BIT203 Programming in Java II

Assignment 2

Due date: Week 13

Value: 15%

Rationale

This assignment is designed to provide you with practical experience in creating a GUI that provides a visual display of a collection object. This assignment relates to the following learning outcome:

- Design and develop a GUI application.

This assignment also provides experience in file IO.

Expected Learning Outcomes Assessed

- LO1: write programs using several classes based on UML class diagrams and other models;

- LO2: apply object-oriented concepts in the design and implementation of the programs;

- LO3: use and extend Java interfaces and classes in the implementation of their programs

- LO4: design and develop a GUI application

SUBMISSION REQUIREMENT

Your assignment has to submit to TurnItIn, with the following all contain in a single file:

1. All your Java source files, printed in Word document format
2. Printed output (showing your interactivity with your program) is to be included at the end of your Java source files, in Word document created in (1)
3. A Turnitin Report, again to be attached within the Word document created in (1)

Turnitin Report (http://www.turnitin.com)
Register yourself in Java2_SS19 using the following details:
          class ID : **22596633**
          Enrollment password: **advancedJava**

Problem statement

A suite of Java programs that form a GUI application is required to simulate the entry and display of owners and boats managed by Bond's Boat Storage system. In Assignment 2, the main class, which is a GUI class, will replace the console display class that you have written in Assignment 1. You will also have to write other classes at your own discretion in order to provide the full functionality.

Requirements explained in the first assignment are extended and refined in this assignment. You should therefore re-read the requirements of the first assignment.

You are required to provide, as a minimum, the following graphical interface components:

- A menu (or sets of buttons) providing the same options as required in Assignment 1, plus the option to save and load the entire database.
- Screens for displaying details of owners, and boats.
- A screen for adding, and updating owners and their details.
- A screen for adding, and updating boats and their details.
- File dialog boxes to choose file names for saving or loading.

You must provide any listings sorted in order appropriately. You may choose to add other screens if you like, as this will result in more marks under the "Visual appearance" criterion below. The GUI may be written in either AWT or Swing, but the use of JList and JTable classes is recommended. You are free to use either text-based file IO or serialisation, but the latter is more likely to result in a stable application. You should write the application so that all details that can possibly be entered by the user are saved to file, including all objects created and their full internal state.

Your source code should contain javadoc comments and tags as follows:
- For classes, to indicate their purpose;
- For methods, to indicate their effect, parameters and return values, as well as any exceptions they throw;
- For fields, to indicate their purpose.

You may submit as many tests of your classes and of your program as you think necessary, though the reason for each should be explained.

Documentation

- You should include comments in your code stating what each method does and explaining any complex sections of code.
- You should include your student name and ID, date, and description about the class/program as comments within the code.
- You should of course use meaningful variable names so that your code is to some extent self-documenting.

What To Submit

You should submit the following:
- A cover-sheet with your student name, and number, plus turnitin originality report.
- Printouts of your source code using **Courier-New 10-point** size font. You may need to indent your code so that any long Java statements are nicely formatted, and not having the second line of a statement printed starting from the left-hand margin. You must print your output in portrait orientation. **Printing in landscape mode will be penalised**.
- Printouts demonstrating that your program produces the correct results
- A CD containing ALL the source files and the respective class files
- Write your name, student number and subject code on the disk; and
- A compressed file containing all the Java source files submitted to elearning.

Marking Scheme

Refer to the Excel file, *203A2MS_SS19.xlsx*, for detailed breakdown of the marks allocated for the requirements.

Note:

If your program does not meet the requirements by the due date you should obtain help from the lecturer and notify the lecturer that you will submit the assignment late (marks will be deducted).

Note about testing and plagiarism

It is very important that you **complete** this assignment **alone**. You may of course obtain general assistance from the lecturing staff in the subject and your peers, but the coding must be carried out yourself. It is normally quite easy to detect when two or more students work together on their coding.

It is also very important that the demonstration of the results of your program using the given test data is produced using the identical version of the program to the printout of your source code. **Students who hand in substantially similar assignments or whose programs do not match their demonstration of testing will fail the assignment.**

Any student suspected of copying, or of not producing the work himself or herself, can be called for **oral examination**, where the student will be expected **to demonstrate sufficient knowledge of the application** to show that it is his or her own original work.