Report of Assignment

Student Name: Sun Zhenhang   Student ID: z5231733

(Using Python 3.7)

1. Program Design:
   In this assignment, I use two major modules: the server program and client program. The server program can serve multiple clients.
   1) First, I simply establish a TCP connection between server and a single client, and implement a "Login System".
   2) Then I add a timeout function to test whether the client is still active, if not, the server will disconnect with this client.
   3) Next, I use the "thread" to allow the server to connect with multiple clients.
   4) After accomplishing this, I start to apply the function of "message" and "broadcast" to achieve the communication among the clients.
   5) And then I achieve the requirement of blacklist and do some modifications in "message" and "broadcast".
   6) Finally, I try to implement "P2P" , which can transmit information without the server, to this architecture.

2. Operation and Message Format:
   1) First, run the server program, when it is ready to accept client, then run the client program.
   2) When you run the client successfully, you will occur these following messages:

```
Username: ethan
Invalid Username. Please try again
Username: yoda
Password: aaa
Invalid Password. Please try again
Password: bbb
Invalid Password. Please try again
Password: ccc
Invalid Password. Your account has been blocked. Please try again later
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```

Then when you want to login in this user again within the block time, you will receive a message like:

```
Username: yoda
Password: aaa
Your account is blocked due to multiple login failures. Please try again later
```

After the block time, you can login this user again.

Suppose now I have logged in the server the client called "yoda", and another client "hans" login after "yoda". And I can do following operations, and each client will output corresponding messages:
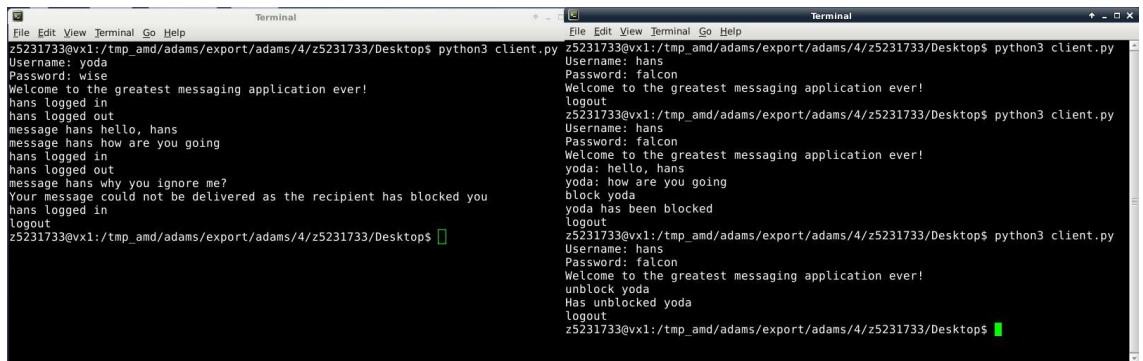
```
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$ python3 client.py
Username: yoda
Password: wise
Welcome to the greatest messaging application ever!
hans logged in
hans: hello, yoda
message hans hello, hans
Your message could not be delivered as the recipient has blocked you
message hans hello, hans
broadcast hi
Your message could not be delivered to some recipients
broadcast hi
hans logged out
logout
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```

```
                                    Terminal                    ↑ _ □ ×
File  Edit  View  Terminal  Go  Help
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$ python3 client.py
Username: hans
Password: falcon
Welcome to the greatest messaging application ever!
message yoda hello, yoda
block yoda
yoda has been blocked
unblock yoda
Has unblocked yoda
yoda: hello, hans
block yoda
yoda has been blocked
unblock yoda
Has unblocked yoda
yoda: hi
logout
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```

For "whoelse" and "whoelsesince" commands:

```
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$ python3 client.py
Username: yoda
Password: wise
Welcome to the greatest messaging application ever!
hans logged in
whoelse
hans
logout
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```

```
                                    Terminal                    ↑ _ □ ×
File  Edit  View  Terminal  Go  Help
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$ python3 client.py
Username: hans
Password: falcon
Welcome to the greatest messaging application ever!
whoelse
yoda
whoelsesince 10
yoda
yoda logged out
whoelsesince 10
yoda
whoelsesince 5
No other users has logged in with the past 5 seconds
whoelsesince 100
yoda
logout
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```

For timeout:

```
File  Edit  View  Terminal  Go  Help
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$ python3 client.py
Username: yoda
Password: wise
Welcome to the greatest messaging application ever!
z5231733@vx1:/tmp_amd/adams/export/adams/4/z5231733/Desktop$
```
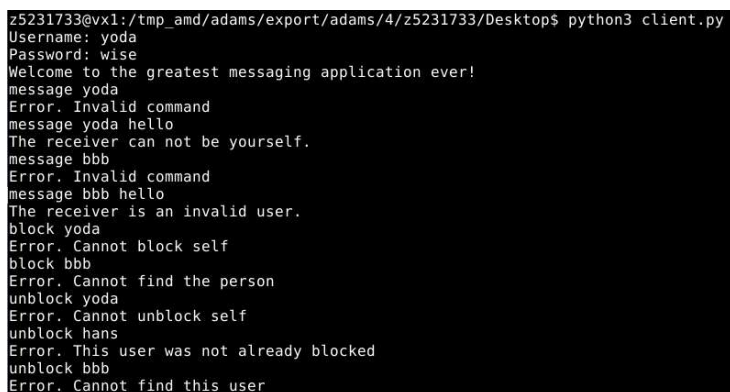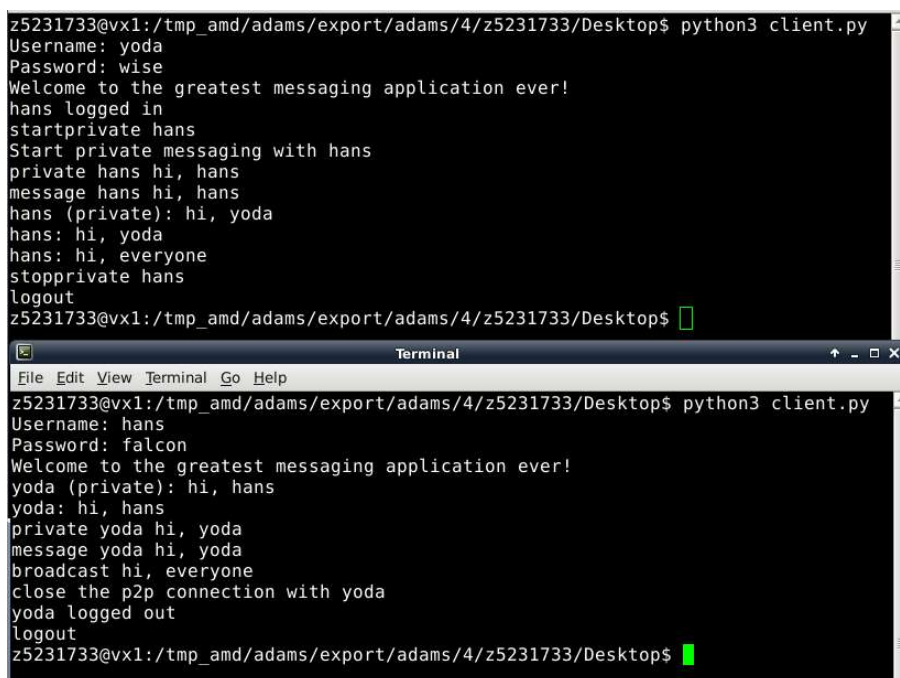
And "yoda" can send message to "hans" if "hans" is offline and does not block yoda:

Any invalid command are shown below:

3) Then, if "yoda" want to establish a private connection with "hans", and they can do like this: