Name: Ankita Ghosh

Reg no: 180905354

Sec: A Roll no: 41

DISTRIBUTED SYSTEMS LAB: WEEK 4

Question: UPI Connectionless Server and Client

Program:

**#server side**

import socket

sock = socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

udp\_host = socket.gethostname()

udp\_port = 12345 # specified port to connect

sock.bind((udp\_host, udp\_port))

while True:

print ("Waiting for client...")

data,addr = sock.recvfrom(1024)

print ("Received Messages:",data.decode()," from",addr)

**#client side**

import socket

sock = socket.socket(socket.AF\_INET,socket.SOCK\_DGRAM)

udp\_host = socket.gethostname()

udp\_port = 12345 # specified port to connect

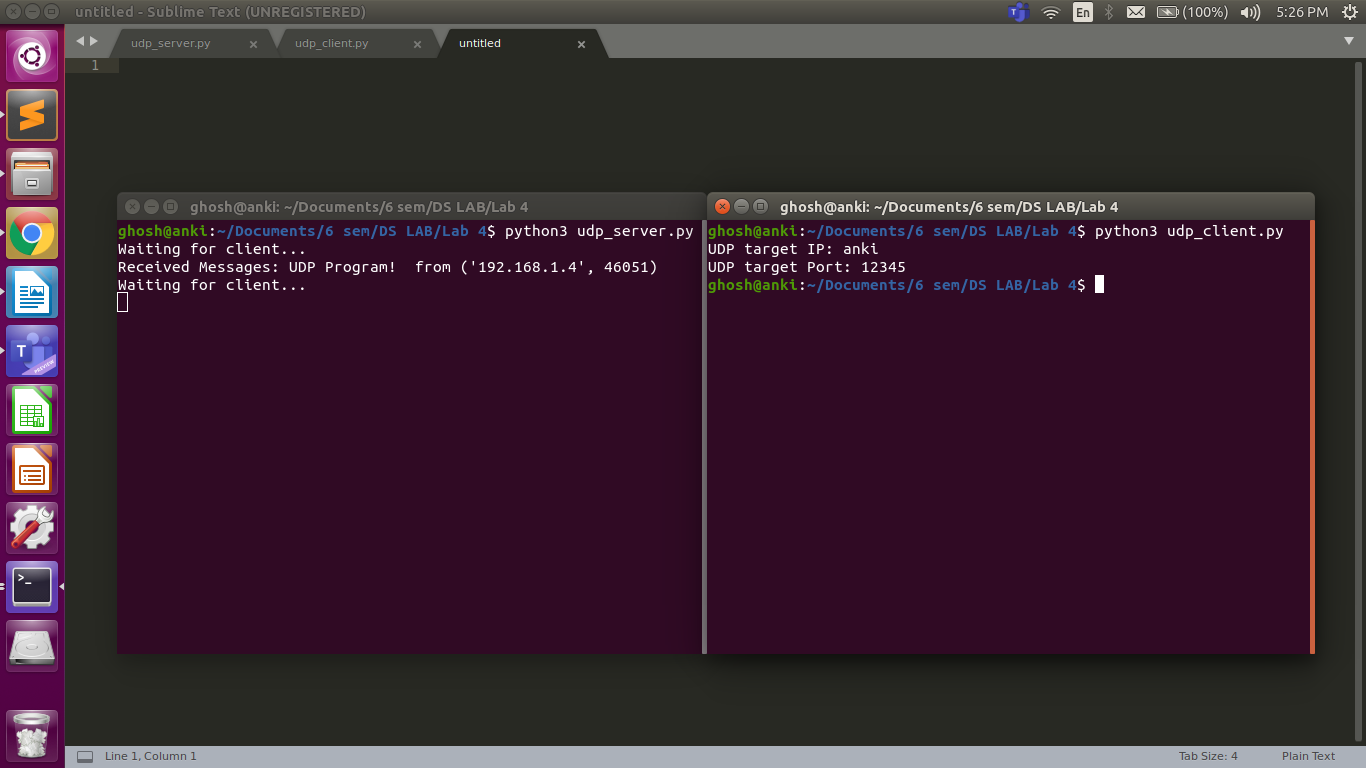
msg = "UDP Program!"

print ("UDP target IP:", udp\_host)

print ("UDP target Port:", udp\_port)

sock.sendto(msg.encode(),(udp\_host,udp\_port))

Output:



Question 1A

Program:

**#Server Side**

import socket

HOST = '127.0.0.1'

PORT = 2053

s = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.bind((HOST,PORT))

s.listen()

conn, addr = s.accept()

with conn:

print('Connected by: ',addr)

while True:

data = conn.recv(1024)

if data:

print("Client: ",data.decode())

data = input("Enter msg:")

if not data:

break;

conn.sendall(bytearray(data,'utf-8'));

conn.close()

**#Client Side**

import socket

HOST = '127.0.0.1'

PORT = 2053

s = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

s.connect((HOST,PORT))

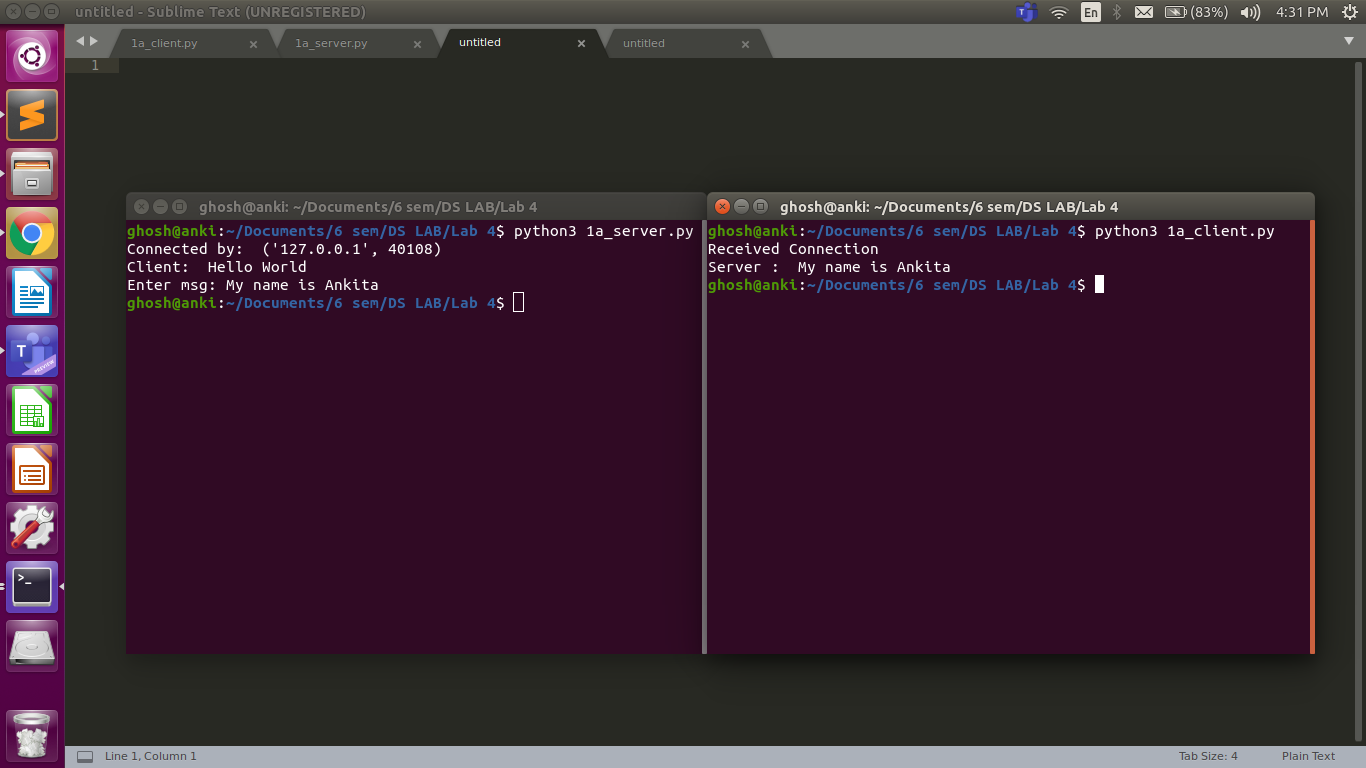
s.sendall(b'Hello World')

data = s.recv(1024)

print("Received Connection")

print("Server :", data.decode())

Output:



Question 2A

Program

**#Server Side**

import time

import socket

servsock = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

host = socket.gethostname()

port = 9991

servsock.bind((host,port))

servsock.listen(5)

while True:

clientsock,addr = servsock.accept()

print("Connected to :",str(addr))

currT = time.ctime(time.time()) + "\r\n"

clientsock.send(currT.encode())

clientsock.close()

**#Client Side**

import socket

s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

host = socket.gethostname()

port = 9991

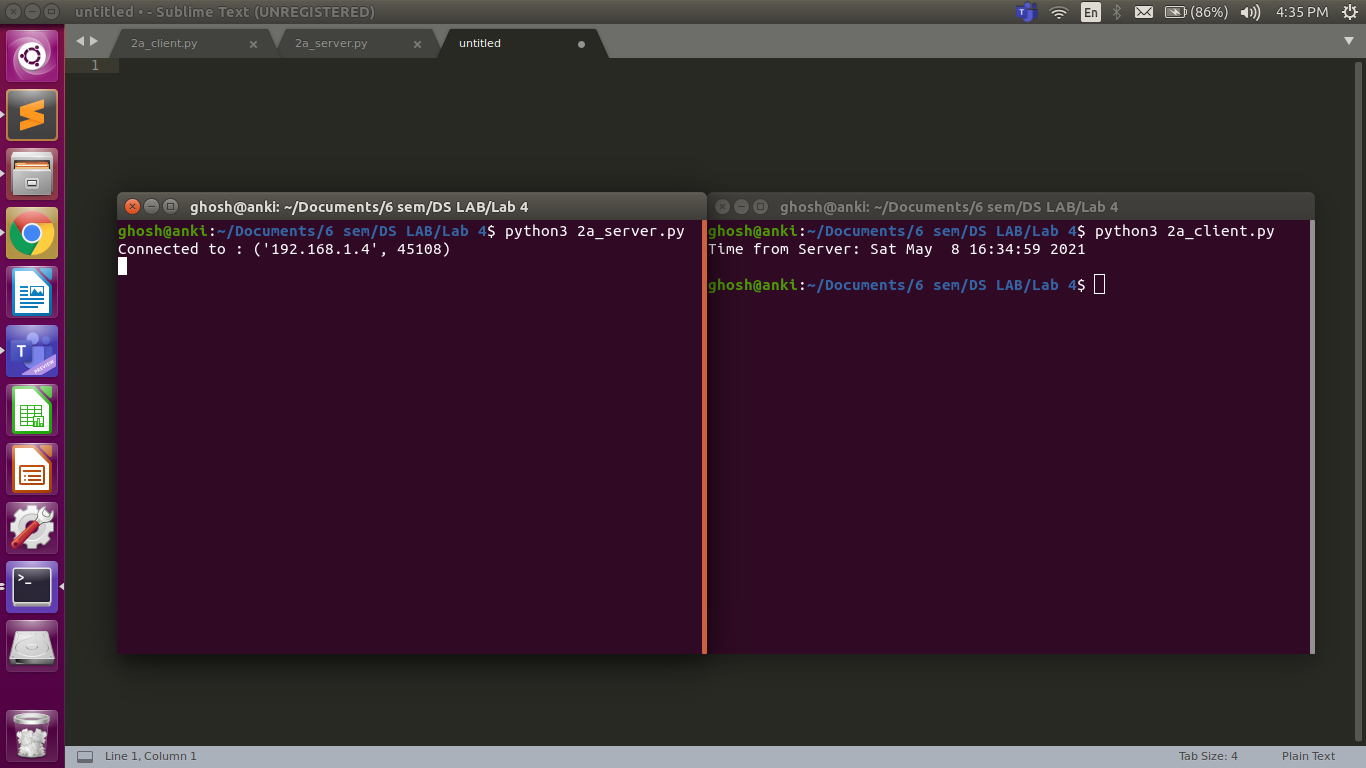
s.connect((host,port))

tm = s.recv(1024)

print("Time from Server:",tm.decode())

s.close()

Output



Question 3A

Program

**#Server Chat:**

import socket

HOST = '127.0.0.1' # Standard loopback interface address (localhost)

PORT = 31621 # Port to listen on (non-privileged ports are > 1023)

s = socket.socket()

s.bind((HOST, PORT))

s.listen()

print("\nWaiting for incoming connections...\n")

conn, addr = s.accept()

print("Received connection from ", addr[0], "(", addr[1], ")\n")

s\_name = conn.recv(1024)

s\_name = s\_name.decode()

print(s\_name, "has connected to the chat room\nEnter [e] to exit chat room\n")

name = input(str("Enter your name: "))

conn.send(name.encode())

while True:

message = input(str("Me : "))

if message == "[e]":

message = "Left chat room!"

conn.send(message.encode())

print("\n")

break

conn.send(message.encode())

message = conn.recv(1024)

message = message.decode()

print(s\_name, ":", message)

**#Client Chat**

import socket

HOST = '127.0.0.1' # Standard loopback interface address (localhost)

PORT = 31621 # Port to listen on (non-privileged ports are > 1023)

s = socket.socket()

name = input(str("\nEnter your name: "))

print("\nTrying to connect to ", HOST, "(", PORT, ")\n")

s.connect((HOST, PORT))

print("Connected...\n")

s.send(name.encode())

s\_name = s.recv(1024)

s\_name = s\_name.decode()

print(s\_name, "has joined the chat room\nEnter [e] to exit chat room\n")

while True:

message = s.recv(1024)

message = message.decode()

print(s\_name, ":", message)

message = input(str("Me : "))

if message == "[e]":

message = "Left chat room!"

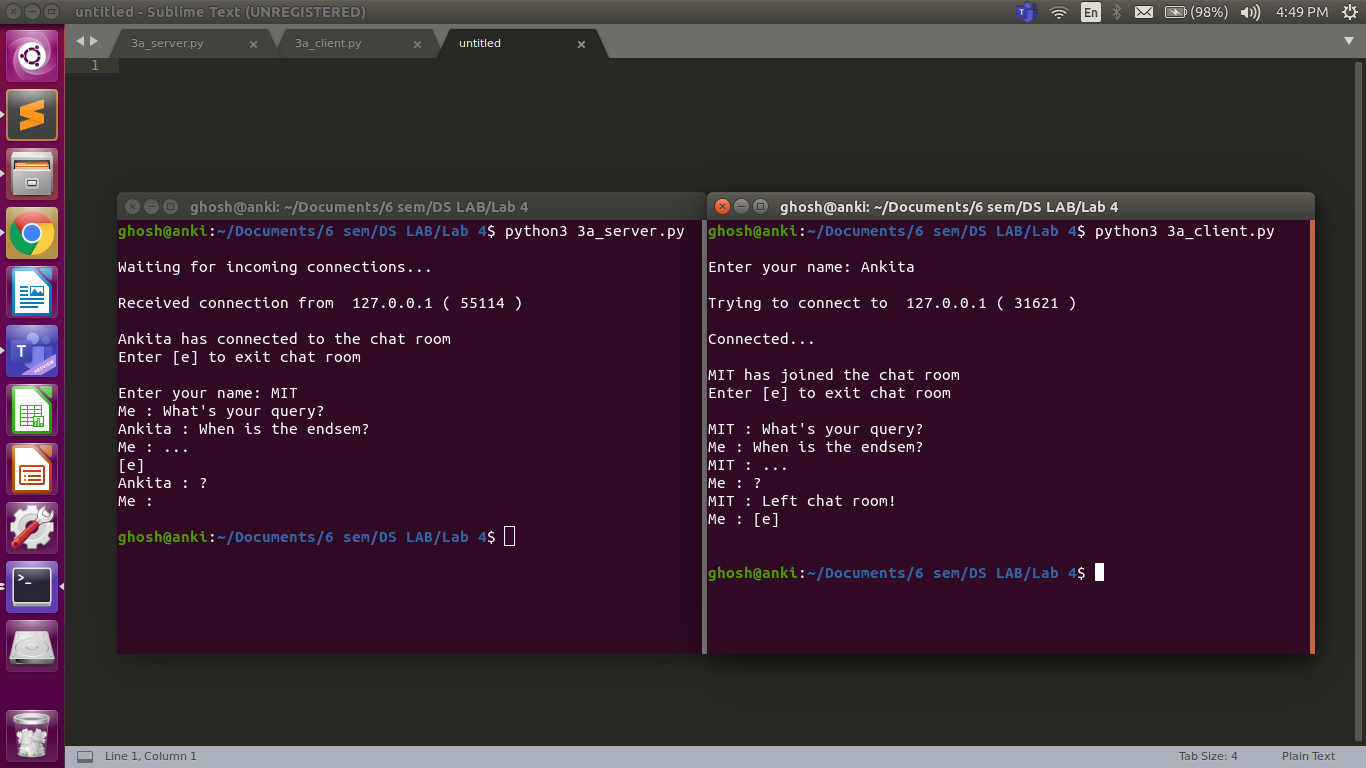
s.send(message.encode())

print("\n")

break

s.send(message.encode())

Output



Question 4

Program

**#Server Side**

import socket

import os

from \_thread import \*

ServerSocket = socket.socket()

host = '127.0.0.1'

port = 11596

ThreadCount = 0

try:

ServerSocket.bind((host, port))

except socket.error as e:

print(str(e))

print('Waitiing for a Connection..')

ServerSocket.listen(5)

def threaded\_client(connection):

connection.send(str.encode('Welcome to the Server'))

while True:

data = connection.recv(2048)

print('Received from client :' + str(ThreadCount) +data.decode())

Inputs = input('Server Says: ')

if not data:

break

connection.sendall(Inputs.encode())

connection.close()

while True:

Client, address = ServerSocket.accept()

print('Connected to: ' + address[0] + ':' + str(address[1]))

start\_new\_thread(threaded\_client, (Client, ))

ThreadCount += 1

print('Thread Number: ' + str(ThreadCount))

ServerSocket.close()

**#client side**

import socket

ClientSocket = socket.socket()

host = '127.0.0.1'

port = 11596

print('Waiting for connection')

try:

ClientSocket.connect((host, port))

except socket.error as e:

print(str(e))

Response = ClientSocket.recv(1024)

while True:

Input = input('Client Say Something: ')

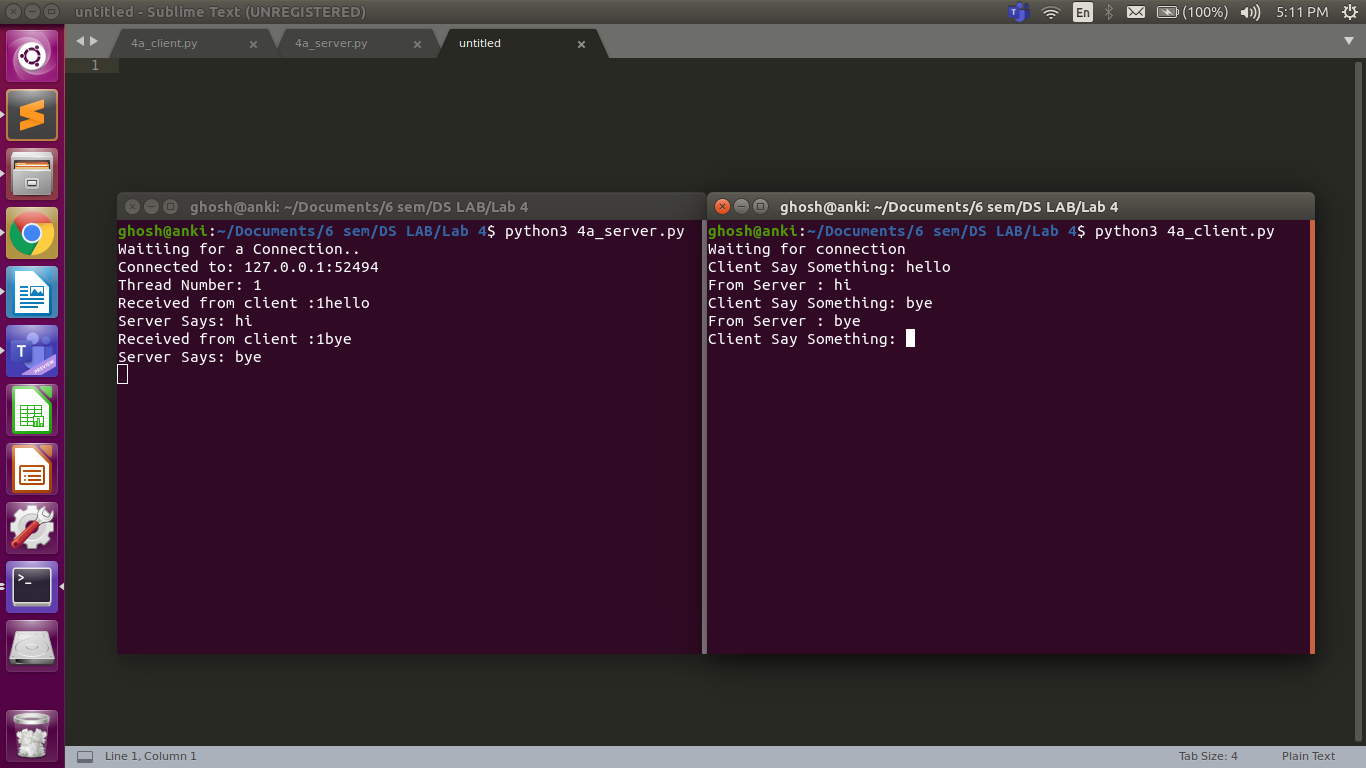
ClientSocket.send(str.encode(Input))

Response = ClientSocket.recv(1024)

print('From Server : ' + Response.decode())

ClientSocket.close()

Output



Question exercise 1

Program

**#Server Side**

import socket

import time

localIP = "127.0.0.1"

localPort = 20001

bufferSize = 1024

currT = time.ctime(time.time()) + "\r\n"

bytesToSend = str.encode(currT)

UDPServerSocket = socket.socket(family=socket.AF\_INET, type=socket.SOCK\_DGRAM)

UDPServerSocket.bind((localIP, localPort))

print("UDP server up and listening")

while(True):

bytesAddressPair = UDPServerSocket.recvfrom(bufferSize)

message = bytesAddressPair[0]

address = bytesAddressPair[1]

clientMsg = "Message from Client:{}".format(message)

clientIP = "Client IP Address:{}".format(address)

print(clientMsg)

print(clientIP)

UDPServerSocket.sendto(bytesToSend, address)

**#Client Side**

import socket

msgFromClient = "Hello UDP Server"

bytesToSend = str.encode(msgFromClient)

serverAddressPort = ("127.0.0.1", 20001)

bufferSize = 1024

UDPClientSocket = socket.socket(family=socket.AF\_INET, type=socket.SOCK\_DGRAM)

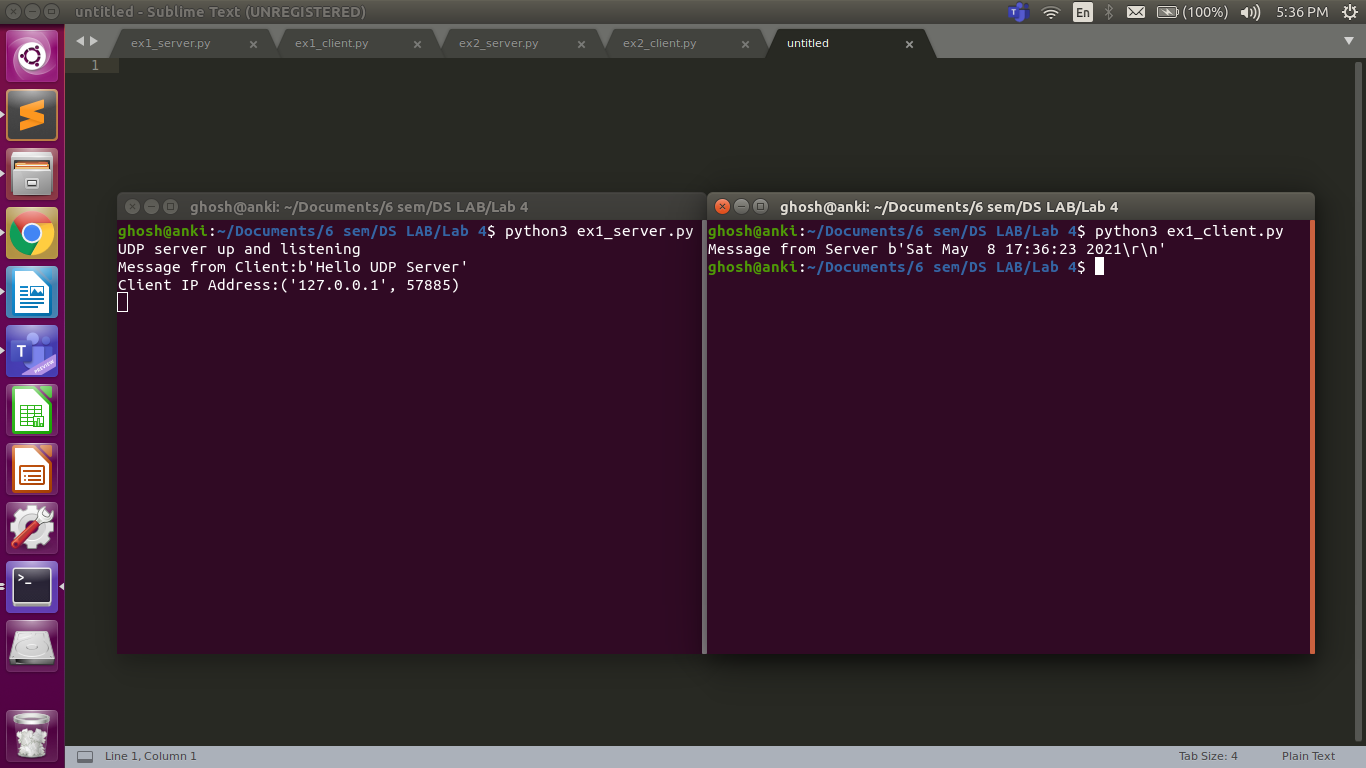
UDPClientSocket.sendto(bytesToSend, serverAddressPort)

msgFromServer = UDPClientSocket.recvfrom(bufferSize)

msg = "Message from Server {}".format(msgFromServer[0])

print(msg)

Output



Question exercise 2

Program

**#Server side**

import socket

localIP = "127.0.0.1"

localPort = 20001

bufferSize = 1024

UDPServerSocket = socket.socket(family=socket.AF\_INET, type=socket.SOCK\_DGRAM)

UDPServerSocket.bind((localIP, localPort))

print("UDP server up and listening")

while(True):

bytesAddressPair = UDPServerSocket.recvfrom(bufferSize)

message = bytesAddressPair[0]

address = bytesAddressPair[1]

clientMsg = "Message from Client:{}".format(message)

print(clientMsg)

Input = input('Server Say Something: ')

bytesToSend = str.encode(Input)

UDPServerSocket.sendto(bytesToSend,address)

**#Client side**

import socket

serverAddressPort = ("127.0.0.1", 20001)

bufferSize = 1024

UDPClientSocket = socket.socket(family=socket.AF\_INET, type=socket.SOCK\_DGRAM)

while True:

Input = input('Client Say Something: ')

bytesToSend = str.encode(Input)

UDPClientSocket.sendto(bytesToSend, serverAddressPort)

msgFromServer = UDPClientSocket.recvfrom(bufferSize)

msg = "Message from Server {}".format(msgFromServer[0])

print(msg)

Output

