Assignment 6:  
Chat Server

CMPS 109 - Winter 2003
Ira Pohl
General Specs

- Write a program that will function as a chat server.
- Allow multiple users to connect
- Any message sent by one user is broadcast to all other users
- Make a GUI with an area to type a message and an area to display the conversation.
No threads needed.

User specifies server address, server port, and login name
- Command line
- GUI Dialog
The Client

- A JTextField is provided for the user to enter new messages.
- A JTextArea is provided to display the overall conversation.
The Server

- Needs to create threads
  - One listens for new connections.
  - Client handling threads
    - One thread per user
    - One thread per X number of users (allows more connections).

- Messages from a client are broadcast to all others.

- Connections are broadcast to all users.
The Server

- Needs a list of connected users
  - Access to the list must be guarded by thread locks.
- A GUI showing the state of the server is optional.
  - The list of connected users.
  - The conversation history.
Messages

- **Client to Server messages**
  - Login: sends username
  - Post: sends a new message
  - Logout: tells the server to remove this client

- **Server to Client messages**
  - Login accept: acknowledge client and username
  - Login reject: sends rejection and reason
  - UserConnected: reports a new client
  - Message: a message sent by a client
Design Hints

- Message superclass
- Share message classes between client and server (use a sub-package)
- Make a text-only version first!
- When broadcasting a message, make a copy of the current client list
Steps

1) Write a server.
2) Make the client connect.
3) Make the server accept or reject the client.
4) Make the client send something to the server (random text?)
Steps

5) Make the server broadcast to all connected users.

6) Write the client GUI.
Questions and Answers