Program 3

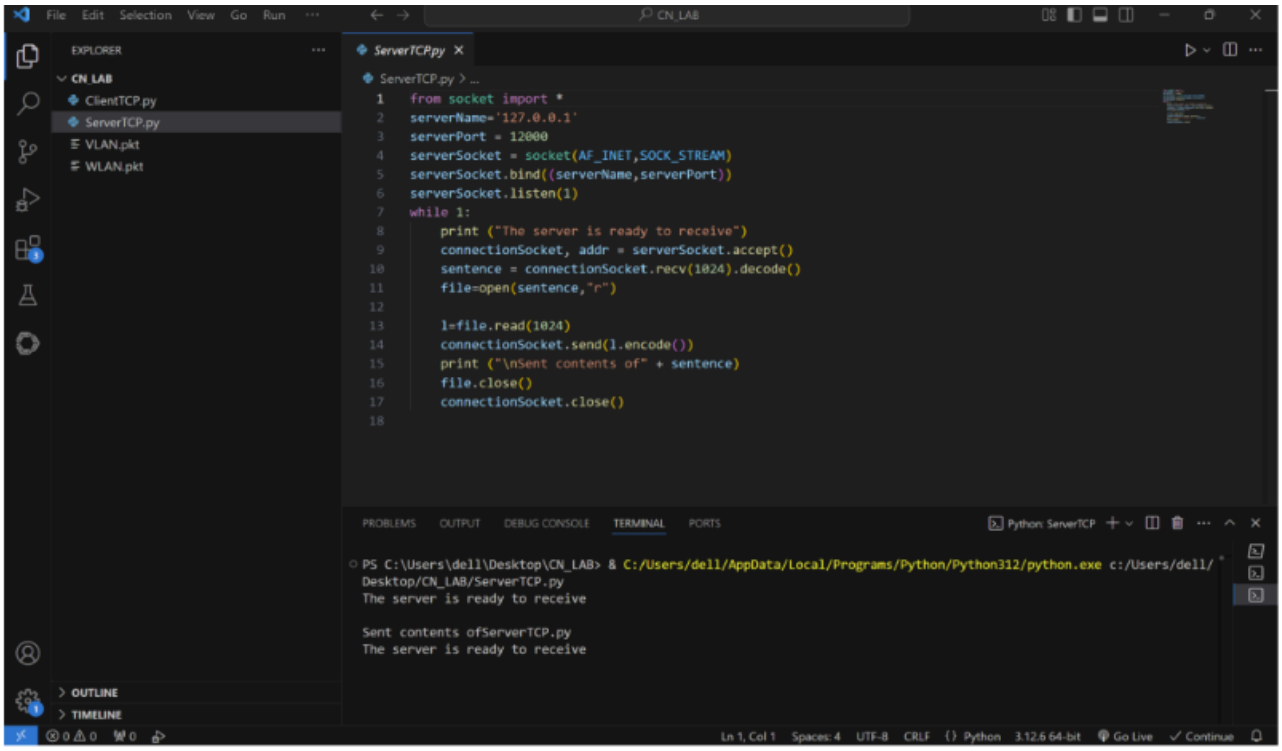
Aim of the program:
Using TCP/IP sockets, write a client-server program to make the client sending the file name and the server to send back the contents of the requested file if present.

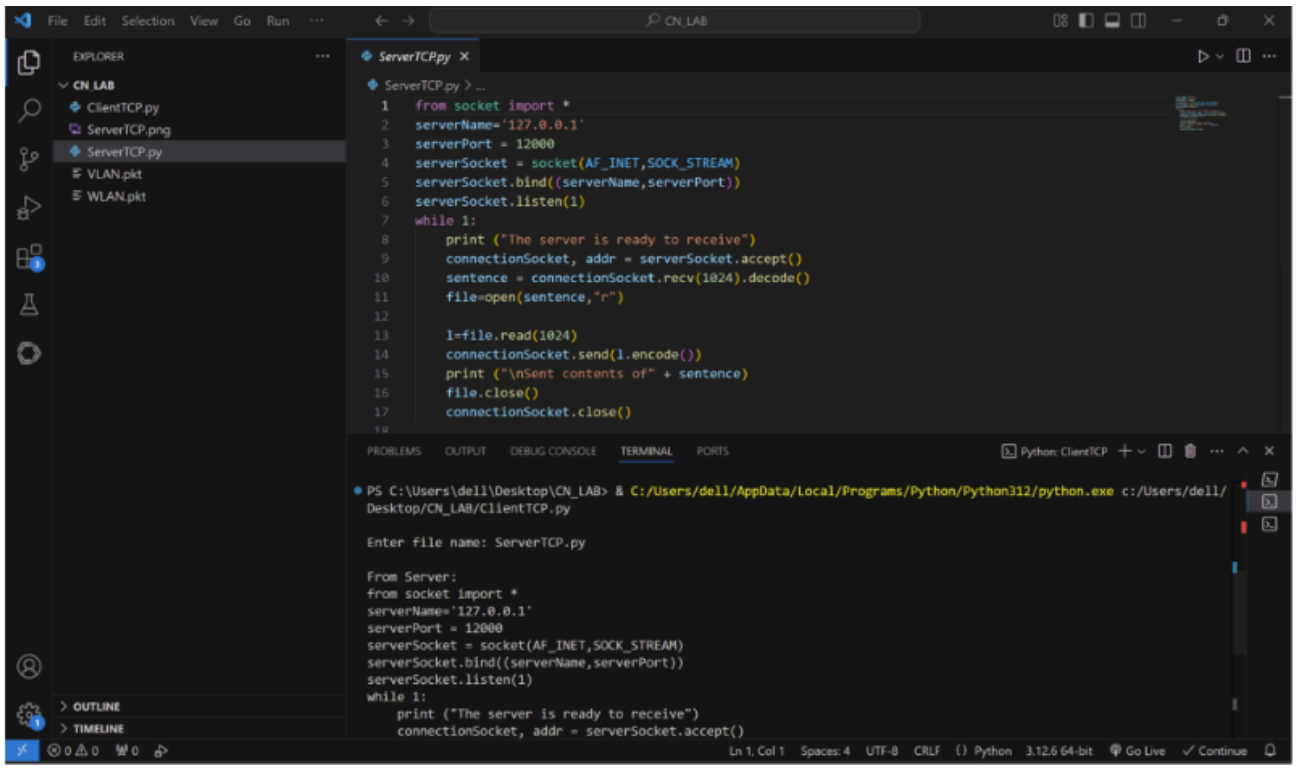
Code:
ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\nEnter file name: ")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print('\nFrom Server:\n')
print(filecontents)
clientSocket.close()
```

ServerTCP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while 1:
print("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file = open(sentence, "r")
l = file.read(1024)
connectionSocket.send(l.encode())
print('\nSent contents of ' + sentence)
file.close()
connectionSocket.close()
```


File Edit Selection View Go Run
CN_LAB
EXPLORER
CN_LAB
ClientTCP.py
ServerTCP.py
VLAN.pkt
WLAN.pkt
ServerTCP.py
ServerTCP.py > ...
from socket import *
serverName='127.0.0.1'
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file=open(sentence,"r")
l=file.read(1024)
connectionSocket.send(l.encode())
print ("\nSent contents of" + sentence)
file.close()
connectionSocket.close()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python: ServerTCP
PS C:\Users\dell\Desktop\CN_LAB> & C:/Users/dell/AppData/Local/Programs/Python/Python312/python.exe c:/Users/dell/Desktop/CN_LAB/ServerTCP.py
The server is ready to receive
Sent contents ofServerTCP.py
The server is ready to receive
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python 3.12.6 64-bit Go Live Continue


File Edit Selection View Go Run
CN_LAB
EXPLORER
CN_LAB
ClientTCP.py
ServerTCP.png
ServerTCP.py
VLAN.pkt
WLAN.pkt
ServerTCP.py
ServerTCP.py > ...
from socket import *
serverName='127.0.0.1'
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
sentence = connectionSocket.recv(1024).decode()
file=open(sentence,"r")
l=file.read(1024)
connectionSocket.send(l.encode())
print ("\nSent contents of" + sentence)
file.close()
connectionSocket.close()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python: ClientTCP
PS C:\Users\dell\Desktop\CN_LAB> & C:/Users/dell/AppData/Local/Programs/Python/Python312/python.exe c:/Users/dell/Desktop/CN_LAB/ClientTCP.py
Enter file name: ServerTCP.py
From Server:
from socket import *
serverName='127.0.0.1'
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
print ("The server is ready to receive")
connectionSocket, addr = serverSocket.accept()
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python 3.12.6 64-bit Go Live Continue

Observation:

24/12/24

Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the request file if present.

client.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("Enter file name");

clientSocket.send(sentence.encode())
fileContents = clientSocket.recv(1024).decode()
clientSocket.close()
```

server.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
print("The server is ready to receive")
while 1:
    connectionSocket addr = serverSocket.accept()
    sentence = connection.socket.recv(1024).decode().

    file = open(sentence, "r")
    l = file.read(1024)
```

```
connectionSocket.send(l.encode())
file.close()
connectionSocket.close()
```

OUTPUT:

The server is ready to receive

connection from ('127.0.0.01', 63844)
Enter the filename: filename.txt
From server: Hello World