TCP-Multi User

Input :

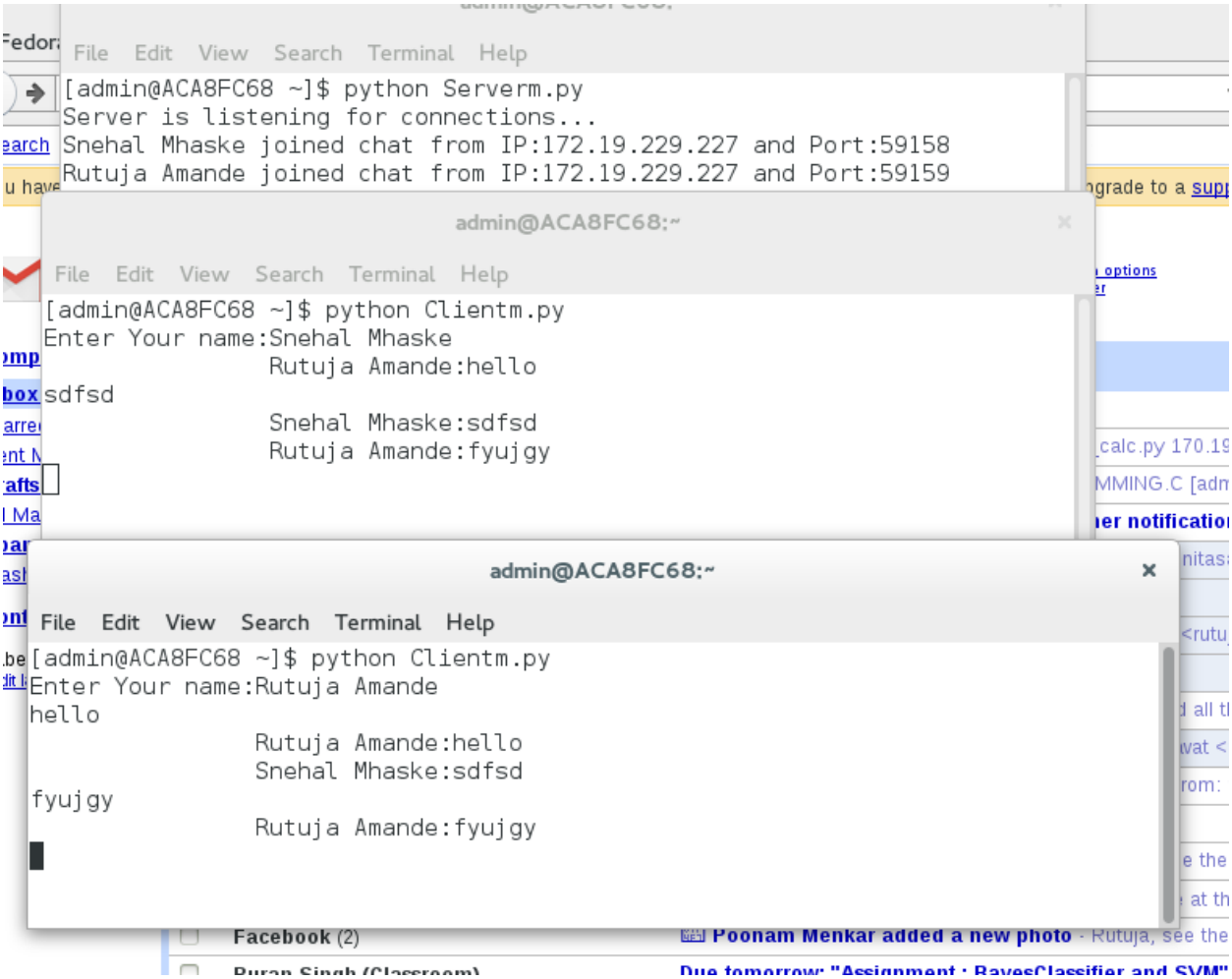
Serverm.py

```
"""
A simple TCP multiuser server
"""
import socket
import os
from threading import Thread
import threading
import thread

clients = set() #add data like list
clients_lock = threading.Lock()
names = {} #like dic

def listener(client, address):
    name=client.recv(1024)   #Receive name of connected client
    names[client] = name          #Add name to names dictionary
    print names[client]+" joined chat from IP:"+str(address[0])+" and
Port:"+str(address[1])
    with clients_lock:
        clients.add(client)       #Adding client socket value to clients
list
        #print clients
    try:
        while True:
            data=client.recv(1024) #Receive data from client
            if not data:
                break
            else:
                data = names[client]+":"+str(data)  #Append data with
client name
                #print data
                with clients_lock:
                    for c in clients:   #Send the data to all the clients
                        c.send(data)
    finally:
        with clients_lock:
            clients.remove(client)#If client exit, then remove the client
value
            client.close()

host = '172.19.229.227'           #Server IP addres
port = 10015     #Port address

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) #Creating a Socket
s.bind((host,port))    #Binding Port and IP address
```

```
s.listen(3)            #Listen to connection

th = []          #Array for threads
print "Server is listening for connections..."
while True:
      '''Accept Connection'''      #Accept a client connecion
      client,address=s.accept()
      try:
        #Launch a thread for each client request
        th.append(Thread(target=listener, args =
(client,address)).start())

      except (KeyboardInterrupt, SystemExit):
        cleanup_stop_thread();
        s.close()       #Closing the socket
        sys.exit()
```

Clientm.py

```
import socket
from threading import Thread, Lock

def receiver(sock):
      while flag:
            data = sock.recv(size)
            print "\t\t"+data

host = '172.19.229.227'           #Server IP addres
port = 10015                #Port address
size = 1024                 #Size of packet data to receive
flag = True

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM) #Creating a
Socket
sock.connect((host,port))   #Creating a Socket'''

'''Connect to Server'''

# Start Receiver Thread
hop = Thread(target=receiver, args=(sock,))
hop.daemon = True
hop.start()

data = raw_input("Enter Your name:")
sock.send(data)
while data != 'quit':       #User will enter 'quit' to exit server
      data = raw_input()          #Enter message to send to server
      sock.send(data)             #Sending data to server

sock.close()     #Closing the socket
```


File Edit View Search Terminal Help
[admin@ACA8FC68 ~]$ python Serverm.py
Server is listening for connections...
Snehal Mhaske joined chat from IP:172.19.229.227 and Port:59158
Rutuja Amande joined chat from IP:172.19.229.227 and Port:59159
admin@ACA8FC68:~
File Edit View Search Terminal Help
[admin@ACA8FC68 ~]$ python Clientm.py
Enter Your name:Snehal Mhaske
Rutuja Amande:hello
sdfsd
Snehal Mhaske:sdfsd
Rutuja Amande:fyujgy
admin@ACA8FC68:~
File Edit View Search Terminal Help
[admin@ACA8FC68 ~]$ python Clientm.py
Enter Your name:Rutuja Amande
hello
Rutuja Amande:hello
Snehal Mhaske:sdfsd
fyujgy
Rutuja Amande:fyujgy
Facebook (2)
Poonam Menkar added a new photo - Rutuja, see the