**TCP Server**

import socket

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server\_address = ('localhost', 10000)

server.bind(server\_address)

server.listen(1)

print("waiting for the connection")

connection, client\_address = server.accept()

print("Connection established with", client\_address)

message = ""

while message != "end":

data = connection.recv(1000).decode()

if data:

print(data)

message = input()

connection.sendall(message.encode())

else:

break

connection.close()

server.close()

**TCP Client**

import socket

client = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server\_address = ('localhost', 10000)

client.connect(server\_address)

print("connecting to port ", server\_address)

message = input()

client.sendall(message.encode())

while message != 'end':

data = client.recv(1000).decode()

if data:

print(data)

message = input()

client.sendall(message.encode())

else:

break

client.close()

**UDP Server**

import socket

server = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM) # Create UDP socket

server\_address = ('localhost', 10000)

server.bind(server\_address)

print("UDP Server is listening on port 10000...")

while True:

data, client\_address = server.recvfrom(1024) # Receive data from client

message = data.decode()

print(f"Received from {client\_address}: {message}")

if message.lower() == "end":

print("Closing server...")

break

response = input("Enter response: ")

server.sendto(response.encode(), client\_address) # Send response to client

server.close()

**UDP Client**

import socket

client = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM) # Create UDP socket

server\_address = ('localhost', 10000)

while True:

message = input("Enter message: ")

client.sendto(message.encode(), server\_address) # Send data to server

if message.lower() == "end":

print("Closing client...")

break

data, \_ = client.recvfrom(1024) # Receive response from server

print("Server:", data.decode())

client.close()