



NAME :: IBRAR BABAR

ROLL NO. :: 19P-0104

COMPUTER NETWORK

LAB # 7

CLINT & SERVER COMMUNICATION

Socket Programming

```
server.py - Assign#2_Sol - Visual Studio Code

EXPLORER
  ASSIGN#2_SOL
    clint.py
    Final SS.jpeg
    server.py

clint.py > ...
1 # This file will be used for recieving files over socket
2 import os
3 import socket
4 import time
5
6 host = input("Host Name: ")
7 sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8
9 # Trying to connect to socket.
10 try:
11     sock.connect((host, 22222))
12     print("Connected Successfully")
13 except:
14     print("Unable to connect")
15     exit(0)
16
17 # Send file details.
18 file_name = sock.recv(100).decode()
19 file_size = sock.recv(100).decode()
20
21 # Opening and reading file.
22 with open("./rec/" + file_name, "wb") as file:
23     c = 0
24     # Starting the time capture.
25     start_time = time.time()
26
27     # Running the loop while file is recieved.
28     while c <= int(file_size):
29         data = sock.recv(1024)
30         if not (data):
31             break
32         file.write(data)
33         c += len(data)

server.py > ...
1 # This file is used for sending the file over socket
2 import os
3 import socket
4 import time
5
6 # Creating a socket.
7 sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8 sock.bind((socket.gethostname(), 22222))
9 sock.listen(5)
10
11 print("Host Name: ", sock.getsockname())
12
13 # Accepting the connection.
14 client, addr = sock.accept()
15
16 # Getting file details.
17 file_name = input("File Name:")
18 file_size = os.path.getsize(file_name)
19
20 # Sending file_name and detail.
21 client.send(file_name.encode())
22 client.send(str(file_size).encode())
23
24 # Opening file and sending data.
25 with open(file_name, "rb") as file:
26     c = 0
27     # Starting the time capture.
28     start_time = time.time()
29
30     # Running loop while c != file_size.
31     while c <= file_size:
32         data = file.read(1024)
33         if not (data):
```

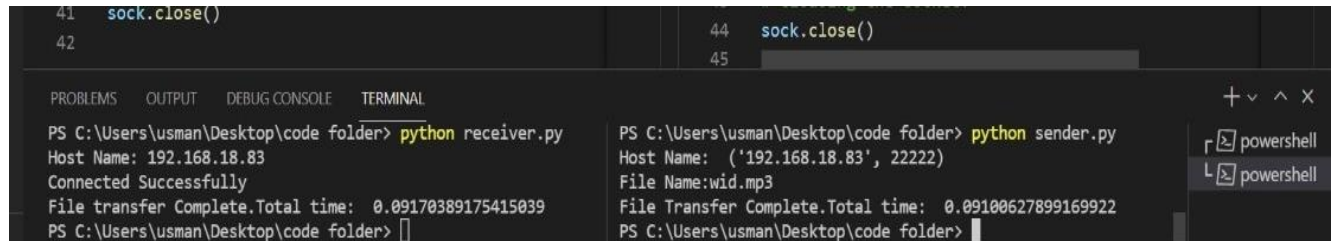
```
server.py - Assign#2_Sol - Visual Studio Code

EXPLORER
  ASSIGN#2_SOL
    1.PNG
    clint.py
    Final SS.jpeg
    server.py

clint.py > ...
34
35 # Ending the time capture.
36 end_time = time.time()
37
38 print("File transfer Complete.Total time: ", end_time - start_time)
39
40 # Closing the socket.
41 sock.close()
42

server.py > ...
37
38 # Ending the time capture.
39 end_time = time.time()
40
41 print("File Transfer Complete.Total time: ", end_time - start_time)
42
43 # Closing the socket.
44 sock.close()
45
```

OUTPUT



```
41 sock.close()
42

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\usman\Desktop\code folder> python receiver.py
Host Name: 192.168.18.83
Connected Successfully
File transfer Complete.Total time: 0.09170389175415039
PS C:\Users\usman\Desktop\code folder>

44 sock.close()
45

PS C:\Users\usman\Desktop\code folder> python sender.py
Host Name: ('192.168.18.83', 22222)
File Name:wid.mp3
File Transfer Complete.Total time: 0.09100627899169922
PS C:\Users\usman\Desktop\code folder>
```

Clint.py

```
import os

import socket

import time


host = input("Host Name: ")

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)


# Trying to connect to socket.
try:
    sock.connect((host, 22222))
    print("Connected Successfully")
except:
    print("Unable to connect")
    exit(0)


# Send file details.

file_name = sock.recv(100).decode()

file_size = sock.recv(100).decode()
```

```
# Opening and reading file.
with open("./rec/" + file_name, "wb") as file:

    c = 0

    # Starting the time capture.
    start_time = time.time()

    # Running the loop while file is recieved.
    while c <= int(file_size):
        data = sock.recv(1024)
        if not (data):
            break
        file.write(data)
        c += len(data)

    # Ending the time capture.
    end_time = time.time()

print("File transfer Complete.Total time: ", end_time - start_time)

# Closing the socket.
sock.close()
```

Server.py

```
# This file is used for sending the file over socket

import os
import socket
import time


# Creating a socket.
sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.bind((socket.gethostname(), 22222))
sock.listen(5)


print("Host Name: ", sock.getsockname())


# Accepting the connection.
client, addr = sock.accept()


# Getting file details.
file_name = input("File Name:")
file_size = os.path.getsize(file_name)


# Sending file_name and detail.
client.send(file_name.encode())
client.send(str(file_size).encode())


# Opening file and sending data.
with open(file_name, "rb") as file:
    c = 0

    # Starting the time capture.
```

```
start_time = time.time()

# Running loop while c != file_size.
while c <= file_size:
    data = file.read(1024)
    if not (data):
        break
    client.sendall(data)
    c += len(data)

# Ending the time capture.
end_time = time.time()

print("File Transfer Complete.Total time: ", end_time - start_time)

# Cloasing the socket.
sock.close()
```

```
20
21 # Opening and reading file.
22 with open("./rec/" + file_name, "wb") as file:
23     c = 0
24     # Starting the time capture.
25     start_time = time.time()
26
27     # Running the loop while file is recieved.
28     while c <= int(file_size):
29         data = sock.recv(1024)
30         if not (data):
31             break
32         file.write(data)
33         c += len(data)
34
35     # Ending the time capture.
36     end_time = time.time()
37
38     print("File transfer Complete.Total time: ", end_time)
39
40 # Closing the socket.
41 sock.close()
42
```

```
22 client.send(str(file_size).encode())
23
24 # Opening file and sending data.
25 with open(file_name, "rb") as file:
26     c = 0
27     # Starting the time capture.
28     start_time = time.time()
29
30     # Running loop while c != file_size.
31 while c <= file_size:
32     data = file.read(1024)
33     if not (data):
34         break
35     client.sendall(data)
36     c += len(data)
37
38     # Ending the time capture.
39     end_time = time.time()
40
41     print("File Transfer Complete.Total time: ", end_time)
42
43 # Cloasing the socket.
44 sock.close()
45
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\usman\Desktop\code folder> python receiver.py
Host Name: 192.168.18.83
Connected Successfully
File transfer Complete.Total time: 0.09170389175415039
PS C:\Users\usman\Desktop\code folder>

PS C:\Users\usman\Desktop\code folder> python sender.py
Host Name: ('192.168.18.83', 22222)
File Name:wid.mp3
File Transfer Complete.Total time: 0.09100627899169922
PS C:\Users\usman\Desktop\code folder>

```
41 sock.close()
42
```

```
44 sock.close()
45
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\usman\Desktop\code folder> python receiver.py
Host Name: 192.168.18.83
Connected Successfully
File transfer Complete.Total time: 0.09170389175415039
PS C:\Users\usman\Desktop\code folder>

PS C:\Users\usman\Desktop\code folder> python sender.py
Host Name: ('192.168.18.83', 22222)
File Name:wid.mp3
File Transfer Complete.Total time: 0.09100627899169922
PS C:\Users\usman\Desktop\code folder>

[NOTE]

There are two files, one is for clint and other for server.

I use python Language for clint and server communication.

I use mp3 file to send across **server** to clint and from **clint** to server.

I just use 1 code for both server & Clint communication and for sending file between them.

If you wanted run the code, simply download the attachment files and then open it on VisualStudioCode or whatever python compiler that suits you. And finally type **python server.py** in Server.py terminal & type python clint.py in clint.py terminal. And see the output.

In case of sending file. Please have a look to the attached screenshots.