

## WEEK 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:

The image shows two separate Python IDLE shells running on the same machine. Both windows have the title "IDLE Shell 3.11.4".

**Left Window (Client Side):**

```

File Edit Shell Debug Options Window Help
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> ===== RESTART: C:\Users\Admin\Desktop\lhm2lcs065\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()
    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()

>>>

```

**Right Window (Server Side):**

```

File Edit Shell Debug Options Window Help
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> ===== RESTART: C:\Users\Admin\Desktop\lhm2lcs065\ServerTCP.py =====
The server is ready to receive
Sent contents ofServerTCP.py
The server is ready to receive

```

## OBSERVATION:

24-8-23

### Exp-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.

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 ("\\n From 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("Sent contents of " + sentence)  
file.close()  
connectionSocket.close()
```

### Output

Server is ready to receive

Client

Enter file name: ServerTCP.py

four client details will come