LAB 15

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.

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()
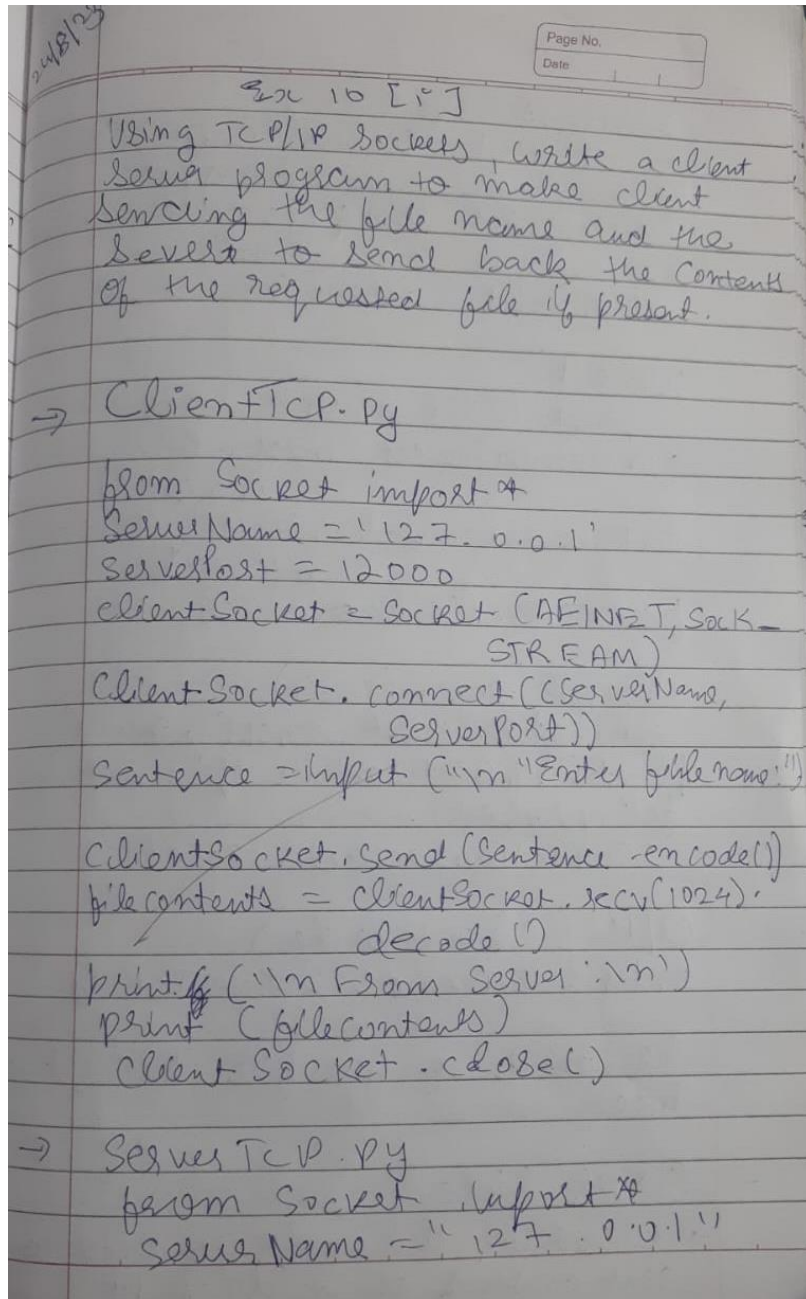
OUTPUT:

```
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()
    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()
```

```
The server is ready to receive

Sent contents ofServerTCP.py
The server is ready to receive
```

'

OBSERVATION:

Ex 10 [i]

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

→ ClientTCP.py

```
from Socket import *
ServerName = '127.0.0.1'
ServerPort = 12000
client Socket = Socket (AF INET, Sock-
                        STREAM)
Client Socket. connect ((ServerName,
                       ServerPort))
Sentence = input ("\n "Enter file name")

ClientSocket. send (Sentence -encode())
file contents = ClientSocket. recv(1024).
                        decode ()
printf ("\n From Server: \n')
print (file contents)
Client Socket .close()
```

→ ServerTCP.py
```
from Socket import *
Server Name = "127.0.0.1"
```

serverPort = 12000
Server Socket = socket (AF_INET, SOCK
                        STREAM)
Server Socket. bind ((serverName, ser..
                      Port))
Server Socket. listen (1)

While 1 :
     print ("The server is ready to recerva
connectionSocket, addr = Server Socket.
                       accept ()
Sentence = connection Socket. recv (1024
                           decode ()
file = open (Sentence, "8")

l = file. read (1024)
connectionSocket. send (l. encode ())
Print ("\n Sent contents of ' + Sentence)
file. close ()
connection Socket. close ()

O/P (client side)
Enter file name: ServerTCP. Py

The contents of the file is displayed
here.

→ Server Side
The server is ready to receive
Sent contents of ServerTCP. Py