Experiment-13

Aim:

To implement your own ping program

Algorithm:

UDP server:

-> Create UDP socket & bind it to a specific address & port

-> wait for message

-> print message & clients address

-> Send back pong to client

UDP client:

-> Create UDP Socket & set a 2sec timeout

-> Send 'ping' to server

-> if no response within 2sec print request timeout.

code:

server.py

```
import socket
def start_server(host='127.0.0.1', port=
                                        12345):
with socket.socket(socket.AF_INET, socket.
   s.bind((host,post)/ socket_D) as s:
print(f"UDP server running on {host}:{port}")
    while True:
    data, addr = s.recieved(1024)
```

```
print(f"Recieved message from {addr}:
                {data.decode()}")
        s.sendto(b'pon', addr)
if __name__ == "__main__":
    start_server()

client.py
import time
import socket
def ping_server(host='127.0.0.1', port=
                                12365"
    with socket.socket(socket.AF_INET, socket.
                   socket.SOCK_DGRAM) as s:
     try:
     s.settime(2)
     start = time.time()
     s.sendto(b'ping', (host, port))
     data, addr = s.recvfrom(1024)
     end = time.time()
     print(f"Recieved {data.decode()}
        from {addr} in {end-start:.2f}
                      seconds")
       except socket.timeout;
       print("Request timed out")
if __name__ == "__main__":
    ping_server()
```

O/P :

Terminal	Terminal
> Python Server.py UDP server running on 127.0.0.1:12345 Recieved message from ('127.0.0.1', 50061): ping	> python Client.py Recieved pong from ('127.0.0.1', 12345) in 0.00 seconds.

Result :

Thus a ping program has been executed successfully.