24/8/23

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.

Python Program -

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 the 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("\nSend contents of"+ sentence)
file.close()
connectionSocket.close()
```

Result:

Client window:
Enter the file name : serverTCP.py
Contents of the file are displayed

Server window:
The server is ready to receive.

Sent contents of serverTCP.py
The server is ready to receive.

10/10

N
28/8/23

24/8/23

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 requited file if present

Python program-

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()
```

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() .
```

Result -

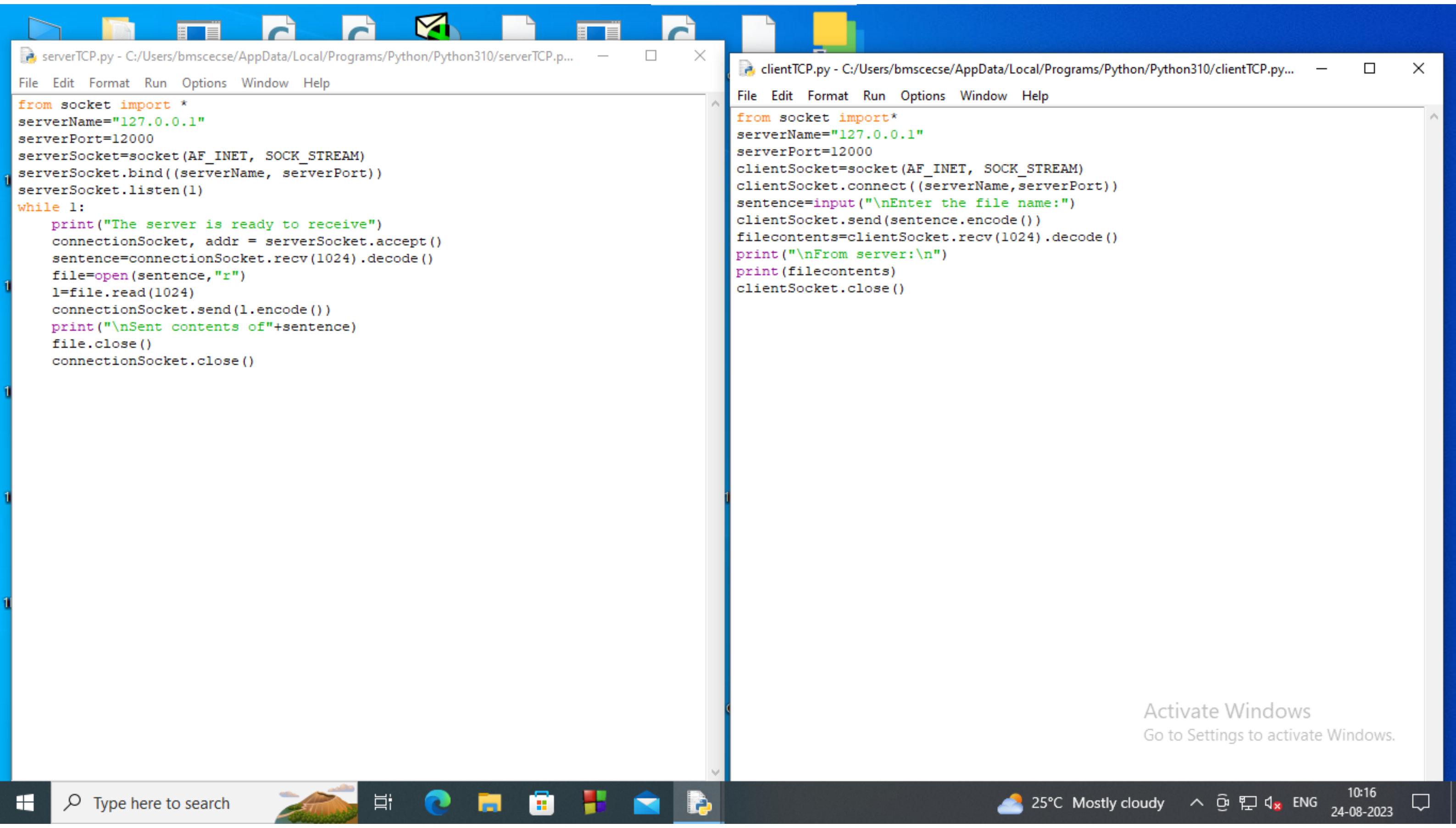
Client window:
Enter the file name: serverTCP.py
Contents of the file are displayed

Server window:
The server is ready to receive

Sent contents of serverTCP.py
The server is ready to receive

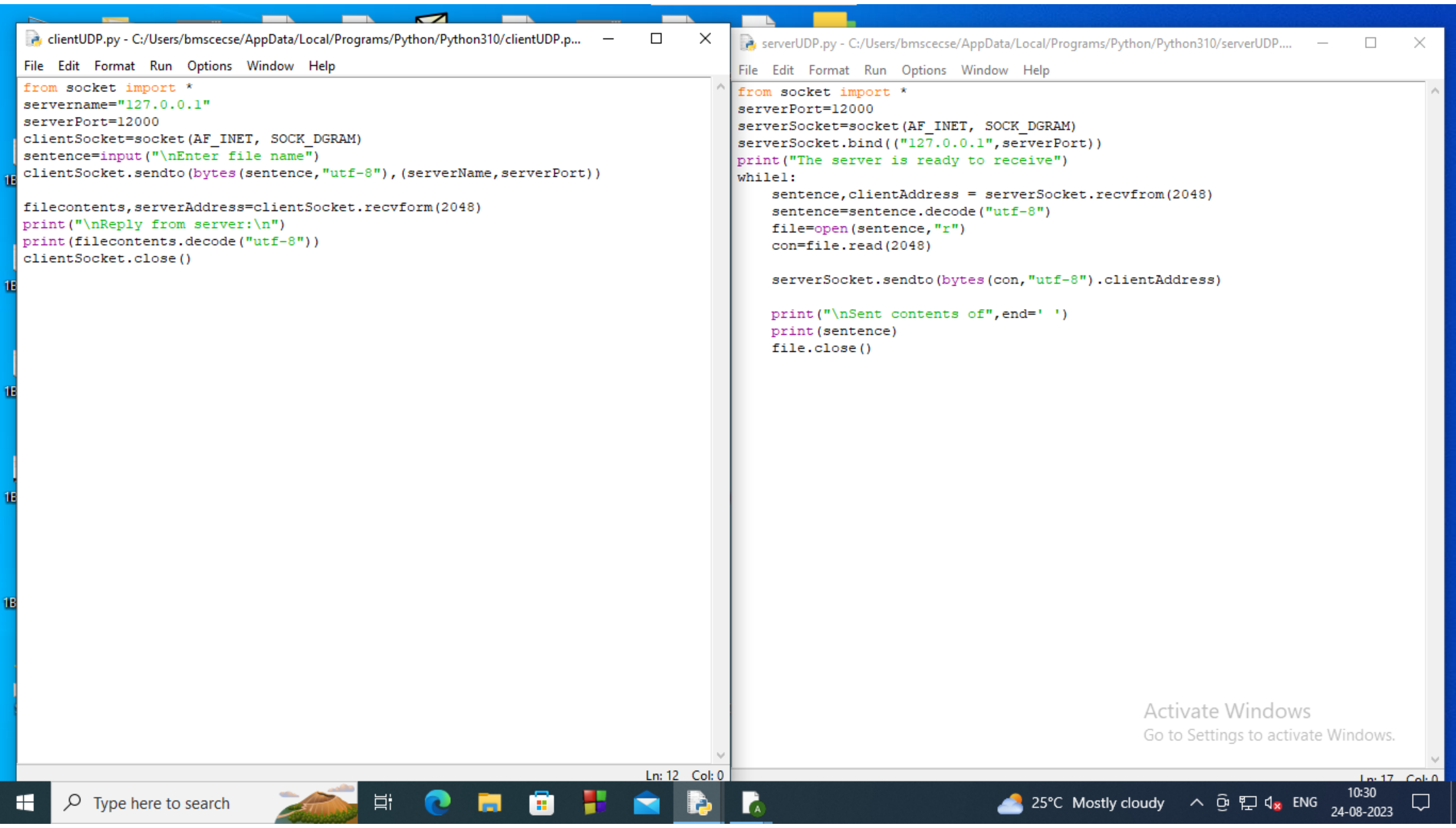
10/10

28/3/23


serverTCP.py - C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverTCP.p...
File Edit Format Run Options Window Help
clientTCP.py - C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientTCP.py...
File Edit Format Run Options Window Help
Activate Windows
Go to Settings to activate Windows.
Type here to search
25°C Mostly cloudy
ENG
10:16
24-08-2023

```
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()
```

```
from socket import*
serverName="127.0.0.1"
serverPort=12000
clientSocket=socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence=input("\nEnter the file name:")
clientSocket.send(sentence.encode())
filecontents=clientSocket.recv(1024).decode()
print("\nFrom server:\n")
print(filecontents)
clientSocket.close()
```

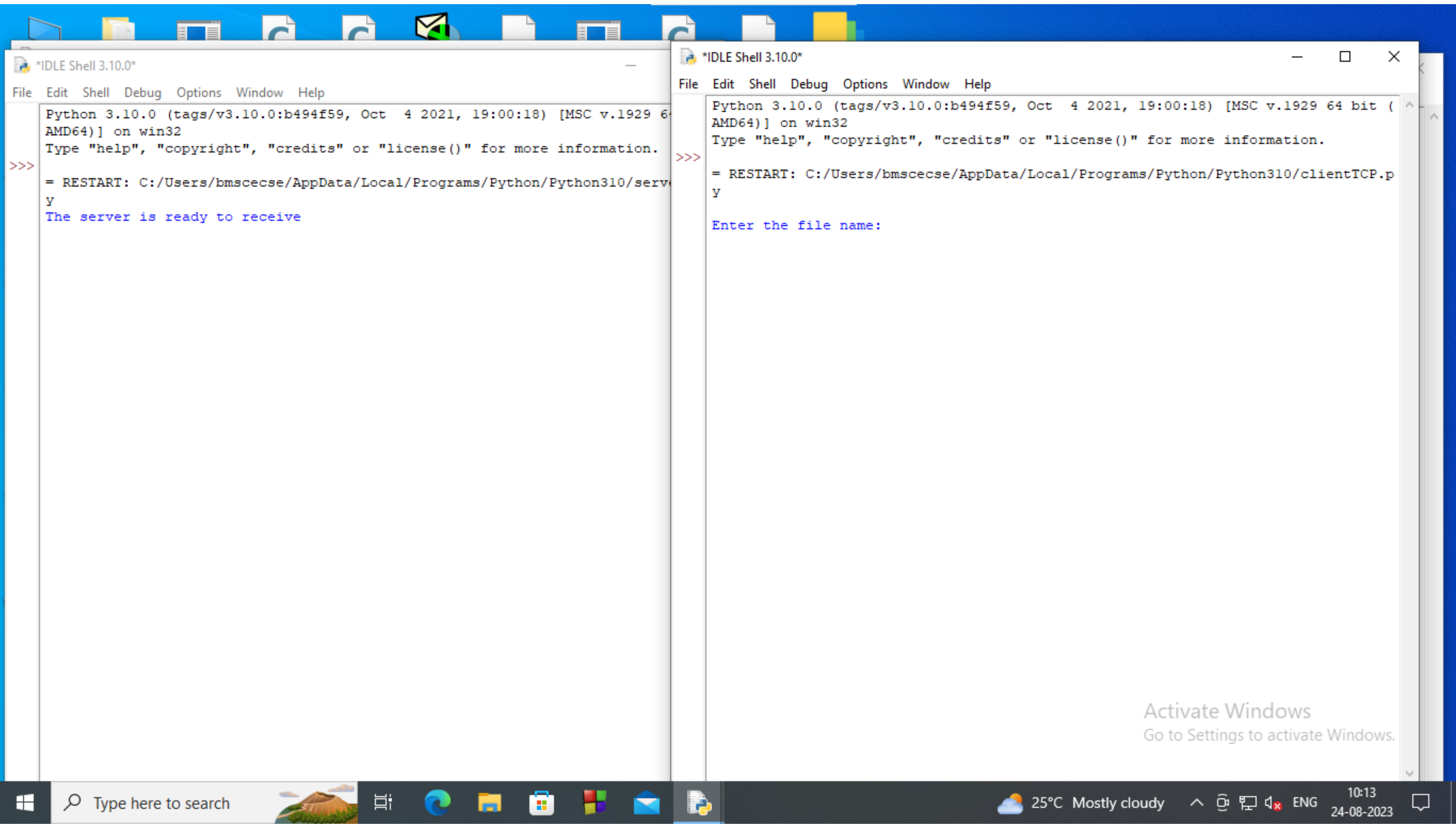


clientUDP.py - C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientUDP.p...
File Edit Format Run Options Window Help

```
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.recvform(2048)
print("\nReply from server:\n")
print(filecontents.decode("utf-8"))
clientSocket.close()
```

Ln: 12 Col: 0
serverUDP.py - C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverUDP....
File Edit Format Run Options Window Help

```
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")
while1:
    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)
    file.close()
```

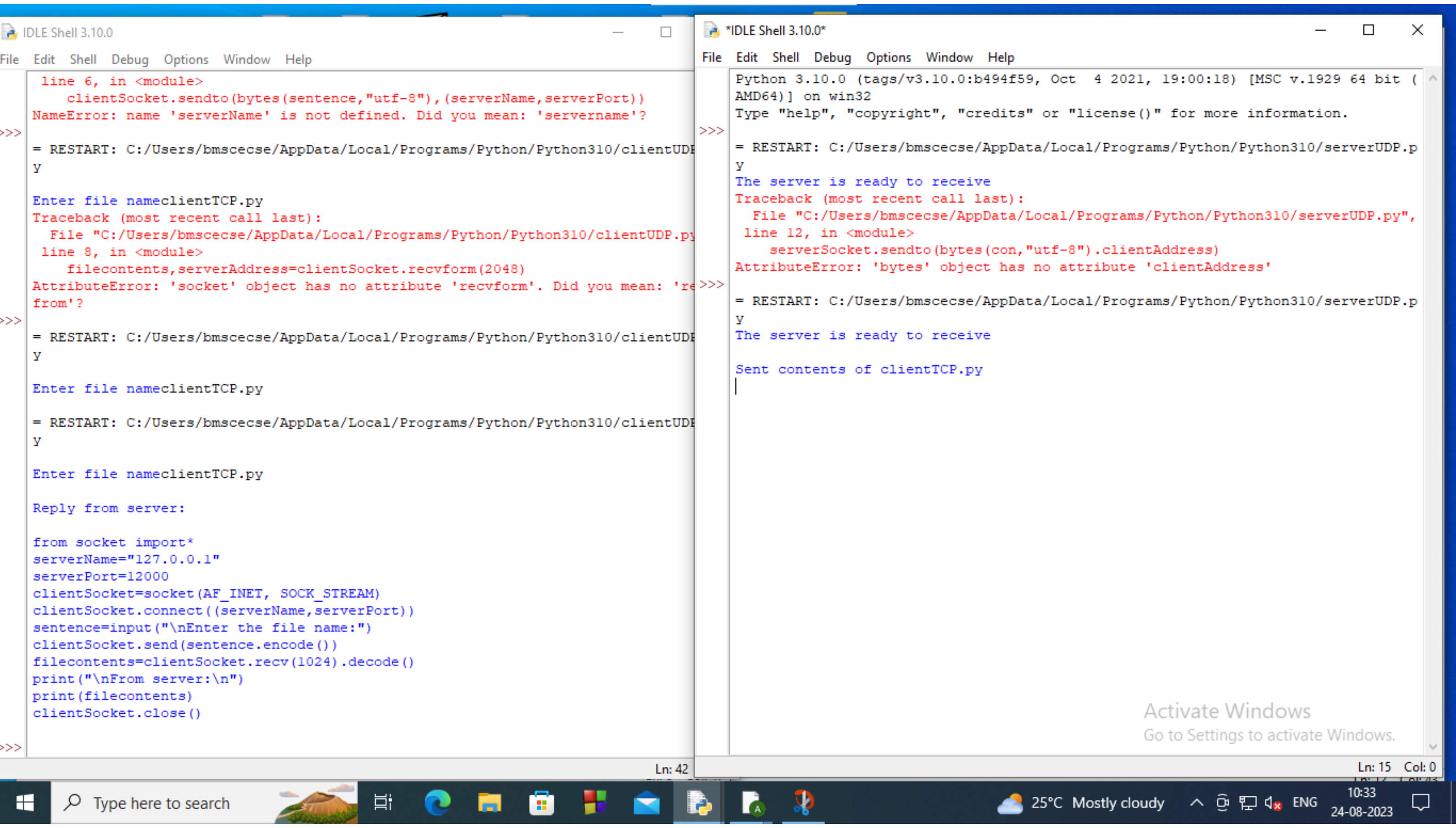


IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help

```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 6
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serv
y
The server is ready to receive
```

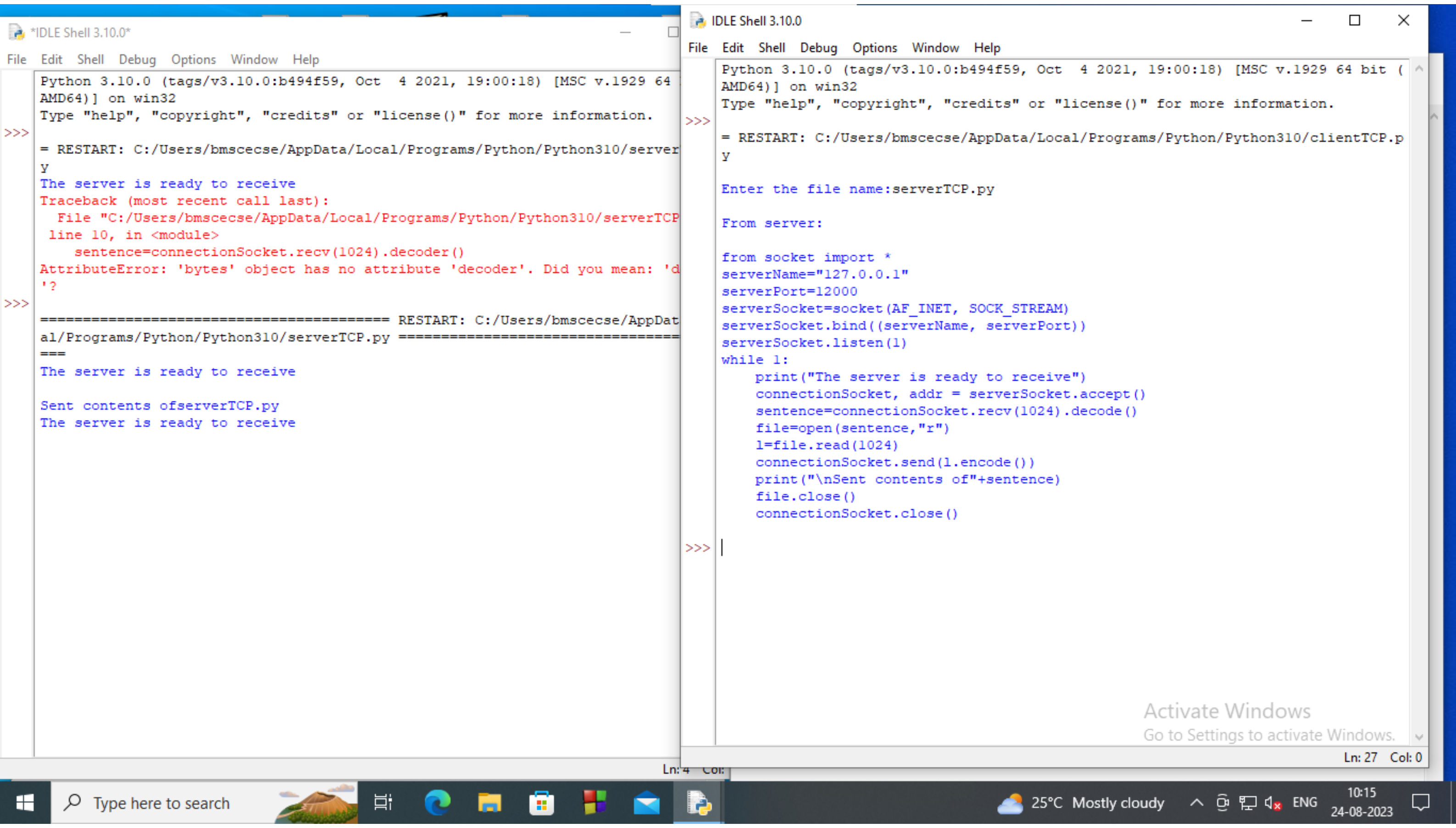
IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help

```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientTCP.p
y

Enter the file name:
```



```
 line 6, in <module>
    clientSocket.sendto(bytes(sentence,"utf-8"),(serverName,serverPort))
NameError: name 'serverName' is not defined. Did you mean: 'servername'?
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientUDE
y

Enter file nameclientTCP.py
Traceback (most recent call last):
  File "C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientUDP.py
 line 8, in <module>
    filecontents,serverAddress=clientSocket.recvform(2048)
AttributeError: 'socket' object has no attribute 'recvform'. Did you mean: 're
from'?
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientUDE
y

Enter file nameclientTCP.py

= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientUDE
y

Enter file nameclientTCP.py

Reply from server:

from socket import*
serverName="127.0.0.1"
serverPort=12000
clientSocket=socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence=input("\nEnter the file name:")
clientSocket.send(sentence.encode())
filecontents=clientSocket.recv(1024).decode()
print("\nFrom server:\n")
print(filecontents)
clientSocket.close()

>>>
```

```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverUDP.p
y
The server is ready to receive
Traceback (most recent call last):
  File "C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverUDP.py",
 line 12, in <module>
    serverSocket.sendto(bytes(con,"utf-8").clientAddress)
AttributeError: 'bytes' object has no attribute 'clientAddress'
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverUDP.p
y
The server is ready to receive

Sent contents of clientTCP.py
```



```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 64
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/server
y
The server is ready to receive
Traceback (most recent call last):
  File "C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/serverTCP
 line 10, in <module>
    sentence=connectionSocket.recv(1024).decoder()
AttributeError: 'bytes' object has no attribute 'decoder'. Did you mean: 'd
'?
>>>
========================================= RESTART: C:/Users/bmscecse/AppDat
al/Programs/Python/Python310/serverTCP.py =================================
===
The server is ready to receive

Sent contents ofserverTCP.py
The server is ready to receive
```

```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct  4 2021, 19:00:18) [MSC v.1929 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/bmscecse/AppData/Local/Programs/Python/Python310/clientTCP.p
y

Enter the 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()

>>> |
```