

ISTANBUL TECHNICAL UNIVERSITY
COMPUTER COMMUNICATION
PROJECT-1



Tugay ACAR

150160511

Computer Engineering

Explanation of Project

1. How to run

- I used cmd for Windows.
- Path to your python.exe
- Then write console in order like that :
python C:path\to\Server.py
python C:path\to\Client.py
- (you can write Client.py infinitely)

2. Explanation of coding

```
def __init__(self, Port):  
    # exceptions for creatng of socket  
    try:  
        serverSocket = socket(AF_INET, SOCK_STREAM)  
    except:  
        exit(1)  
  
    try:  
        serverSocket.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)  
    except:  
        exit(1)  
  
    try:  
        serverSocket.bind(('', Port))  
    except:  
        exit(1)  
  
    try:  
        serverSocket.listen(45)  
    except:  
        exit(1)  
  
    print("The server is ready")
```

Figure 1

In the figure 1 : The init part in Server.py try to catch errors. If the server don't has error, prints "The server is ready"

```

client = client()
while True:
    #threading a new thread for every client who is joined
    connectionSocket, addr = serverSocket.accept()
    #add clients to our list
    client.add_client(connectionSocket,addr)
    threading.Thread(target=self.listen_Client, args=(connectionSocket, addr, client)).start()
    time.sleep(1)

```

Figure 2

In the Figure 2: Add our clients to server and thread them. Also, i keep all clients in class.list()

```

def receive():
    #to reveive messages from server
    while True:
        try:
            msg = clientSocket.recv(1024)
            print(msg.decode('utf-8'))
        except OSError: # Possibly client has left the chat.
            break

def send():
    #to send messages to server and then the other clients
    msg = input('')
    timestamp = ' '+str(time.ctime(time.time()))
    msg = msg + timestamp
    clientSocket.send(msg.encode())

    if msg == "{quit}":
        clientSocket.close()
        exit(0)

while True:
    #i wrote an algorithm like that
    if write_flag==1:
        send_thread = Thread(target=send)
        send_thread.start()
        time.sleep(1)
        write_flag=0
        read_flag=1
        continue
    if read_flag==1:
        receive_thread = Thread(target=receive)
        receive_thread.start()
        time.sleep(1)
        read_flag =0
        write_flag = 1
        continue

```

Figure 3

In the Figure 3: Send and Receive messages between clients and server.

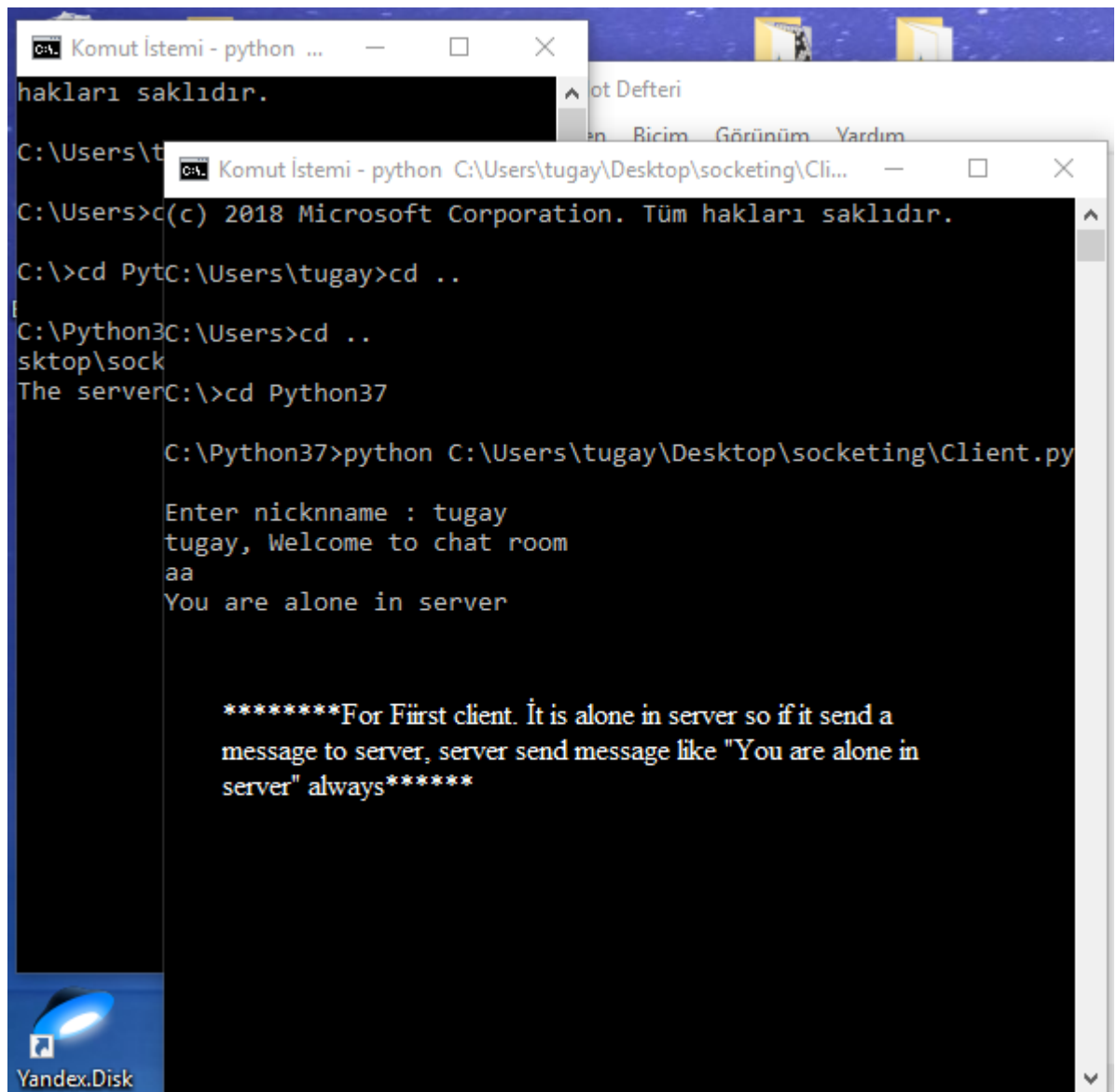
```
#Send messages to all of other clients
def broadcast(msg, clients, conn):
    for i in clients.list_sockets:
        if conn != i:
            i.send(msg.encode())

class Server():
    def listen_Client(self, connectionSocket, addr, cc):
        #first i keep a nickname to understand who send message
        nickname = connectionSocket.recv(1024)
        name[connectionSocket] = nickname.decode('utf-8')
        #send a welcome message to only itself
        welcome_message = nickname.decode('utf-8') + ', Welcome to chat room '
        connectionSocket.send(welcome_message.encode())
        #send joining message to all of other clients
        message = nickname.decode('utf-8') + ', joined chat room '
        broadcast(message, cc, connectionSocket)
    while True:
        #server takes an input
        data = connectionSocket.recv(1024)
        # if data(message) != exit, it sends the data to all of other client
        if data != bytes("exit", "utf8"):
            # we have just one client, the client is alone in server. So i :
            if (len(cc.list_sockets) == 1):
                msg = 'You are alone in server'.encode()
                connectionSocket.send(msg)
            #if we have more clients than 1, sen messages to other clients
            data = name[connectionSocket]+' : '+data.decode('utf-8')
            broadcast(data, cc, connectionSocket)
        else:
            #if message exit. Client exit
            connectionSocket.send(bytes("exited", "utf8"))
            connectionSocket.close()
            cc.list_sockets.remove(connectionSocket)
```

c

Figure 4

In the Figure 4: Server.py takes input from 1 client and send them other clients(broadcast)



```
Komut İstemi - python ...
hakları saklıdır.
C:\Users\tugay\Desktop\socketing\

Komut İstemi - python C:\Users\tugay\Desktop\socketing\Cli...
C:\Users>c(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\>cd PytC:\Users\tugay>cd ..
C:\Python3C:\Users>cd ..
sktop\sock
The serverC:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py
Enter nickname : tugay
tugay, Welcome to chat room
aa
You are alone in server

*****For First client. It is alone in server so if it send a
message to server, server send message like "You are alone in
server" always*****
```

Figure 5

Figure 5: The first client joins the server. The client is alone in server. So server always send message like that “You are alone in server” ,if client send message to server.Takes a nickname and keep it.

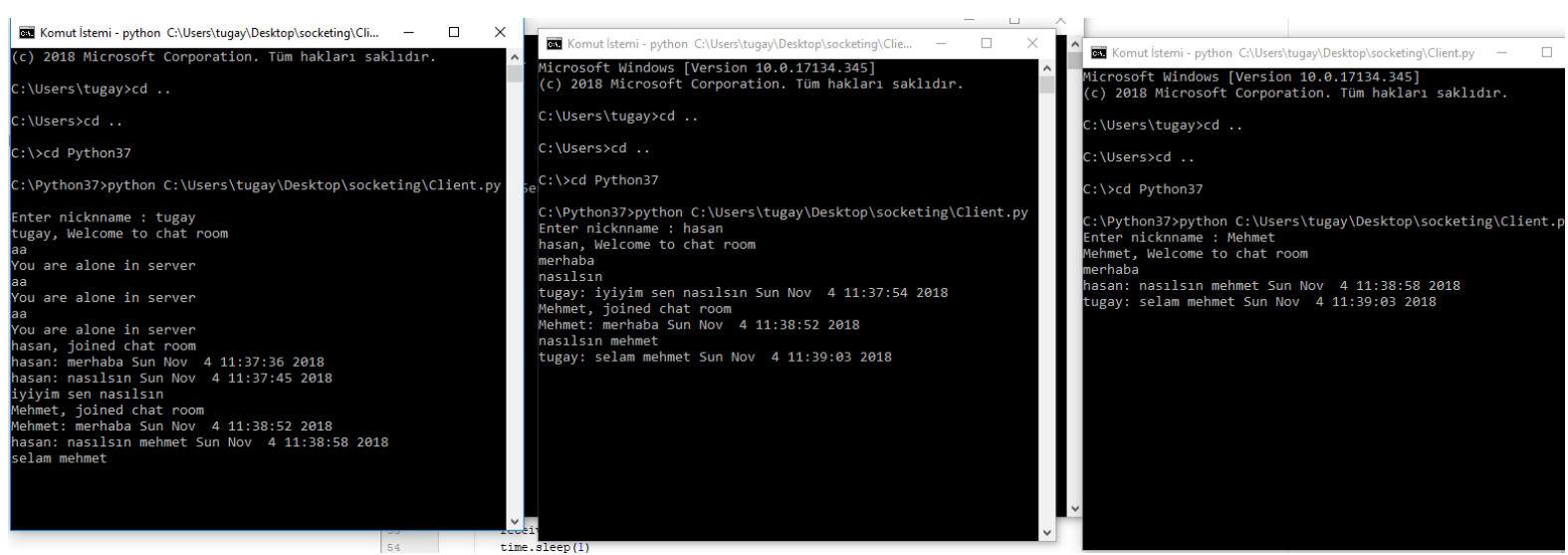
```
Komut İstemi - python C:\Users\tugay\Desktop\socketing\Cli...
(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\Users\tugay>cd ..
C:\Users>cd ..
C:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py server
Enter nickname : tugay
tugay, Welcome to chat room
aa
You are alone in server
aa
You are alone in server
aa
You are alone in server
hasan, joined chat room
hasan: merhaba Sun Nov 4 11:37:36 2018
hasan: nasılsın Sun Nov 4 11:37:45 2018
iyiyim sen nasılsın

Joining message

Microsoft Windows [Version 10.0.17134.345]
(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\Users\tugay>cd ..
C:\Users>cd ..
C:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py
Enter nickname : hasan
hasan, Welcome to chat room
merhaba
nasılsın
tugay: iyiyim sen nasılsın Sun Nov 4 11:37:54 2018
Who is      Message      Date
send th
message
```

Figure 6

Figure 6: The second client joins the server. Firstly takes a nickname then server send client welcome message. After that, Server send joining message of Client 2(hasan) to other clients. If anyone send a message, it will be transmitted to other clients as like that nickname + message + timestamp



```
Komut İstemi - python C:\Users\tugay\Desktop\socketing\Cli...
(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\Users\tugay>cd ..
C:\Users>cd ..
C:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py
Enter nickname : tugay
tugay, Welcome to chat room
aa
You are alone in server
aa
You are alone in server
aa
You are alone in server
hasan, joined chat room
hasan: merhaba Sun Nov  4 11:37:36 2018
hasan: nasılsın Sun Nov  4 11:37:45 2018
iyiyim sen nasılsın
Mehmet, joined chat room
Mehmet: merhaba Sun Nov  4 11:38:52 2018
hasan: nasılsın mehmet Sun Nov  4 11:38:58 2018
selam mehmet
time.sleep(1)
```

```
Komut İstemi - python C:\Users\tugay\Desktop\socketing\Cli...
Microsoft Windows [Version 10.0.17134.345]
(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\Users\tugay>cd ..
C:\Users>cd ..
C:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py
Enter nickname : hasan
hasan, Welcome to chat room
merhaba
nasılsın
tugay: iyiyim sen nasılsın Sun Nov  4 11:37:54 2018
Mehmet, joined chat room
Mehmet: merhaba Sun Nov  4 11:38:52 2018
nasılsın mehmet
tugay: selam mehmet Sun Nov  4 11:39:03 2018
```

```
Komut İstemi - python C:\Users\tugay\Desktop\socketing\Client.py
Microsoft Windows [Version 10.0.17134.345]
(c) 2018 Microsoft Corporation. Tüm hakları saklıdır.
C:\Users\tugay>cd ..
C:\Users>cd ..
C:\>cd Python37
C:\Python37>python C:\Users\tugay\Desktop\socketing\Client.py
Enter nickname : Mehmet
Mehmet, Welcome to chat room
merhaba
hasan: nasılsın mehmet Sun Nov  4 11:38:58 2018
tugay: selam mehmet Sun Nov  4 11:39:03 2018
```

Figure 7

Figure 7: Same procedure with Figure 6. Takes a nickname, server send client welcome message and then send joining message to other clients. If one of clients writes a message, server send them to all of other clients.