

WEEK 16

Using UDP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.

CODE:

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\nEnter file name: ")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName, serverPort))
filecontents,serverAddress = clientSocket.recvfrom(2048)
print ("\nReply from Server:\n")
print (filecontents.decode("utf-8"))
# for i in filecontents:
# print(str(i), end = " ")
clientSocket.close()
clientSocket.close()
```

ServerUDP.py

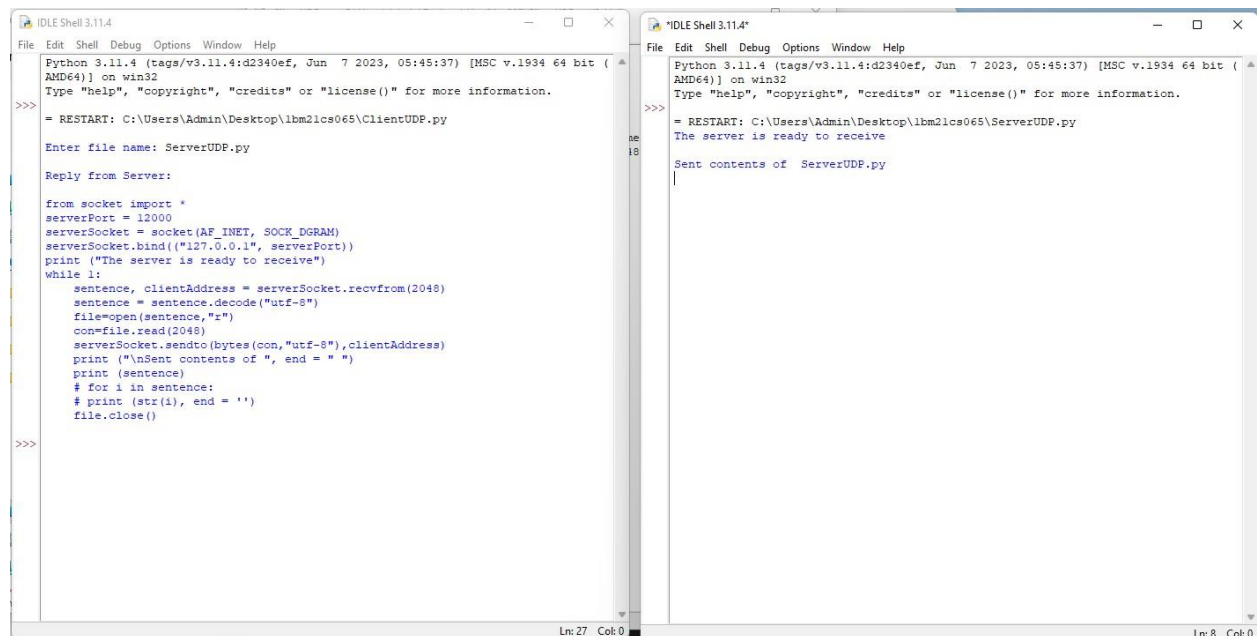
```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode("utf-8")
```

```

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()

```

OUTPUT:



The image shows two side-by-side screenshots of the IDLE Shell 3.11.4 window. The left window displays the Python code for a server that receives a file path via UDP and sends its contents back. The right window shows the output of the script, indicating that the server is ready to receive and has successfully sent the contents of the file 'ServerUDP.py'.

```

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\ClientUDP.py
Enter file name: ServerUDP.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()
>>>

```

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\ServerUDP.py
The server is ready to receive
Sent contents of ServerUDP.py
|

OBSERVATION:

Experiment - 16

Date / /
Page

File)

Aim: 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.

clientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("Enter filename: ")
clientSocket.sendto(bytes(sentence, 'utf-8'), (serverName,
serverPort))
fileContent, serverAddress = clientSocket.recvfrom(2048)
print("In Reply from server: ")
print(fileContent.decode('utf-8'))
clientSocket.close()
```

run here

server.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 receive")
while 1:
    sentence, clientAddress = serverSocket.recvfrom(
2048)
    sentence = sentence.decode('utf-8')
    file = open(sentence, "r")
    data = file.read(2048)
```

```
serverSocket.sendto(bytes(content), (IP, 8080), clientAddress)
print("I'm send content of ", end="")
print(fileName)
file.close()
```

output:

>>> Server.py

The server is ready to receive
sent content of ServerUDP.py.
The server is ready to receive.

>>> client.py

Enter file name: ServerUDP.py

Reply from server:

~~"Content of ServerUDP.py"~~

12/1/22