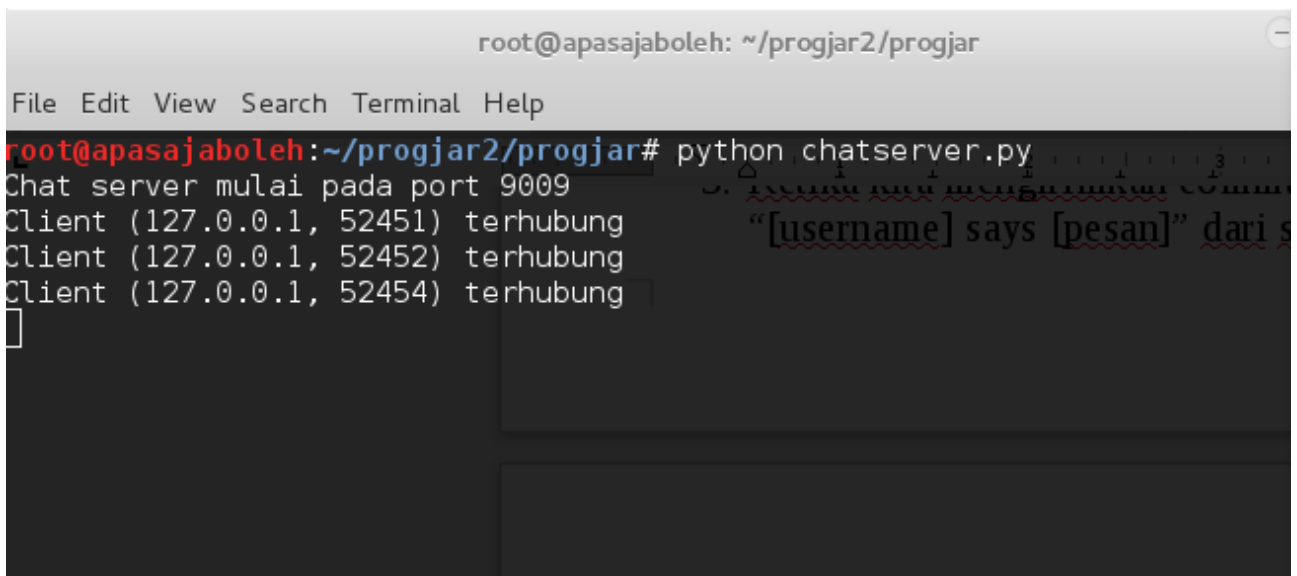


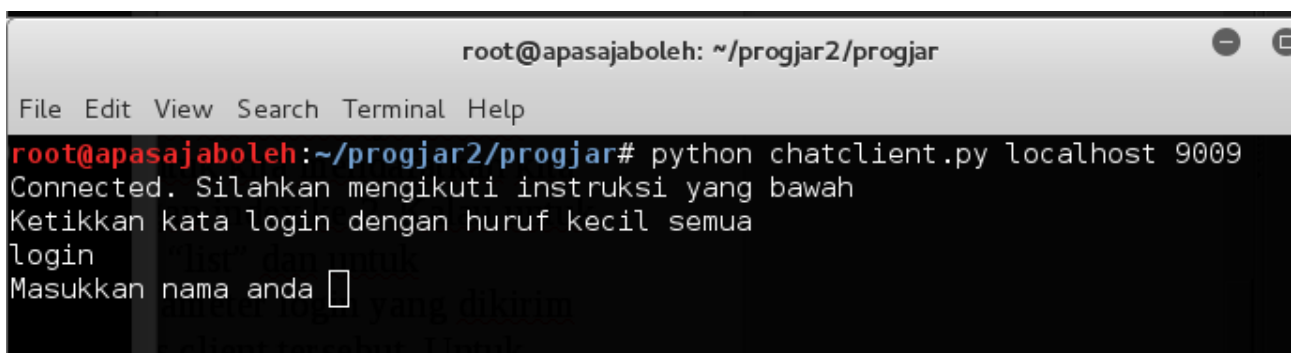
Dokumentasi Tugas 2 Progjar B  
Chatting Client – Server  
Freddy Hermawan Yuwono  
5113100040

1. Pertama tama kita jalankan chat servernya terlebih dahulu melalui terminal linux (kalau punya saya) python chatserver.py. Tetapi sebelumnya kita harus mengetahui bahwa dia sudah berada didirectory tempat kita menyimpan chatserver.py.



```
root@apasajableh: ~/progjar2/progjar
File Edit View Search Terminal Help
root@apasajableh:~/progjar2/progjar# python chatserver.py
Chat server mulai pada port 9009
Client (127.0.0.1, 52451) terhubung
Client (127.0.0.1, 52452) terhubung
Client (127.0.0.1, 52454) terhubung
```

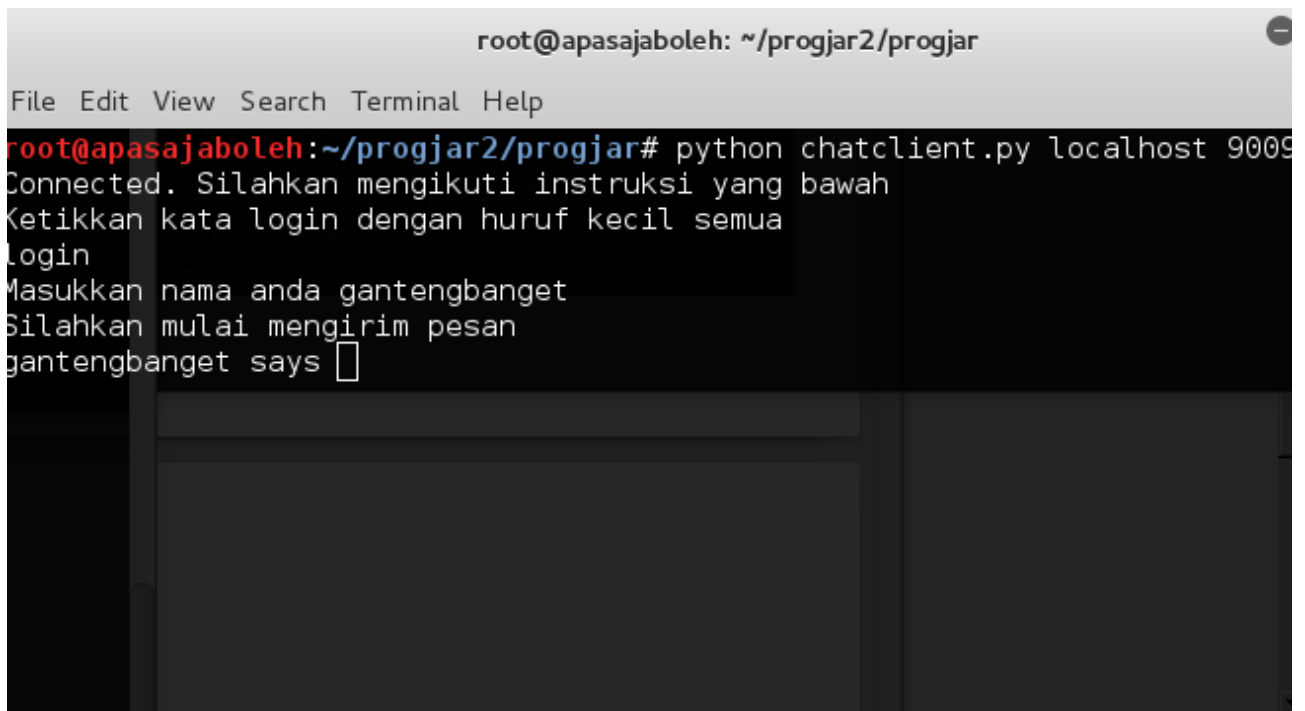
2. Kalau sudah kita tinggal buka tiga terminal baru diluar dari terminal yang sudah ada dan arahkan dahulu pada directory yang tepat dan kita langsung run (kalau punya saya) python chatclient.py localhost 9009. kenapa ada tambahan localhost 9009. Jadi localhost sendiri adalah alamat yang akan kita tuju dan 9009 adalah port yang akan kita buka untuk melakukan chatting roomnya.



```
root@apasajableh: ~/progjar2/progjar
File Edit View Search Terminal Help
root@apasajableh:~/progjar2/progjar# python chatclient.py localhost 9009
Connected. Silahkan mengikuti instruksi yang bawah
Ketikkan kata login dengan huruf kecil semua
login
Masukkan nama anda
```

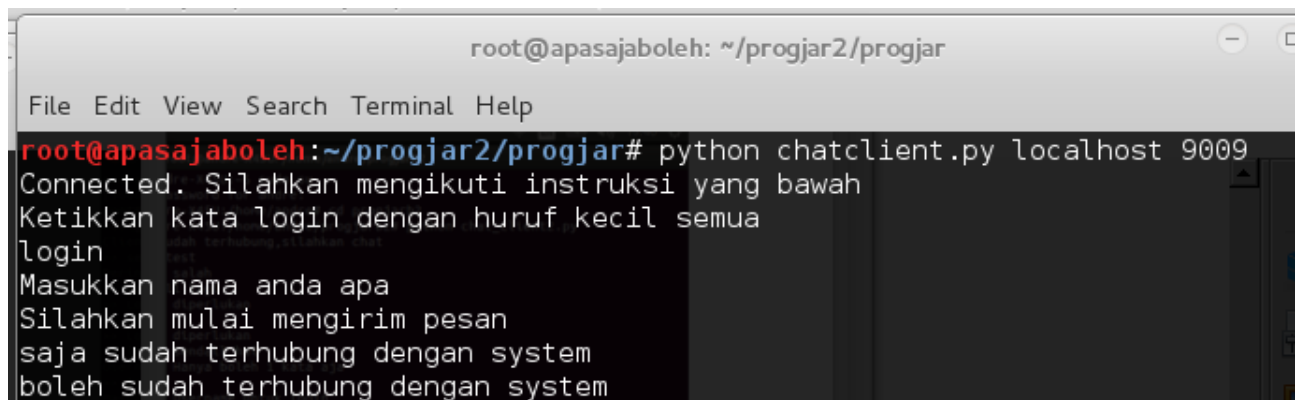
3. Kalau programnya sudah berjalan dengan baik langkah selanjutnya

adalah kita mengetikkan kata “login” pada program yang telah jalan. Sebagai catatan bahwa apabila kita tidak mengetikkan kata “login” baik itu dengan spasi ataupun salah penulisan maka programnya akan tutup sendiri. Selanjutnya kita mengetikkan username / nama yang akan digunakan dalam chatting room. Dengan syarat bahwa usernamenya jangan sama dan dalam usernamenya jangan ada karakter “ ” / spasi. Soalnya jika tidak programnya bisa error dan tidak bisa berjalan dengan baik lagi karena saya menggunakan string split.



```
root@apasajaboleh: ~/progjar2/progjar
File Edit View Search Terminal Help
root@apasajaboleh:~/progjar2/progjar# python chatclient.py localhost 9009
Connected. Silahkan mengikuti instruksi yang bawah
Ketikkan kata login dengan huruf kecil semua
login
Masukkan nama anda gantengbanget
Silahkan mulai mengirim pesan
gantengbanget says
```

4. Jika kita sudah berada didalam server tersebut dan sebagai tambahan apabila kita sudah berhasil login maka akan timbul notifikasi kepada semua user yang online selain kita. Dan kita sudah bisa berkomunikasi dengan yang lainnya.



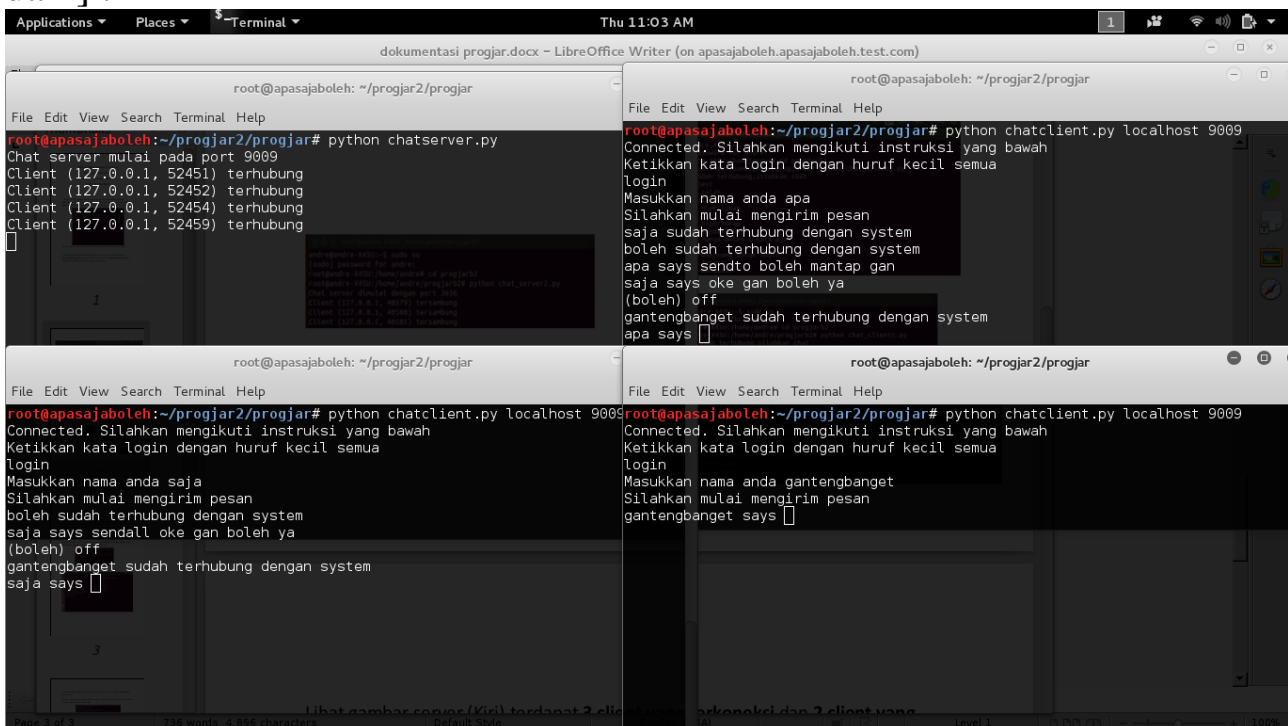
```
root@apasajaboleh: ~/progjar2/progjar
File Edit View Search Terminal Help
root@apasajaboleh:~/progjar2/progjar# python chatclient.py localhost 9009
Connected. Silahkan mengikuti instruksi yang bawah
Ketikkan kata login dengan huruf kecil semua
login
Masukkan nama anda apa
Silahkan mulai mengirim pesan
saja sudah terhubung dengan system
boleh sudah terhubung dengan system
```

- Salah satu contoh pada salah satu user yang aktif lainnya.
5. Perintah perintah yang ada adalah sebagai berikut dengan format sebagai berikut. Jika itu salah perintah atau kita langsung tulis pesan

saja maka yang akan keluar adalah “invalid command”. Perintah yang tersedia :

1. list adalah untuk melihat user yang aktif siapa saja. Formatnya adalah list
2. sendall adalah untuk mengirimkan kesemua user yang aktif pada saat itu. Formatnya adalah sendall [spasi] pesan
3. sendto adalah mengirimkan pesan secara tersendiri kepada user yang kita tuju. Formatnya adalah sendto [spasi] [user yang dituju] [spasi] [pesan].

Semuanya komentarnya dengan huruf kecil semuanya dan tanpa kurung [ dan ] .



The screenshot shows a terminal window with a chat server and client interaction. The server is running on port 9009 and the client is running on port 9008. The server output shows several clients connecting and sending messages. The client output shows the user logging in and sending messages.

```
root@apasajaboleh: ~/progjar2/progjar
File Edit View Search Terminal Help
root@apasajaboleh:~/progjar2/progjar# python chatserver.py
Chat server mulai pada port 9009
Client (127.0.0.1, 52451) terhubung
Client (127.0.0.1, 52452) terhubung
Client (127.0.0.1, 52454) terhubung
Client (127.0.0.1, 52459) terhubung
root@apasajaboleh:~/progjar2/progjar# python chatclient.py localhost 9008
Connected. Silahkan mengikuti instruksi yang bawah
Ketikkan kata login dengan huruf kecil semua
login
Masukkan nama anda saja
Silahkan mulai mengirim pesan
boleh sudah terhubung dengan system
saja says sendall oke gan boleh ya
(boleh) off
gantengbanget sudah terhubung dengan system
saja says
```

Ini adalah salah satu hasil screenshotan dari programnya yang berjalan.

note : line 72 dan 73 sebenarnya tidak terlalu berguna pada chatserver.py. Jalan kerja programnya adalah sebagai berikut :

1. Ketika program servernya sudah jalan maka dia akan terus listen dan menunggu respon masuk keserver.
2. ketika program client sudah dijalankan maka pertama kali dia akan melihat input pertama kita adalah kata “login” atau tidak. Apabila itu bukan kata “login” sebagai inputan pertama maka program tersebut akan tutup. Apabila inputan pertama kita adalah “login” selanjutnya kita tinggal mengetikkan username kita. Apabila sudah maka dia

akan mengecek apakah itu adalah " " / spasi atau tidak. Kalau spasi maka programnya tutup sedangkan apabila bukan maka dia akan mengirimkan username kita. Jadi bentuk datanya adalah "username says username" kenapa hal ini terjadi itu karena dalam program server aku set untuk array index ke 2 adalah sama dengan array index ke 1 adalah login. Jadi dicek dulu dalam server apakah itu sudah ada atau tidak dalam mapping antara socket object dan username yang ada. Jika sudah ada maka server akan merespon "username sudah dipake" terus diterjemahkan kedalam pada program client bahwa itu artinya username sudah ada maka program akan ditutup. Jika itu masih belum ada usernamenya maka selanjutnya adalah kita memappingkan antara socket objectnya itu dengan username dan kita memappingkan antara address yang kita gunakan dengan username yang kita pakai terus kita broadcastnya ke client yang sisanya selain dari yang sekarang kita ada.

3. Kalau sudah semuanya maka kita sudah terdaftar dan kita bisa mengesekusi command yang lain seperti list, sendto, sendall.
4. Penjelasan untuk command yang ada bahwa sebenarnya program yang aku buat untuk menyimpan namanya baik pada sisi client dan sisi servernya. Pada sisi clientnya itu sudah kita sudah menyimpan sejak memasukkan usernamenya sedangkan pada sisi servernya ketika kita sudah mengisikan kata usernamenya itu apa dan ketrima itu kita sudah menyimpannya.
5. Ketika kita mengirimkan command nantinya itu bentuknya seperti ini "[username] says [pesan]" dari sisi client. Jadi pada server aku memecahnya berdasarkan spasi yang ada. Kalau sudah aku mencocokkan pada array index kedua karena itu semuanya adalah kunci dari command yang aku gunakan untuk kita mendafatkan kita akan melihat bahwa index ke 0 sama dengan index ke 2. Kalau untuk list dari lihat dari index ke 2 sama dengan "list" dan untuk sendall, sendto semuanya sama. Untuk parameter login yang dikirim adalah socketnya dia, socket server, address client tersebut. Untuk parameter sendall adalah socketnya dia, socket server, pesan. Untuk sendto it parameternya adalah socket tujuan, pesan (yang terdiri atas dari mana dan pesan yang ada). Untuk parameter list adalah socket kita dan socket server.

Untuk codingan yang client

```
import sys, socket, select
```

```
def chat_client():
```

```
    if(len(sys.argv) < 3) :
```

```
        print 'pakai alamat dan juga portnya ya...'
```

```
        sys.exit()
```

```
    host = sys.argv[1]
```

```
    port = int(sys.argv[2])
```

```
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

```
    s.settimeout(2)
```

```
    # connect to remote host
```

```
    print 'Connected. Silahkan mengikuti instruksi yang bawah'
```

```
    print 'Ketikkan kata login dengan huruf kecil semua'
```

```
    #sys.stdout.write('Masukkan nama anda '); sys.stdout.flush()
```

```
    test=raw_input()
```

```
    #s.send(test)
```

```
    if test == "login" :
```

```
        sys.stdout.write('Masukkan nama anda '); sys.stdout.flush()
```

```
        test =raw_input()
```

```
        if test == ":
```

```
            sys.exit()
```

```
        try :
```

```
        s.connect((host, port))
```

```
        except :
```

```
        print 'Unable to connect'
```

```
        sys.exit()
```

```
        print 'Silahkan mulai mengirim pesan'
```

```
        sys.stdout.write(test+" says "); sys.stdout.flush()
```

```
        s.send(test+" says "+ test)
```

```
        while 1:
```

```
            socket_list = [sys.stdin, s]
```

```
            read_sockets, write_sockets, error_sockets =
```

```
select.select(socket_list , [], [])
```

```

for sock in read_sockets:
    if sock == s:
        data = sock.recv(4096)
        if not data :
            print '\nDC dari chat server'
            sys.exit()
        else :
            sys.stdout.write(data)
            if data == "\rusername sudah dipake\n":
                sys.exit()
            else :
                sys.stdout.write(test+" says "); sys.stdout.flush()

    else :

        msg = sys.stdin.readline()
        msg = test+" says "+msg
        s.send(msg)
        sys.stdout.write(test+" says "); sys.stdout.flush()

else :
    sys.exit()
if __name__ == "__main__":

    sys.exit(chat_client())

```

Untuk codingan yang server

```

import sys, socket, select

HOST = "
SOCKET_LIST = []
RECV_BUFFER = 4096
PORT = 9009
MAPPING = {}
alamat = {}
def chat_server():

```

```

server_socket = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)
server_socket.setsockopt(socket.SOL_SOCKET,
socket.SO_REUSEADDR, 1)
server_socket.bind((HOST, PORT))
server_socket.listen(10)
SOCKET_LIST.append(server_socket)

print "Chat server mulai pada port " + str(PORT)

while 1:
    ready_to_read,ready_to_write,in_error =
select.select(SOCKET_LIST,[],[],0)

    for sock in ready_to_read:
        if sock == server_socket:
            sockfd, addr = server_socket.accept()
            SOCKET_LIST.append(sockfd)
            print "Client (%s, %s) terhubung" % addr
            #broadcast(server_socket, sockfd, "\r[%s:%s] memasuki chatting
room kami\n" % addr)
        else:

            try:
                data = sock.recv(RECV_BUFFER)
                if data:
                    global MAPPING
                    global alamat
                    ji = data.split()
                    if ji[2] == "sendto" :
                        test =False
                        for key,value in MAPPING.iteritems():
                            if value == sock :
                                test=True
                        if test==True :
                            sendto(ji[3],ji)

                    elif ji[2] == "list" :
                        test =False

```

```

for key,value in MAPPING.iteritems():
    if value == sock :
        test=True
if test==True :
    daftar(sock,server_socket)

```

```

elif ji[2]=="sendall" :
    datajadian = "\r" + ji[0] + " says "
    count = 0
    while count !=len(ji):
        if count >2:
            datajadian += (ji[count]+" ")
            count+=1
        datajadian += "\n"
        test =False
    for key,value in MAPPING.iteritems():
        if value == sock :
            test=True
    if test==True :
        broadcast(server_socket, sock, datajadian)

```

```

elif (ji[0]==ji[2]):
    login(sock,ji[0],server_socket,addr)
"""elif ji[2]=="login" :
    login(sock,ji[3],server_socket,addr)"""
else:
    sock.send("\rinvalid command\n")

```

```

else:
    if sock in SOCKET_LIST:
        SOCKET_LIST.remove(sock)
        simpan = MAPPING[sock]
        broadcast(server_socket, sock, "\rClient (%s) off\n" %

```

simpan)

except:

```

for key,value in alamat.iteritems() :
    if value == addr:

```



```

        broadcast(server_socket, sock, "\r(%s) off\n"%key )
    continue

server_socket.close()

def daftar(sock1,server_socket):
    data = "\rlist yang online\n"
    for socket in SOCKET_LIST :
        for key,value in MAPPING.iteritems():
            if socket == value :
                data+=(key + " is online \n")
    try:
        sock1.send(data)
    except:
        socket.close()
        if socket in SOCKET_LIST :
            SOCKET_LIST.remove(socket)

def sendto(destination,message):
    data = "\r"+message[0] + " says "
    count = 0
    while count!=len(message):
        if count>3 :
            data+=(message[count]+" ")
            count +=1
    data += "\n"
    socket = MAPPING[destination]
    try :
        socket.send(data)
    except :
        socket.close()
        if socket in SOCKET_LIST :
            SOCKET_LIST.remove(socket)

def login(sock,name,server_socket,addr):
    test =True
    for key,value in MAPPING.iteritems():
        if key == name :
            test=False
    if(test==False):

```

```

        sock.send("\rusername sudah dipake\n")
    else :
        MAPPING[name]=sock
        alamat[name]=addr
        broadcast1(sock,server_socket,name)
def broadcast (server_socket, sock, message):

    for socket in SOCKET_LIST:
        if socket != server_socket and socket != sock :
            try :
                socket.send(message)
            except :
                socket.close()
                if socket in SOCKET_LIST:
                    SOCKET_LIST.remove(socket)
def broadcast1 (sock,server_socket,name):
    broadcast(server_socket,sock,"\r%s sudah terhubung dengan system
\n"%name)

if __name__ == "__main__":

    sys.exit(chat_server())

```