

Ex NO:120

ECHO CLIENT SERVER USING TCP/UDP

DATE: 23.09.25

USING SOCKETS

Aim:

To implement echo client server using TCP/UDP sockets.

Algorithm:

a) Client

```
import socket
def ping_server(host = '127.0.0.1', port = 12345):
    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
        s.settimeout(2)
        message = input("Enter message to send: ")
        message = message.encode()
        s.sendto(message, (host, port))
        try:
            data, addr = s.recvfrom(1024)
            print("Echo from server: ", data.decode())
        except socket.timeout:
            print("Request timed out")
    if __name__ == "__main__":
        ping_server()
```

b) 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("UDP Server running on host: {} port: {}".format(host, port))
        while True:
            data, addr = s.recvfrom(1024)
            print(f"Received message from {addr}: {data.decode()}")
            s.sendto(data, addr)
```

```
if __name__ == "__main__":
    start_server()
```

Input or output:

Client: The original developer builds tests based upon the original requirements.

Enter message to send: Hello

Echo from server: Hello

lower:

UDP Server running on 129.0.0.1:12345

Received message from (1124.0.0.1:54321): Hello

C. confidens auf Böden mit sandigem Substrat

(elbo, "gel lastchino") 1 mtrq.

about 'time' is created about me, about me, too.

(C. torb. with before trials) thing

Output Input

(also " : brevis") trinque

(["]: νον["]) Βαρζι = βεν

Wadsworth - Penn Distress - 1903

the following = a Original copy file

C. below "top") may

~~for out~~

1981: *Levi's* launches its first television commercial.

Result : The process has been completed successfully.

~~The program has been executed successfully.~~

63/60/2

Aim:

To implement chat client server using TCP/UDP
sockets.

a) Server

```

import socket
def start_server(host='127.0.0.1', port=12345):
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.bind((host, port))
    s.listen(1)
    print("Server is waiting for connection...")
    conn, addr = s.accept()
    print("Connected by ", addr)
    while True:
        data = conn.recv(1024).decode()
        if not data or data.lower() == 'exit':
            print("Client ended the chat.")
            break
        print("Client:", data)
        msg = input("You: ")
        conn.sendall(msg.encode())
        if msg.lower() == 'exit':
            print("Chat ended!")
            break
    conn.close()
    s.close()
start_server()

```

b) Client

```
import socket
def start_client(host = '127.0.0.1', port = 12345):
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect((host, port))
    print("Connected to server.")
    while True:
        msg = input("You: ") + '\n'
        s.sendall(msg.encode())
        if msg.lower() == 'exit':
            print("Chat ended.")
            break
        data = s.recv(1024).decode()
        if not data or data.lower() == 'exit':
            break
        print("Server: ", data)
    s.close()
```

start_client()

Output:

```
Server is waiting for connection
Connected by ('127.0.0.1', 61773)
Client: Hello
You: Hi
Client ended the chat
Client: 
You: Hello
Server: Hi
You: exit
chat ended.
Result: The program has been executed successfully.
```