

Garv Ghai
180905232
DS Lab - 4
CSE B – 32

1A]

#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())
```

#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()
```

O/P:

```
180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~$ cd Desktop/Lab4
180905232@project-lab:~/Desktop/Lab4$ python3 q1Accli.py
Received Connection
Server : Hello
180905232@project-lab:~/Desktop/Lab4$

180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~$ cd Desktop/Lab4
180905232@project-lab:~/Desktop/Lab4$ python3 q1Aserv.py
Connected by: ('127.0.0.1', 41332)
Client: Hello World
Enter msg:Hello
180905232@project-lab:~/Desktop/Lab4$
```

2A]

#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()
```

#Server Side

```
import time
import socket
```

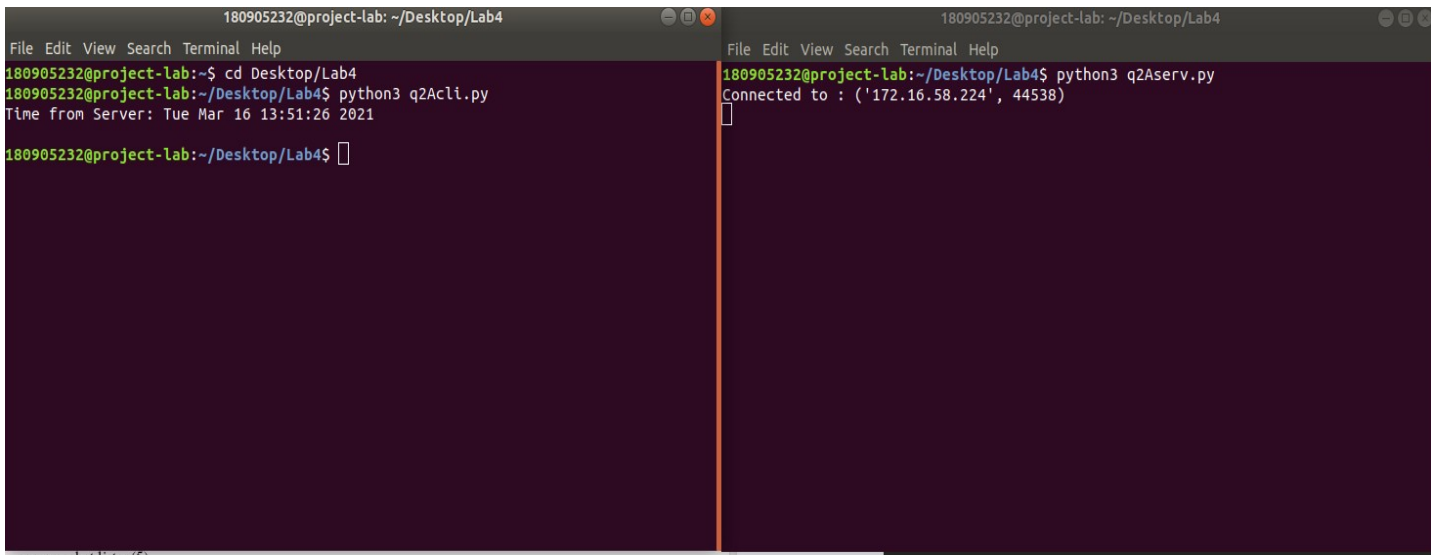
```
servsock = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
host = socket.gethostname()
```

```
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()
```

O/P:



```
180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~$ cd Desktop/Lab4
180905232@project-lab:~/Desktop/Lab4$ python3 q2Acli.py
Time from Server: Tue Mar 16 13:51:26 2021
180905232@project-lab:~/Desktop/Lab4$
```

```
180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 q2Aserv.py
Connected to : ('172.16.58.224', 44538)
```

3A]

#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())
```

#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)

```

O/P:

The image shows two terminal windows side-by-side, both titled '180905232@project-lab: ~/Desktop/Lab4'.

Left Terminal (Client):

```

180905232@project-lab:~$ cd Desktop/Lab4
180905232@project-lab:~/Desktop/Lab4$ python3 q3Accli.py
Enter your name: Garv
Trying to connect to 127.0.0.1 ( 31621 )
Connected...
Abc has joined the chat room
Enter [e] to exit chat room
Abc : hey
Me : HI
Abc : Left chat room!
Me : [e]
180905232@project-lab:~/Desktop/Lab4$ 

```

Right Terminal (Server):

```

180905232@project-lab:~$ cd Desktop/Lab4
180905232@project-lab:~/Desktop/Lab4$ python3 q3Aserv.py
Waiting for incoming connections...
Received connection from 127.0.0.1 ( 48280 )
Garv has connected to the chat room
Enter [e] to exit chat room
Enter your name: Abc
Me : hey
Garv : HI
Me : [e]
180905232@project-lab:~/Desktop/Lab4$ 

```

4A]

#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()
```

#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('Waiting 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()

```

O/P:

```

180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 q4Accli.py
Waiting for connection
Client Say Something: Hi
Received from Server : hello
Client Say Something: 

```

```

180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 q4aserv.py
Waiting for a Connection..
Connected to: 127.0.0.1:46506
Thread Number: 1
Received from client :1Hi
Server Says: hello

```

Exercise:

1]

#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)

```

```

# Create a datagram socket

UDPServerSocket = socket.socket(family=socket.AF_INET, type=socket.SOCK_DGRAM)

# Bind to address and ip

UDPServerSocket.bind((localIP, localPort))

print("UDP server up and listening")

# Listen for incoming datagrams
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)

    # Sending a reply to client

    UDPServerSocket.sendto(bytesToSend, address)

#Client Side

import socket

msgFromClient = "Hello UDP Server"

bytesToSend = str.encode(msgFromClient)

serverAddressPort = ("127.0.0.1", 20001)

bufferSize = 1024

```

Create a UDP socket at client side

```
UDPClientSocket = socket.socket(family=socket.AF_INET, type=socket.SOCK_DGRAM)
```

Send to server using created UDP socket

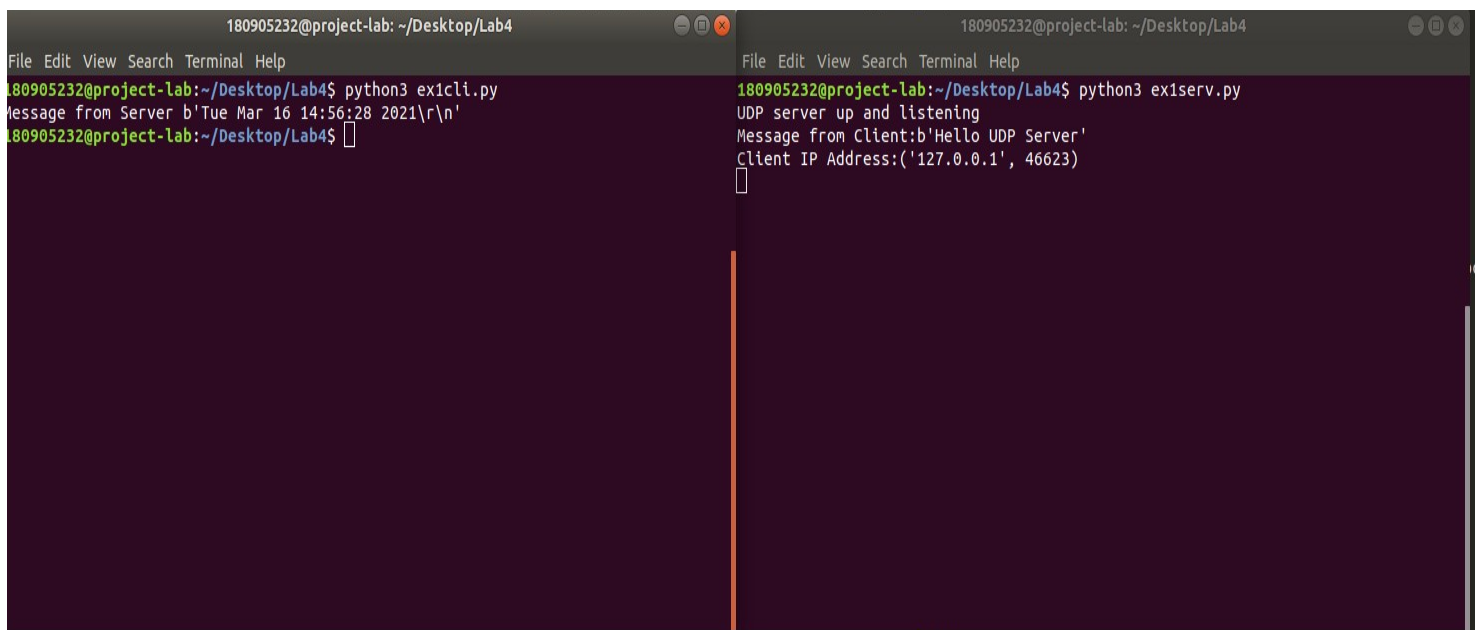
```
UDPClientSocket.sendto(bytesToSend, serverAddressPort)
```

```
msgFromServer = UDPClientSocket.recvfrom(bufferSize)
```

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

```
print(msg)
```

O/P:



The image shows two terminal windows side-by-side. The left window is titled '180905232@project-lab: ~/Desktop/Lab4' and shows the command 'python3 ex1cli.py' being executed. The output is 'Message from Server b'Tue Mar 16 14:56:28 2021\r\n''. The right window is also titled '180905232@project-lab: ~/Desktop/Lab4' and shows the command 'python3 ex1serv.py' being executed. The output is 'UDP server up and listening', 'Message from Client:b'Hello UDP Server'', and 'Client IP Address:('127.0.0.1', 46623)'.

```
180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 ex1cli.py
Message from Server b'Tue Mar 16 14:56:28 2021\r\n'
180905232@project-lab:~/Desktop/Lab4$

180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 ex1serv.py
UDP server up and listening
Message from Client:b'Hello UDP Server'
Client IP Address:('127.0.0.1', 46623)
180905232@project-lab:~/Desktop/Lab4$
```

2]

#Client

```
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)
```

```
#Server
```

```
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")
```

```
# Listen for incoming datagrams
```

```
while(True):
```

```
    bytesAddressPair = UDPServerSocket.recvfrom(bufferSize)
```

```
    message = bytesAddressPair[0]
```

```
    address = bytesAddressPair[1]
```

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

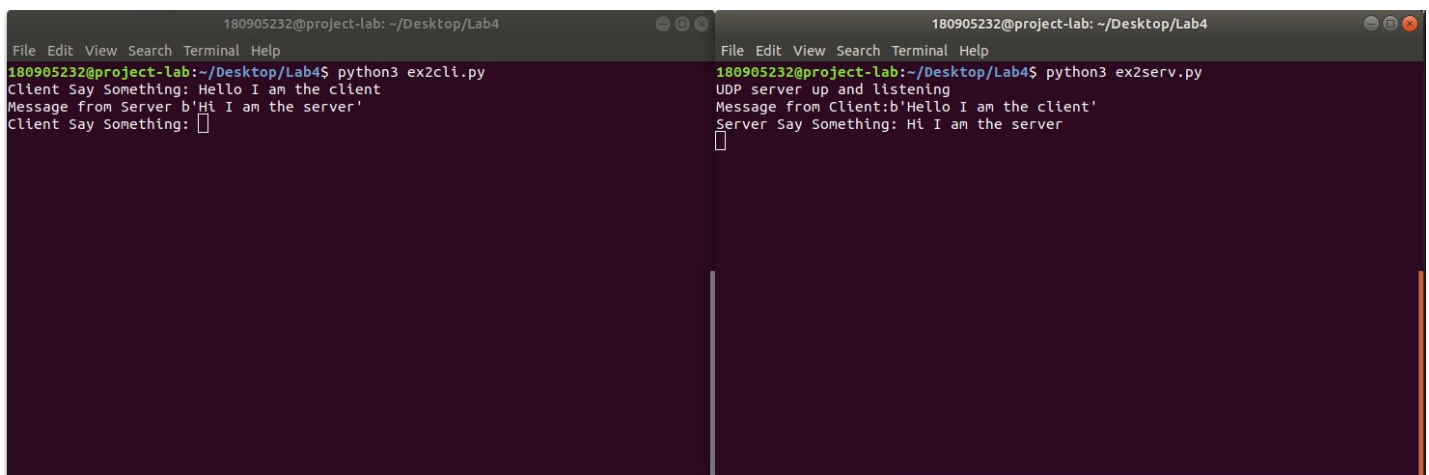
```
    print(clientMsg)
```

```
    Input = input('Server Say Something: ')
```

```
    bytesToSend = str.encode(Input)
```

```
    UDPServerSocket.sendto(bytesToSend, address)
```

O/P:



The image shows two terminal windows side-by-side. The left window is titled '180905232@project-lab: ~/Desktop/Lab4' and shows the execution of 'python3 ex2cli.py'. The user enters 'Hello I am the client' and receives the response 'Message from Server b'Hi I am the server''. The right window is titled '180905232@project-lab: ~/Desktop/Lab4' and shows the execution of 'python3 ex2serv.py'. It displays 'UDP server up and listening', receives 'Message from Client:b'Hello I am the client'', and sends 'Server Say Something: Hi I am the server'.

```
180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 ex2cli.py
Client Say Something: Hello I am the client
Message from Server b'Hi I am the server'
Client Say Something: █

180905232@project-lab: ~/Desktop/Lab4
File Edit View Search Terminal Help
180905232@project-lab:~/Desktop/Lab4$ python3 ex2serv.py
UDP server up and listening
Message from Client:b'Hello I am the client'
Server Say Something: Hi I am the server
█
```

3]

#Client

import socket

serverAddressPort = ("172.16.58.224", 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)

#Server

import socket

localIP = "172.16.58.72"

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")

Listen for incoming datagrams

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)