09/23 Lab-10

3) Using TCP/IP sockets, write a client-server prog -ram to make client sending the file name and the server to send back the contents of the requested file if present.

Server:

- A server has a bind() method which binds it to a specific IP and port so that it can listen to incom -ing requests on that IP and port.
- A server has a listen() method which puts the server into listening mode. This allows the server to listen to incoming connections.
- And Server has an accept() and close() method. The accept method initiates a connection with the client and the close method closes the connec -tion with the client.

Step 1: We import socket which is necessary for client-server program.
Then we made a socket objects and reserved a port on our pc.

step 2: Bind the server to specified port. passing an empty string means that server can listen to incoming connections from other compu -ters as well. Here we pass 127.0.0.1 so, it will listen to only those calls made by within the local computer.

step 3: We put the server into listening mode. At last, we make while loop and start to accept all incoming connections and close connections after thank you message.

=> open a new file in a IDLE and write the following code and save as "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)
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()
```

Run the file server.py

o/p => The server is ready to recieve.

This shows that our server is working.

Client:

step1: make a socket object.

step2: Establish a connection with the server and lastly we will recieve data from the server and close the connection

=> open a new file in a IDLE and write the following code and save as "client.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()
```

Run the file client.py

```
O/P => Enter file name : server.py
```

when the client requests for server.py the server will send a file which is requested and the filecontents will be stored in 'filecontents' file and in the output it will be printed.

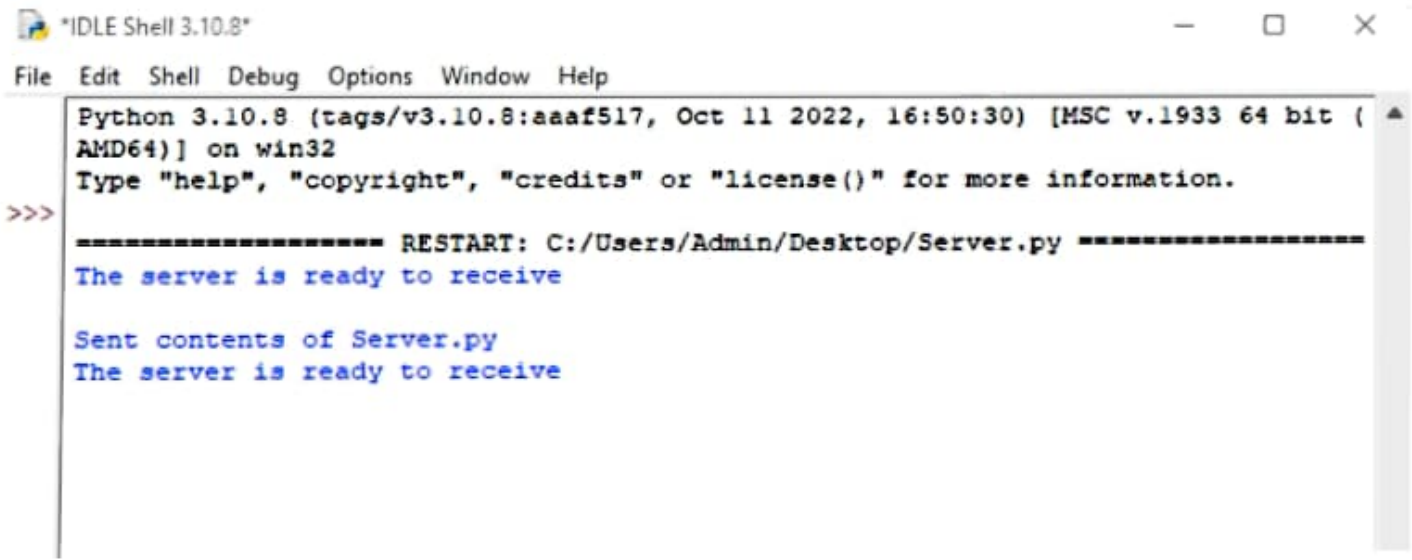
Server:

```
The server is ready to recieve
sent contents of server.py
The server is ready to recieve.
```

10/10

1/9/23

OUTPUT:


IDLE Shell 3.10.8
File Edit Shell Debug Options Window Help
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/Admin/Desktop/Server.py ==========
The server is ready to receive
Sent contents of Server.py
The server is ready to receive

IDLE Shell 3.10.8

File Edit Shell Debug Options Window Help

```
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/Admin/Desktop/Client.py ==========

Enter file name: Server.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()

>>>
```

4) Using UDP 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.

=> Server:

Here, Like in TCP/IP we create socket object and bind it to specified port and server will be continuously listening when the client sends request it responds accordingly.

ServerUDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket (AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to recieve")
while 1:
    sentence, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode ("utf-8")
    file = open (sentence, "r")
    con = file.read (2048)
    serverSocket.sendto (bytes (con, "utf-8"), clientAddress)
    print ('\n Sent contents of ', end = ' ')
    print (sentence)
    file.close()
```

When you run the above file, the output window O/p => The server is ready to recieve.

clientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\n Enter file name: ")
clientSocket.sendto(bytes(sentence, "utf-8"), (serverN
    -ame, serverPort))
fileContents, serverAddress = clientSocket.recvfrom
                                        (2048)
print ("\nReply from server:\n")
print (fileContents.decode("utf-8"))
clientSocket.close()
```

o/p =>

```
enter file name: serverUDP.py
Reply from server:
. . .
. . .
. . .
. . .
```

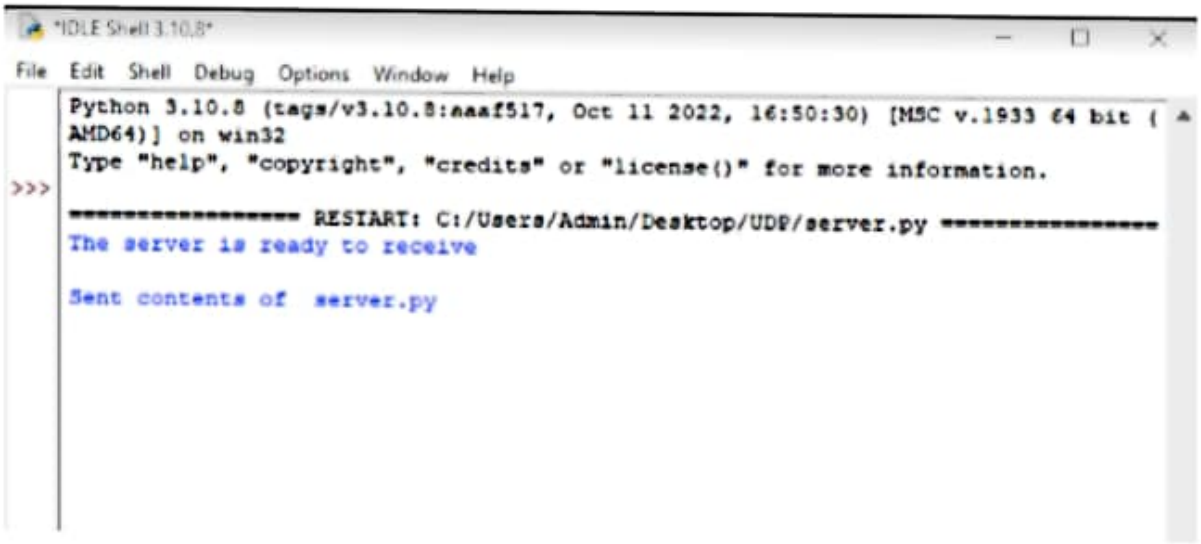
server:

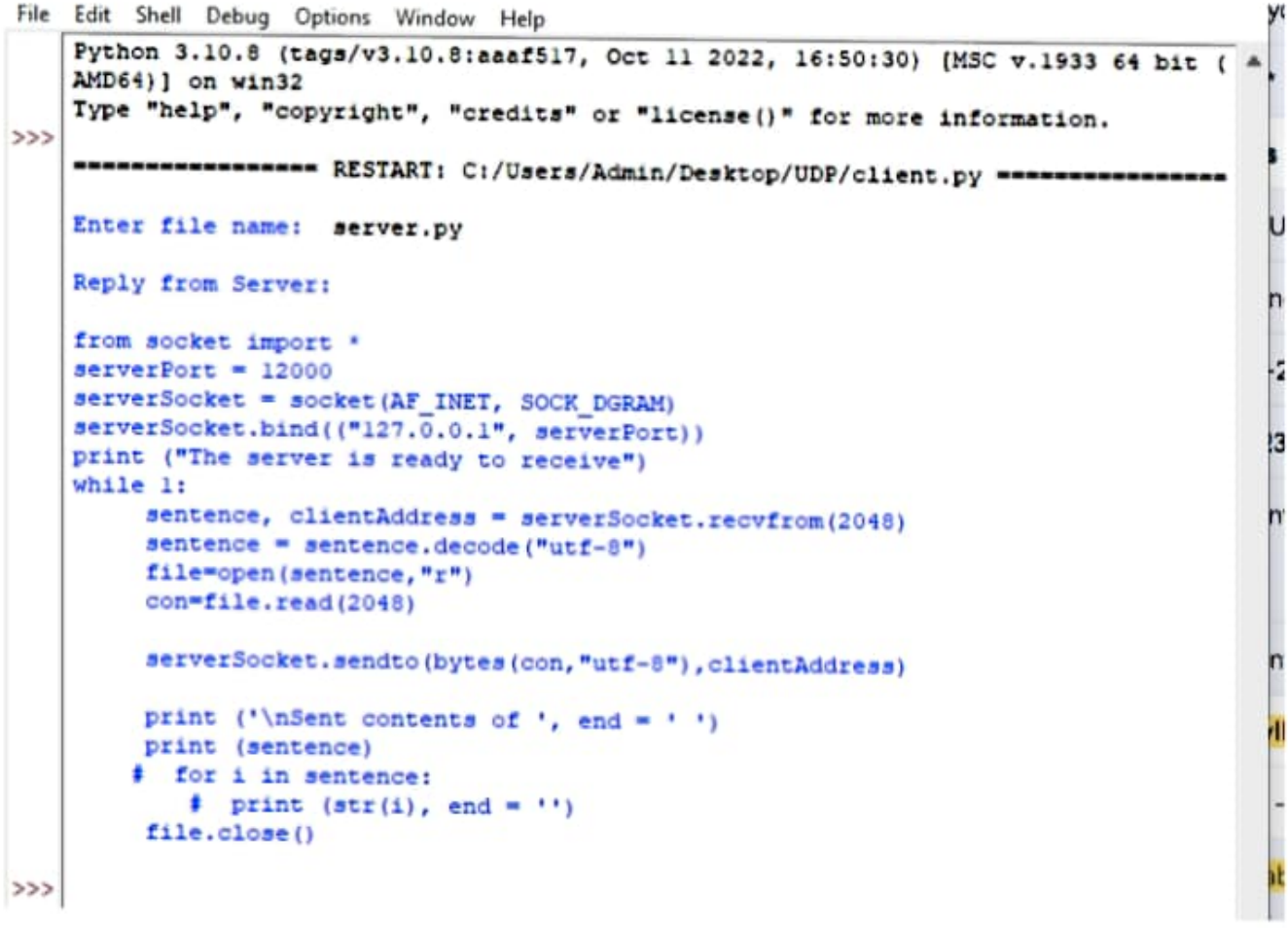
```
The server is ready to ~~receive~~
Sent contents of serverUDP.py
```

10/10

1/9/23

OUTPUT:


IDLE Shell 3.10.8
File Edit Shell Debug Options Window Help
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/Admin/Desktop/UDP/server.py ==========
The server is ready to receive
Sent contents of server.py


File Edit Shell Debug Options Window Help
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/Admin/Desktop/UDP/client.py ==========
Enter file name: server.py
Reply from Server:
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode("utf-8")
file=open(sentence,"r")
con=file.read(2048)
serverSocket.sendto(bytes(con,"utf-8"),clientAddress)
print ('\nSent contents of ', end = ' ')
print (sentence)
# for i in sentence:
# print (str(i), end = '')
file.close()
>>>