Testcase 2001

System: Simple Chat Phase: 2 to 4

Server startup check with default arguments

Severity: 1

Instructions:

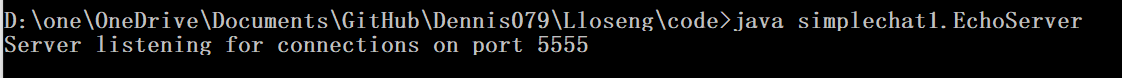
1. At the console, enter: java EchoServer.

Expected result:

1. The server reports that it is listening for clients by displaying

the following message:

Server listening for clients on port 5555



2. The server console awaits for user input.

Cleanup:

Hit CTRL+C to kill the server.

Testcase 2002

System: Simple Chat Phase: 2

Client startup check without a login

Severity: 1

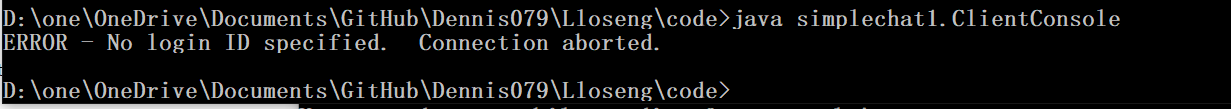
Instructions:

1. At the console, enter: java ClientConsole.

Expected result:

1. The client reports it cannot connect without a login by displaying:

ERROR - No login ID specified. Connection aborted.



2. The client terminates.

Cleanup: (if client is still active)

Hit CTRL+C to kill the client.

Testcase 2003

System: Simple Chat Phase: 2

Client startup check with a login and without a server

Severity: 1

Instructions:

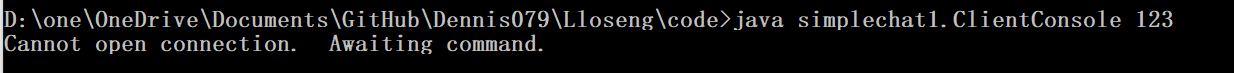
1. At the console, enter: java ClientConsole <loginID> where loginID

is the name you wish to be identified by.

Expected result:

1. The client reports it cannot connect to a server by displaying:

Cannot open connection. Awaiting command.



2. The client waits for user input

Cleanup:

Hit CTRL+C to kill the client.

Testcase 2004

System: Simple Chat Phase: 2

Client connection with default arguments

Severity: 1

Instructions:

1. Start a server (Testcase 2001, instruction 1)

2. On a different console on the same computer, start a

client (Testcase 2003, instruction 1)

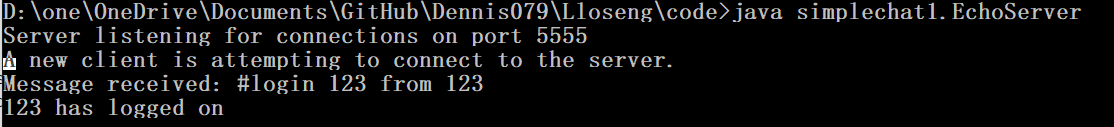
Expected results:

1. The server displays the following messages in sequence:

A new client is attempting to connect to the server.

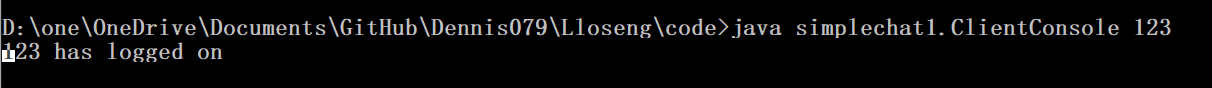
Message received #login <loginID> from null.

<loginID> has logged on.



2. The client displays message:

<loginID> has logged on.



3. The client and the server wait for user input.

Cleanup: (unless proceeding to Testcase 2005)

Hit CTRL+C to kill the client.

Hit CTRL+C to kill the server.

Testcase 2005

System: Simple Chat Phase: 2

Client Data transfer and data echo

Severity: 1

Instructions:

1. Start a server and a client on the same computer and connect using

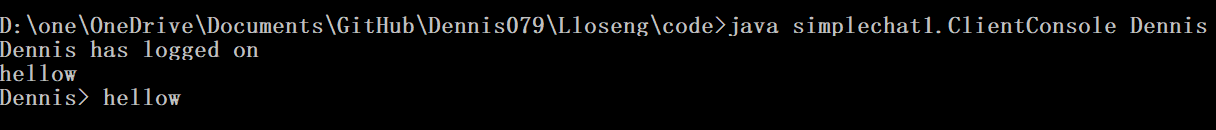
default arguments (Testcase 2004 instructions).

2. Once connected, type in data on the client console and press ENTER.

Expected results:

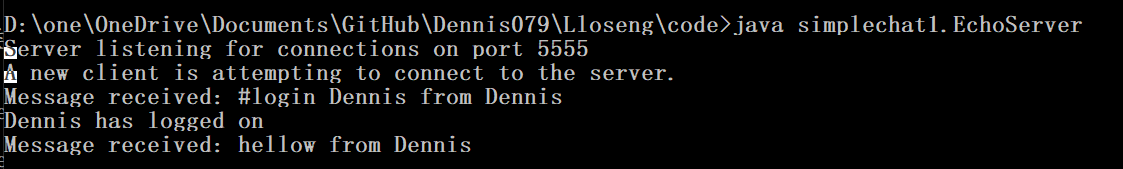
1. The message is echoed on the client side, but is preceded by the

sender's loginID and the greater than symbol(">").



2. The server displays a message similar to the following:

Message received: <user input> from <loginID>



Cleanup: (unless proceeding to Testcase 2006)

Hit CTRL+C to kill the client.

Hit CTRL+C to kill the server.

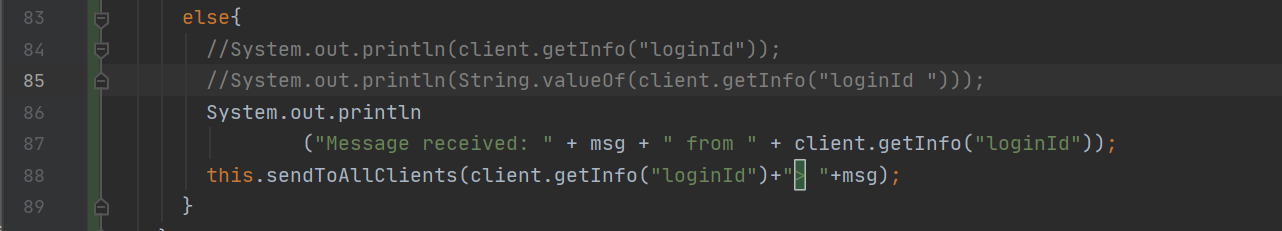
2005 problem

I meet a problem: (EchoServer.java)

System.out.println(client.getInfo("loginId")); “123” as out put

Is not equals to

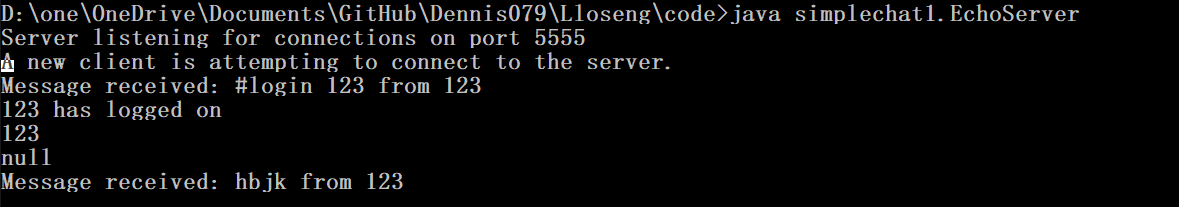
System.out.println(String.valueOf(client.getInfo("loginId "))); “null” as out put



ClientConsole ( loginid: 123) input hbjk

But in echoServer, when I try to get the id by “this.sendToAllClients(client.getInfo("loginId"))”

it gives me “null” as feedback



Testcase 2006

System: Simple Chat Phase: 2

Multiple local connections

Severity: 1

Instructions:

1. Start a server and multiple clients with DIFFERENT loginIDs and connect

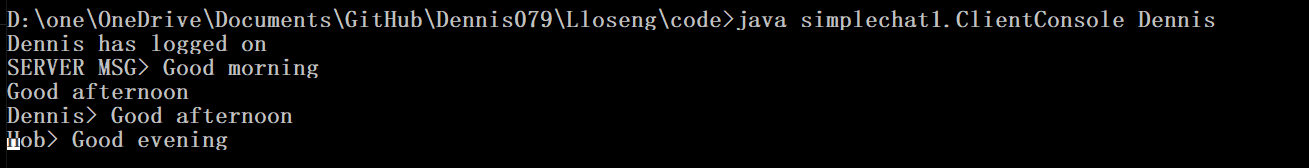
them to the server using default arguments. (Testcase 2005 instructions).

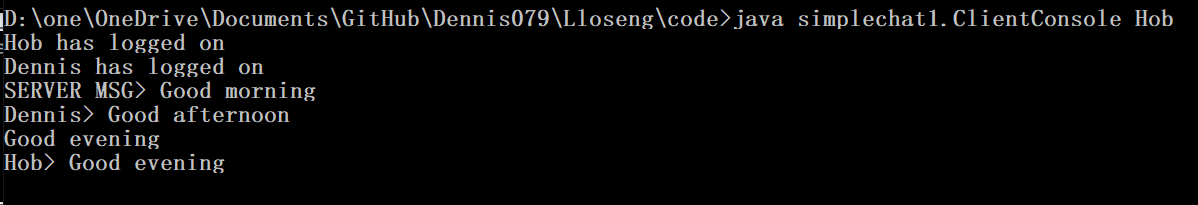
2. Start typing on all the client consoles AND the server console,

pressing ENTER to send each message.

Expected results:

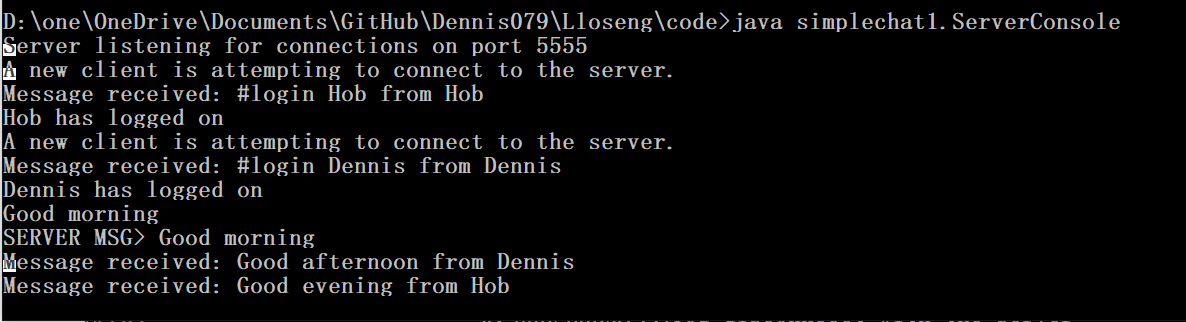
1. All client messages are echoed as in Testcase 2005.





2. All messages from the server console are echoed on the server console

and to all clients, but are preceeded by "SERVER MESSAGE> ".



Cleanup:

Hit CTRL+C to kill the clients

Hit CTRL+C to kill the servers

Testcase 2007

System: Simple Chat Phase: 2 and subsequent

Server termination command check

Severity: 1

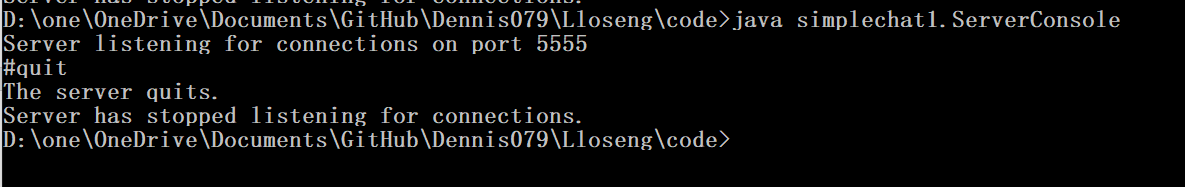
Instructions:

1. Start a server (Testcase 2001 instruction 1) using default arguments.

2. Type "#quit" into the server's console.

Expected result:

1. The server quits.



Cleanup (If the server is still active):

Hit CTRL+C to kill the server.

Testcase 2008

System: Simple Chat Phase: 2 and 3

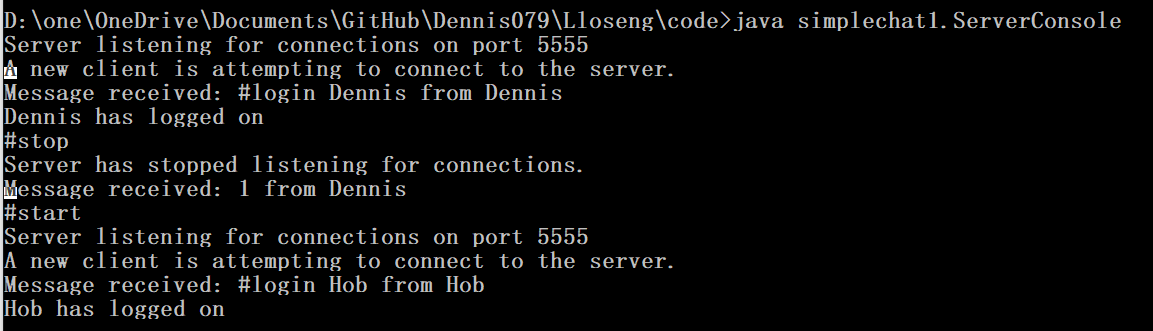
Server stop check

Severity: 2

Expected results:

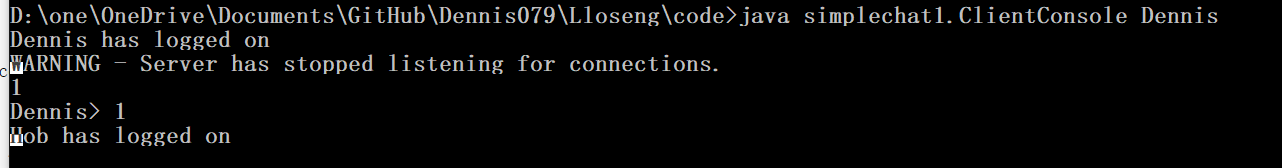
1. The server displays:

Server has stopped listening for connections.



2. The client displays:

WARNING - Server has stopped listening for connections.

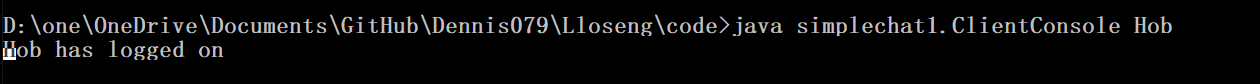


3. Data echoing still works normally.

4. Attempting to connect other clients will hang the console as these new

connections are put on backlog until the max backlog (default of 10) is reached.

5. Restarting the server will cause the backlogged clients to connect properly.



Cleanup: (Unless proceeding to Testcase 2008)

Type #quit to kill the server

Hit CTRL+C to kill the client

Testcase 2009

System: Simple Chat Phase: 2 and 3

Server close command check

Severity: 2

Instructions:

1. Start a server and connect a client to it. (Testcase 2007 instruction 1)

2. Stop the server using the #stop command.

3. Type "#close" into the server's console.

Expected result:

1. Server displays in sequence:

Server has stopped listening for connections.

<loginID> has disconnected.



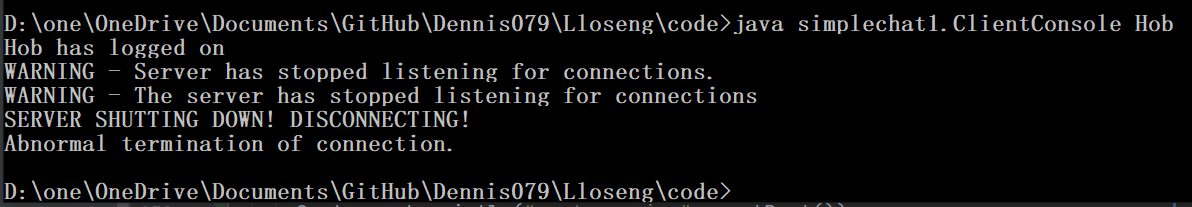
2. The client displays:

WARNING - The server has stopped listening for connections

SERVER SHUTTING DOWN! DISCONNECTING!

Abnormal termination of connection.

and then waits for input.



Cleanup: (Unless proceeding to Testcase 2009)

Hit CTRL+C to kill the client.

Hit CTRL+C to kill the server.

Testcase 2009

System: Simple Chat Phase: 2 and subsequent

Server restart

Severity: 1

Instructions:

1. Start a server, connect a client, and close the server. (Testcase 2008)

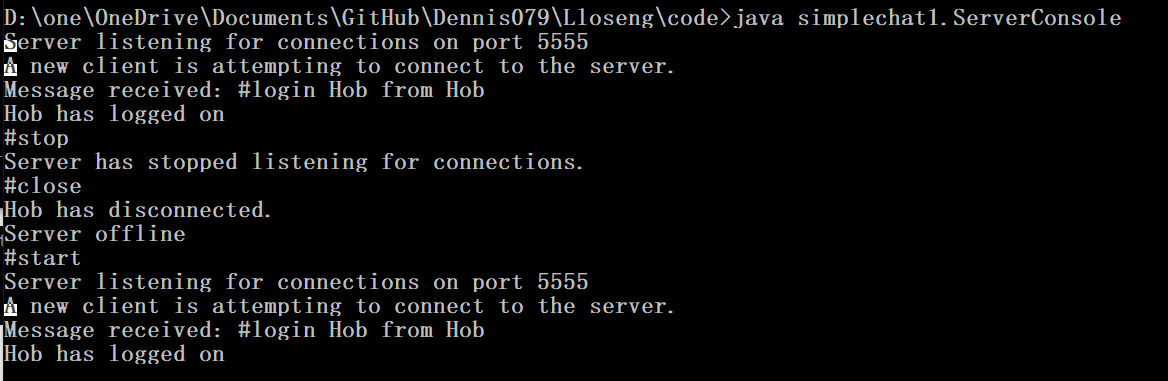
2. Type "#start" into the server's console.

3. Attempt to connect a client.

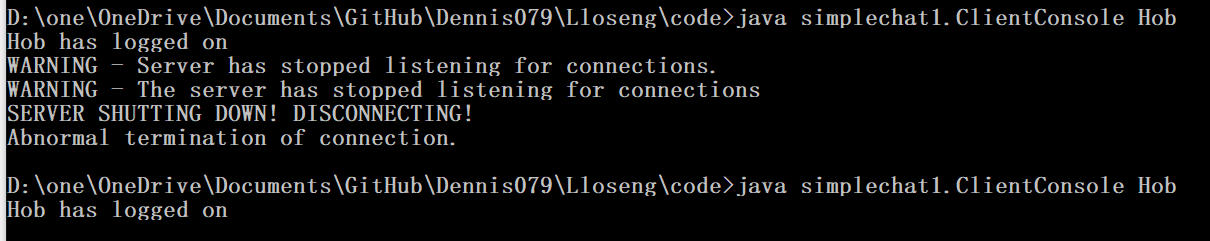
Expected result:

1. The server closes, restarts and then displays:

Server listening for connections on port 5555.



2. The client connects normally as described in Testcase 2004.



Cleanup: (Unless proceeding to Testcase 2010)

Hit CTRL+C to kill the client.

Type #quit to kill the server.

Testcase 2010

System: Simple Chat Phase: 2 and subsequent

Client termination command check

Severity: 1

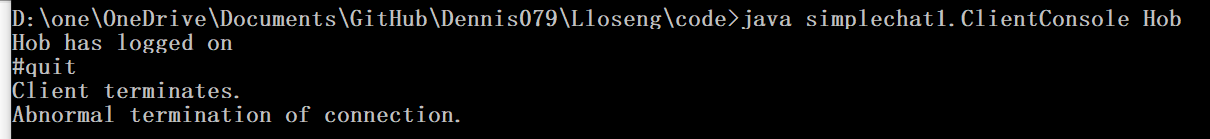
Instructions:

1. Start a client (Testcase 2002, instruction 1).

2. Type "#quit" into the client's console.

Expected result:

1. Client terminates.



Cleanup: (If client is still active)

Hit CTRL+C to kill the client.

//start a serve first

Testcase 2011

System: Simple Chat Phase: 2 and subsequent

Client logoff check

Severity: 1

Instructions:

1. Start a server (Testcase 1001, instruction 1), and then connect a

single local client to this server.

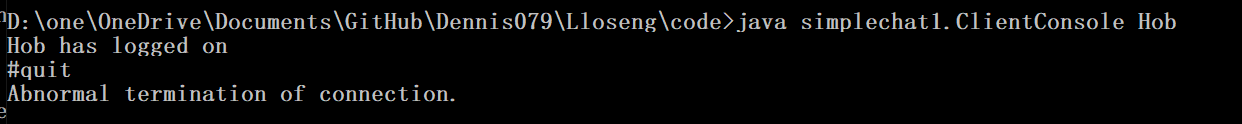
2. Type "#logoff" into this client's console.

Expected results:

1. Client disconnects and displays

Connection closed. (Under NT, it will display

Abnormal termination of connection.)



Cleanup: (Unless proceeding to Testcase 2012)

Type "#quit" to kill the client.

Testcase 2012

System: Simple Chat Phase: 2 and subsequent

Client host and port setup commands check

Severity: 1

Instructions:

1. Start a client but no servers, and attempt to connect using default

arguments.

2. At the client's console, type

"#sethost <newhost>"

where <newhost> is the name of a computer on the network

3. At the client's console, type

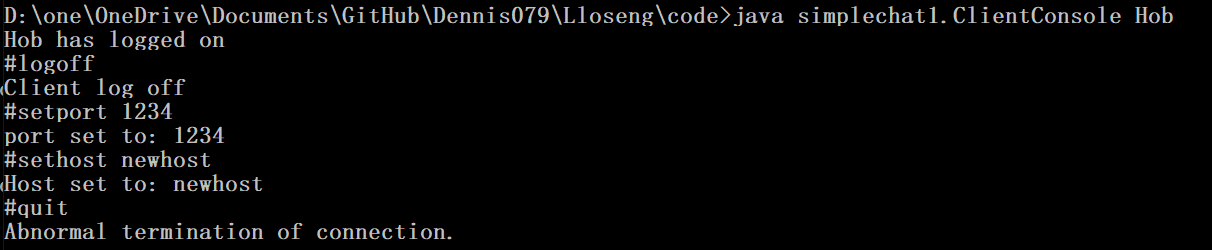
".

Expected result:

1. The client displays

Host set to: <newhost>

port set to: 1234.



Cleanup:

Type #quit to kill the client.

Testcase 2013

System: Simple Chat Phase: 2 and subsequent

Starting a server on a non-default port

Severity: 1

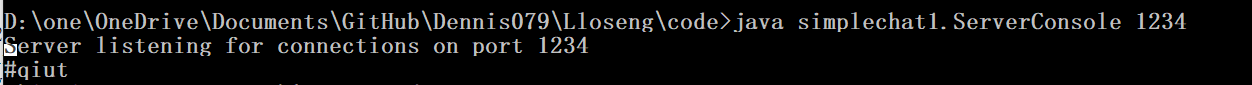
Instructions:

1. Start a server by typing java ServerConsole 1234.

Expected result:

1. The server displays

Server listening for connections on port 1234.



Cleanup (Unless proceeding to Testcase 2014)

Type #quit to kill the server.

Testcase 2014

System: Simple Chat Phase: 2 and subsequent

Connecting a client to a non-default host or port

Severity: 1

Instructions:

1. Start a server on port 1234 (Testcase 2013)



2. On a different computer, start a client by typing

"java ClientConsole <loginID> <host> 1234"

replacing the parameters by appropriate values.

黑色的屏幕截图

描述已自动生成

Expected Result:

1. The connection occurs normally.

Testcase 2015

System: Simple Chat Phase: 2

Multiple remote clients disconnections and reconnections

Severity: 1

Instructions:

1. Start a server (Testcase 2001, instruction 1).

2. On different computers, start clients (1 or 2 per computer) and connect

them to the server.

3. Exchange data.

4. Close the server using the #close command.

5. Change the server's port by typing

"#setport <newport>"

6. Restart the server using the #start command.

7. Change the ports of each clients using the #setport command.

8. Reconnect the clients to the server by using the #login <loginID>

command.

9. close the server by using the #quit command.

Expected results:

1. The first set of connections occur normally.

2. When the server is closed, all clients are disconnected.

3. The server displays the following message when the #setport command

is used:

port set to: <newport>.

4. The server restarts and displays:

Server listening for connections on port <newport>.

5. The clients change port as in Testcase 2012.

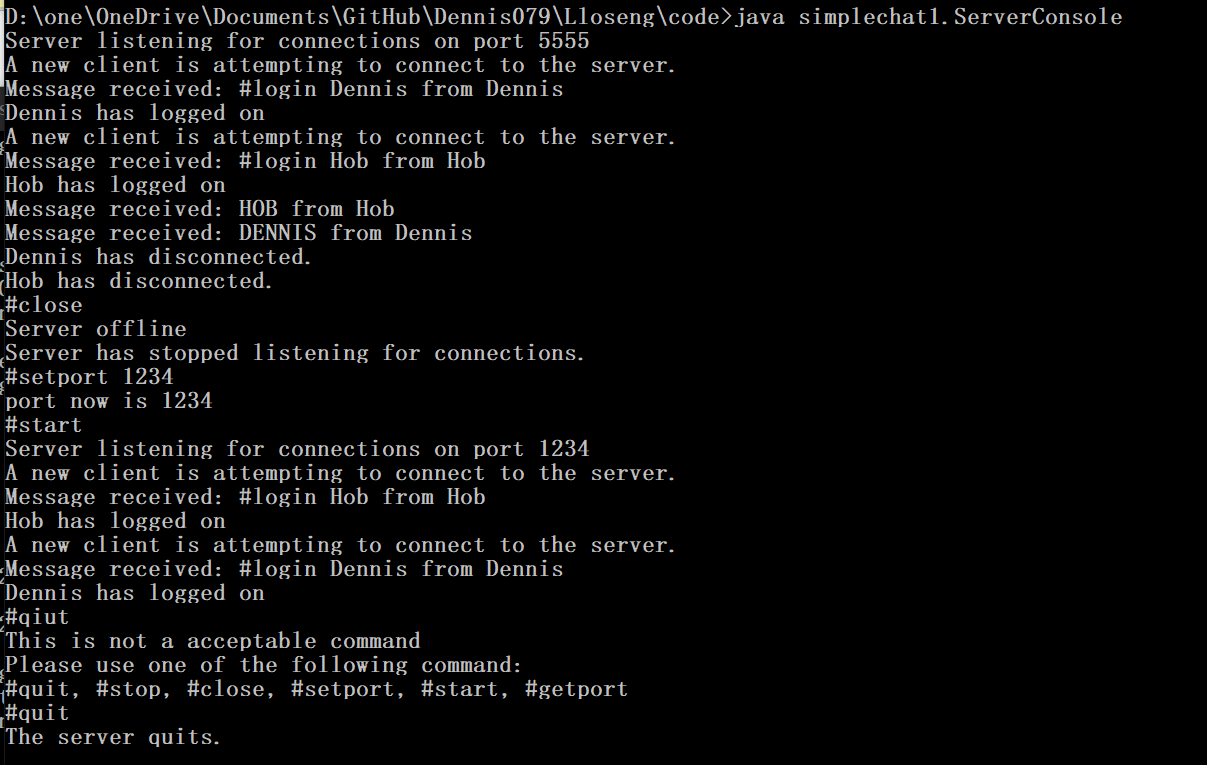
6. The clients reconnect normally.

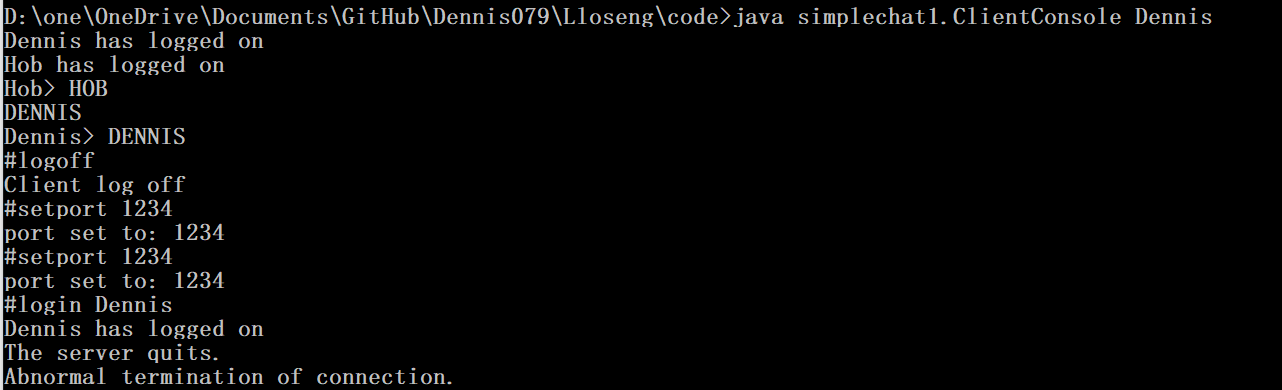
7. The clients are disconnected when the server quits.

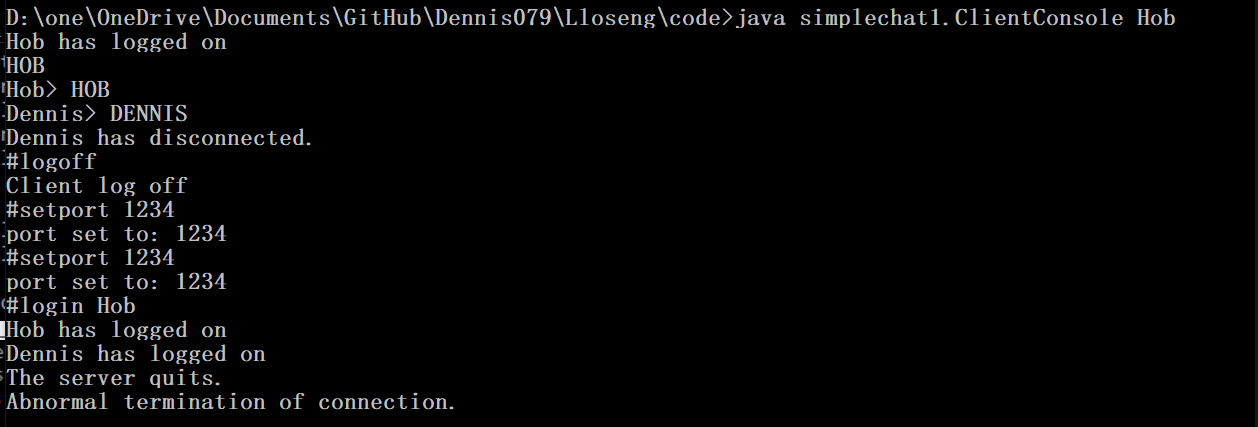
Cleanup:

Type #quit to kill the clients

Type #quit to kill the server (if still active)







Testcase 2017

System: Simple Chat Phase: 2 and subsequent

Client quitting or logging off a server with multiple connections

Severity: 1

Instructions:

1. Start a server and connect multiple (at least 3) clients.

2. In one client's console, type "#quit".

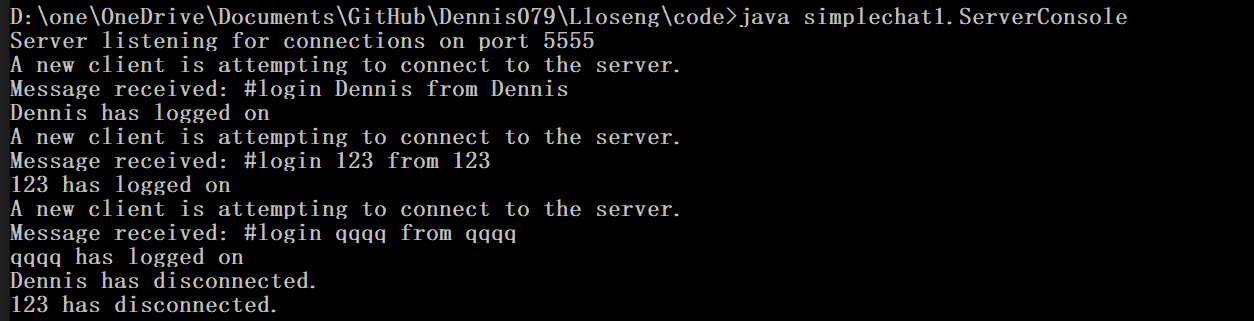
3. In a second client's console type "#logoff".

Expected results:

1. In both cases, all remaining clients and the server get the following

message:

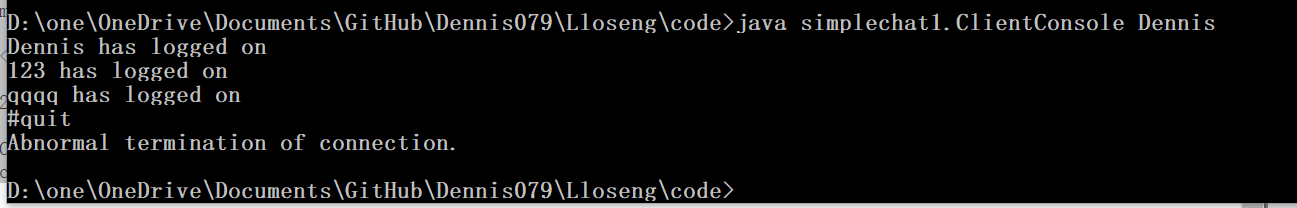
<loginID> has disconnected.

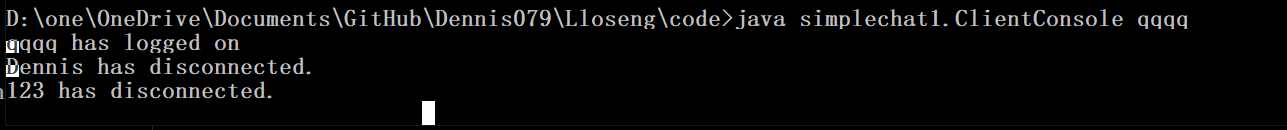


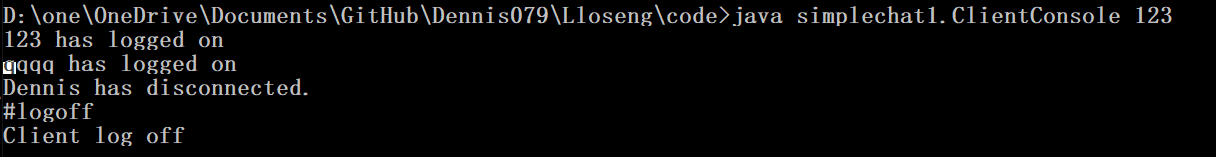
2. The clients display:

Connection Closed. (Exception: NT will display "Abnormal termination of

connection." when the logoff command is used.)







For 2016, 2018, 2019, I do not have condition to do it.