Cycle II

LAB 15:

Aim : 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 requested file if present.

3. 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 requested file if present.

ServerTCP.py

```
from socket import *
serverName = "127.0.0.1" → loopback address
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('\n Sent contents of ' + sentence)
    file.close()
    connectionSocket.close()
```

ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
```

```
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\n Enter file name: ")

clientSocket.send(sentence.encode())
fileContents = clientSocket.recv(1024).decode()
print('\nFrom Server: \n')
print(fileContents)
clinetSocket.close()
```

Procedure:

- Create 2 IDLE instances and write client and server files.
- Run server first and then the client.

Output:-

Server Instance :-

The server is ready to receive

Client Instance :-

Enter file name : ServerTCP.py

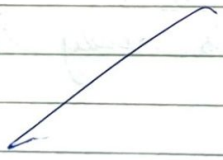
From Server :

The contents of ServerTCP.py is displayed here

Server Instance:-

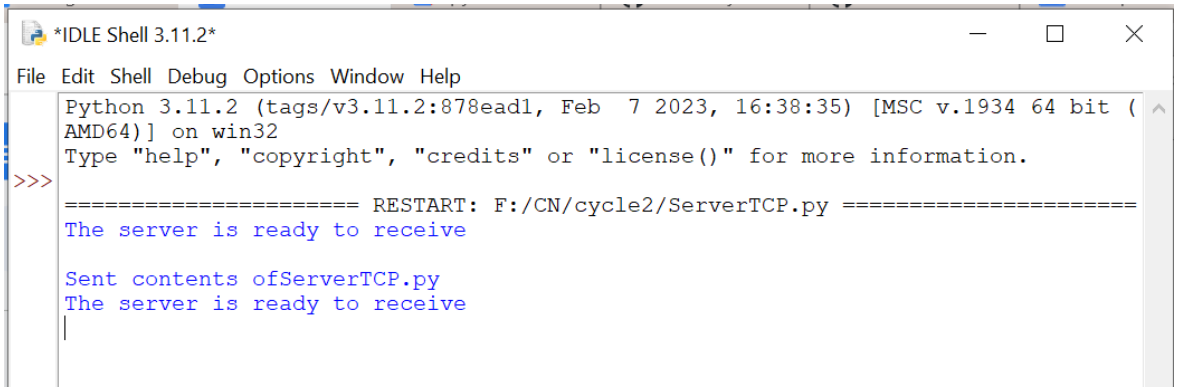
The server is ready to receive

Sent contents of ServerTCP.py
The server is ready to receive.



Output :

Server instance:



Client instance: