Ex: No: 12

Date: 28/10/24

Practical - 12

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):
    with socket.socket (socket.AF_INET,
        Socket.Soc_dgram) as s:
    try:
     s.sendto (6 "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()
```

CIP:- 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
```

5) Implement chat client server using tcp/ucp sockets:

chat serv.py

```
import socket
def sent():
    port = 12345
    host = '127.0.0.1'
    with socket.socket(socket.AF_INET, socket.
        SOCK_DGRAM) as s:
    s.bind((host, port))
    while (True):
        d, add = s.recvfrom(1024)
        print("client", +d.decode()+)
        a = input("Enter reply")
        s.sendto(a.encoded(), add)
        if (a == "end")
            break
            exit.
```

receive2.py:

```
import socket
import time
def recv2(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(f{d.decode()})

while(true):
    a = input("enter message")
    if(a == "end"):
        recv2(a)
        break
    else:
        recv2(a)
```

O/P:

```
python .\chart_serv.py
client {'hi'}
Enter reply hello.
client {'how are you'}
Enter reply: Imfine
```

```
python .\recv.py
Enter message hi
    {'hello'}
Enter messages how are
you {'Imfine'}
Enter message.
```

Result:

Thus the program is executed successfully & the output is verified.

19/11/24