson()` reads a `Person` from the user interface and adds them to the database

2. `deletePerson()` reads a Person's name from the user interface and tries to delete them from the database. If not successful, this method displays an error message on the user interface. If successful, this method displays the name of the Person deleted and the words " was deleted successfully" on the user interface.
3. `findPerson()` reads a Person's name from the user interface and tries to find them in the database. If not found the error message "No such person" is displayed.
4. `displayAll()` displays all people in the database on the user interface
5. `display()` displays the requested person on the user interface

## 4 Marking Guidelines

84% 6% per method  $\times$  14 methods

16% Comments and style (follow your lab instructor's guidelines)

Good luck, and have fun!