12/10/25 Experiment 13

AIM : Implement your own ping program

Algorithm :

1) Create a UDP Socket
2) Set a timeout of 2 seconds
3) Record start time
4) Send the message "ping" to the server
5) wait to receive a response from the server.
   * If received, record end time and display the reply with round-trip time
   * If timeout occur, print "Request time out".
6) close the Socket.

Input

```
import socket
import time
def ping-server (host = '127.0.0.1', port = 12345):
    with
  socket. socket (socket. AF_ INET, socket . sock_DGR
                                          as s :
        try :
            s. settimeout (2)
            Start = time . time()
            s. sendto ( b 'ping': (host , port ))
            data, addr = s. recvfrom (1024)
            end = time. time()
            print (f " Received {data. decode ()} from
              {addr} in {end-start : .2f } seconds')
                Except Socket. timeout :
                    print ( " Request timed out ")
```

```
if _name_ == "__main__":
    ping_server ().
```

<u>output</u>:

```
Received Pong from ('127.0.0.1', 12345) in
                          0.00 seconds

Request timed out
```

Experiment 13 b

<u>Algorithm</u>:

```
Import socket
def start_server ( host = '127.0.0.1', port = 12345):
    with
    socket.socket (socket.AF_INET, socket.sock_DGRAM)
    as s:
        s.blind ((host, port))
        print (f "UDP server running on {host}:
                                        {port}")
    while True :
        data, addr = s.recv from (1024)
        print (f "Received message from {addr}:
                          {data.decode ()}")
            s.send to (b'Pong', addr)

    if__name__ == "__main__":
        start_server ()
```

Output

```
UDP server running on 127.0.0.1 : 12345
Received message from ('127.0.0.1", 52345):
                                          ping.
```

Result : Thus the ping program has been implemented successfully.

[illegible]