WEEK 4: Distributed Systems Labs
Name: Sagnik Chatterjee
Reg: 180905478
Roll No: 61
VI 'B'

Q1. Write a simple Echo server and client in Python.(TCP Echo Server)

simpleServer.py

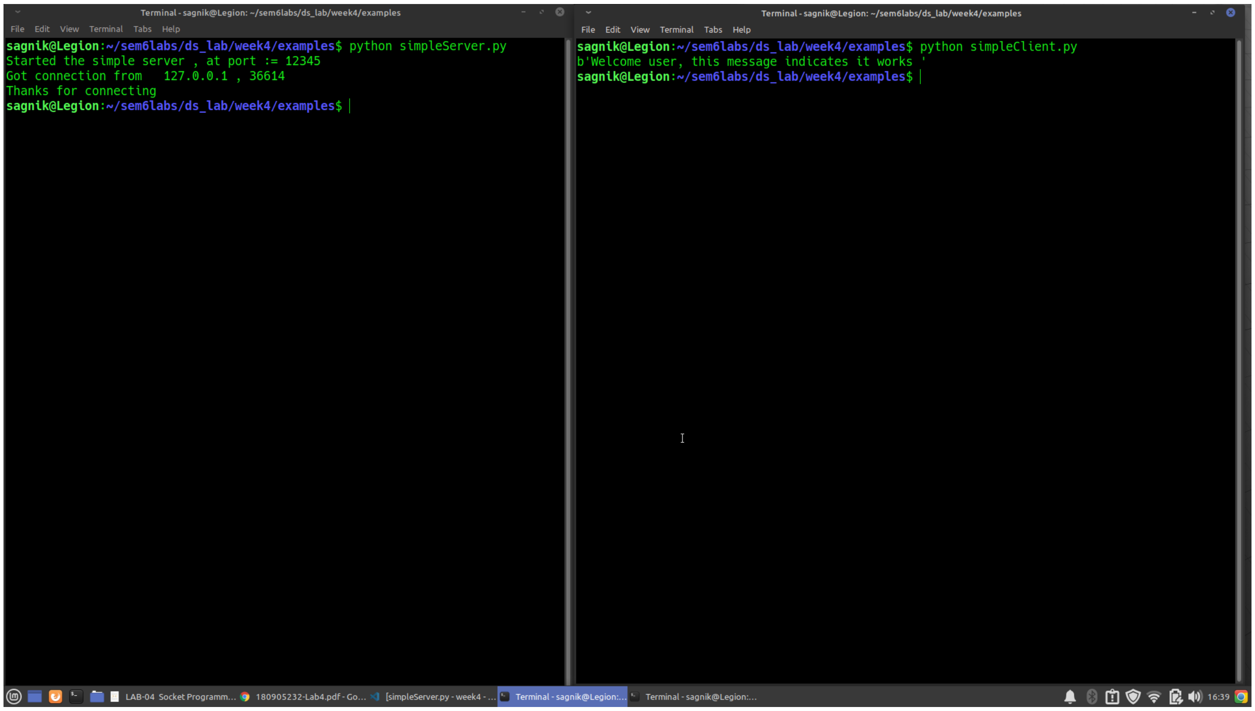
```
# echo server program
import socket

host = socket.gethostname()
port = 12345
s = socket.socket()
s.bind((host, port))
s.listen(5)
print(f"Started the simple server , at port := {port}")
conn, addr = s.accept()
print(f"Got connection from   {addr[0]} , {addr[1]}")
print("Thanks for connecting ")
while True:
   data = conn.recv(1024)
   if not data:
       break
   conn.sendall(data)
conn.close()
```

simpleClient.py

```
import socket

host = socket.gethostname()
port = 12345
s = socket.socket()
```

```
s.connect((host, port))
s.sendall(b'Welcome user, this message indicates it works ')
data = s.recv(1024)
s.close()
print(repr(data))
```

Output:


Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples
File Edit View Terminal Tabs Help
LAB-04 Socket Programm...
180905232-Lab4.pdf - Go...
[simpleServer.py - week4 - ...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:39

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$ python simpleServer.py
Started the simple server , at port := 12345
Got connection from   127.0.0.1 , 36614
Thanks for connecting
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$ python simpleClient.py
b'Welcome user, this message indicates it works '
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$
```

Q2 Write an connectionless echo server in Python using UDP.

connlessServer.py

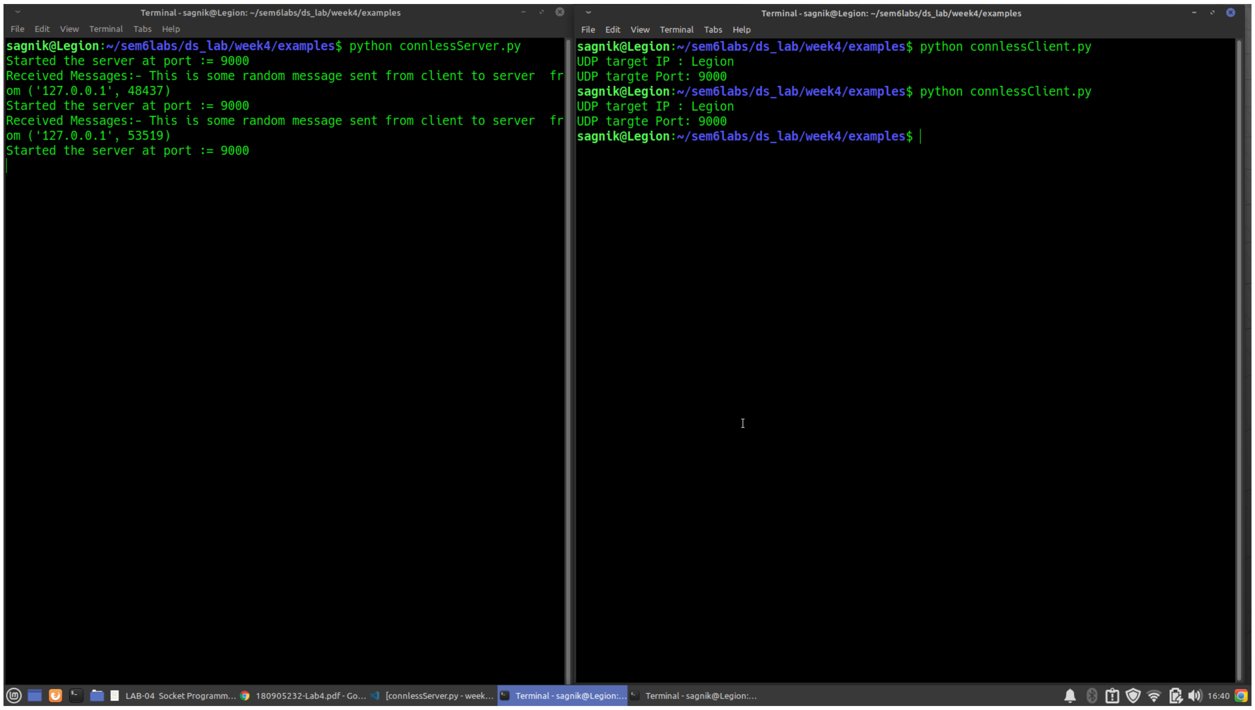
```
import socket

# udp base conenctions
sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

udp_host = socket.gethostname()  #getting the host ip
udp_port = 9000   # this port should be connected

sock.bind((udp_host, udp_port))
```

```
while True:
    # stay connected until client found and received data
    print(f"Started the server at port := {udp_port}")
    data, addr = sock.recvfrom(1024)
    print(f"Received Messages:- {data.decode()} from {addr}")
```

connlessClient.py

```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

udp_host = socket.gethostname()  # getting the host ip

udp_port = 9000  # this port should be connected

msg = "This is some random message sent from client to server "
print(f"UDP target IP : {udp_host}")
print(f"UDP targte Port: {udp_port}")

sock.sendto(msg.encode(), (udp_host, udp_port))
```

Output:


Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples
File Edit View Terminal Tabs Help
LAB-04 Socket Programm...
180905232-Lab4.pdf - Go...
[connlessServer.py - week...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:40

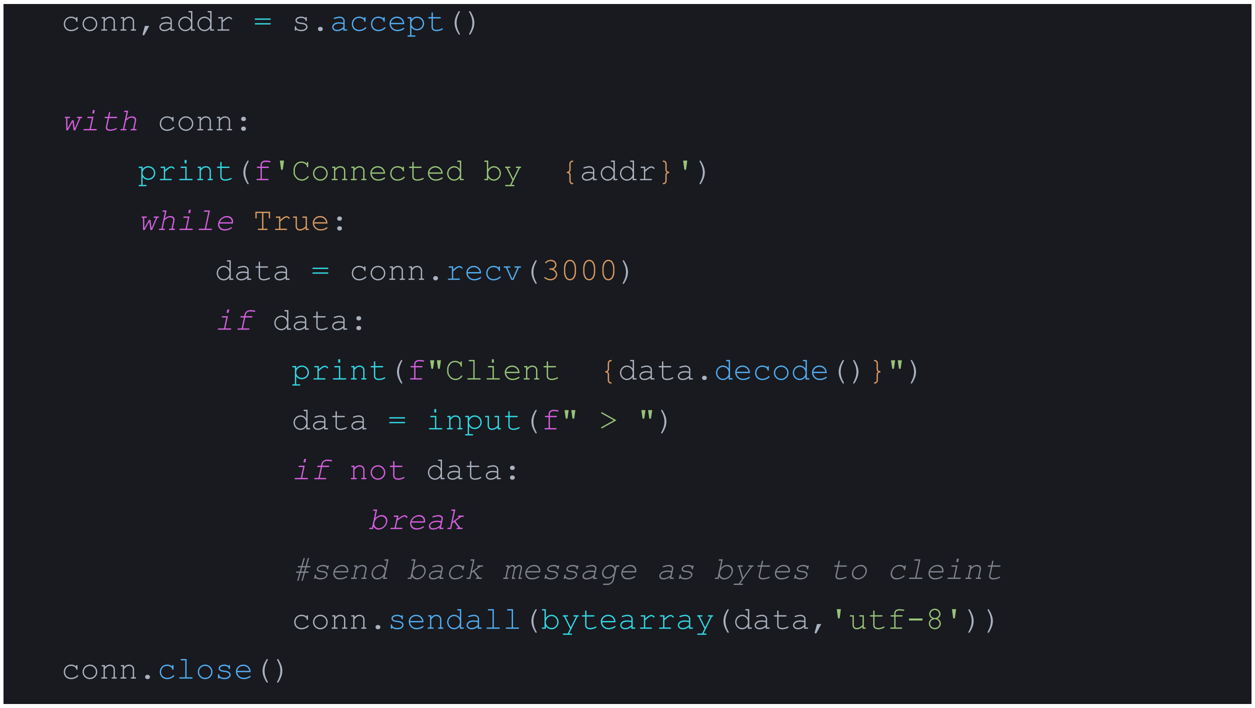
```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$ python connlessServer.py
Started the server at port := 9000
Received Messages:- This is some random message sent from client to server  fr
om ('127.0.0.1', 48437)
Started the server at port := 9000
Received Messages:- This is some random message sent from client to server  fr
om ('127.0.0.1', 53519)
Started the server at port := 9000
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$ python connlessClient.py
UDP target IP : Legion
UDP targte Port: 9000
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$ python connlessClient.py
UDP target IP : Legion
UDP targte Port: 9000
sagnik@Legion:~/sem6labs/ds_lab/week4/examples$
```

Q3 Write a program where client can send a message to the server and the server can receive
the message and send, or echo, it back to the client.

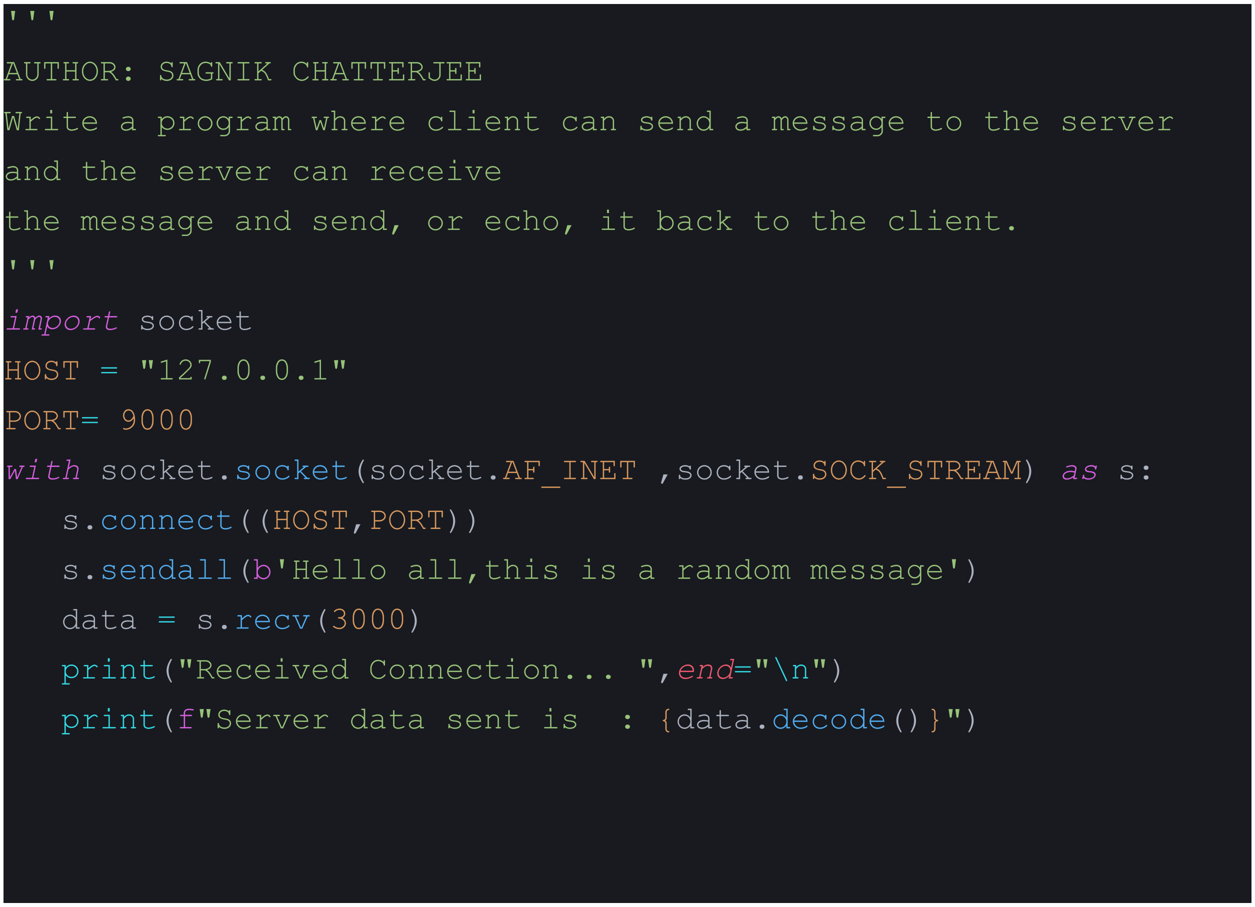
q3server.py

```
'''
AUTHOR: SAGNIK CHATTERJEE
Write a program where client can send a message to the server
and the server can receive
the message and send, or echo, it back to the client.
'''

import socket

HOST="127.0.0.1"
PORT=9000
with socket.socket(socket.AF_INET,socket.SOCK_STREAM) as s:
    s.bind((HOST,PORT))
    s.listen()
```



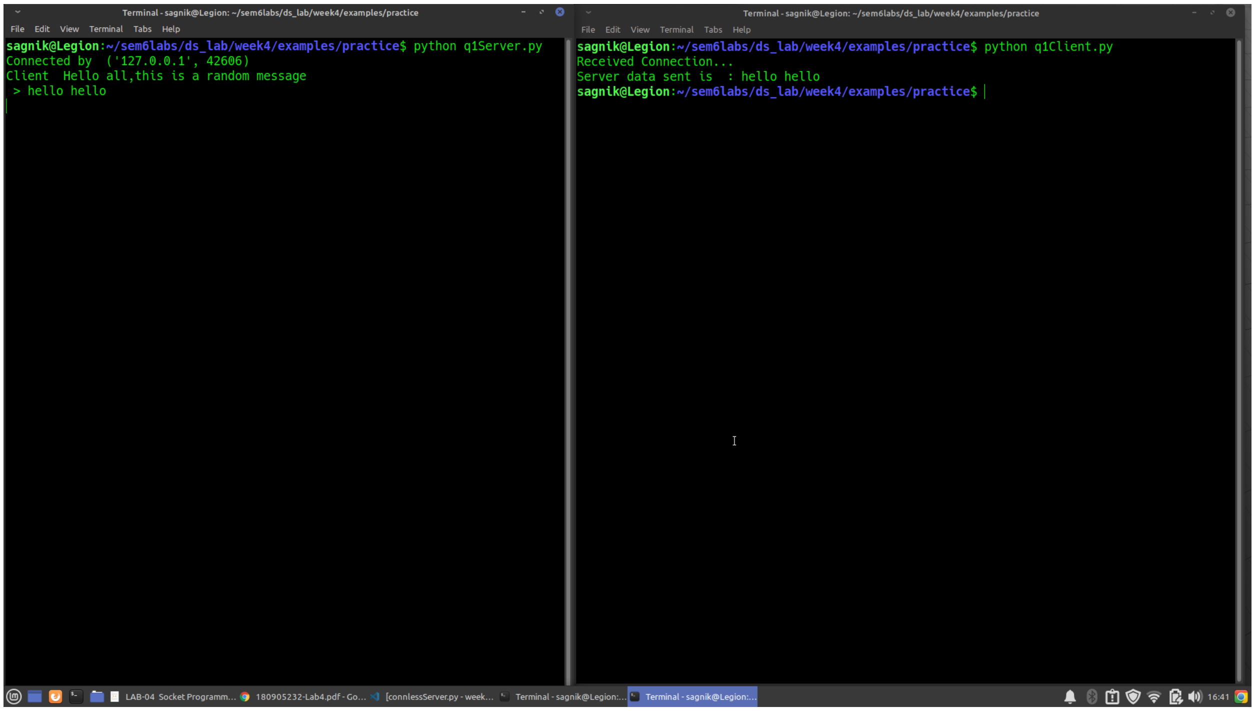
```
    conn,addr = s.accept()

    with conn:
        print(f'Connected by  {addr}')
        while True:
            data = conn.recv(3000)
            if data:
                print(f"Client  {data.decode()}")
                data = input(f" > ")
                if not data:
                    break
                #send back message as bytes to cleint
                conn.sendall(bytearray(data,'utf-8'))
    conn.close()
```

q3client.py



```
'''
AUTHOR: SAGNIK CHATTERJEE
Write a program where client can send a message to the server
and the server can receive
the message and send, or echo, it back to the client.
'''
import socket
HOST = "127.0.0.1"
PORT= 9000
with socket.socket(socket.AF_INET ,socket.SOCK_STREAM) as s:
    s.connect((HOST,PORT))
    s.sendall(b'Hello all,this is a random message')
    data = s.recv(3000)
    print("Received Connection... ",end="\n")
    print(f"Server data sent is  : {data.decode()}")
```

Output:


Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help
LAB-04 Socket Programm...
180905232-Lab4.pdf - Go...
[connlessServer.py - week...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:41

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q1Server.py
Connected by  ('127.0.0.1', 42606)
Client  Hello all,this is a random message
 > hello hello
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q1Client.py
Received Connection...
Server data sent is  : hello hello
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$
```

Q4 Write a program to create TCP time server in Python

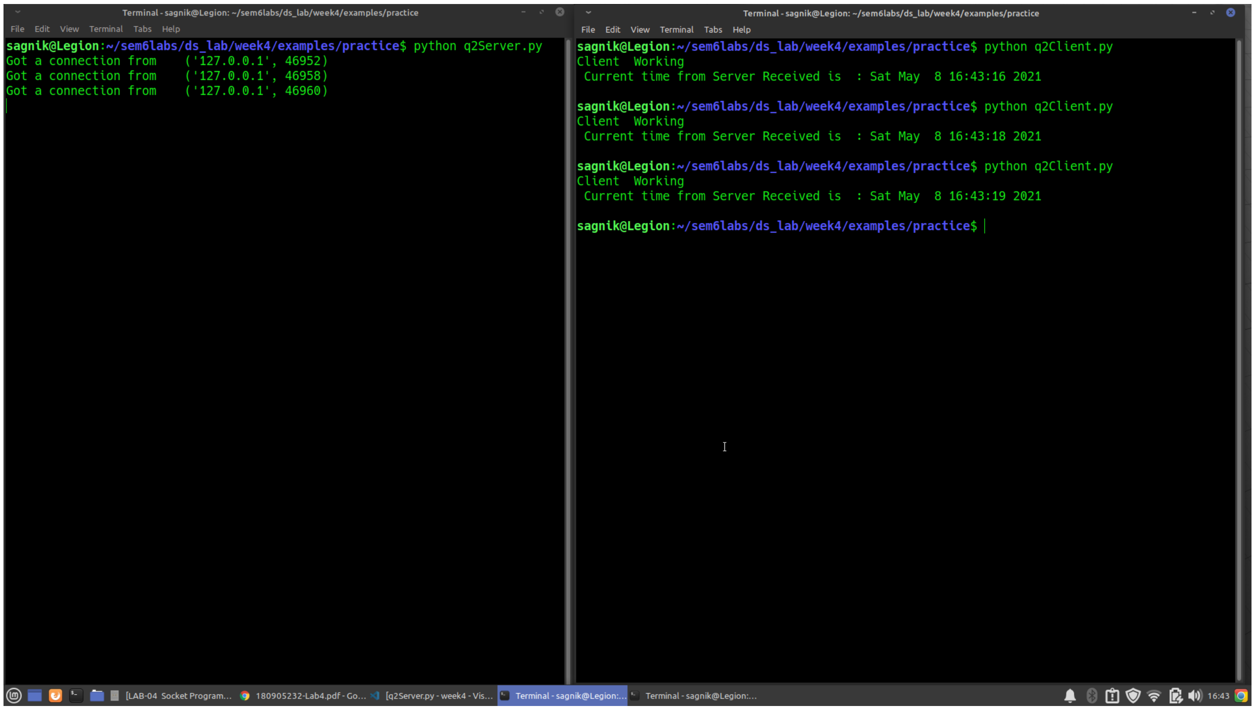
q4server.py

```
'''
Author: Sagnik Chatterjee
Write a program to create TCP time server in Python
'''

import socket
import time

#socket object creation
serversocket = socket.socket(
socket.AF_INET, socket.SOCK_STREAM)

host = socket.gethostname()
port = 9456

# bind to the port
serversocket.bind((host, port))
serversocket.listen(5) #queue 5 requests

while True:
   # establish a connection
   clientsocket,addr = serversocket.accept()
   print(f"Got a connection from    {str(addr)}")
   currentTime = time.ctime(time.time()) + "\r\n"
   clientsocket.send(currentTime.encode('ascii'))
   clientsocket.close()
```

q4client.py

```
"""
Author: Sagnik Chatterjee
Write a program to create TCP time server in Python
"""

import socket

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) #socket
object ,tcp type

host = socket.gethostname() #local macchine name
port = 9456

print("Client  Working   ")
s.connect((host, port)) #connect

tm = s.recv(1024)
print(" Current time from Server Received is  :", tm.decode())
s.close()
```

Output:



Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help
[LAB-04 Socket Program...
180905232-Lab4.pdf - Go...
[q2Server.py - week4 - Vis...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:43

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q2Server.py
Got a connection from    ('127.0.0.1', 46952)
Got a connection from    ('127.0.0.1', 46958)
Got a connection from    ('127.0.0.1', 46960)
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q2Client.py
Client  Working
 Current time from Server Received is  : Sat May  8 16:43:16 2021

sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q2Client.py
Client  Working
 Current time from Server Received is  : Sat May  8 16:43:18 2021

sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q2Client.py
Client  Working
 Current time from Server Received is  : Sat May  8 16:43:19 2021

sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$
```

Q5 Write a TCP chat server in python using socket programming.

q5server.py

```
import socket

HOST = '127.0.0.1'
PORT = 31621

s = socket.socket()
s.bind((HOST, PORT))
s.listen()

print("  Waiting for incoming connections...\n")
conn, addr = s.accept()

print(f"Received connection from {addr[0]} ,{addr[1]}
",end="\n")
s_name = conn.recv(1024)
s_name = s_name.decode()
```

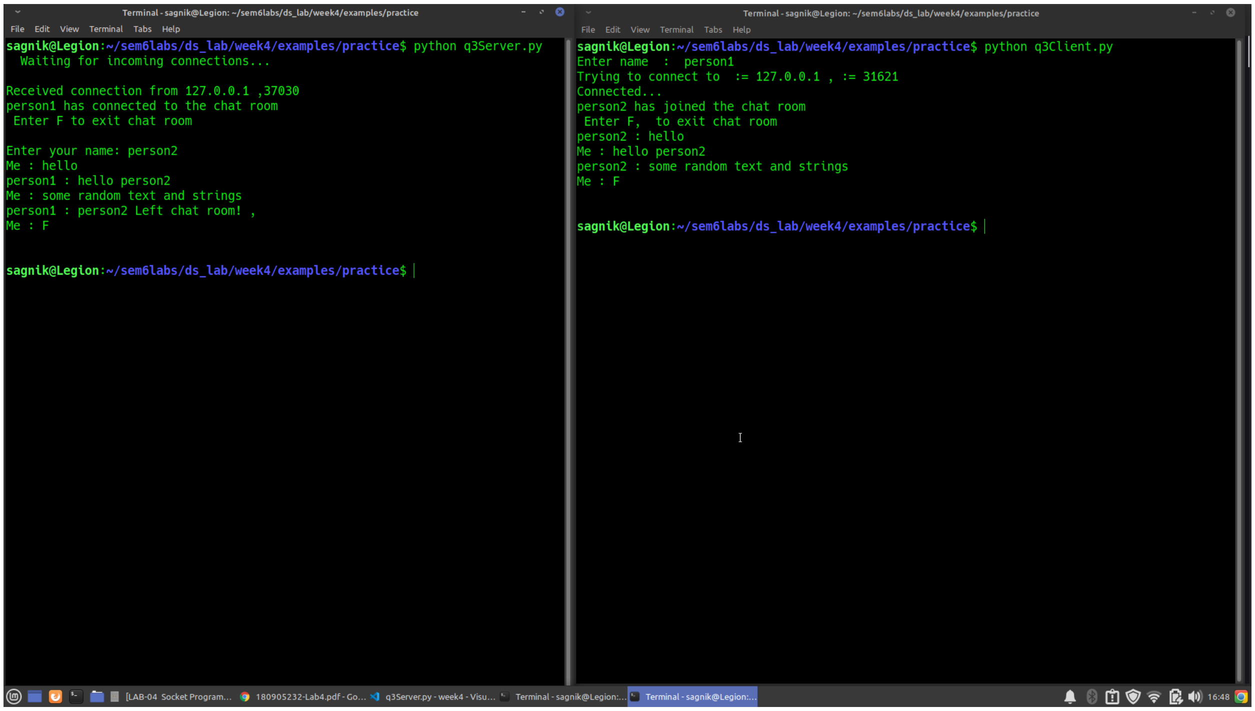
```
print(f"{s_name} has connected to the chat room \n Enter F to
exit chat room\n")
name = input(str("Enter your name: "))
conn.send(name.encode())
while True:
   message = input(str("Me : "))
   if message == "F":
       message = "Left chat room! "
       conn.send(message.encode())
       print("\n")
       break
   conn.send(message.encode())
   message = conn.recv(1024)
   message = message.decode()
   print(f"{s_name} : {message}")
```

q5client.py

```
"""
AUTHOR: SAGNIK CHATTERJEE
Write a TCP chat server in python using socket programming.
"""

import socket

HOST = "127.0.0.1"
PORT = 31621

s = socket.socket()
name = input(str("Enter name  :  "))
print(f"Trying to connect to  := {HOST} , := {PORT}", end="\n")

s.connect((HOST, PORT))
print("Connected...", end="\n")
```

```python
s.send(name.encode())
s_name = s.recv(1024)
s_name = s_name.decode()
print(f"{s_name} has joined the chat room \n Enter F,  to exit chat room", end="\n")

while True:
   message = s.recv(1024)
   message = message.decode()
   print(s_name, ":", message)
   message = input(str("Me : "))
   if message == "F":
       message = f"{s_name} Left chat room! , "
       s.send(message.encode())
       print("\n")
       break
   s.send(message.encode())
```

Output:



Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q3Server.py
  Waiting for incoming connections...

Received connection from 127.0.0.1 ,37030
person1 has connected to the chat room
 Enter F to exit chat room

Enter your name: person2
Me : hello
person1 : hello person2
Me : some random text and strings
person1 : person2 Left chat room! ,
Me : F


sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ 
```

Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q3Client.py
Enter name  :  person1
Trying to connect to  := 127.0.0.1 , := 31621
Connected...
person2 has joined the chat room
 Enter F,  to exit chat room
person2 : hello
Me : hello person2
person2 : some random text and strings
Me : F


sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ 
```

[LAB-04 Socket Program...
180905232-Lab4.pdf - Go...
q3Server.py - week4 - Visu...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:48

Q6 Write a Forking/ Threading (Concurrent Server) in Python using TCP.

q6Server.py

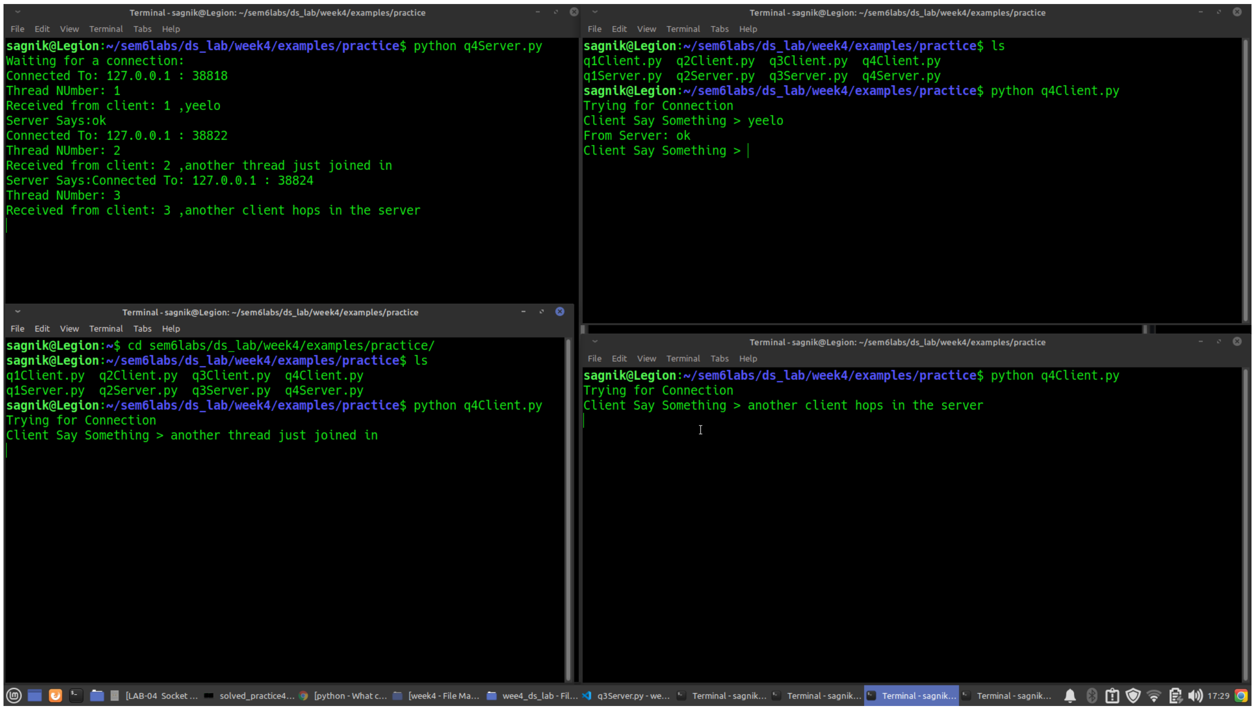
```
import socket
import os
from _thread import *

server_socket = socket.socket()
host = "127.0.0.1"
port = 11596
thread_count = 0
try:
   server_socket.bind((host, port))
except socket.error as e:
   print(str(e))

print("Waiting for a connection:")
server_socket.listen(5)


def thread_client(connection):
   connection.send(str.encode("Welcome to the Server"))
   while True:
       data = connection.recv(2048)#max upto 2048 bytes receive
       print(f"Received from client: {str(thread_count)}
,{data.decode()}")

       inputs = input("Server Says:")
       if not data:
           break
       connection.sendall(inputs.encode())
   connection.close()



while True:
   client, address = server_socket.accept()
```

```
    print(f"Connected To: {address[0]} : {str(address[1])}")
    start_new_thread(thread_client, (client,))
    thread_count += 1
    print(f"Thread NUmber: {str(thread_count)}")
server_socket.close()
```

q6client.py

```
import socket

client_socket= socket.socket()
host='127.0.0.1'
port = 11596

print("Trying for Connection")
try:
    client_socket.connect((host,port))
except socket.error as e:
    print(str(e))

response = client_socket.recv(1024)
while True:
    data_input = input("Client Say Something > ")
    client_socket.send(str.encode(data_input))
    response = client_socket.recv(1024)
    print(f"From Server: {response.decode()}")

client_socket.close()
```

Output:


Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4/examples/practice
File Edit View Terminal Tabs Help

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q4Server.py
Waiting for a connection:
Connected To: 127.0.0.1 : 38818
Thread NUmber: 1
Received from client: 1 ,yeelo
Server Says:ok
Connected To: 127.0.0.1 : 38822
Thread NUmber: 2
Received from client: 2 ,another thread just joined in
Server Says:Connected To: 127.0.0.1 : 38824
Thread NUmber: 3
Received from client: 3 ,another client hops in the server
```

```
sagnik@Legion:~$ cd sem6labs/ds_lab/week4/examples/practice/
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ ls
q1Client.py  q2Client.py  q3Client.py  q4Client.py
q1Server.py  q2Server.py  q3Server.py  q4Server.py
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q4Client.py
Trying for Connection
Client Say Something > another thread just joined in
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ ls
q1Client.py  q2Client.py  q3Client.py  q4Client.py
q1Server.py  q2Server.py  q3Server.py  q4Server.py
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q4Client.py
Trying for Connection
Client Say Something > yeelo
From Server: ok
Client Say Something > 
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4/examples/practice$ python q4Client.py
Trying for Connection
Client Say Something > another client hops in the server
```

Q7 Write a UDP time server to display the current time and day.

q7server.py

```
"""
Author: Sagnik Chatterjee
UDP time server to display the current time and day
"""
import socket
import time

host_udp = "127.0.0.1"
port_udp  = 20001
bufferSize = 1024

currT = time.ctime(time.time())
bytesToSend = str.encode(currT)

# Create a datagram socket
udpServerSocket = socket.socket(family=socket.AF_INET,
type=socket.SOCK_DGRAM)
```

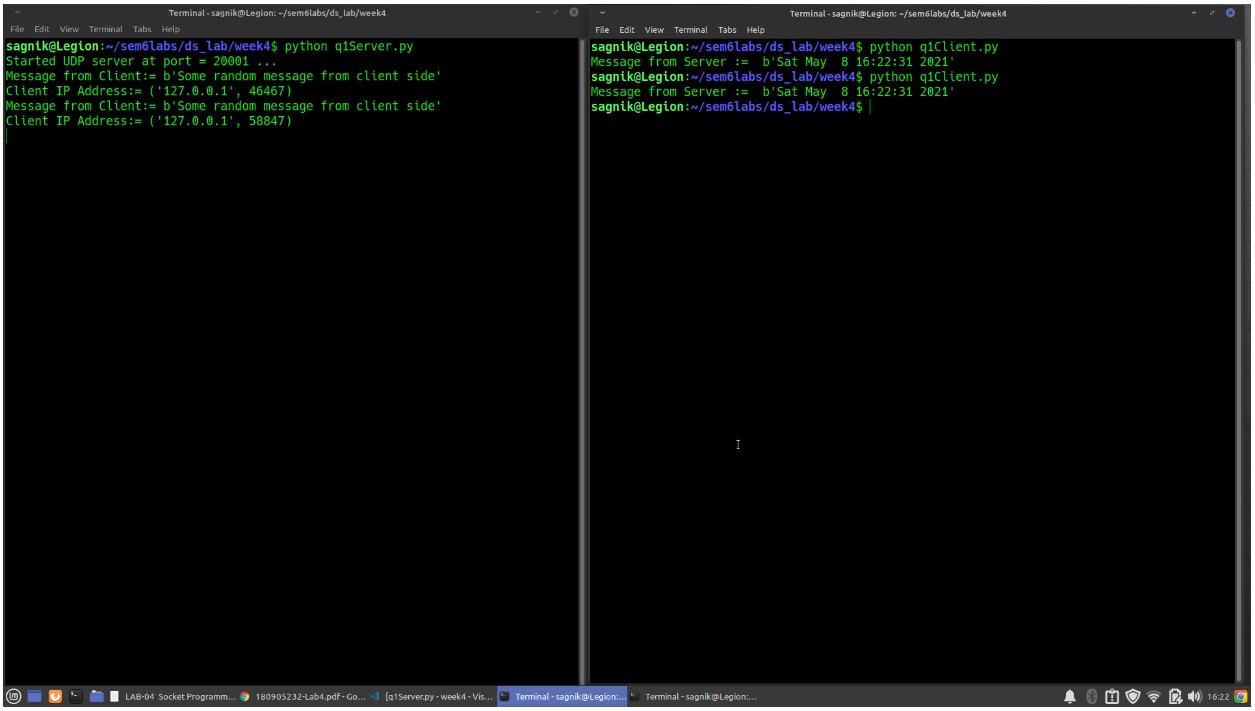
```
# Bind to address and ip
udpServerSocket.bind((host_udp, port_udp))

print(f"Started UDP server at port = {port_udp} ...")
# Listen for incoming datagrams
while True:
   bytesAddressPair = udpServerSocket.recvfrom(bufferSize)
   message = bytesAddressPair[0]
   address = bytesAddressPair[1]
   print(f"Message from Client:= {message}")
   print(f"Client IP Address:= {address}")
   # Sending a reply to client
   udpServerSocket.sendto(bytesToSend, address)
```

q7client.py

```
"""
Author: Sagnik Chatterjee
UDP time server to display the current time and day
"""
import socket

host_udp="127.0.0.1"
port_udp=20001
bufferSize = 1024



msgFromClient = "Some random message from client side"
bytesToSend = str.encode(msgFromClient)
serverAddressPort = (host_udp,port_udp)

# Create a UDP socket at client side
udpClientSocket = socket.socket(family=socket.AF_INET,
type=socket.SOCK_DGRAM)
```

```python
# Send to server using created UDP socket
udpClientSocket.sendto(bytesToSend, serverAddressPort)

msgFromServer = udpClientSocket.recvfrom(bufferSize)
print(f"Message from Server :=  {format(msgFromServer[0])}")
```

Output:


Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4
File Edit View Terminal Tabs Help
LAB-04 Socket Programm...
180905232-Lab4.pdf - Go...
[q1Server.py - week4 - Vis...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
16:22

```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q1Server.py
Started UDP server at port = 20001 ...
Message from Client:= b'Some random message from client side'
Client IP Address:= ('127.0.0.1', 46467)
Message from Client:= b'Some random message from client side'
Client IP Address:= ('127.0.0.1', 58847)
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q1Client.py
Message from Server :=  b'Sat May  8 16:22:31 2021'
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q1Client.py
Message from Server :=  b'Sat May  8 16:22:31 2021'
sagnik@Legion:~/sem6labs/ds_lab/week4$ 
```

Q8 Write a UDP simple chat program for message send and receive.

q8server.py

```python
"""
AUTHOR: SAGNIK CHATTERJEE
Write a UDP simple chat program for message send and receive.

"""

import socket
```

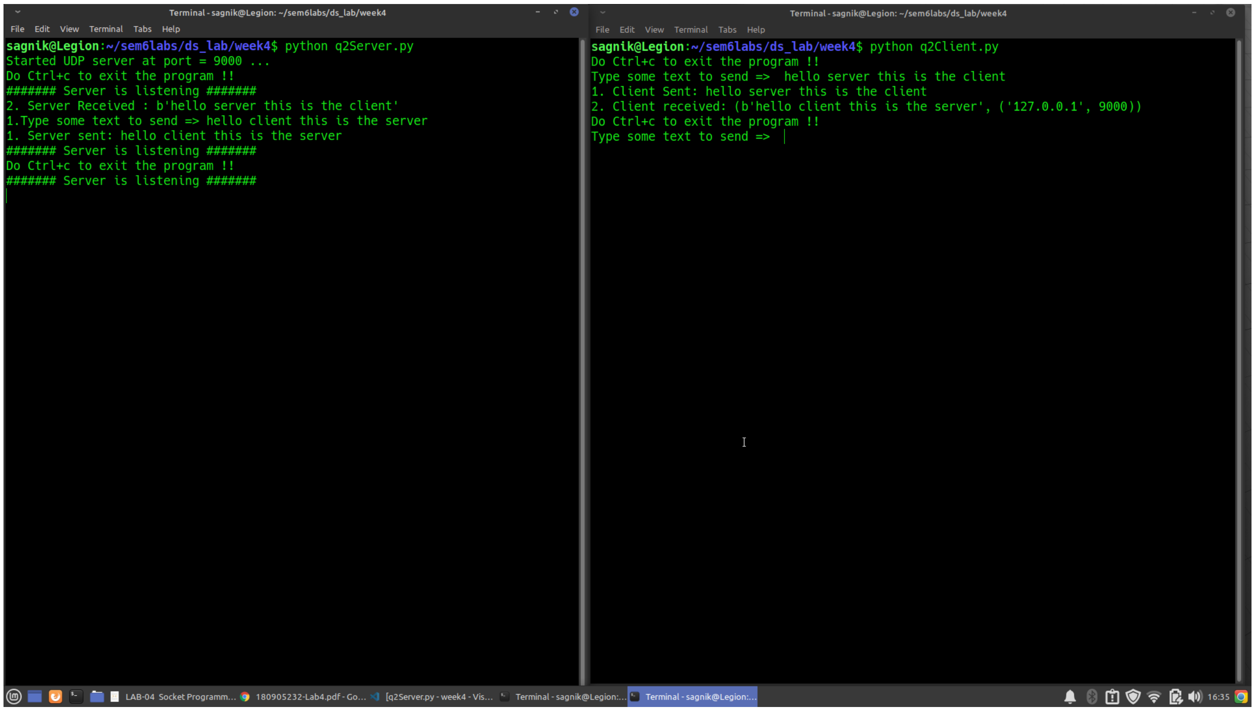
```
host_udp = "127.0.0.1"
port_udp = 9000
bufferSize = 1024

udpServerSocket = socket.socket(family=socket.AF_INET,
type=socket.SOCK_DGRAM)
udpServerSocket.bind((host_udp,port_udp))

print(f"Started UDP server at port = {port_udp} ...")
# Listen for incoming datagrams
while True:
   print("Do Ctrl+c to exit the program !!")
   print("####### Server is listening #######")
   bytesAddressPair = udpServerSocket.recvfrom(bufferSize)
   message = bytesAddressPair[0]
   address = bytesAddressPair[1]
   print(f"2. Server Received : {message}")
   input_value = input("1.Type some text to send => ")
   bytesToSend = str.encode(input_value)
   udpServerSocket.sendto(bytesToSend, address)
   print(f"1. Server sent: {input_value}")
   print("####### Server is listening #######")
```

q8client.py

```
"""
AUTHOR: SAGNIK CHATTERJEE
Write a UDP simple chat program for message send and receive.

"""

import socket

# address is a tuple of the port and the host
```

```python
address = ("127.0.0.1", 9000)
bufferSize = 1024

udpClientSocket = socket.socket(family=socket.AF_INET,
type=socket.SOCK_DGRAM)
while True:
    print("Do Ctrl+c to exit the program !!")
    input_value = input("Type some text to send =>  ")
    print(f"1. Client Sent: {input_value} ")
    bytesToSend = str.encode(input_value)
    udpClientSocket.sendto(bytesToSend, address)
    msgFromServer = udpClientSocket.recvfrom(bufferSize)
    print(f"2. Client received: {msgFromServer}")
```

Output:



```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q2Server.py
Started UDP server at port = 9000 ...
Do Ctrl+c to exit the program !!
####### Server is listening #######
2. Server Received : b'hello server this is the client'
1.Type some text to send => hello client this is the server
1. Server sent: hello client this is the server
####### Server is listening #######
Do Ctrl+c to exit the program !!
####### Server is listening #######
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q2Client.py
Do Ctrl+c to exit the program !!
Type some text to send =>  hello server this is the client
1. Client Sent: hello server this is the client
2. Client received: (b'hello client this is the server', ('127.0.0.1', 9000))
Do Ctrl+c to exit the program !!
Type some text to send =>  
```

Q9. Write a TCP/UDP peer to peer chat system between two different machines.

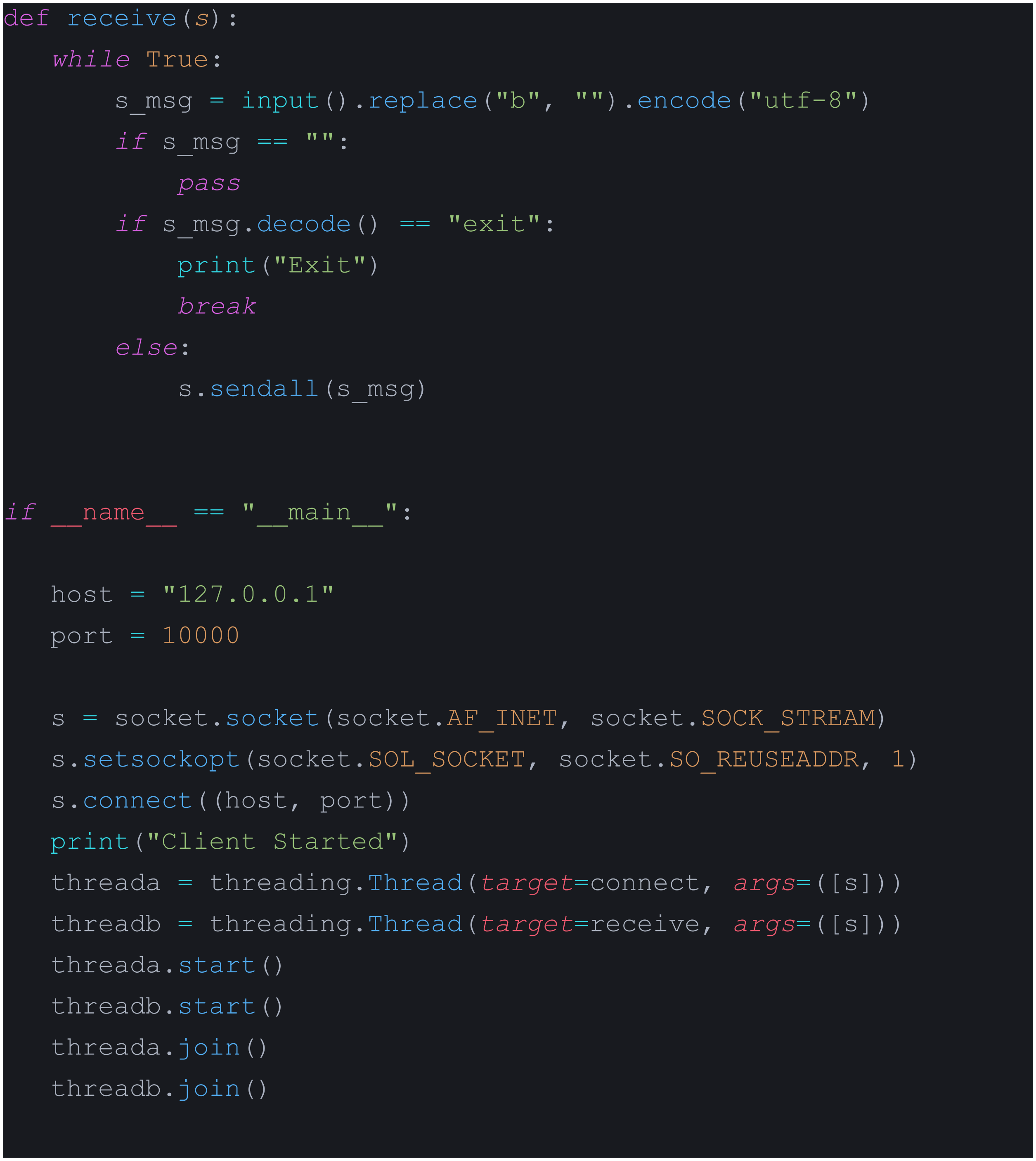
q9server.py

```python
"""
```

```
AUTHOR: SAGNIK CHATTERJEE

Write a TCP/UDP peer to peer chat system between two different
machines.

"""

import sys
import socket
import threading


def connect(conn):
    while True:
        received = conn.recv(1024)
        if received == " ":
            pass
        else:
            print(received.decode())


def sendMsg(conn):
    while True:
        send_msg = input().replace("b", "").encode()
        if send_msg == " ":
            pass
        else:
            conn.sendall(send_msg)


if __name__ == "__main__":
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    s.bind(("", 10000))
    s.listen()
    (conn, addr) = s.accept()
    print("Server Started...")
    threada = threading.Thread(target=connect, args=([conn]))
    threadb = threading.Thread(target=sendMsg, args=([conn]))
    threada.start()
    threadb.start()
    threada.join()
    threadb.join()
```

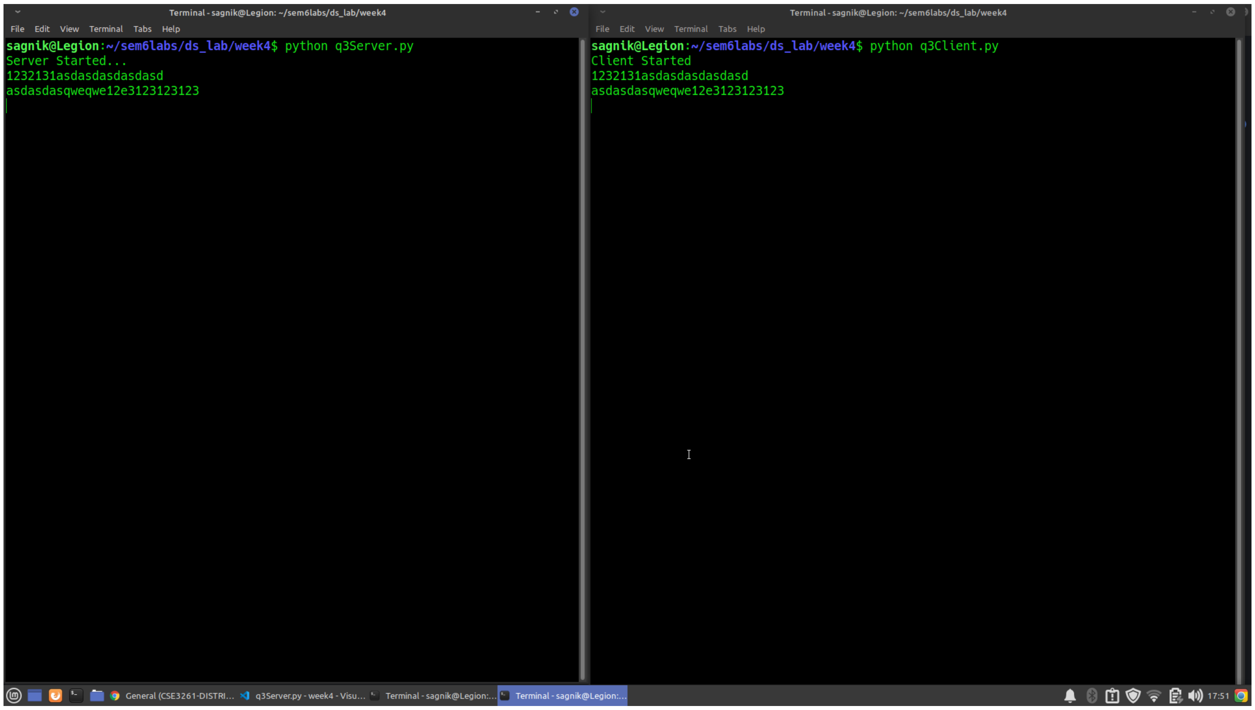
Q9client.py

```
"""
AUTHOR: SAGNIK CHATTERJEE

Write a TCP/UDP peer to peer chat system between two different
machines.

"""
import sys
import socket
import threading


def connect(s):
    while True:
        r_msg = s.recv(1024)
        if not r_msg:
            break
        if r_msg == "":
            pass
        else:
            print(r_msg.decode())
```



```
def receive(s):
    while True:
        s_msg = input().replace("b", "").encode("utf-8")
        if s_msg == "":
            pass
        if s_msg.decode() == "exit":
            print("Exit")
            break
        else:
            s.sendall(s_msg)


if __name__ == "__main__":

    host = "127.0.0.1"
    port = 10000

    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    s.connect((host, port))
    print("Client Started")
    threada = threading.Thread(target=connect, args=([s]))
    threadb = threading.Thread(target=receive, args=([s]))
    threada.start()
    threadb.start()
    threada.join()
    threadb.join()
```

Output:



Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4
File Edit View Terminal Tabs Help
Terminal - sagnik@Legion: ~/sem6labs/ds_lab/week4
File Edit View Terminal Tabs Help
General (CSE3261-DISTRI...
q3Server.py - week4 - Visu...
Terminal - sagnik@Legion:...
Terminal - sagnik@Legion:...
17:51

```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q3Server.py
Server Started...
1232131asdasdasdasdasd
asdasdasqweqwe12e3123123123
```

```
sagnik@Legion:~/sem6labs/ds_lab/week4$ python q3Client.py
Client Started
1232131asdasdasdasdasd
asdasdasqweqwe12e3123123123
```