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:

```
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 file name: ")
clientSocket.send(sentence.encode())
filecontents: clientSocket.recv(1024).decoder()
print("\nFrom Server:\n")
print(filecontents)
clientSocket.close()
```

```
ServerTCP.py
from socket import*
serverName = "127.0.0.1" -> loopback address
serverPort=12000
serverSocket= socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while(1):
    print("The server is ready to recieve")
    connectionSocket, addr = ServerSocket.accept()
    Sentence = connectionSocket.recv(1024).decode()
      file= open(sentence,"r")
        l = file.read(1024)
      connectionSocket.send(l.encoder())
      print('\n Sent contents of '+ sentence)
      file.close()
      connectionSocket.close()
```

OUTPUT:

server side:
The server is Ready to receive

client side:
Enter file name: serverTCP.py
from server:
forom socket import*
⋮

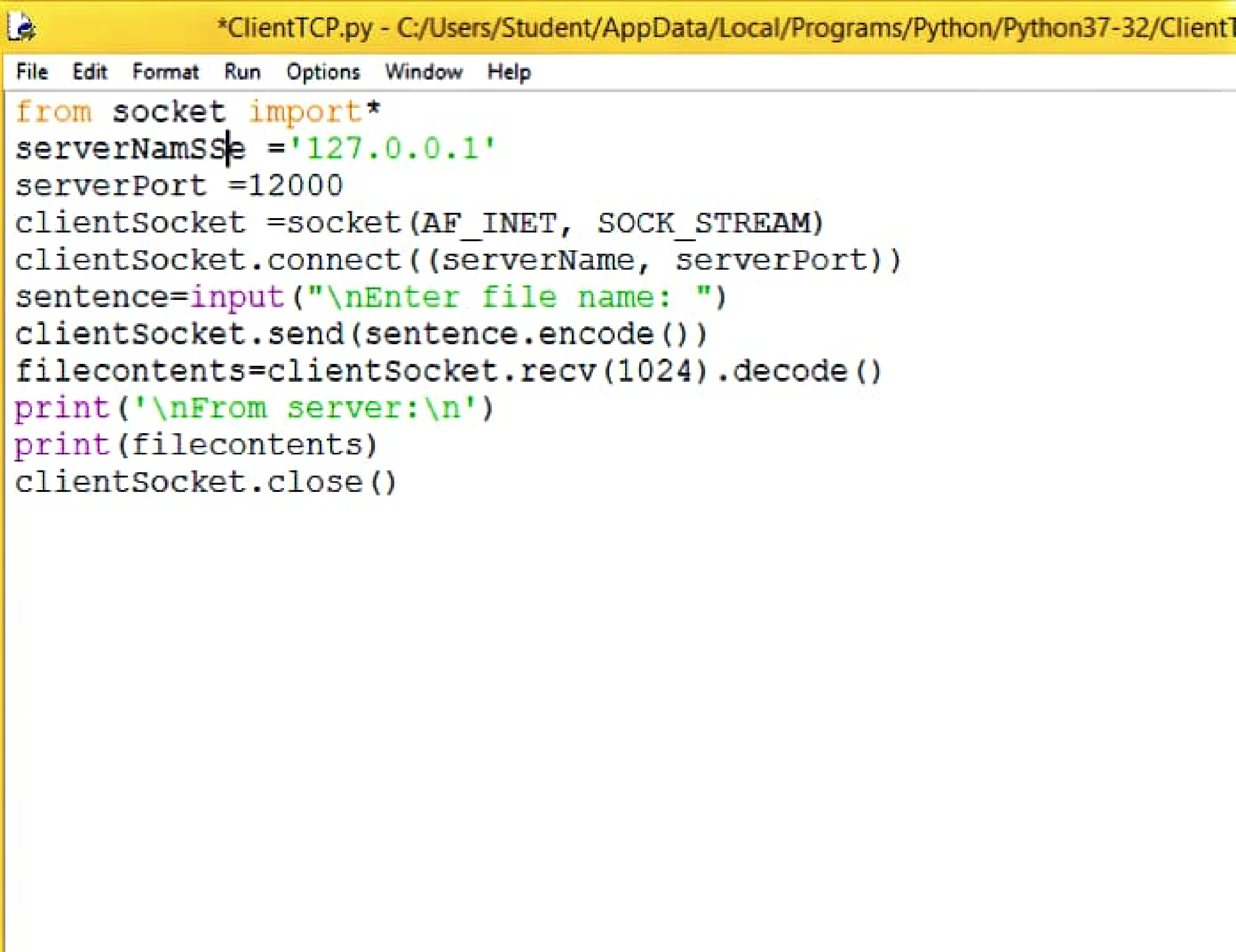
(codes under serverTCP.py is printed as written above)

Server side:
The server is ready to recive

sent contents of serverTCP.py
The server is ready to recive

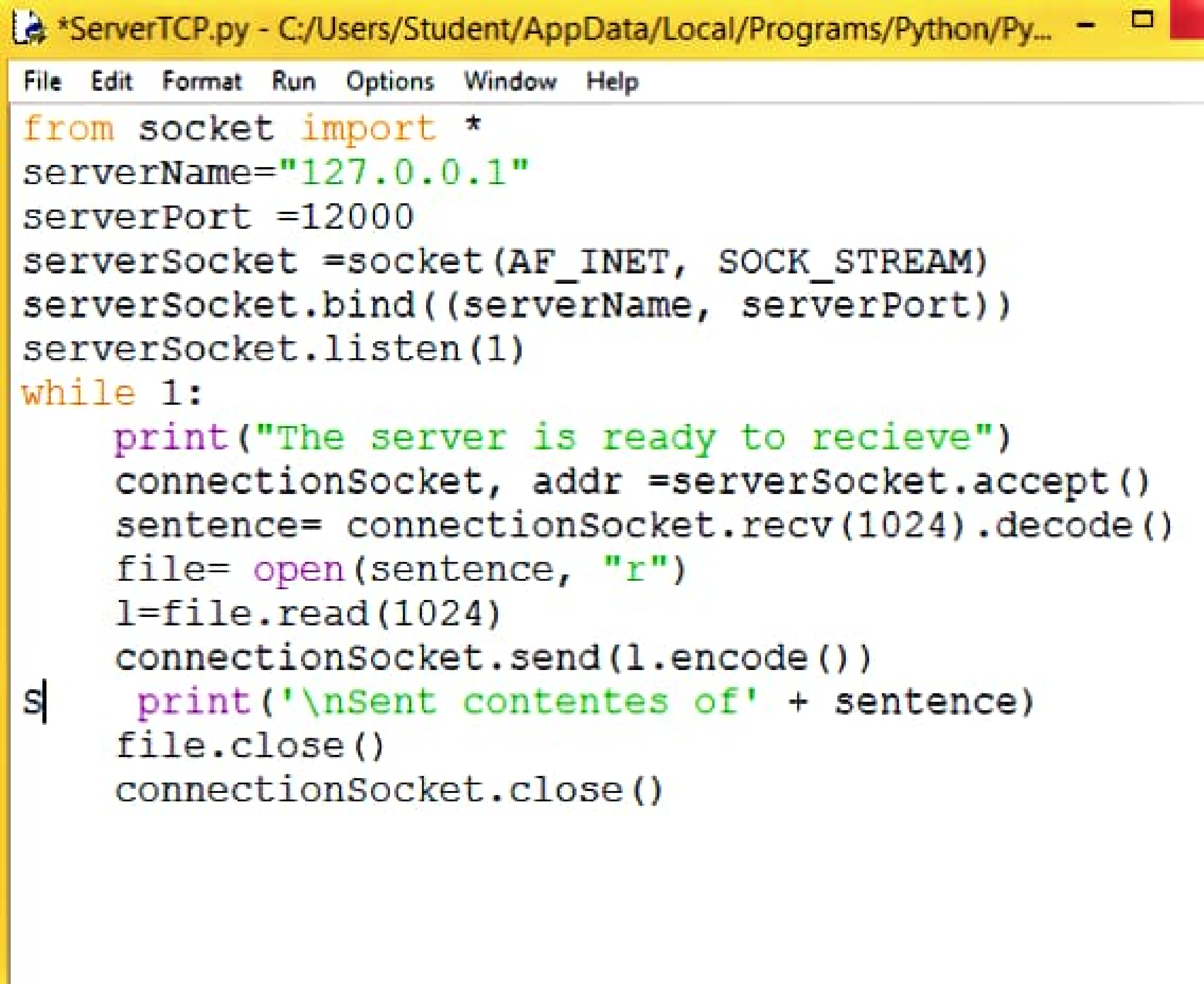
10/10

N
28/3/23

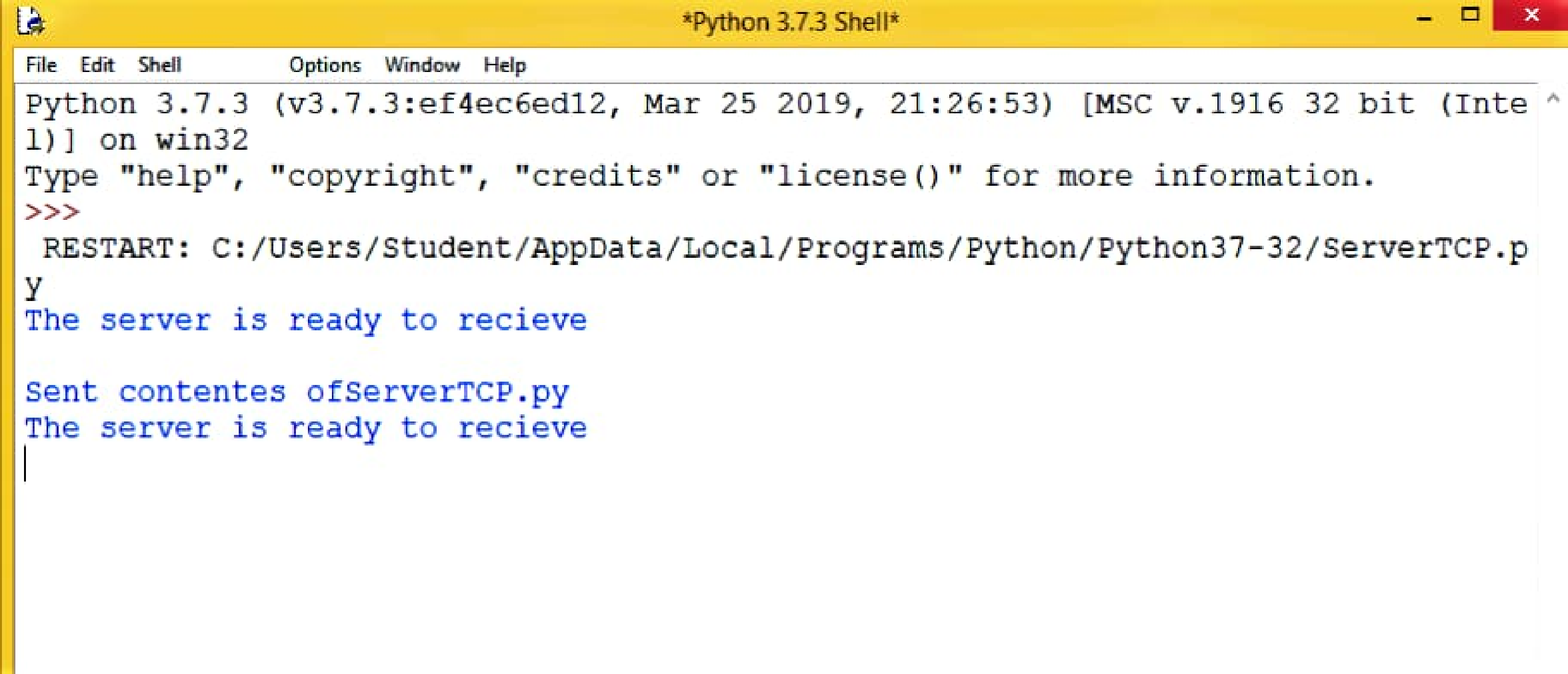


*ClientTCP.py - C:/Users/Student/AppData/Local/Programs/Python/Python37-32/ClientT
File Edit Format Run Options Window Help

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

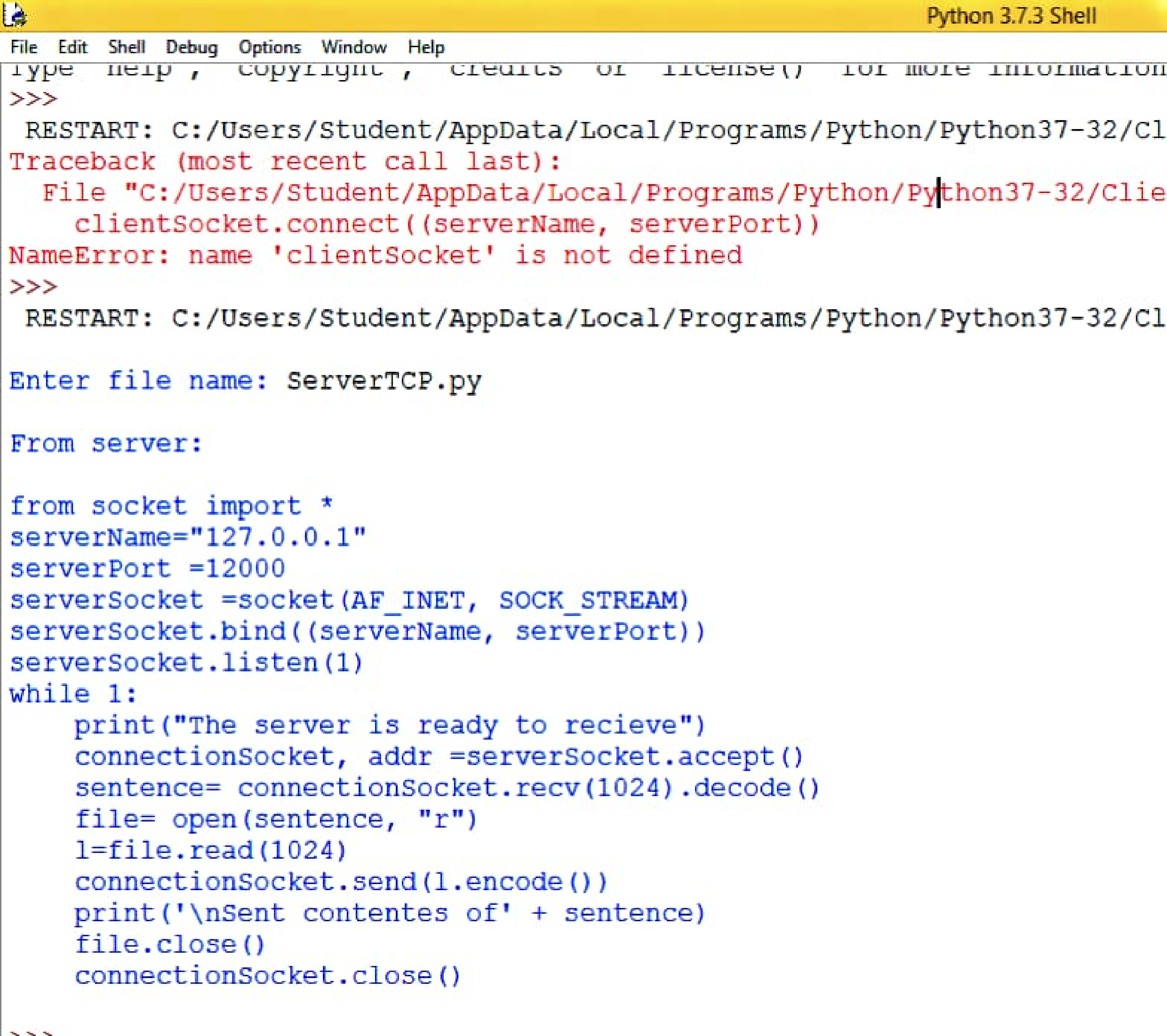

*ServerTCP.py - C:/Users/Student/AppData/Local/Programs/Python/Py...
File
Edit
Format
Run
Options
Window
Help

```
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 recieve")
    connectionSocket, addr =serverSocket.accept()
    sentence= connectionSocket.recv(1024).decode()
    file= open(sentence, "r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
S    print('\nSent contentes of' + sentence)
    file.close()
    connectionSocket.close()
```



```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte
l)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
 RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/ServerTCP.p
y
The server is ready to recieve

Sent contentes ofServerTCP.py
The server is ready to recieve
```



```
Type "help", "copyright", "credits" or "license()" for more information
>>>
 RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Cl
Traceback (most recent call last):
  File "C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Clie
    clientSocket.connect((serverName, serverPort))
NameError: name 'clientSocket' is not defined
>>>
 RESTART: C:/Users/Student/AppData/Local/Programs/Python/Python37-32/Cl

Enter 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 recieve")
    connectionSocket, addr =serverSocket.accept()
    sentence= connectionSocket.recv(1024).decode()
    file= open(sentence, "r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
    print('\nSent contentes of' + sentence)
    file.close()
    connectionSocket.close()

>>>
```