Computer Science 2A

Practical Assignment 05

2016-03-15

Time: Deadline — 2016-03-22 12h00 Marks: 50

This practical assignment must be uploaded to eve.uj.ac.za **before** 2016-03-22 12h00. Late or incorrect submissions **will not be accepted**, and will therefore not be marked. You are **not allowed to collaborate** with any other student.

Good coding practices include a proper coding convention and a good use of JavaDoc comments. Marks will be deducted if these are not present. Every submission **must** include a batch file. See the reminder page for more details.

The Java Development Kit (JDK) has been installed on the laboratory computers along with the Eclipse Integrated Development Environment (IDE).

This practical aims to solidify your understanding of Java GUI

The story continues from Practical04. After obtaining the detailed roster you now have access to the crew members who have the required skills to help you open the door on your crew quarters. However, you are unable to determine the location of the crew members in relation to your room. Viewing other statistics on the information screen you find the layout of the ship. In order to display the layout of the ship you will need to build a GUI.

Create a **`ShipFrame`** class which extends **`JFrame`**. This will be the main application frame of the GUI. Create two buttons, one for opening a file and one for saving a file. When the open button is clicked then user is presented with an open file dialog. When a file is chosen then the file must be read using the **`CrewRoster`** class. The contents of the file must be displayed in a text area that will be in the center of the **`ShipFrame`**. The save button will allow the user to save the contents of the text area to file, using a save file dialog.

The main method will create and display the **`ShipFrame`**.

Mark sheet

1. **`ShipFrame`**
   (a) Open button shows dialog [05]
   (b) Display file contents in text area [05]
   (c) Save button [05]
   (d) Save text area to file [05]
2. **`Main`**
   (a) Create **`ShipFrame`** [01]
   (b) Set location [01]
   (c) Set close operation [01]
   (d) Show frame [01]
3. Coding convention (structure, layout, OO design) and commenting (normal and JavaDoc commenting). [10]
4. Correct execution [16]

NB

Submissions which **do not compile** will be capped at 40%

Execution marks are awarded for a correctly functioning application and not for having some related code.

Reminder

Your submission must follow the naming convention as set out in the general learning guide.

`SURNAME_INITIALS_STUDENTNUMBER_SUBJECTCODE_YEAR_PRACTICALNUMBER`

Example

Surname	Berners-Lee
Initials	TJ
Student number	209912345
Module Code	CSC2A10
Current Year	2016
Practical number	P00

`Berners-Lee_TJ_209912345_CSC2A10_2016_P00`

Your submission must include the following folders:

- `bin` - *(Required)* Should be empty at submission but will contain runnable binaries when your submission is compiled.
- `docs` - *(Required)* Contains the batch file to compile your solution, UML diagrams, and any additional documentation files. Do not include generated JavaDoc.
- `src` - *(Required)* Contains all relevant source code. Source code must be places in relevant sub-packages!
- `data` - *(Optional)* Contains all data files needed to run your solution.
- `lib` - *(Optional)* Contains all libraries needed to compile your solution.

NB

Every submission **must** include a batch file. This batch files must contain commands which will compile your Java application source code, compile the associated application JavaDoc and run the application. **Do not** include generated JavaDoc in your submission. All of the classes/methods which were created/updated need to have JavaDoc comments.