01.10.25 Practical - 12 - TCP/UDP packets

Aim: (a) Implement echo client server using TCP/UDP Socket

Code:

TCP Server:

```
import socket
HOST = '127.0.0.1'
PORT = 65432
with socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.bind((HOST, PORT))
    s.listen()
print(f'TCP server listening on {HOST}:{PORT}')
conn, addr = s.accept()
with conn:
    print:
        print(f"connected by {addr}")
    while True:
        data = conn.recv(1024
        if not data:
            break
print(f"Received {data.decode()}")
conn.sendall(data)
```

TCP client

```
import socket
Host = '127.0.0.1'
port = 5432
with socket(socket.AF_INET,
socket.SOCK_STREAM)
s.connect((HOST,PORT))
while TRUE:
    message = input("You:")
    if message.lower() = 'quit':
        break
s.sendall(message.encode())
    data = s.recv(1024)
print('Echo: {data.decode()}')
```

Input: Client types something

Server output: Client: Ping

Client output: You: Ping server: ACK: Ping

Result:

Implementation of echo client server using TCP/IP sockets was completed successfully

07.10.25 Practical 12-b · Chat client server.

Aim b) Implement chat client server using TCP, UDP sockets.

Code:

UDP Server

```
import socket
Host = '127.0.0.1'
Port = 65433
with socket.socket(socket.AF_INET, socket.SOCK_DGRAM):
    s.bind((Host, PORT))
    print(f"UDP Server listening on {Host}:{PORT}");
    while True:
        data, addr = s.recvfrom(1024)
        print(f"Received from {addr}: {data.decode()}")
        s.send(data, addr)
```

UDP client:

```
import socket
Host = '127.0.0.1'
Port = 65433
```

```
with socket.socket(socket.AF_INET,
    socket_stream.programs:
while True:
    message = input("You:")
    if message.lower() == 'exit':
        break
    s.sendto(message.encode(), (HOST, PORT))
    data server = s.recvfrom(1024)
    print(f"Echo: {data.decode()}")
```

Output.

```
PS python TCP_server.py
server output:
TCP server listening on 127.0.0.1:65432
Client output
Connected to server 127.0.0.1:65432.
type 'exit' to quit
```

Result:

Implementation of chat client & server using TCP/UDP sockets was executed Successfully.