IST 256
Lab Week 12, Monday, November 15, 2010

1. Complete Wednesday's lab with Student data from a file.
Finish the last part with the display of all students.

2. Extending the program with Student data from a file
In this example, we will extend the Students example that we did in the last lab. Today we will add a button on the GUI so that we can search for students by (partial) name.

a. Start by **opening the Students** project (or whatever you called it).

b. Extend the form interface.
On the form, we will keep the previous buttons and labels for reading the file and saving the data, and for displaying the students. Now we will add a textfield to allow the user to type in a search name and another button to perform the search. The form can look something like this:

```
|__ Read Student File__|      File Status

|___Display Students___|
          Results:

Enter (partial) search name:  |___________|        (label and textfield)
          |__Search Student Names__|                  (button)
Results:                                           (multiline label)
```

For the new button, select Event -> action -> actionPerformed.
(You may find that you explicitly need to resize the Results label in the Design pane for all the results to show.)

c. Add accessor method to the Student data class

In the search, we will need to get the student name from each student class instance. So we should add an accessor method to the class.

Go to the code for Student.java and add the following method where the comment says "//no accessor methods so far".

```
public String getStudentname ()
   {
       return studentname;
   }
```

As usual, adjust the indentation, and change the comment to say "// accessor methods"

d. Add Code for the new Search button

This button will search the array of Students to find ones whose name matches according to the following criteria:

> the search string can be found as a starting substring of the student's name and
> case should be ignored.

Find and display all the students that match.

- Declare a variable to hold the search name String and get the search name from the TextField.
- Declare and initialize two string variables to have the result header and the result names. The result names should be initialized to the empty string "".
- Write a for loop that iterates over all the students in the array up to numstudents. Compare each student name (using the getStudentname accessor method) with the search name according to the criteria described above.
- If the student name matches, add that student to the resultnames.
- After the loop, if no students were found, add a No Match message to the resultnames.
- Display the resultheader and resultnames.

e. Test your program.

Test your program with by typing in example search strings that match and don't match.

Write here examples that you tested:

1. An example that didn't match anything:

2. An example that matched one student:

3. An example that matched more than one student:

4. An example that shows the search is ignoring case:

Hand in your lab from last Wednesday, as it is due today. (This lab can either be handed in today or on Wednesday, Nov. 17.)