Practical 12

Expt : 12

Date : 6.10.25

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

Code:

```
import socket
import time
def ping_server (host ='127.0.0.1', port = 12345):
    with socket.socket (socket.AF_INET, socket.SOCK_DGRAM)
                                                  as s:
        s.settimeout (2)
        try:
            s.sentto ( b'Hello', (host, port))
            print (f"Message sent to {host}:{port}")
            data, addr = s.recvfrom (1024)
            print (f"Received reply from {addr}:
                                  {data.decode ()}")
        except socket.timeout :
            print ("Request timed out")
ping_server ()
```

Input:

```
ping_server()
      or
ping_server('192.168.1.5', 9999)
```

Output:

When a server is running, then OP is:

```
Message sent to 127.0.0.1:12345
Received reply from ('127.0.0.1', 12345): Hello client
```

RESULT:

The UDP client code is written and executed. The UDP client tries to send message and is verified successfully.

Expt : 12 b

Date: 6.10.25

AIM: Implement chat client server using TCP/ UDP sockets

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

start_server()
```

Input:

The server is started

start_server ()

Output:

After running the server

```
- UDP server running on 127.0.0.1:12345
```

After the client sends message, the server output:

```
- Received message from ('127.0.0.1', 55544):
                                       Hello
```

RESULT:

The UDP server was successfully implemented and executed. It received the message sent by client and was verified.