Computer Science 3053

Human-Computer Interaction

Assignment 3

Using Java SWT (from `http://www.eclipse.org/swt`), develop a desktop application that gives the user a series (progressively disclosed) of questions and multiple choice answers. Your application must:

1. (2 points) Read possible questions and multiple choice options (along with the correct answer option) from a File with the format given below. The number of possible questions may be variable (e.g. don't assume three questions) but the number of multiple choice options per question will always be four. Consider using `java.io.BufferedReader` and `java.io.FileReader` classes for reading from the File. The file to read from must be selectable by the user using a `org.eclipse.swt.widgets.FileDialog` (the SWT equivalent of Swing's `javax.swing.JFileChooser`).

2. (1 point) Show the user a single question at any given time, along with the four possible options to choose from and a way for the user to make a selection of one (and only one) option (e.g. using a radio button).

3. (1 point) Record the number of correctly answered questions and, after all questions are answered, show the user their ”score” in terms of percentage of correctly answered questions.

4. (2 points) The applications shall have two ”modes” of operation which change the behavior after the user has answered a given question. This mode should be toggle-able via some GUI element (e.g. a menu item in the menu bar). In the first mode, once the user makes a selection of an option the application should ”hide” that question and options and ”show” the next question with its options. In the second mode, once the user makes a selection of an option the application should give reasonable feedback on whether the user's selection was correct and if not which option was correct (e.g. with changes to the text such as style/boldness, color, or with added images to indicate the correct answer).

5. (1 point) The order of the displayed options should be randomly determined (i.e. the options should not always be displayed in the same order as listed in the file). This order should generally change between executions of the program on the same input file.

In addition to the above functional requirements, your code must also be organized using traditional object-oriented patterns. I.e. turning in code that implements the entire application in a single class (or single class with anonymous inner classes) is inappropriate. This non-functional requirement is worth 3 points of credit.

0.1 File Format

The format of the questions/options file shall be the following. Each question text appears on a single line followed immediately by four lines, one for each answer option text. Additional questions must be preceeded with a blank line followed by five more lines (one for the question text, four for the options' text). The line of text for the correct option will begin with a capital 'C' character followed by a space character. Lines with incorrect answers will begin with two space characters.

Example:

```
What is the capital of the Assyrian empire?
  Rome
  Washington, D.C.
C Nineveh
  Cairo

What is the air-speed velocity of an unladen swallow?
  1,000,000 mph
  1,000,000 kph
  1,000,000 * speed of light
C Depends on whether it's an African or European swallow
```

0.2 Due Date

You must complete the assignment by Thursday, March 3 at 11:59 PM and submit the source code (not compiled class files, nor a JAR or ZIP) on D2L in the appropriate DropBox. If your program utilizes other resource files (e.g. icon images, etc.) then they can be submitted as individual files too. For this assignment it is **not** necessary to come to office hours or a special appointment to demonstrate your application, unless you feel that only by your demonstration could the program be used correctly and graded fairly.

Some extra credit can be granted for creative extensions to the basic functional requirements listed above.