Introduction to Computer Science 2

Lab 5: Advanced User Interfaces

Learning Goals:

- To learn how to build a complex GUI.

Exercise 1 (5 points)

Write a class named `Converter` with a graphical interface that allows the user to convert an amount of money between U.S. dollars (USD), euros (EUR), and British pounds (GBP). The user interface should have the following elements: a text box to enter the amount to be converted, two combo boxes to allow the user to select the currencies, a button to make the conversion, and a label to show the result. Display a warning if the user does not choose different currencies. Use the following conversion rates:

- 1 EUR is equal to 1.18 USD
- 1 GBP is equal to 1.32 USD
- 1 GBP is equal to 1.12 EUR

Make sure that all the components and functionalities are properly shown in the video to be submitted!

(For this exercise, Codegrade will not autograde your source code. It will only give you a grade for the compilation automatically. A teacher will then assign the grade for the correctness of your program)

Exercise 2 (5 points)

Write a class named `Login` which is a graphical interface that implements a login window with text fields for the username and password and a button to check them (you can also include a label to indicate the number of attempts left). When the login is successful, hide the login window and open a new window with a welcome message.

Follow these rules for validating the password:

1. The user name is not case sensitive.
2. The password is case sensitive.
3. The user has three opportunities to enter valid credentials.

Otherwise, display an error message and terminate the program.

To handle the login functionality of the program, write a second class named `LoginManager` with two `static` methods:

- `boolean login(String userName, String password)`

  Which returns true if the user was successfully authenticated, or false otherwise. IMPORTANT: The number of attempts is updated in this method, increasing it if the login process fails or setting it up to zero otherwise.

- `int getAttempts()`

  Which returns the number of **failed login attempts** (incrementally from 0 to 3). The attempts reset only when the program is closed and started again.

The code must include two arrays storing (as strings) the database of usernames and corresponding passwords. Therefore, both arrays will contain the same number of elements.

Use the following usernames and passwords in your program:

Username	Password
jerry	123Welcome
tom	Qwerti
enrique	Azertu2

HINT: If you receive the error below for the last test, the problem could be related to how we are testing your methods. If this is the case, it should be solved by:

- Monitoring the number of attempts in the login method (returning false if the number of attempts is not correct) and
- Increasing the counter for attempts also in the login method.
- The attempts are incremental (starting from 0, the error must be generated when the attempt reaches 3)

With these changes, your code should pass the last test. We apologise for the inconvenience.

```
Error:  I tried to hack your system 3 times, you shouldn't allow me to login anymore.
==> expected:  <false> but was:  <true>
```

Make sure that all the components and functionalities are properly shown in the video to be submitted!

NOTE: In Lecture 6 you will learn how to implement this process nicer using other files to store and retrieve the username-password information. Optionally, you can revisit this assignment and apply it here.

Honor code, coding style, and deliverable:

Try to solve the exercises with what you already know. You are welcome to expand your program to do extra things but they are not mandatory.

Plagiarism is not allowed! We will run sophisticated software that automatically detects similarities on source code among students. All plagiarism incidents will be immediately reported to the Board of Examiners

Submission!

Submit your java files to CodeGrade (see more information in the Syllabus).
Ask your instructor in case there is a problem with your submission.

DO NOT SEND SUBMISSIONS VIA EMAIL
YOUR LAB WILL NOT GET GRADED!