CSCI 260 Assignment 1. Due 5:00pm 14 September

Pleas do all work for this assignment individually, and feel free to send me e-mail (allen@bowdoin.edu) if you have problems. For the Please e-mail me your answers as well.

1. Answer the following Exercises in your text.

   Chapter 2, Exercises E4, E6, E7, E9, E10, E12aceg

   Chapter 3, Exercises E42, E44, E45de, E47de.

2. The knock-knock client-server system *kk2Client* and *kk2Server* can be run using Eclipse and the workspace folder *workspace260*. Exercise this system using two or more computers in the lab. Here, the server should be running on one computer and one client should be independently running on each of the others.

   a. This system is an example of a "thin client" system, since all the code that controls interaction with a user resides on the server side. Briefly describe the revisions needed for it to become a "fat client" system, where that code resides on the client side instead.

   b. What trade-offs occur when choosing between a fat client and a thin client design?

3. Modify the "knock-knock" client-server system so that it has the following added capabilities:

   a. Currently, if the server shuts down while a client is connected, the client continues to wait for messages. Modify the client's hooks `connectionClosed` and `connectionException` so that it responds to server shutdown by displaying the message `"server shut down"` and quitting.

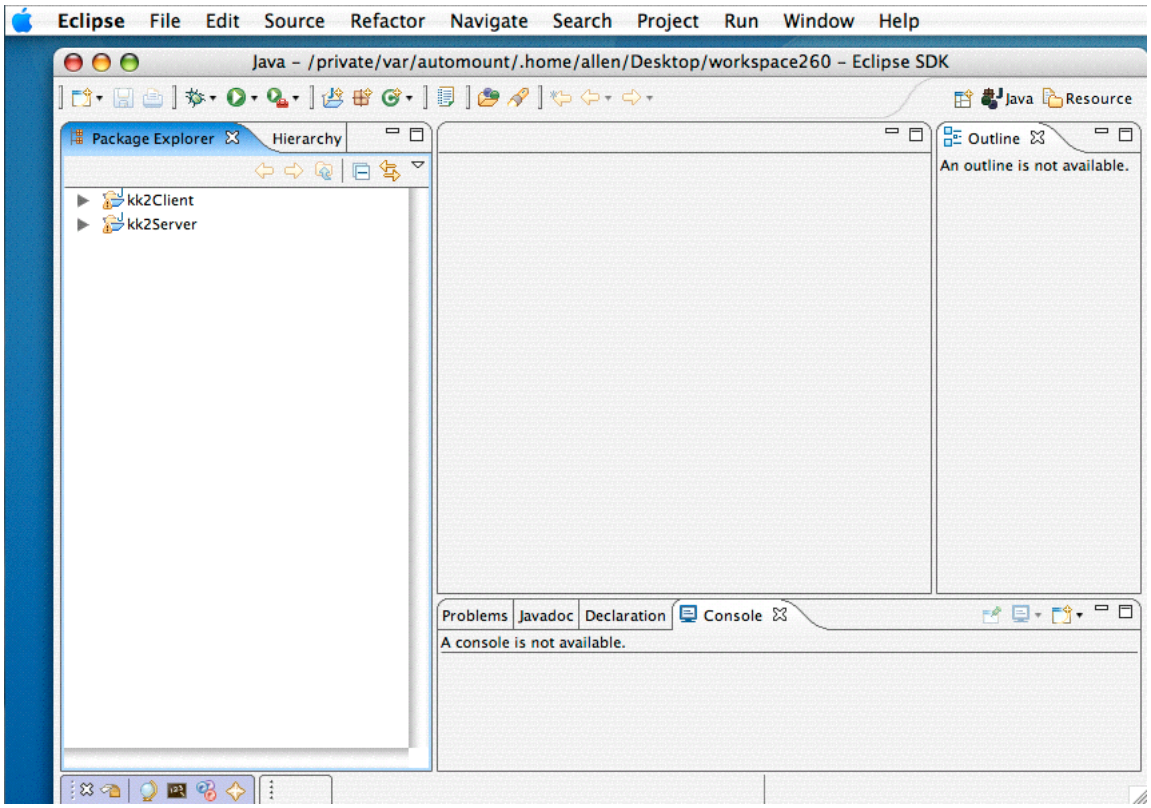
   b. Currently, the client always uses a default port. Modify the client so that it obtains the port number from the command line, in addition to the host name (which it already obtains from the command line).

   c. Currently, the server does not allow user input, and so it cannot (for instance) be shut down. Study the way user input is obtained by the client using the *ClientConsole* class. Create a similar mechanism on the server side (call it *ServerConsole*) that also implements the *ChatIF* interface. Now implement the following commands for the server user:

      `#quit` causes the server to shut down gracefully.

      `#stop` causes the server to stop listening for new clients and disconnect all existing clients.

      `#start` causes the server to restart listening for new clients (but only if the server is stopped).



Starting Eclipse in the Searles 224 lab

1. From your browser, link to the course Web site and change the URL suffix **syllabus.html** to **workspace260.zip**.

2. Selecting this link should download and unzip the folder **workspace260** onto your desktop.

3. Start Eclipse 3.0 (not 3.1) from **Macintosh HD** → **Applications** and select **workspace260** when prompted. The Eclipse window should appear as shown above.