Exp. no: 12
Date: 23/10/24

a) Echo client server
b) chat client server using TCP/IP

Aim:

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

Client:

```
import socket
import time
def ping_server(host='127.0.0.1', port=12345):
    socket.socket(socket.AF_INET,
        socket.SOCK_DGRAM) as s:
    try:
        s.sendto("Hello", (host, port))
    except s.timeout:
        print("Request timed out")
if __name__ == "__main__":
    ping_server()
```

Server:

```
import socket
def start_server(host='127.0.0.1', port=12345):
with socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    as s:
    s.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__":
    start_server()
```

O/P

```
python server.py
UDP, server running on 127.0.0.1:12345
Received message from ('127.0.0.1', 59290).
                                    Hello
```

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

b) Implement ~~client~~ chat client server using TCP/IP sockets:

```
chat_serv.py
    import socket
    def server():
        port = 12345
        host = '127.0.0.1'
        with socket.socket(socket.AF_INET,
```

```
s.bind((host, port))
while (True):
    d, addr = s.recvfrom(1024)
    print("client", d.decode())
    a = input("Enter reply")
    s.sendTo(a.encoded(), addr)
    if(a=="end")
        break
        exit
```

Receiver 2.py:

```
import socket
import time
def recver2(a):
    host = '127.0.0.1'
    port = 12345
    with socket
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"):
```

```
            recv2(c)
            break
        else:
            recv2(a)
```

o/p

```
python .\chat_serv.py
Client {'hi'}
Enter Reply Hello
Client {"How are you"}
Enter reply : Im fine
python .\ recv.py
    Enter message hi
    {'hello'}
    Enter messages How are you
    {'I'm fine'}
    Enter Message.
```

Result:

Thus the program is executed sucessfully and output is verified.