Designing Interactive Systems II - Summer Semester 2010

Assignment 1: Hello GUI
Due: Monday, May 3 2010 @ 10:00am
This assignment is to be completed in *groups of two.*

In this assignment, you will be introduced to the basics of GUI programming.

Part 1: Console UI

As a warm-up exercise, we will ask you to first write a simple console (text-based) UI program. This program should present the user with a list of languages. After the user selects an option by typing in the first letter of that language, followed by “enter”, the system will output “Hello” in that language.

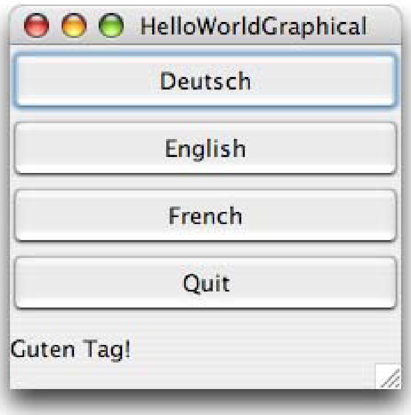
A usage session should look like the following:

```
Macintosh:~/$ java HelloWorldConsole

Select one of the following:
  [D]eutsch
  [E]nglish
  [F]rancais
  [Q]uit
> d
Guten Tag!

Select one of the following:
  [D]eutsch
  [E]nglish
  [F]rancais
  [Q]uit
> q
Quitting...
Macintosh:~/$
```

Place your code into the class `HelloWorldConsole`. *Hint:* Use `System.in.read()`.

Part 2: Graphical UI

Now you will take the same functionality and place it into a simple *graphical* user interface, `HelloWorldGraphical`. This graphical user interface should look something like the following:



Guten Tag!

The text at the bottom of the window should reflect which button was last pressed, similar to your `HelloWorldConsole` program. The UI should be developed using Java Swing. If you have never programmed in Java Swing before, don't despair! There are a number of resources on the web to help get you started with Java Swing programming – you can find these resources via Google. *Hint:* Use event listeners.

Testing Your Understanding

Answer the following questions:

1. What is the key difference in the way `HelloWorldConsole` retrieves user input from the way `HelloWorldGraphical` retrieves user input?
2. (a) What is the *minimum* number of event listeners required to implement the functionality in `HelloWorldGraphical`? Explain.

   (b) What is the *maximum* number of event listeners that could be used to implement the required functionality in `HelloWorldGraphical`? Explain.

   (c) If your answers to part (a) and (b) are different, what are some possible reasons for using more or less event listeners for a particular application, if they all result in the same functionality?

Extra Credit

For extra credit (each will give you a maximum of one grade level bonus):

- Make it such that you don't have to press an "enter" key to select an option in `HelloWorldConsole`.
- Use different widgets and/or widget layouts in `HelloWorldGraphical` to give better affordances, feedback, etc... (e.g. button groupings, highlight state for the buttons, use of radio buttons). In your `README` file (see below), explain why your redesigned interface is better than the one shown in the above figure.

Submission

Email a **ZIP archive** of your assignment to `dis2_submissions@cs.rwth-aachen.de` before the due date.
Name your archive ⟨lastnamegroupmember1_lastnamegroupmember2⟩.zip where you fill in your last names. The subject of your email should be "DIS2 Assignment 1"; be sure to use this *exact* subject line as it will be used to filter assignment submissions for grading.

Your assignment archive should include your source code and everything necessary to compile and run your program. Be sure to document your source code. Include a **PDF document** with the name `README.pdf` that contains:

- names and email addresses of all group members
- instructions on how to compile and run your source code
- answers to the questions (see *Testing Your Understanding*)
- non-obvious things you did in your code (if any)
- anything that you think makes your design particularly optimized and/or elegant

Make sure that your submission compiles and runs on the lab machines.

Please note that assignments that do not meet the above submission criteria will not be graded. Be prepared to discuss your solution in the next lab.

Grading

The assignment will be graded on the following rough scale:

- 1.0 - exceptional work that clearly went above and beyond what was given on the exercise
- 2.0 - exercise was completed satisfactorily as per the assignment specification
- 3.0 - exercise was completed, but has some problems
- 4.0 - incomplete exercise
- 5.0 - little or no effort was put into the exercise

Late assignments will *not be graded.*