## **EXPERIMENT 16**

NAME: VARUN RAJ S USN: 1BM21CS264

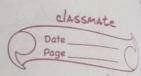
Using UDP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.

```
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:
    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()
>>>
   - kESTAKT: C:\USeTS\AGMITT\DeSKLOP\IX
  The server is ready to receive
  Sent contents of ServerUDP.py
```

## **OBSERVATION:**

	classmate a
	Dafe Page
4 8 2	
( )	and made on Japan and AB-16 by made
	entime samue acust & Diet & D
	AIM- ( C Distant) moja - Dif-
	using UDP sockets, write a client server program to make
	dient sinding the fill name and the sorver to sind back
	the contents of the suguested file if present.
	( windows) source
	CODE - DANAS AL 1 NO - 41-
	client UDP. by ( Das (1) at ) long it
	from socket import * ( ) seed when
	SorverName = "187.0.0.1"
	server Port = 19000
	client Socket = socket (AFINET, SOCK_DGRAM)
	sentina = input (" m Enta fill name");
	client saket . sendto (bytis (sentina, "utf-8"), (server name,
	sorverPort))
	file contents, Servor Address = client Socket. recvfrom (2048)
	print ("In Riply from server: 'm')
	print (fill con lints. duade ("utf-8"))
	# for e in filecontents:
	# print (str(i), end = ")
	client socket-close()
	client saket.cluse()
	ServerUDP.py
	from socket import *
	SurverPort = 12000
	8 orver Socket = socket (AF-INET, SOCK-DGRAM)
	sorverSocket. bind (("197.0.0.1", sorverPort))
	print ("The surver is ready to receive")
	while 1:



3entina, diene Addrus = 8021	en 30cket. recy from (2048)		
sentina = sentina. dubdi ("1	utf-8")		
Jill - oben (sentina, "r")	17119		
ion liv rend (2048)	une gau paveu		
Burver Socket. Bend to (bytes (un, "Utf-8"), client Address)			
print ( 'In sent contents of,	end= 11) may self		
print (sintina)			
# for i in sentence:	- 3000		
# print (str(i), end="	) pg gau troil		
Jul. UBSeC) + trad	ni 20108 nasi		
	1 3molysyrs		
	SoverPort & Boo		
BRELL (AF INTET, SCHILDERAM)	a = 130000 \$10000		
( me cotto fill octor)	tudio mannos		
ose), (12 fau" Dadas) wyd) oth			
	((Influence)		
g) most your . Herostrade - courbb Asovio	E dannapit		
( ov : covide most)			
links death ( orf 8.)			
(fatha)			
( to bin ( ) ide	1 600		
	- 199008 ta m		
(1030(1)	Survey Family		
	and 900 marine		
	The state of the s		