**Client Server Programming using TCP and UDP Sockets**

Server.py

from socket import \*

# Define server details

serverName = "127.0.0.1"

serverPort = 12000

# Create and configure the server socket

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

serverSocket.bind((serverName, serverPort))

serverSocket.listen(1)

print("The server is ready to receive")

while True:

    # Accept a client connection

    connectionSocket, addr = serverSocket.accept()

    print(f"Connection established with {addr}")

    try:

        # Receive the file name from the client

        sentence = connectionSocket.recv(1024).decode()

        # Try to open and read the requested file

        with open(sentence, "r") as file:

            fileContents = file.read(1024)

            connectionSocket.send(fileContents.encode())

    except FileNotFoundError:

        # Handle case where file does not exist

        errorMessage = f"Error: File '{sentence}' not found"

        connectionSocket.send(errorMessage.encode())

    except Exception as e:

        # Handle other exceptions

        errorMessage = f"Error: {str(e)}"

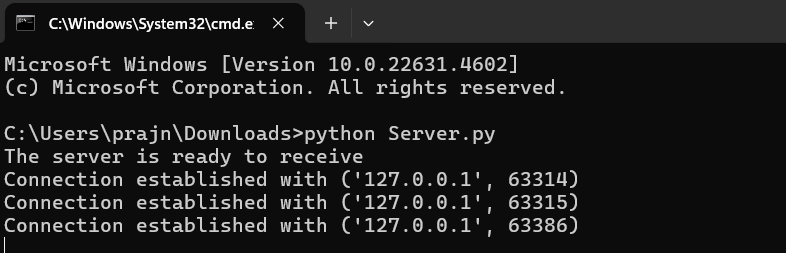
        connectionSocket.send(errorMessage.encode())

    finally:

        # Close the client connection

        connectionSocket.close()

Output:



Client.py

from socket import \*

# Define server details

serverName = "127.0.0.1"

serverPort = 12000

# Create and connect the client socket

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

clientSocket.connect((serverName, serverPort))

# Get the file name from the user

sentence = input("Enter file name: ")

# Send the file name to the server

clientSocket.send(sentence.encode())

# Receive the server's response

filecontents = clientSocket.recv(1024).decode()

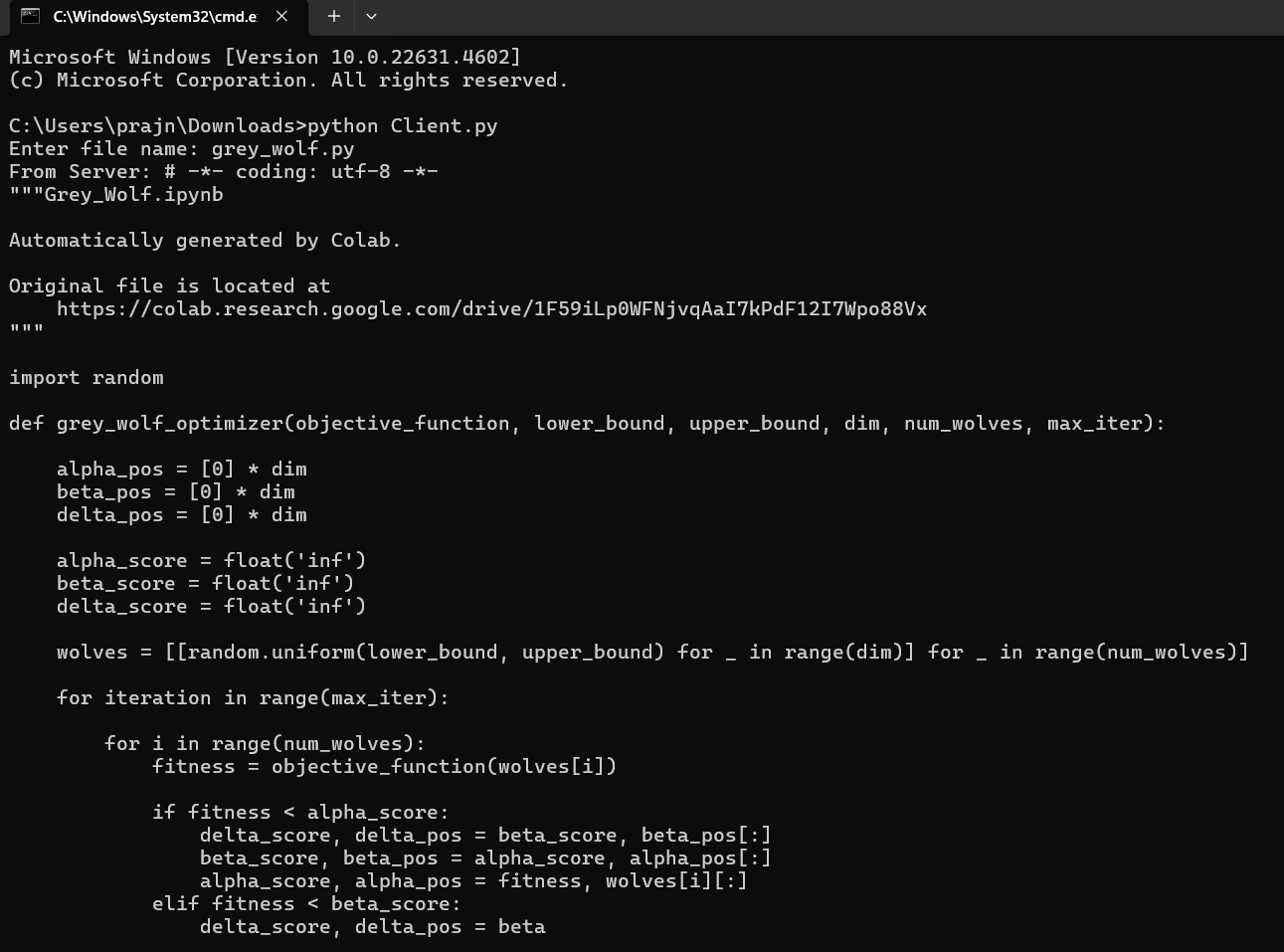
# Print the server's response

print("From Server:", filecontents)

# Close the client socket

clientSocket.close()

Output :



ServerUDP.py

from socket import \*

# Define server details

serverPort = 12000

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

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

print("The server is ready to receive")

while True:

    # Receive file name from the client

    sentence, clientAddress = serverSocket.recvfrom(2048)

    filename = sentence.decode("utf-8")  # Decode the file name

    try:

        # Try to open and read the requested file

        with open(filename, "r") as file:

            fileContents = file.read(2048)

            serverSocket.sendto(fileContents.encode("utf-8"), clientAddress)

        print(f"Sent back to client: {fileContents}")

    except FileNotFoundError:

        # Handle case where file does not exist

        errorMessage = f"Error: File '{filename}' not found"

        serverSocket.sendto(errorMessage.encode("utf-8"), clientAddress)

        print(errorMessage)

    except Exception as e:

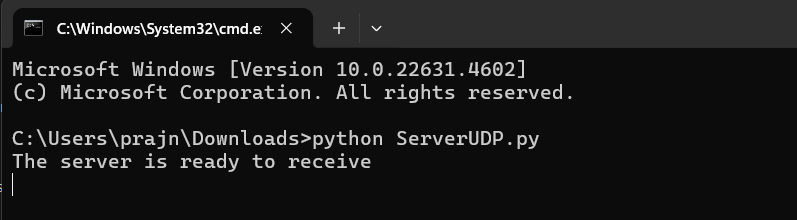
        # Handle other exceptions

        errorMessage = f"Error: {str(e)}"

        serverSocket.sendto(errorMessage.encode("utf-8"), clientAddress)

        print(errorMessage)

Output:



ClientUDP.py

from socket import \*

# Define server details

serverName = "127.0.0.1"

serverPort = 12000

# Create the UDP client 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(sentence.encode("utf-8"), (serverName, serverPort))

# Receive the server's response

filecontents, serverAddress = clientSocket.recvfrom(2048)

# Print the server's response

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

# Close the client socket

clientSocket.close()

Output:

