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.

Went program:

client TCP. Py

from socket import *

server Name = 127.0.0.1

serverport = 12000

client socket = Socket (AE INET, SOCK_STREAM)

client Socket-connect ((server Name, server Port))

sentence = input ("In Enter gile name :")

client Socket send csentence encode (1)

filecontents = client Socket. recv (1024). decode()

print ("In from server: In").

print (filicontents)

client Socket. close ()

```
Server program:
```

SeTVENTCP. PY

from sockel import *

Server Name = 127, 0.0.1

Server Port = 12000

Server Socket = Socket (Af - INET, SOCK - STREAM)

server Socket. bind ((server Name, server Port))

server Socket. Listen (1)

while 1:

print ("The server is ready to receive")

connection Socket, addr: server Socket. accept()

sentence: connection Socket. recv (1024), decoder

file: open Csentence, "7")

1: file: read (1024)

connection Socker send (1.encode())

print ("In Sent Contents of '+ sentence)

jile dose ()

connection Socket dose()

output

server side:

The server is ready to receive

client side:

Enter tile name: SomerTCP.py

From server:

grom Socket import *

(serverace py is printed)

server side:

The server is ready to receive sent contents of servertcp. py
The server is ready to receive.

ATTOMATE AS 35

A first total to an all and the

```
File Edit Format Run Options Window Help
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 recieve")
    connectionSocket, addr =serverSocket.accept()
    sentence= connectionSocket.recv(1024).decode()
    file= open(sentence, "r")
    l=file.read(1024)
    connectionSocket.send(1.encode())
     print('\nSent contentes of' + sentence)
S
    file.close()
    connectionSocket.close()
```

*ServerTCP.py - C:/Users/Student/AppData/Local/Programs/Python/Py...

ClientTCP.py - C:/Users/Student/AppData/Local/Programs/Python/Python37-32/ClientT File Edit Format Run Options Window Help from socket import serverNamSSe = '127.0.0.1' serverPort = 12000

```
serverNamSS\[eq = '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()
```

File Edit Shell Options Window Help

Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte ^ 1)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/ServerTCP.p

Y
The server is ready to recieve

Sent contentes ofServerTCP.py
The server is ready to recieve

Python 3.7.3 Shell

6

_ 🗆 ×

```
File Edit Shell Debug Options Window Help
Type
              CODALIGHT '
                            crearra
                                     or treeuse() for more fulformactous
     merb '
>>>
RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Cl
Traceback (most recent call last):
  File "C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Clie
    clientSocket.connect((serverName, serverPort))
NameError: name 'clientSocket' is not defined
>>>
RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Cl
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 recieve")
    connectionSocket, addr =serverSocket.accept()
    sentence= connectionSocket.recv(1024).decode()
    file= open(sentence, "r")
    l=file.read(1024)
    connectionSocket.send(1.encode())
    print('\nSent contentes of' + sentence)
    file.close()
    connectionSocket.close()
```

Python 3.7.3 Shell

a