Assignment - 10 Socket Programming - File Transfer (Multiple Clients)

K.CHARAN
2023000608

File Transferring for Multiple Clients in Socket Programming includes:

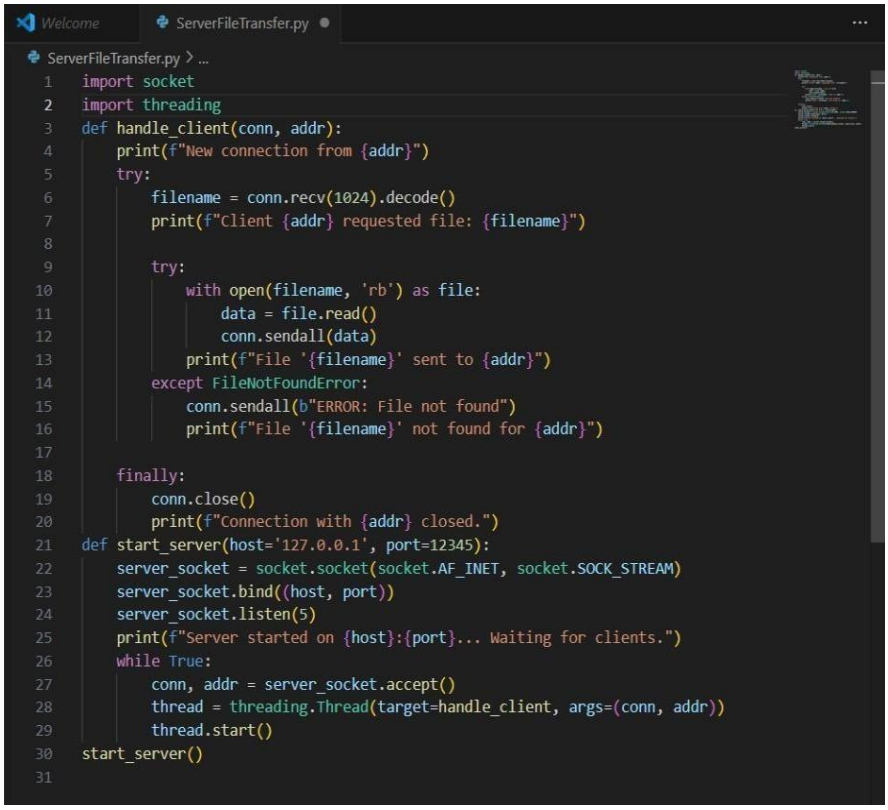
Server

- Listens for incoming client connections.
- Uses **multithreading** to handle multiple clients concurrently.
- Sends a requested file to the connected client.

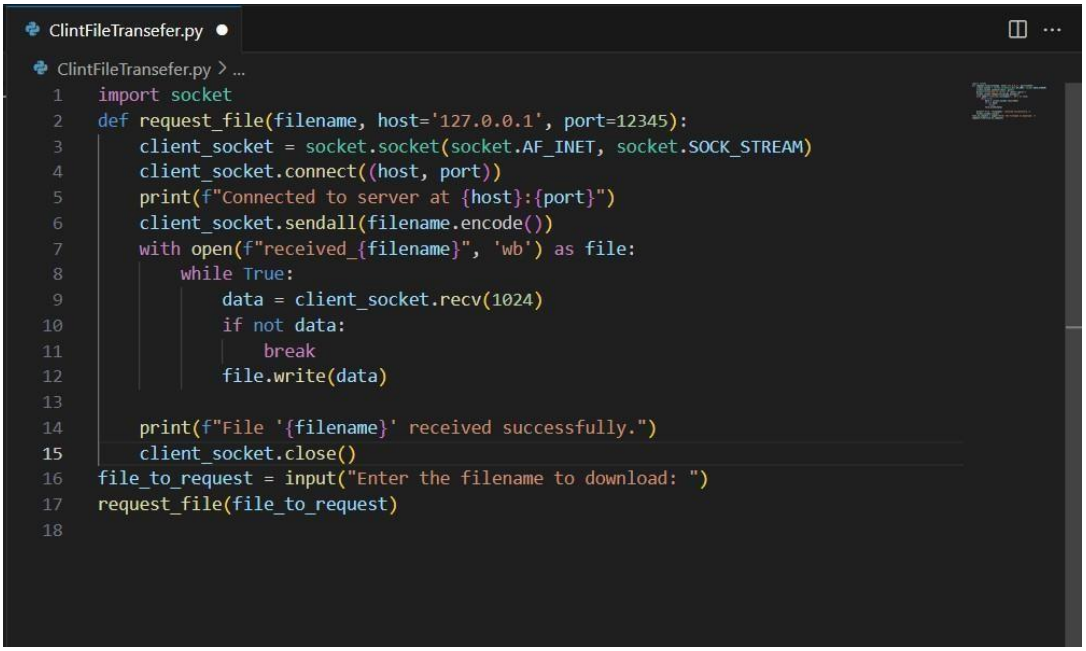
Client

- Connects to the server.
- Requests a file from the server.
- Receives and saves the file locally.

Server code:



Clint Code:



Its Works like:

Server Side

1. Creates a TCP socket and binds it to a port.
2. Waits for incoming client connections.
3. When a client connects:

Spawns a new thread using threading.Thread(target=handle client, args=(conn, addr)).

This allows handling multiple clients simultaneously.

4. Receives the filename from the client.

5. Checks if the file exists:
    - If found, reads and sends the file.
    - If not, sends an error message.

Client Side

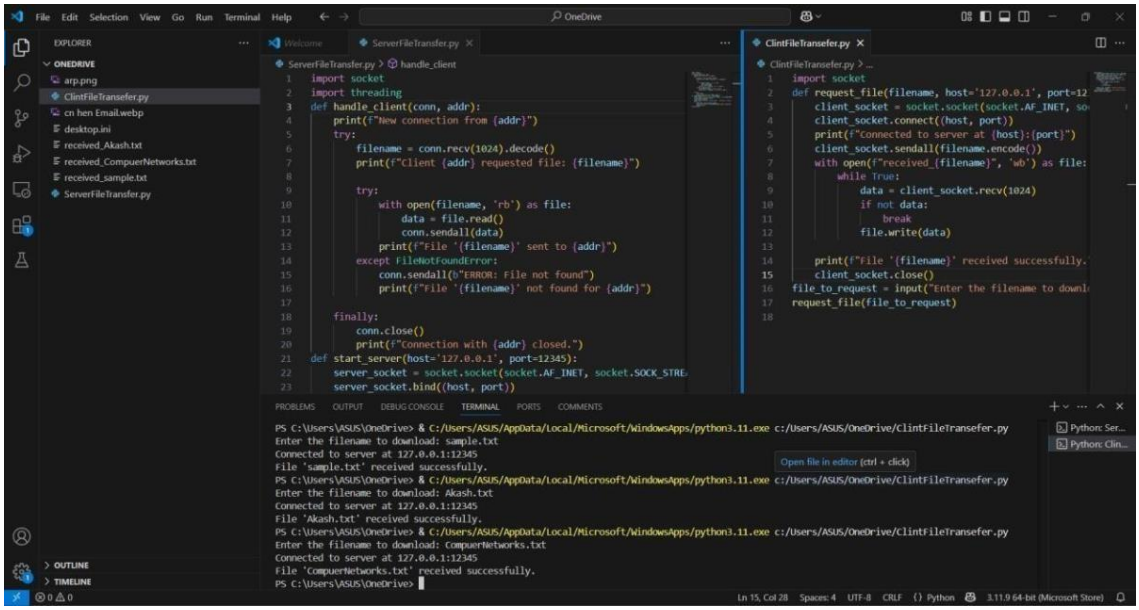
1. Connects to the server.

2. Sends the filename it wants to download.

3. Receives the file and saves it locally.

4. Handles errors if the file is not found.

OUTPUT:


File Edit Selection View Go Run Terminal Help
OneDrive
EXPLORER
ONEDRIVE
arp.png
ClintFileTransefer.py
cn hen Email.webp
desktop.ini
received_Akash.txt
received_CompuerNetworks.txt
received_sample.txt
ServerFileTransfer.py
Welcome
ServerFileTransfer.py
ServerFileTransfer.py > handle_client
import socket
import threading
def handle_client(conn, addr):
print(f"New connection from {addr}")
try:
filename = conn.recv(1024).decode()
print(f"Client {addr} requested file: {filename}")
try:
with open(filename, 'rb') as file:
data = file.read()
conn.sendall(data)
print(f"File '{filename}' sent to {addr}")
except FileNotFoundError:
conn.sendall(b"ERROR: File not found")
print(f"File '{filename}' not found for {addr}")
finally:
conn.close()
print(f"Connection with {addr} closed.")
def start_server(host='127.0.0.1', port=12345):
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STRE
server_socket.bind((host, port))
ClintFileTransefer.py
ClintFileTransefer.py > ...
import socket
def request_file(filename, host='127.0.0.1', port=12
client_socket = socket.socket(socket.AF_INET, so
client_socket.connect((host, port))
print(f"Connected to server at {host}:{port}")
client_socket.sendall(filename.encode())
with open(f"received_{filename}", 'wb') as file:
while True:
data = client_socket.recv(1024)
if not data:
break
file.write(data)
print(f"File '{filename}' received successfully.
client_socket.close()
file_to_request = input("Enter the filename to downl
request_file(file_to_request)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\ASUS\OneDrive> & C:/Users/ASUS/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/ASUS/OneDrive/ClintFileTransefer.py
Enter the filename to download: sample.txt
Connected to server at 127.0.0.1:12345
File 'sample.txt' received successfully.
PS C:\Users\ASUS\OneDrive> & C:/Users/ASUS/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/ASUS/OneDrive/ClintFileTransefer.py
Enter the filename to download: Akash.txt
Connected to server at 127.0.0.1:12345
File 'Akash.txt' received successfully.
PS C:\Users\ASUS\OneDrive> & C:/Users/ASUS/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/ASUS/OneDrive/ClintFileTransefer.py
Enter the filename to download: CompuerNetworks.txt
Connected to server at 127.0.0.1:12345
File 'CompuerNetworks.txt' received successfully.
PS C:\Users\ASUS\OneDrive>
Open file in editor (ctrl + click)
Python: Ser...
Python: Clin...
OUTLINE
TIMELINE
Ln 15, Col 28 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)