Programming Assignment #2
COMP150CBD
Fall 2004
Due before class on 9/28/2004

Assignment #2 introduces you to encapsulation, inheritance, polymorphism, events, and exception handling in Java. This assignment is worth 75 project points. You may also earn up to 10 extra credit points. It will be graded on completeness, readability of code (use capitalization, indentation, and commenting conventions), correctness, and robustness.

ChatRoomClient[1]

A chat room is a place where multiple users can interact by alternately entering lines of text into their client and then making them visible to other users. In this assignment you will build onto the code, provided via link from the course home page named ChatRoomClient.java, that includes the beginning of an implementation of a ChatRoomClient. You will add a mechanism for a user to select a chat room name, and a menu that supports a few basic functions.

To complete this assignment you should import the javax.swing package. I recommend that you look through the Swing APIs to find out what methods and services will be available to you before you begin the assignment.

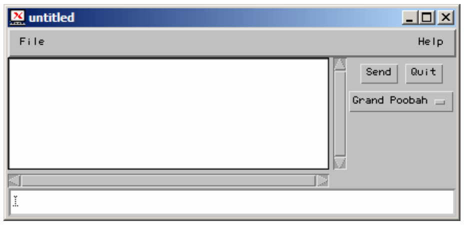
All work should be your own but feel free to discuss the assignment with colleagues and to consult resources on the web. DO NOT copy solutions from any where. Your solution must be original work by you.

Exercises:

1. Copy the ChatRoomClient.java source file from the COMP150CBD web site to a suitable location in your work area and move into that area yourself.
2. Environmental differences will influence your answers to the following questions. Specify now what operating system you are using. Either answer the following questions or state that the question is not applicable to your situation.
3. Run javadoc on this file. If you are running this on a windows machine, this probably produces a warning. What is the warning and how can you fix it?
4. Now try to compile the source file using javac. No matter what machine you are running on you will get an error. The source of the problem is the same thing that caused the warning when running javadoc in Step 3. What is the error and how can it be fixed?
5. If you still get an error you didn't completely fix the problem. Why did you get the error and what did you do to fix it. Look through the code for other places where this same error manifests itself and fix them too.
6. Okay, so now your program compiles. Run the program to make sure everything works okay. You should see a new window on your desktop that is the beginning of a chat client. Play around with it to see what it is currently capable of and study the source file to get an idea of how things work and what methods are being used.
7. Move into a different directory and type "java ChatRoomClient". Does the program run? If not, what is the problem and how can you fix it?
8. What package provides the *Frame*?
9. Where does *ActionListener* come from and what does it do?

[1] This assignment is adapted from the ChatClient exercise found in course WJ-2755 on the Sun Learning Center

10. On your own, explore the rest of the code asking yourself similar questions.
11. Now it is time to begin extending the functionality of the ChatRoomClient. Your goal is to have your chat room look something like this:



12. First, add the ChatRoomClient to a new package of your choice. You may name your package whatever you want as long as you follow the package naming convention discussed in class.
13. The ChatRoomClient currently uses AWT for all of its GUI components (e.g. Frame, Panel, Button, etc.). Change it so that it includes Swing. You should replace all AWT GUI elements such as buttons, text areas, frames, panels, etc. with their Swing equivalents. **All further GUI modifications in this assignment should use Swing and not AWT.**
14. What difference does changing from AWT to Swing make with respect to the platform the client runs on?
15. Create a combobox that lies below the Send and Quit buttons. Have it include a number of name options to allow the user to choose a user name to be used while chatting. Then modify the *ActionListener* for the Send button. Have it prepend the selected user name to the message prior to adding it to the output text area.
16. Create a menubar. Add a File menu and a Help menu.
17. Add a menu item called "Quit" to the File menu. When that menu item is selected the program should be terminated.
18. Experiment with changing the color or font of the various components within the chat room frame. Your final code must reflect a change in each from the original ChatRoomClient code.
19. Add a menu item called "About" to the Help menu. When that menu item is selected, a “modal dialog” should be displayed that tells the user about the chat program. If you don’t know what a model dialog is you should consult the online java documentation at java.sun.com.

Extra Credit Possibilities:

(10 EC) If you are feeling ambitious, try adding support for client to client interaction.

Submit your answers the written questions, the ChatRoomClient.html page produced in number 2, and your modified source file to **provide**. Feel free to submit the whole directory instead of all the individual files. You should use the following command line to submit your homework:

- provide comp150cbd a2 [files/directory]