**UDP socket:**

**Code:**

**Server:**

from socket import \*

# Server port and address

serverPort = 12000

# Create the server UDP socket

serverSocket = socket(AF\_INET, SOCK\_DGRAM)

# Bind the server socket to localhost (127.0.0.1) and port 12000

serverSocket.bind(("127.0.0.1", serverPort))

# Print message to confirm the server is ready

print("The server is ready to receive")

# Infinite loop to keep the server running

while True:

# Receive the file name from the client

sentence, clientAddress = serverSocket.recvfrom(2048)

try:

# Open the file requested by the client

with open(sentence.decode("utf-8"), "r") as file:

# Read the contents of the file

file\_contents = file.read(2048)

# Send the contents back to the client

serverSocket.sendto(bytes(file\_contents, "utf-8"), clientAddress)

print(f"Sent back to client: {file\_contents}")

except FileNotFoundError:

# Send error message if the file is not found

error\_message = f"File '{sentence.decode('utf-8')}' not found."

serverSocket.sendto(bytes(error\_message, "utf-8"), clientAddress)

print(f"Error: {error\_message}")

**Client:**

from socket import \*

# Server address and port

serverName = "127.0.0.1" # Localhost address

serverPort = 12000 # Port number

# Create the client UDP socket

clientSocket = socket(AF\_INET, SOCK\_DGRAM)

# Get the file name from the user

sentence = input("Enter file name: ")

# Send the file name to the server

clientSocket.sendto(bytes(sentence, "utf-8"), (serverName, serverPort))

# Receive the file contents or error message from the server

filecontents, serverAddress = clientSocket.recvfrom(2048)

# Print the received contents

print('From Server:', filecontents.decode("utf-8"))

# Close the client socket

clientSocket.close()

**Output:**



