S. No.	Date	Title	Page No.	Teacher's Sign / Remarks
12.	02·11·2024	echo Client Server using TCP/UDP		
13.	8·11·2024	Ping Program.		
14	6·11·2024	RAW socket		
15.	8·11·2024	Web Analyzer.		
		Completed		

Exp No: 12   Echo client / Server using
Date :   TCP / UDP.

Aim:
a). Implement echo client server using TCP/UDP
sockets.

client:
___

import socket
import time
def ping_server (host = '127.0.0.1', Port = 12245):

        socket. socket (socket. AF_INET)

        socket. SOC DGRAM) as s :

        try :

              s. send to (6 "Hello", (host, port))

        except s.timeout:

              print ("Request time out");

        if_name = "main" :

              ping_server ()

Server :
___

        import socket
        def start_server (host = '127.0.0..1';
              port = 12345);
        with socket. socket (socket. AR_NET ; socket.
              Sock_DGRAM) as s :

```
c.bind ((host, port))
print (f "UDP server running on {host}")

while True:
    data, addr = s.recvfrom(1024)
    print (f "Received message from {addr}:
            {data.decode}")


if __name__ = "main":
    client_server()
```

O/P: python server.py

UDP server running on 127.0.0.1: 12345
Received message from ('121.0.0.1', 59290).

python client.py
Received reply from server: Hello, Client.

b). Implement a chat client server using TCP/UDP sockets!

```
import socket
def server ():
    port = 12345
    host = '127.0.0.1'
    with socket.socket(socket.AK_INET, socket.
                SOCK_DGRAM) on s:
        s.bind((host, port))
        while (True):
```

```python
        aold = s.recvfrom(1024)
        print("client", {d.decode{}})
        a = input("enter reply")
        s.sendto(a.encode(), odd)
        if(a == "end")
            break
            exit
```

recvr2.py:

```python
import socket
import time
def recvr2(a):
    host = '127.0.0.1'
    port = 12345
    with socket.socket(socket.AF_INET, socket.
        sock-DGRAM) as s:
        s.sendto(a.encode(), (host, port))

        d, addr = s.recvfrom(1024)
        print({d.decode{}})

        while(true):
            a = input("Enter message"):
            if(a == "end"):

                recvr2(a)
                break
            else:
                recvr(2)
```

O/P:
python .\ chat_serv.py
client { 'hi'}
enter Reply : hello
client {" How are you ?"}
enter Reply : Im fine

python .\recvr.py
enter message hi
{'hello'}
enter message How are you
{'Im fine'}
Enter Message

Result:
Thus the program is executed successfully and
the O/P is verified.