

Ex: NO: 12

Practical - 12.

Date: 28.10.24

Aim:

Implementing 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.SOCK_DGRAM ) as s :
    try :
        s.sendto ( "Hello", (host, port))
    except s.time out :
        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 ( "UDP server running on port" )
while True:
    data, addr : s.recvfrom (1024)
    print f ( "Received message" )
    if __name__ == "main":
        start_server()
```



OID: python receiver.py

UDP server running on 127.0.0.1:12345

Received message from (127.0.0.1, 59290)

python client.py

Received reply from server: Hello, client

b) Implement chat client server using TCP / UDP sockets

chat serv.py

```
import socket
```

```
def receiver():
```

```
    port = 12345
```

```
    host = '127.0.0.1'
```

```
    with socket.socket(socket.AF_INET,  
                        SOCK_STREAM) as s:
```

```
        s.bind((host, port))
```

```
    while True:
```

```
        d = s.recvfrom(1024)
```

```
        print('Client:')
```

```
        a = input("Enter Reply")
```

```
        s.sendto(a.encode(), d)
```

```
        if a == 'end':
```

```
            break
```

```
            exit
```

```
receiver()
```

receiver2.py

```
import socket
```

```
import time
```

```
def receiver():
```

```
    host = '127.0.0.1'
```



while (True):

a = input ("Enter Message")

if (a == "end"):

receiver (0)

break

else:

receiver (a)

O/P:

python .\chat\_serv.py

client {'hi'}

client {'How are you'}

Enter reply I'm fine

python

.\recr.py

Enter Message hi

{ 'Hello' }

Enter Messages How are you

{ "I am fine" }

Enter Message .