# **Instant Messaging System**

IDE Used: Intellj Ultimate Edition

Operating System: Mac OS

Language: Java

ID:1001444149

Name: Krishna Chinya

Something’s to note before executing the program.

1)Whole project is implemented in JAVA.

2)Both Client and Server are **Multithreaded** and Any number of clients can connect to server and chat simultaneously with other client.

3)All the Chat data is stored in **LOG FILE.**

4) GET and PUT headers are displayed only once when Client connects to Server. For the consequent chats I’m displaying headers as it is quite disturbing for the user to read so I’m removing the user Chat and displaying on console. **If you want, see the header for each chat it can been seen LOG FILE.**

5) **Server.Java file needs to be running for client to connect. So its pre-requisite to start server before starting the client.**

6) Next, Client.java needs to be executed depending on how many clients are required.

7) As soon as client joins. It asks for UserName before proceeding futher. **Program doesn’t allow multiple clients to have same name** and **No second client connection allowed.**

8)In order for client to leave the chat type ”Bye“.

**More detailed explanation is provided with screen shots in output section**

How to Compile the program and Run the program?

There are 2 two ways

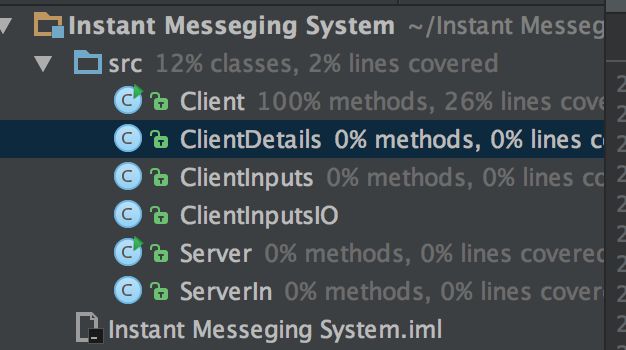
1)Using IDE.

2)compiling from command Prompt.

# Using IDE.

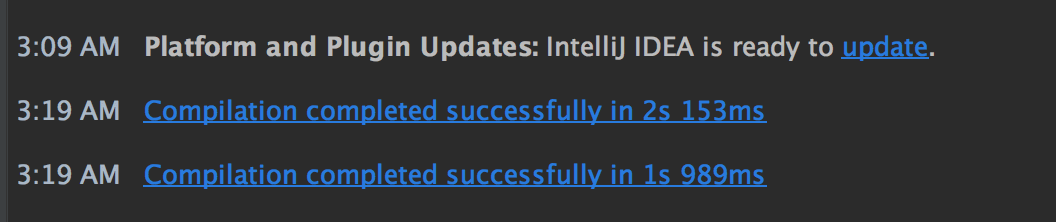
1)Load the Project in Intellj, Folder Name: **Instant Messeging System**

2)Once the project is loaded you should be able to see the following files.

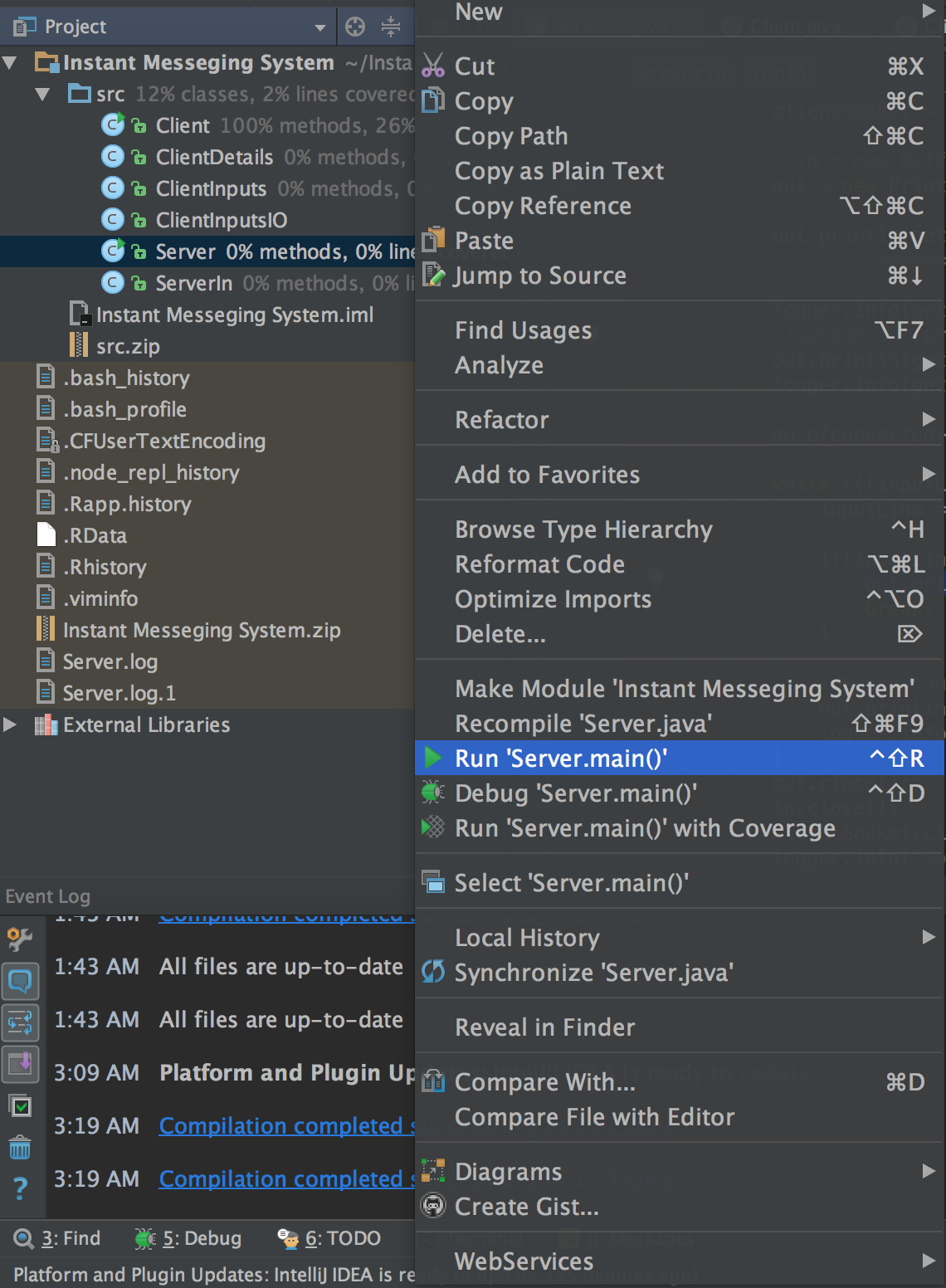
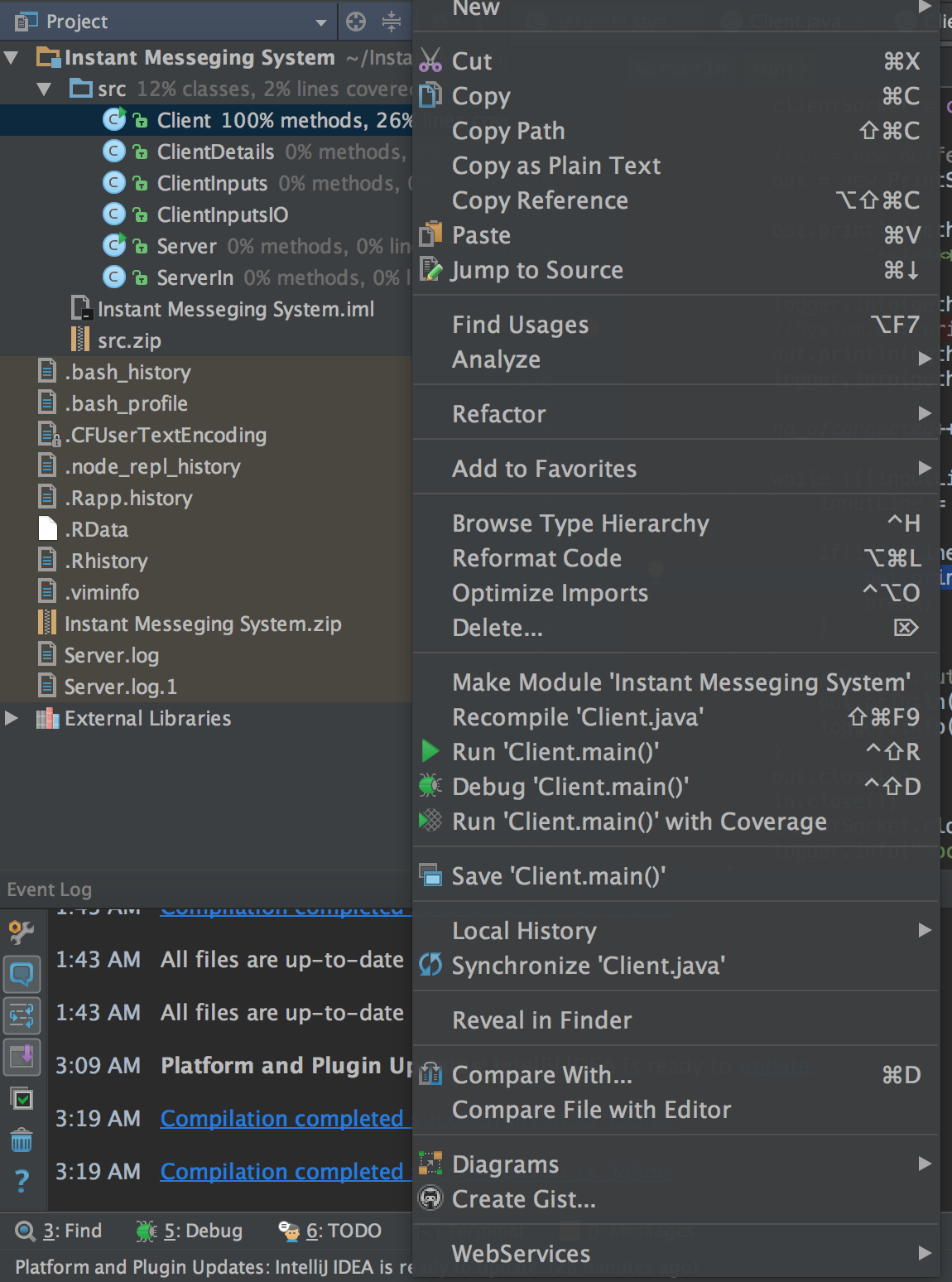


3) Goto Build 🡪 Rubuild Project

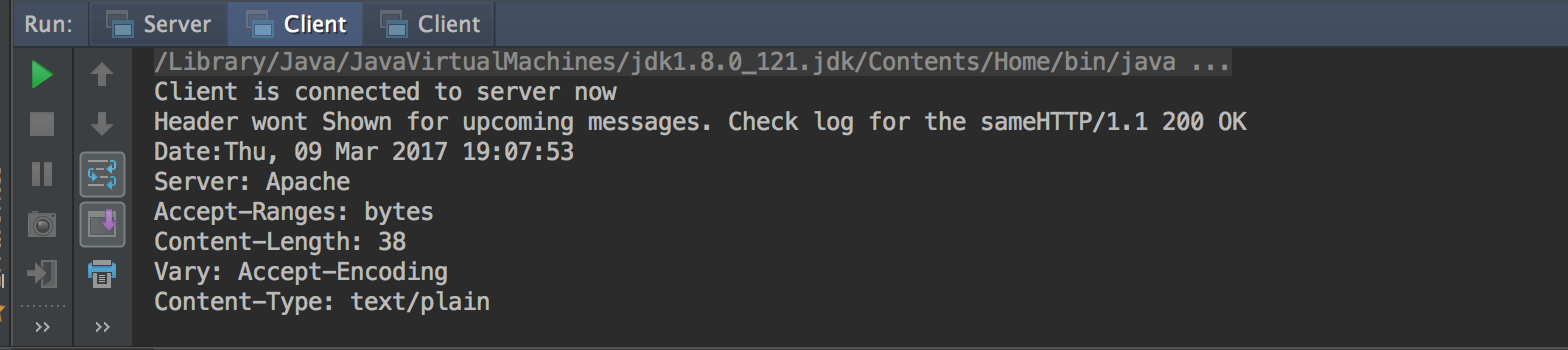
IDE checks for the error and displays succeeded



4) Then click on server.java and do run. And similarly click on client.java(Any number client can run, its multithreaded) and do run.

5)Then it will open command promt windows as below in the IDE itself where user can perform chatting.



**Output will be explained at the later point of the document.**

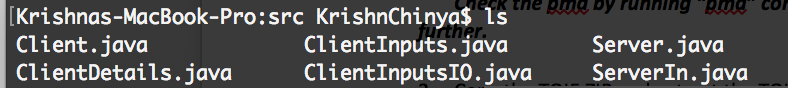
# Using Command Prompt.

1)I have given 2 sets of folders. One if you are running via IDE and another for compiling via terminal.

2)Extract the folder – Instant Messeging System – CMD

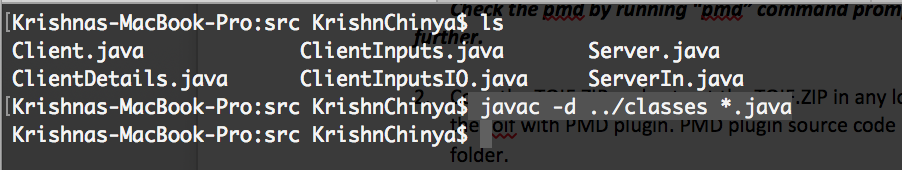
3)Then Open Terminal and Goto **/Instant Messaging System – CMD/src** folder.

4) Type ls and you should be able to see below files.

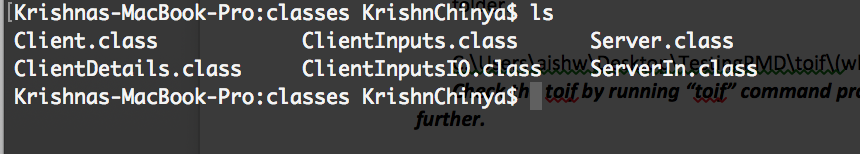


5)In order compile and generate classes, type in the below command

javac -d ../classes \*.java



This would generate all the required classes in ../classes folder as shown below



6)In order to run the server program and Client Program. Type in below command.

For Server: java Server



For Client:java Client



# Output

1. Run the Server as above in separate terminal

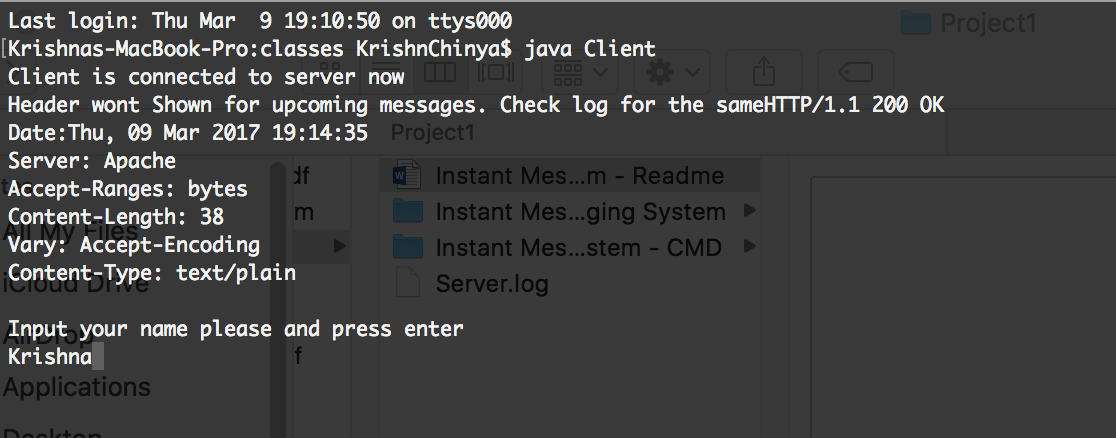


1. Then run the as clients as required. For this example: In order to show multithreading (**Extra Credit**)Im running 4 client, Even more client can be executed if required. **Open the client in 4 different terminals**



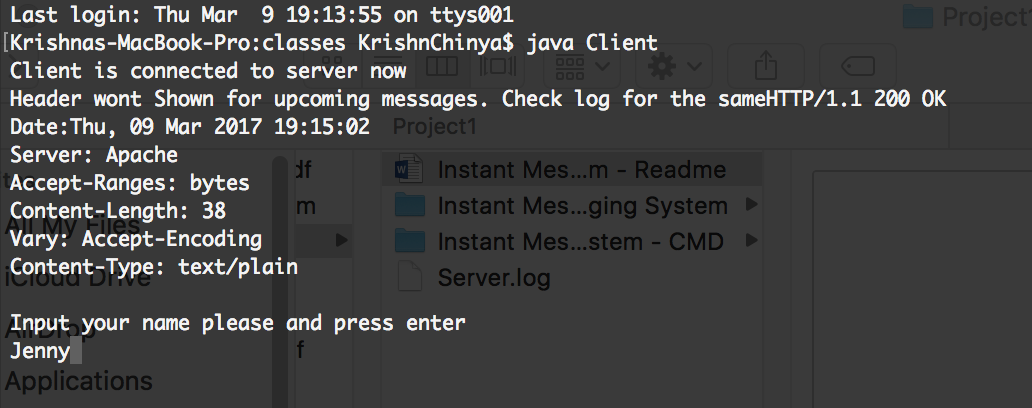
1. Now start all 4 client or 2 client in terminal.
2. As soon as terminal starts it asks for username for that terminal with the header.

**Client: Krishna**

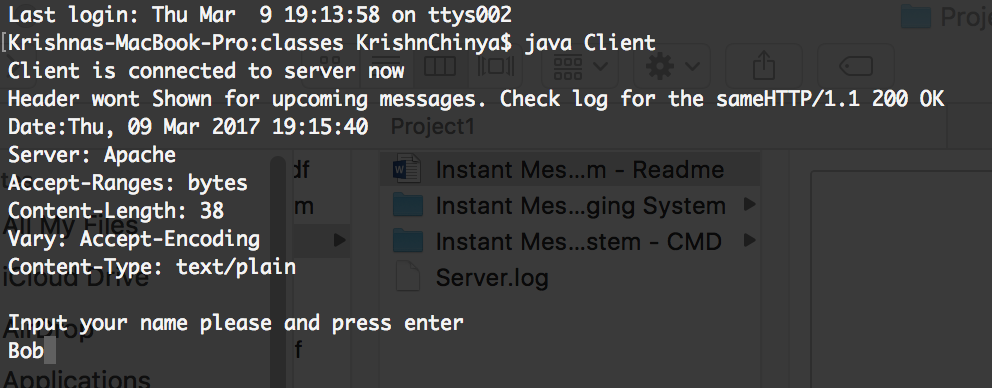


Now press enter and start client two,three and four in different terminal and enter name similarly.

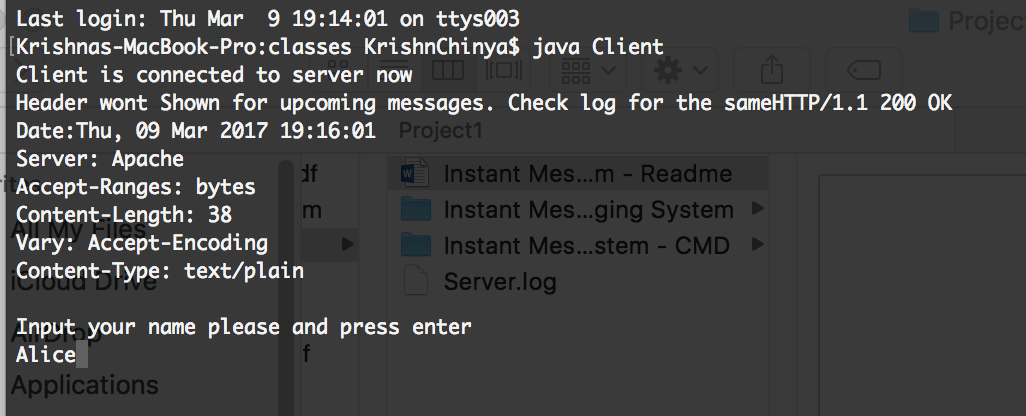
**Client: Jenny**



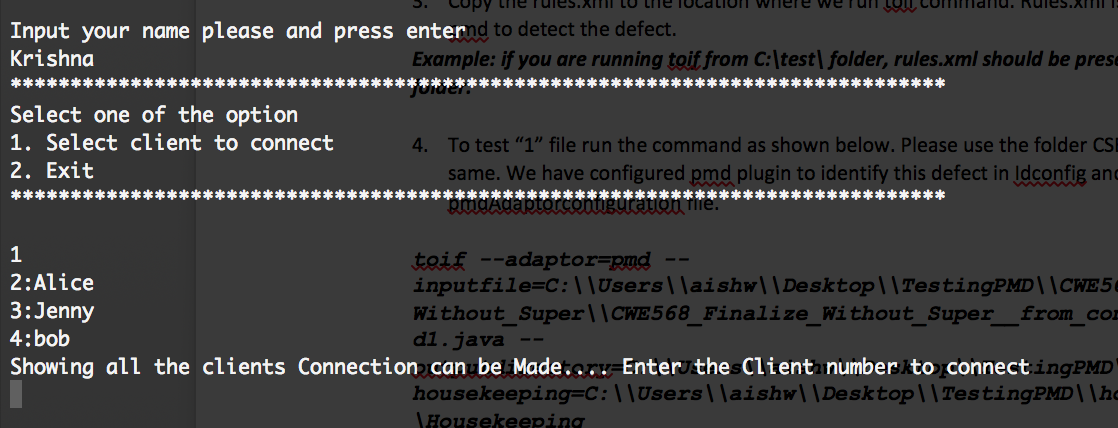
**Client:Bob**

****

**Client:Alice**

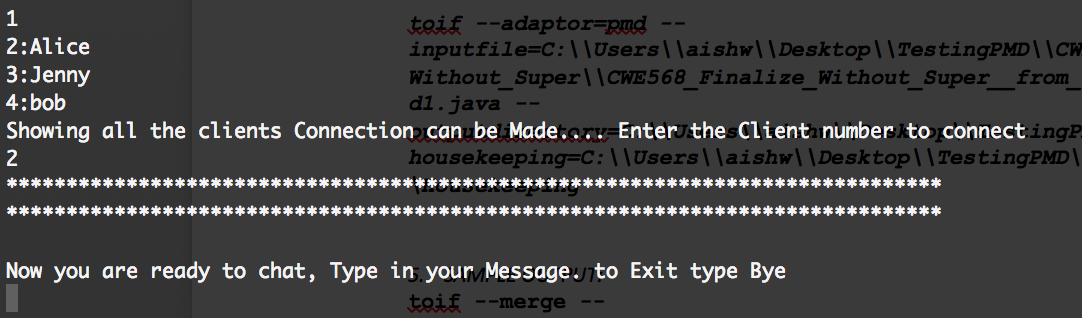
****

1. Now go to. Client/user – Krishna terminal window. And select option 1 from the menu. Menu is visible as soon as you press enter.
2. It will show all the user connected as below.(This for user: Krishna) Since all the users are not connected to any user it will show all user names.



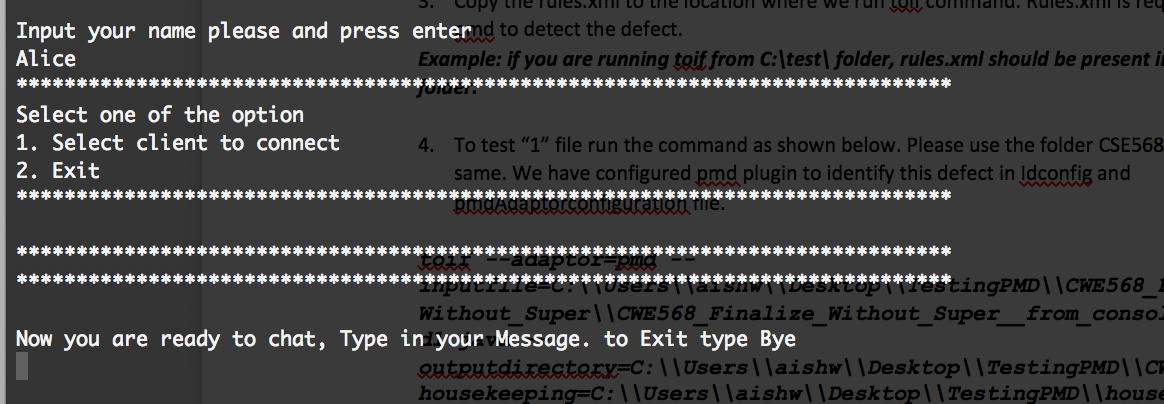
1. Now Select/Enter the user you want to connect. Here Im selecting 2. So **Krishna** will be connected to **Alice:**

**Krishna Terminal Window**

****

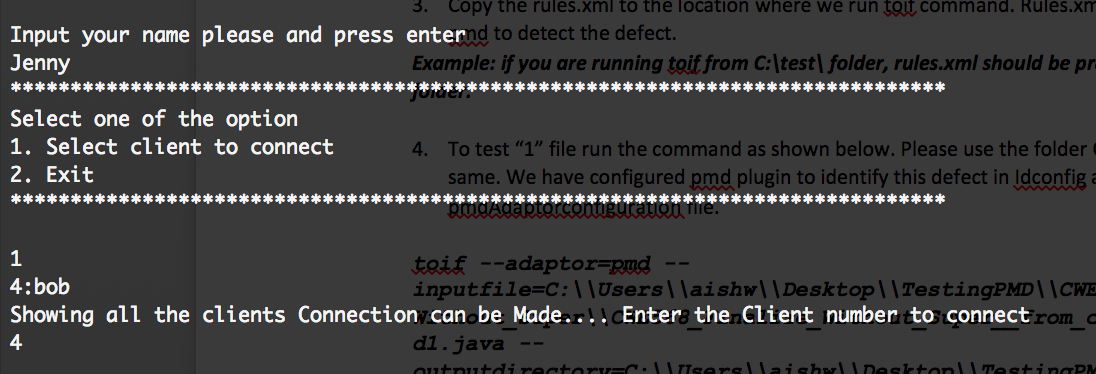
At the same moment allice will be update with information that new user has been connected.

Alice Terminal Window

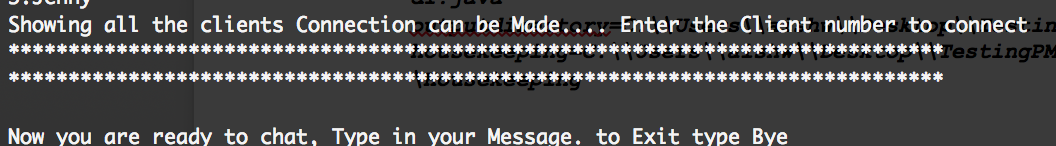


1. Similarly, Type in 1 for Jenny terminal and select bob to connect**.(Here for Jenny – Alice and Krishna wont be visible as they are already connected only free users will be displayed)**

**Jenny Terminal**

****

**Bob terminal**



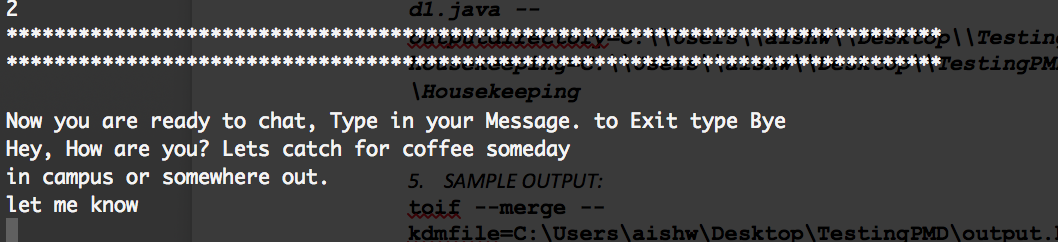
1. Now users are connected as below.

Krishna 🡪 Alice

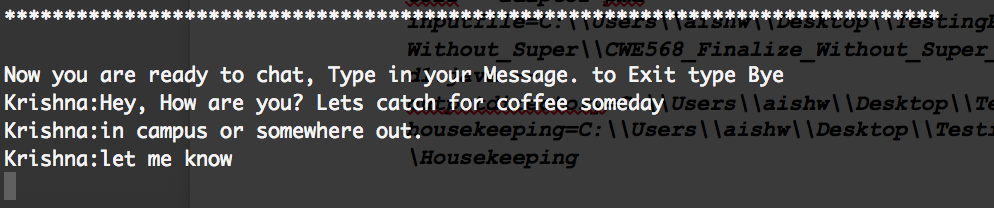
Jenny 🡪 Bob

1. Now we can start chatting and below are screen shots for the same

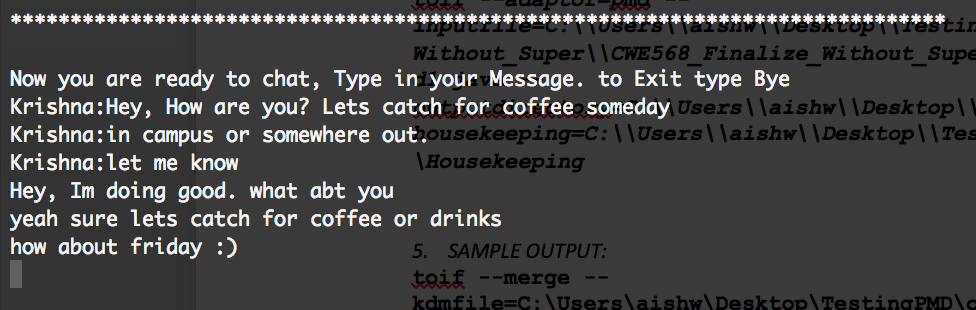
**Krishna Terminal**



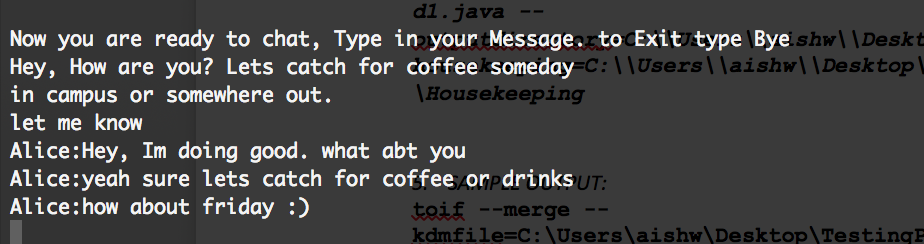
**Alice Terminal**



**Alice Terminal after typing in the message**

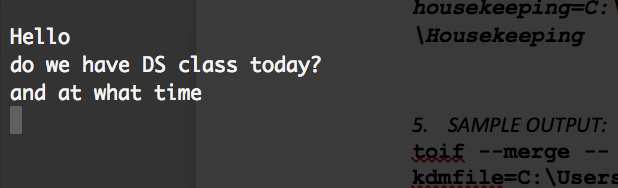


**Krishna Terminal**

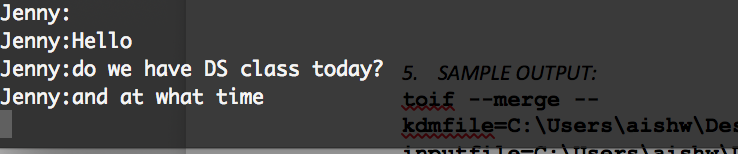


similarly, for other users.

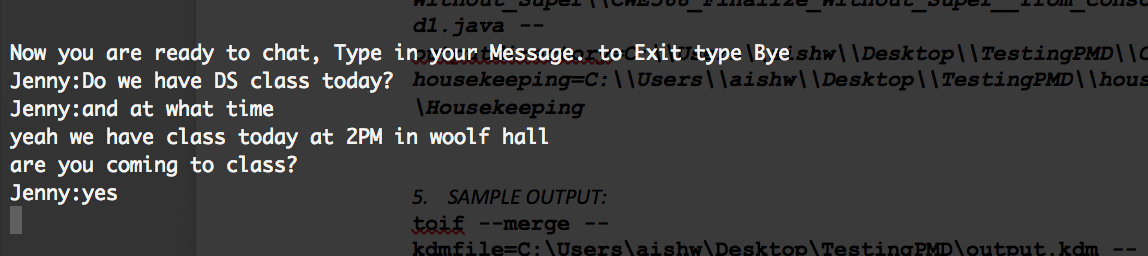
**Bob**



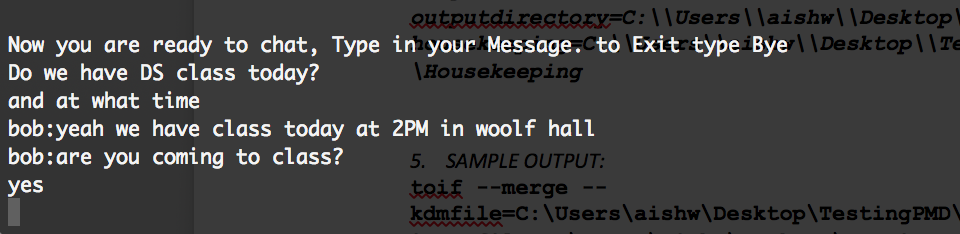
**Jenny Terminal**



**Jenny Terminal**



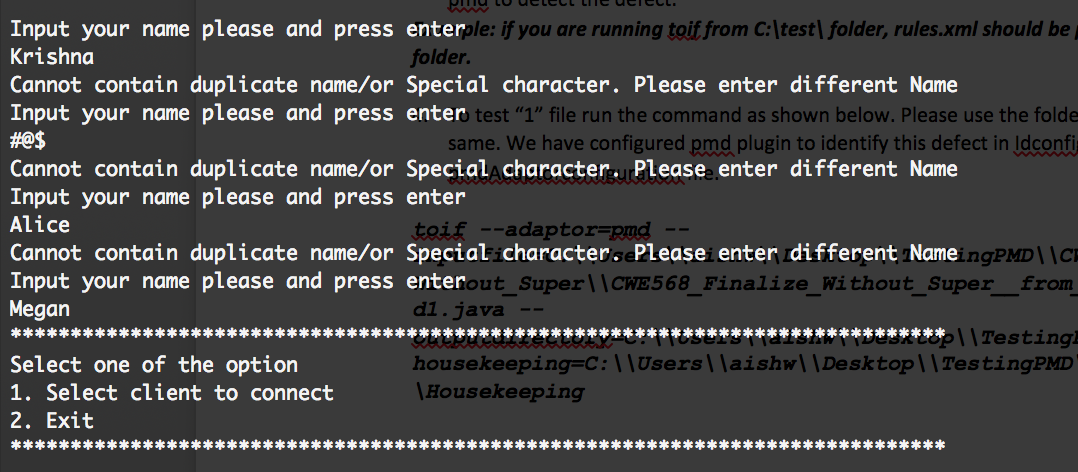
**Bob terminal**



# Handling the Exceptions

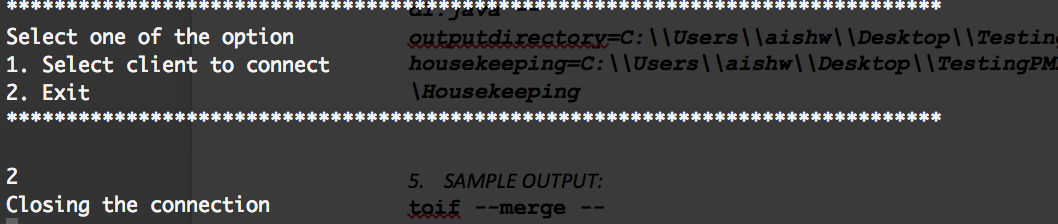
1. Duplicates Users are not allowed and Special symbols are not allowed(Bad Names)

When user enter duplicate/existing name he will be warned and asked to enter another name.



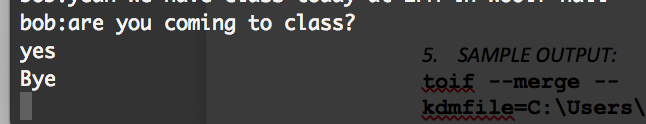
User wont accepted until he gives different names(Upper and Lower cases of same names are allowed).

1. In order to exit, select the option 2.

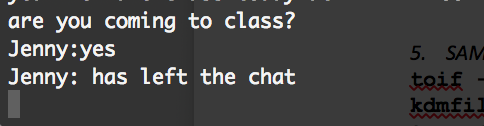


1. In order to exit while chatting. Press Bye(Case doenst matter). Other connected user will be notified.

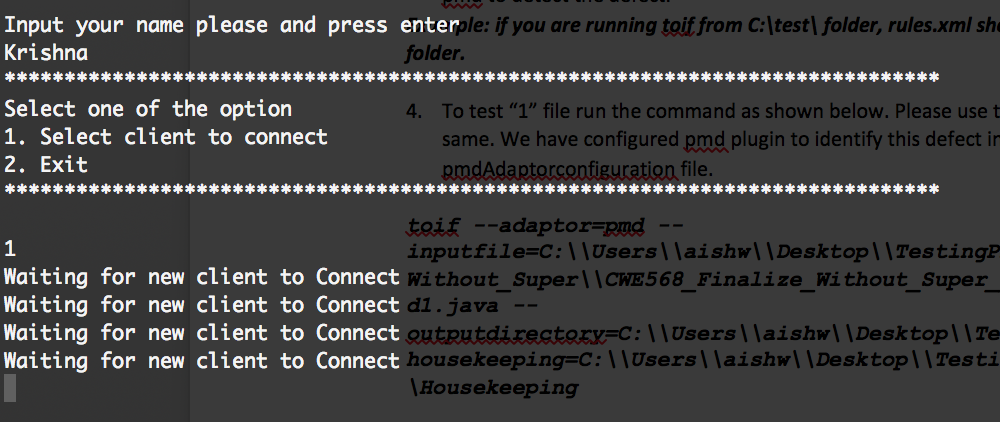
**Jenny enters bye.**



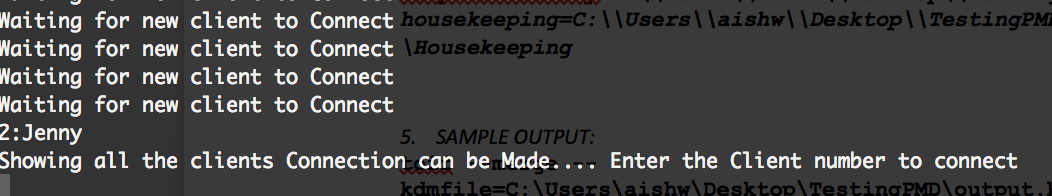
**Bob terminal – shows Jenny left.**

****

1. If no New users are connected. Then he will asked to wait until new user connected(5 secs checks for new user)

****

as soon as new user connects it will be updated here.



5)Server logs has been created(**Extra credit**). Here all message will be stored.

You can find the log file in the extracted folder. Sample output below

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Mar 09, 2017 7:14:35 PM Server main

INFO: New Client Connected

Mar 09, 2017 7:14:35 PM ServerIn run

INFO: HTTP/1.1 200 OK

Date:Thu, 09 Mar 2017 19:14:35

Server: Apache

Accept-Ranges: bytes

Content-Length: 38

Vary: Accept-Encoding

Content-Type: text/plain

Input your name please and press enter

Mar 09, 2017 7:14:59 PM ServerIn parseheader

INFO: GET Krishna HTTP/1.1

User-Agent: curl/7.16.3 libcurl/7.16.3 OpenSSL/0.9.7l zlib/1.2.3

Host: InstantMessaging System

Accept-Language: en, mi

Mar 09, 2017 7:14:59 PM ServerIn parseheader

INFO:

Mar 09, 2017 7:14:59 PM ServerIn run

INFO: Krishna

Mar 09, 2017 7:14:59 PM ServerIn run

INFO: HTTP/1.1 200 OK

Date:Thu, 09 Mar 2017 19:14:59

Server: Apache

Accept-Ranges: bytes

Content-Length: 222

Vary: Accept-Encoding

Content-Type: text/plain

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Select one of the option

1. Select client to connect

2. Exit

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Mar 09, 2017 7:15:02 PM Server main

INFO: New Client Connected

Mar 09, 2017 7:15:02 PM ServerIn run

INFO: HTTP/1.1 200 OK

Date:Thu, 09 Mar 2017 19:15:02

Server: Apache

Accept-Ranges: bytes

Content-Length: 38

Vary: Accept-Encoding

Content-Type: text/plain

Input your name please and press enter

Mar 09, 2017 7:15:35 PM ServerIn parseheader

INFO: GET Jenny HTTP/1.1

User-Agent: curl/7.16.3 libcurl/7.16.3 OpenSSL/0.9.7l zlib/1.2.3

Host: InstantMessaging System

Accept-Language: en, mi

Mar 09, 2017 7:15:35 PM ServerIn parseheader

INFO:

Mar 09, 2017 7:15:35 PM ServerIn run

INFO: Jenny

Mar 09, 2017 7:15:35 PM ServerIn run

INFO: HTTP/1.1 200 OK

Date:Thu, 09 Mar 2017 19:15:35

Server: Apache

Accept-Ranges: bytes

Content-Length: 222

Vary: Accept-Encoding

Content-Type: text/plain

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Select one of the option

1. Select client to connect

2. Exit

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*