

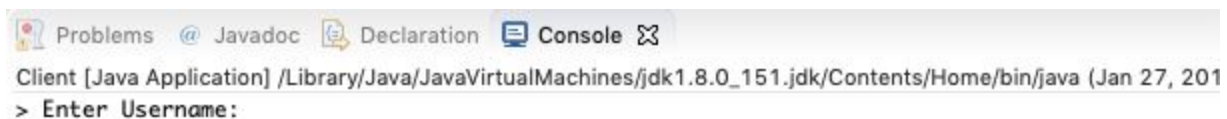
Solution Approach

My solution approach was to follow the guidelines of the lab and to have both a client file and a server file. These files work together to create a functioning chat application that can host multiple users, allowing them to chat with each other. The class `Runnable()` was implemented to utilize threads and transfer data to the client, along with redistributing the data to both the client and the server. In order to add more users to the chat room, the client file must be run multiple times.

Required Output

1.1 - Connect to Server

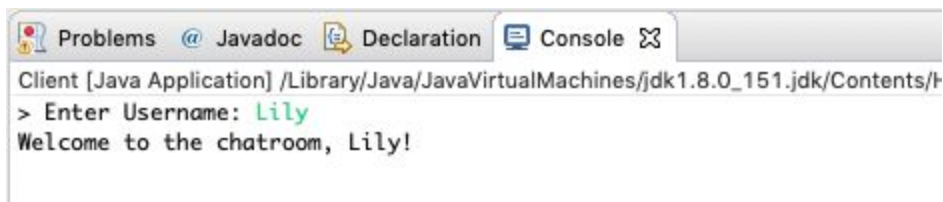
- a. Upon starting the client, it comes up with a prompt asking for a username.



The screenshot shows a Java IDE with a console window. The console title bar includes 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console text reads: 'Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 201...)' followed by a prompt '> Enter Username:'.

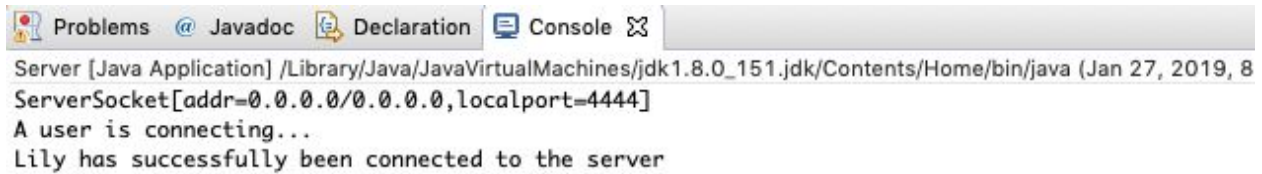
- b. After the user enters a name, the client is connected to the server.

- Client View:



The screenshot shows the same Java IDE console window. The prompt '> Enter Username:' has been followed by the user input 'Lily' in green text. Below the input, the output 'Welcome to the chatroom, Lily!' is displayed in green text.

- Server View:



```
Problems @ Javadoc Declaration Console
Server [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2019, 8:00:00 AM)
ServerSocket[addr=0.0.0.0/0.0.0.0,localport=4444]
A user is connecting...
Lily has successfully been connected to the server
```

1.2 - Send a message to the server

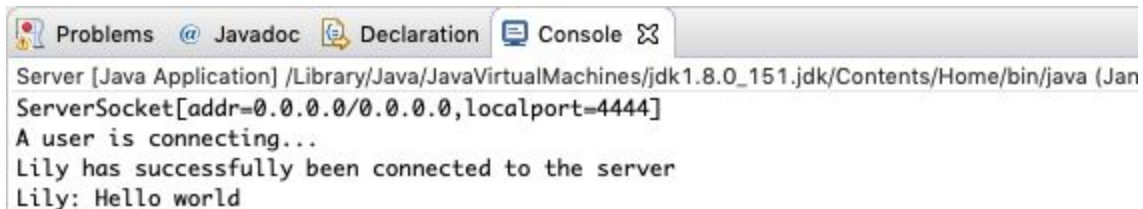
- a. The user name and chat message are shown on the client console, and then sent to the server.

- Client View:



```
Problems @ Javadoc Declaration Console
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2019, 8:00:00 AM)
> Enter Username: Lily
Welcome to the chatroom, Lily!
Hello world
|
```

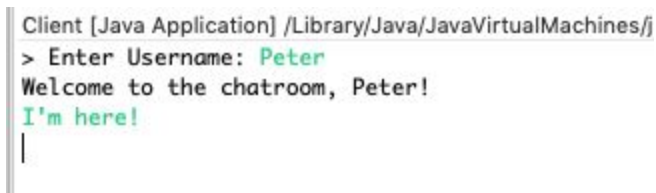
- Server View:



```
Problems @ Javadoc Declaration Console
Server [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2019, 8:00:00 AM)
ServerSocket[addr=0.0.0.0/0.0.0.0,localport=4444]
A user is connecting...
Lily has successfully been connected to the server
Lily: Hello world
```

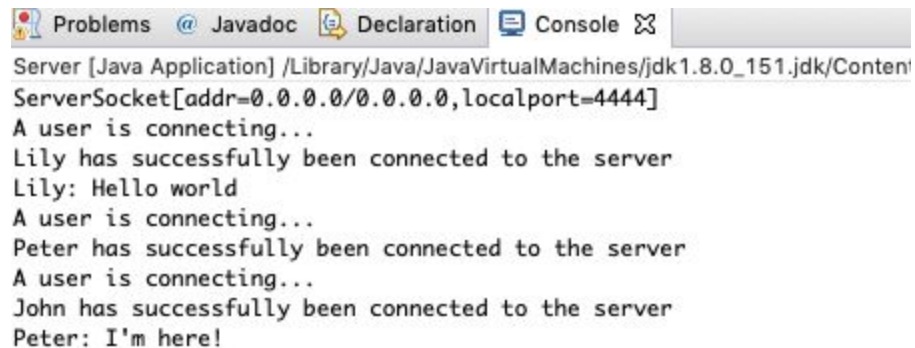
Added users can view the messages as well. For example:

- Client View (new user Peter):



```
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2019, 8:00:00 AM)
> Enter Username: Peter
Welcome to the chatroom, Peter!
I'm here!
|
```

- Server View (new user Peter and John):



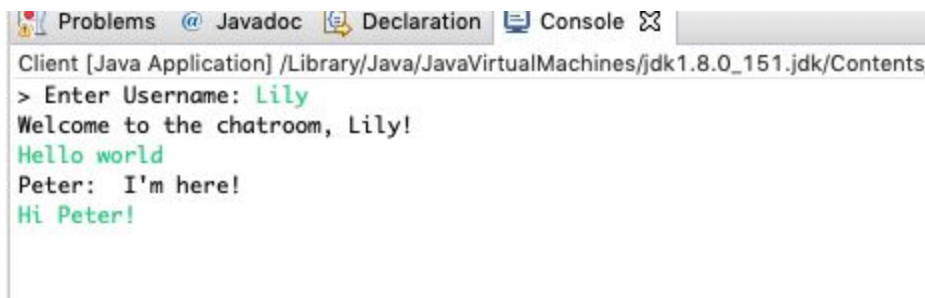
The screenshot shows the IDE's Console window with the following text:

```

Server [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Resources/Java
ServerSocket[addr=0.0.0.0/0.0.0.0,localport=4444]
A user is connecting...
Lily has successfully been connected to the server
Lily: Hello world
A user is connecting...
Peter has successfully been connected to the server
A user is connecting...
John has successfully been connected to the server
Peter: I'm here!
  
```

- The server then broadcasts messages to all the other connected clients.

- Client View (Lily):



The screenshot shows the IDE's Console window with the following text:

```

Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Resources/Java
> Enter Username: Lily
Welcome to the chatroom, Lily!
Hello world
Peter: I'm here!
Hi Peter!
  
```

- Client View (John):



The screenshot shows the IDE's Console window with the following text:

```

Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Resources/Java
> Enter Username: John
Welcome to the chatroom, John!
Peter: I'm here!
Lily: Hi Peter!
  
```

- Client View (Peter):

```
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2017)
> Enter Username: Peter
Welcome to the chatroom, Peter!
I'm here!
Lily: Hi Peter!
```

c. Messages are then printed in both the client and server consoles.

- Client View (Lily):

```
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2017)
> Enter Username: Lily
Welcome to the chatroom, Lily!
Hello world
Peter: I'm here!
Hi Peter!
Peter: How is everyone doing today?
John: I'm doing well, what about you Peter?
```

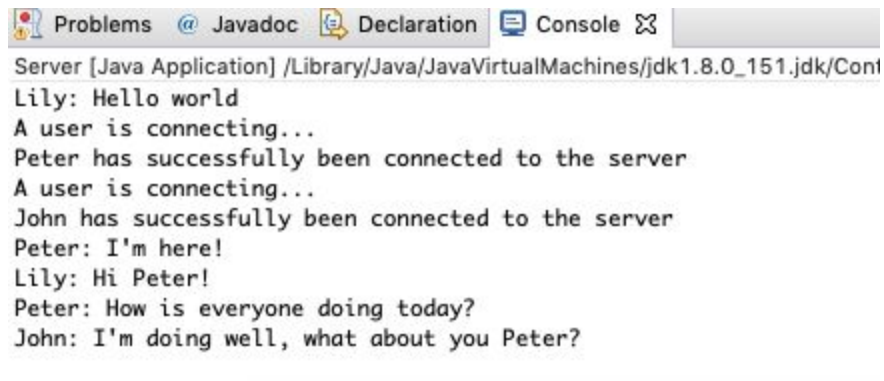
- Client View (John):

```
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2017)
> Enter Username: John
Welcome to the chatroom, John!
Peter: I'm here!
Lily: Hi Peter!
Peter: How is everyone doing today?
I'm doing well, what about you Peter?
```

- Client View (Peter):

```
Client [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home/bin/java (Jan 27, 2017)
> Enter Username: Peter
Welcome to the chatroom, Peter!
I'm here!
Lily: Hi Peter!
How is everyone doing today?
John: I'm doing well, what about you Peter?
```

- Server View:



The screenshot shows the Eclipse IDE's Console window. The title bar includes tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console output shows a Java application running on a Windows system. The output text is as follows:

```
Server [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Resources/Java
Lily: Hello world
A user is connecting...
Peter has successfully been connected to the server
A user is connecting...
John has successfully been connected to the server
Peter: I'm here!
Lily: Hi Peter!
Peter: How is everyone doing today?
John: I'm doing well, what about you Peter?
```