

### **Program 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.

#### 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()
print ('From Server:', filecontents)
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 = connectionSocket.recv(1024).decode()
    file=open(sentence,"r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
file.close()
connectionSocket.close()
```

#### Output

```
= RESTART: C:/Users/Dell/Desktop/5th sem/CN/LAB/Client.py
Enter file name: example.txt
From Server: Hello World!!
|
```

*Figure 83: Output of Client.py*

```
===== RESTART: C:/Users/Dell/Desktop/5th sem/CN/LAB/Server.py =====
The server is ready to receive
|
```

*Figure 84: Output of Server.py*

### Experiment no 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:

##### • 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()
print("From Server: ", fileContents)
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 = connectionSocket.recv(1024).decode()
    file = open(sentence, 'r')
    data = file.read(1024)
    connectionSocket.send(data.encode())
    file.close()
    connectionSocket.close()
```

Figure 85: Observation Book

### Output:

Server.py - The server ready to receive.

client.py - Enter file name: example.txt

From Server: Hello, world!

Q. 11  
3/12/21

Figure 86: Observation Book