



OCTOBER 2021

“LAB -1 REPORT”

Submitted for the course

Of

DISTRIBUTED SYSTEMS

Under the guidance of

Dr. CHANCE R EARY

Submitted by

**SHIVANI MANOJKUMAR
PANCHIWALA**

– **1001982478**

IMPLEMENTATION DETAILS

I implement this project into python language and used PyCharm for the Programming.

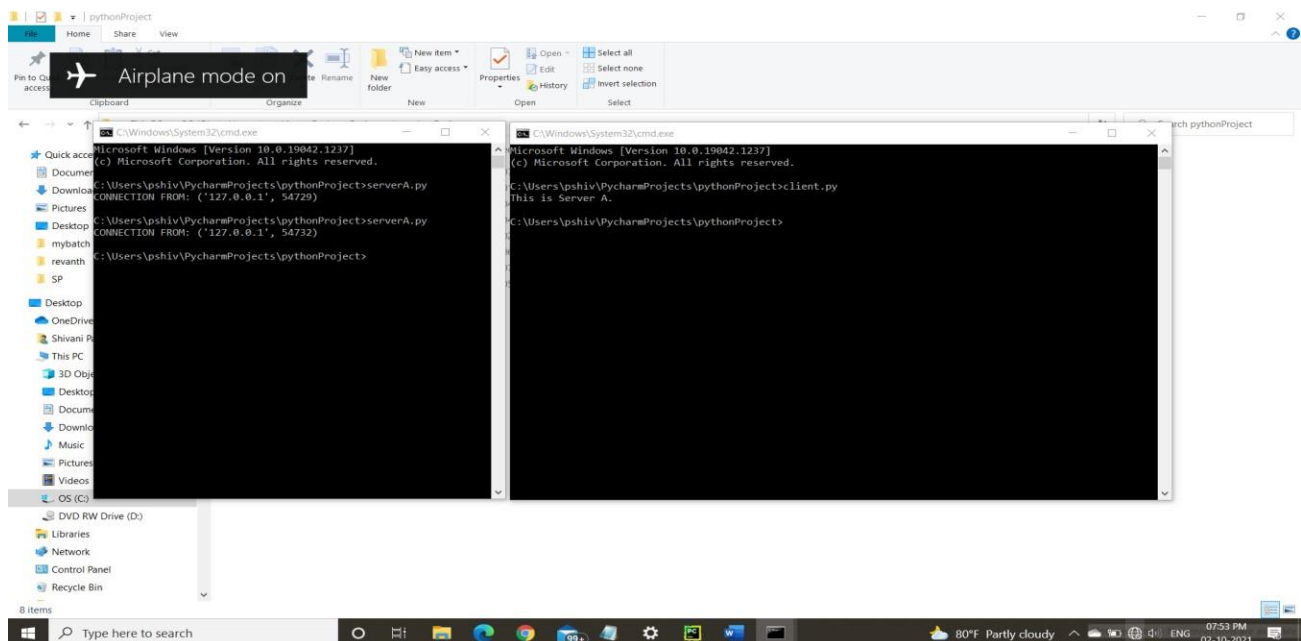
In this project, first I was trying to simple basic establish the connection between client to server. For establish the connection I import the socket and OS module. And then write the server code and client code for as per below from the given reference.
<https://stackoverflow.com/questions/47539028/transfer-contents-of-a-folder-over-network-by-python>

```
# SERVER CODE
sock = socket()          # Build Socket Object
sock.bind(('', 5000))    # bind the socket with server and port number
sock.listen(2)           #allow maximum 2 connection to the socket
client, addr = sock.accept() #wait till a client accept and establish the
                             connection
print("CONNECTION FROM:", str(addr)) # display client address
```

```
# Client Code
# Make a directory for the received files.
os.makedirs('client',exist_ok=True)

sock = socket()          # Build Socket Object
sock.connect(('localhost',5000)) #bind host address and port together and
connect to the server A
with sock,sock.makefile('rb') as clientfile:
    while True:
        raw = clientfile.readline()    # read the file
        if not raw: break # no more files, server closed connection.
        print(raw.decode()) # print and decode the serverA file
        break # no more files, server closed connection
```

Then I got the output like this.



Then I write the same client & server code for server B and establish the connection. After, Establish the connection between Server B to Server A to Client, In Server A, I give the path of directory of my folder and list out the files from directory.

```
dir_name = 'C:\\Users\\pshiv\\PycharmProjects\\pythonProject\\MP' # path of the folder
arr = os.listdir(dir_name) # list out the files from directory
```

After that I create the for loop and join the file path with directory and display the last modification of date of file from the given reference.

<https://thispointer.com/python-get-list-of-files-in-directory-sorted-by-date-and-time/>

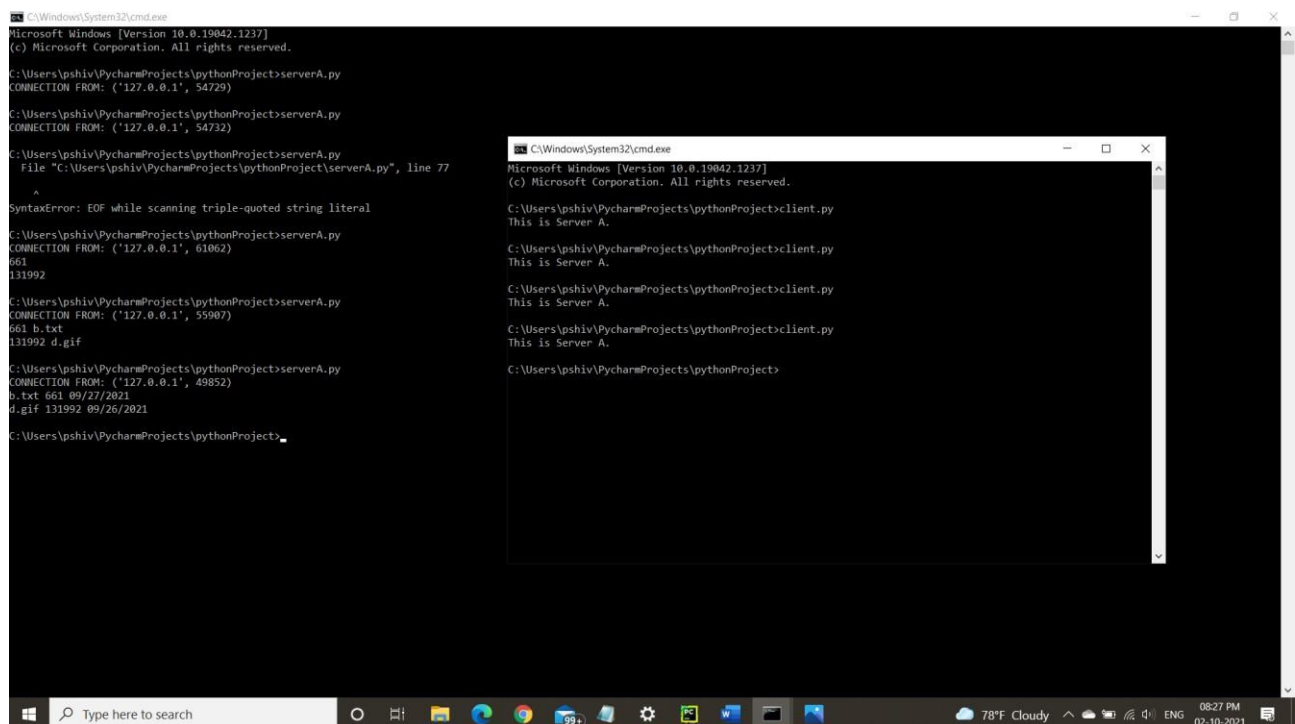
```
file_path = os.path.join(dir_name, file_name) # join the file path with directory
timestamp_str = time.strftime('%m/%d/%Y', # last modification date of file
                             time.gmtime(os.path.getmtime(file_path)))
```

After that I display the size into byte using os.stat().st_size as per given reference.

<https://stackoverflow.com/questions/40783029/os-stat-st-size-gives-me-incorrect-size-in-python> and <https://www.journaldev.com/32067/how-to-get-file-size-in-python>

```
files with size = (os.stat(file_path).st_size) # Get file Size in bytes
```

In Output, I got Filename, Size, and Date as per below.



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 54729)

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 54732)

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
File "C:\Users\pshiv\PycharmProjects\pythonProject\serverA.py", line 77
^
SyntaxError: EOF while scanning triple-quoted string literal

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 61062)
661
131992

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 55907)
661 b.txt
131992 d.gif

C:\Users\pshiv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 49852)
b.txt 661 09/27/2021
d.gif 131992 09/26/2021

C:\Users\pshiv\PycharmProjects\pythonProject>

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\pshiv\PycharmProjects\pythonProject>client.py
This is Server A.

C:\Users\pshiv\PycharmProjects\pythonProject>client.py
This is Server A.

C:\Users\pshiv\PycharmProjects\pythonProject>client.py
This is Server A.

C:\Users\pshiv\PycharmProjects\pythonProject>client.py
This is Server A.

C:\Users\pshiv\PycharmProjects\pythonProject>
```

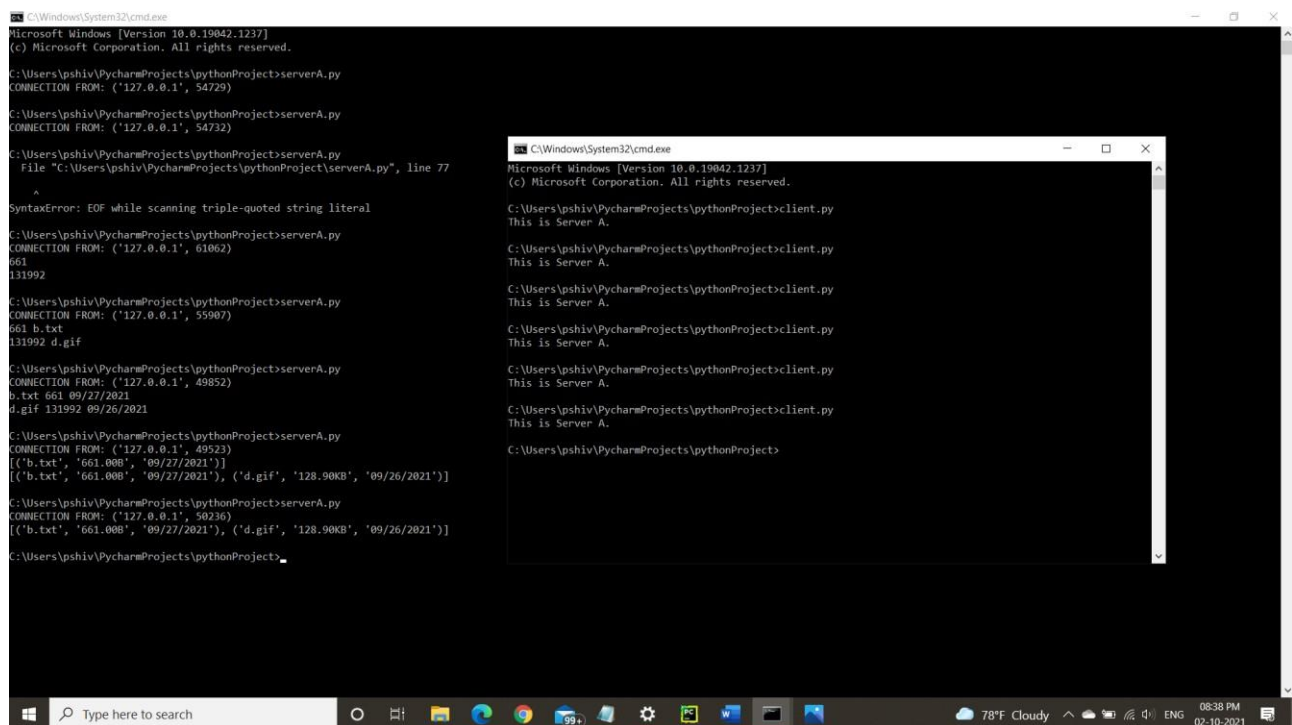
After got the size, I converted that size into Human readable size and convert into 'KB', 'MB', 'GB' etc. using given reference.

<https://stackoverflow.com/questions/1094841/get-human-readable-version-of-file-size>

```
def human_readable_size(size, decimal_places=2):    # Get human readable version
of file size
    for unit in ['B', 'KB', 'MB', 'GB', 'TB']:
        if size < 1024.0:
            break
        size /= 1024.0
    return f"{size:.{decimal_places}f}{unit}"
human_size = (human_readable_size(files_with_size))
```

After applying above code, I append the data and got the output like this,

```
data1 = (file_name, human_size, timestamp_str)    # Get filename, human readable
size, date into data
d.append(data1)    # append all three file into d
```



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 54729)

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 54732)

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
File "C:\Users\psliv\PycharmProjects\pythonProject\serverA.py", line 77
SyntaxError: EOF while scanning triple-quoted string literal

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 61062)
661
131992

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 55907)
661 b.txt
131992 d.gif

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 49523)
b.txt 661 09/27/2021
d.gif 131992 09/26/2021

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 50230)
[('b.txt', '661.00B', '09/27/2021')]
[('b.txt', '661.00B', '09/27/2021'), ('d.gif', '128.90KB', '09/26/2021')]

C:\Users\psliv\PycharmProjects\pythonProject>serverA.py
CONNECTION FROM: ('127.0.0.1', 50230)
[('b.txt', '661.00B', '09/27/2021'), ('d.gif', '128.90KB', '09/26/2021')]

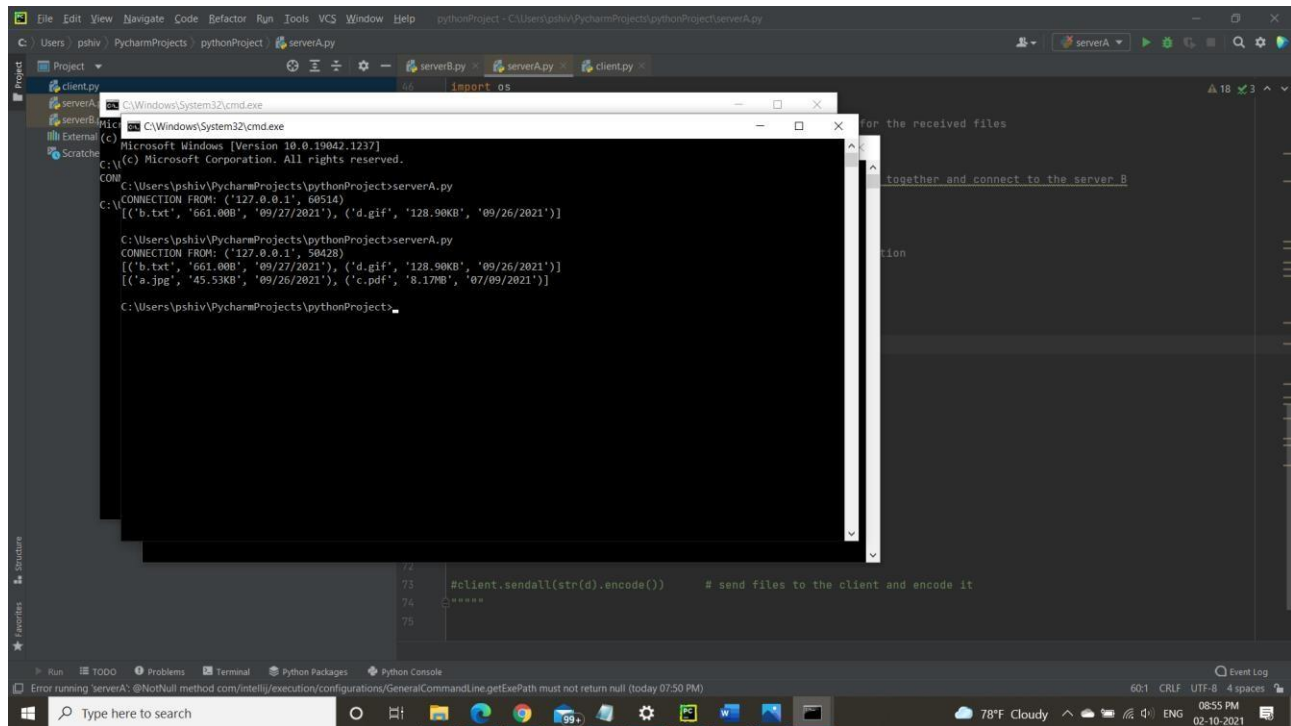
C:\Users\psliv\PycharmProjects\pythonProject>
```

After get the list of files with files metadata then I used same server A code for server B using different directory of folder and also write same client code for server B into the server A file.

After getting server B file I send that server B file to the client and encode it.

```
client.sendall(str(sending).encode())    # send files to the client and encode
it
```

Now, Server A has both server files.



The screenshot shows the PyCharm IDE interface. A terminal window is open, displaying network logs for a server. The logs show two connections from '127.0.0.1'. The first connection receives a list of files: ['b.txt', '661.00B', '09/27/2021'], ('d.gif', '128.90KB', '09/26/2021'). The second connection receives a list of files: ['b.txt', '661.00B', '09/27/2021'], ('d.gif', '128.90KB', '09/26/2021'), ('a.jpg', '45.53KB', '09/26/2021'), ('c.pdf', '8.17MB', '07/09/2021'). The code editor shows a Python script with the following lines:

```
72
73 #client.sendall(str(d).encode()) # send files to the client and encode it
74
75
```

After get the both server files on server A, I append both server files and sort the files by file name.

```
# Merge Server A and Server B file

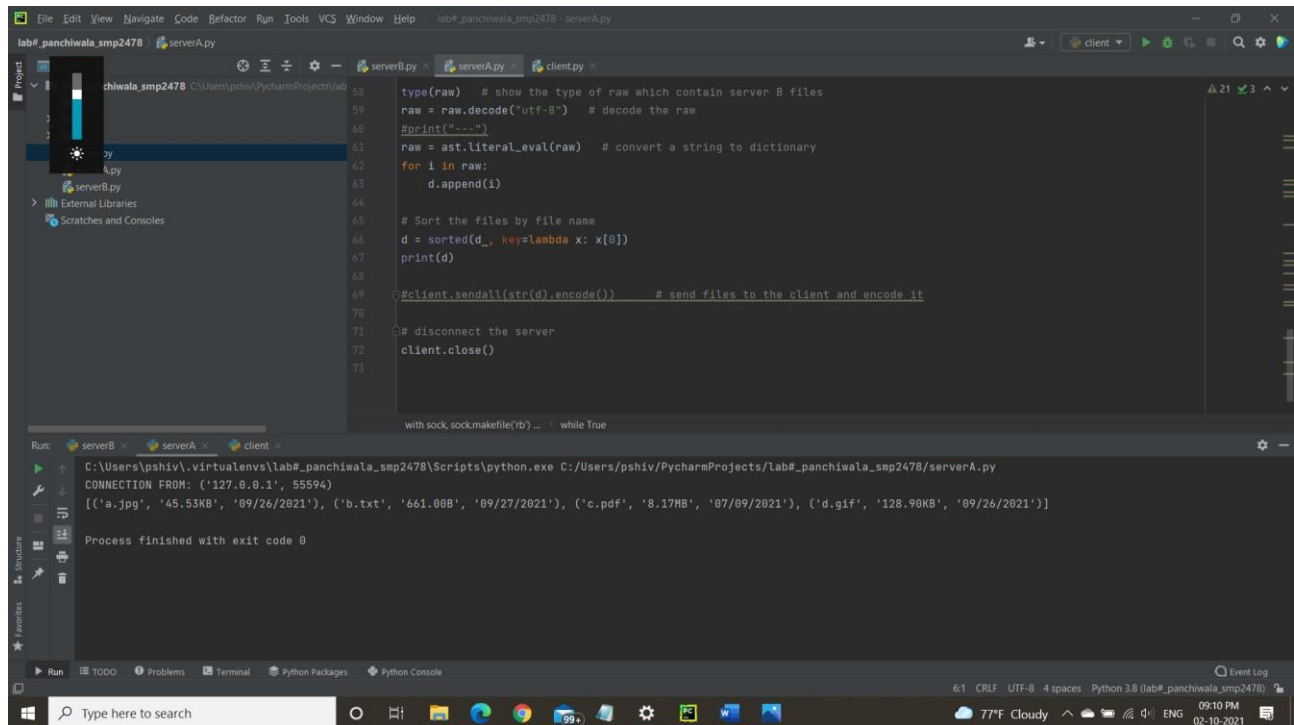
type(raw) # show the type of raw which contain server B files
raw = raw.decode("utf-8") # decode the raw
#print("---")
raw = ast.literal_eval(raw) # convert a string to dictionary
for i in raw:
    d.append(i)

# Sort the files by file name
d = sorted(d, key=lambda x: x[0])
```

For convert a string to dictionary format I used `ast.literal_eval()` using given reference.

https://www.kite.com/python/docs/ast.literal_eval

Then I got the output like this



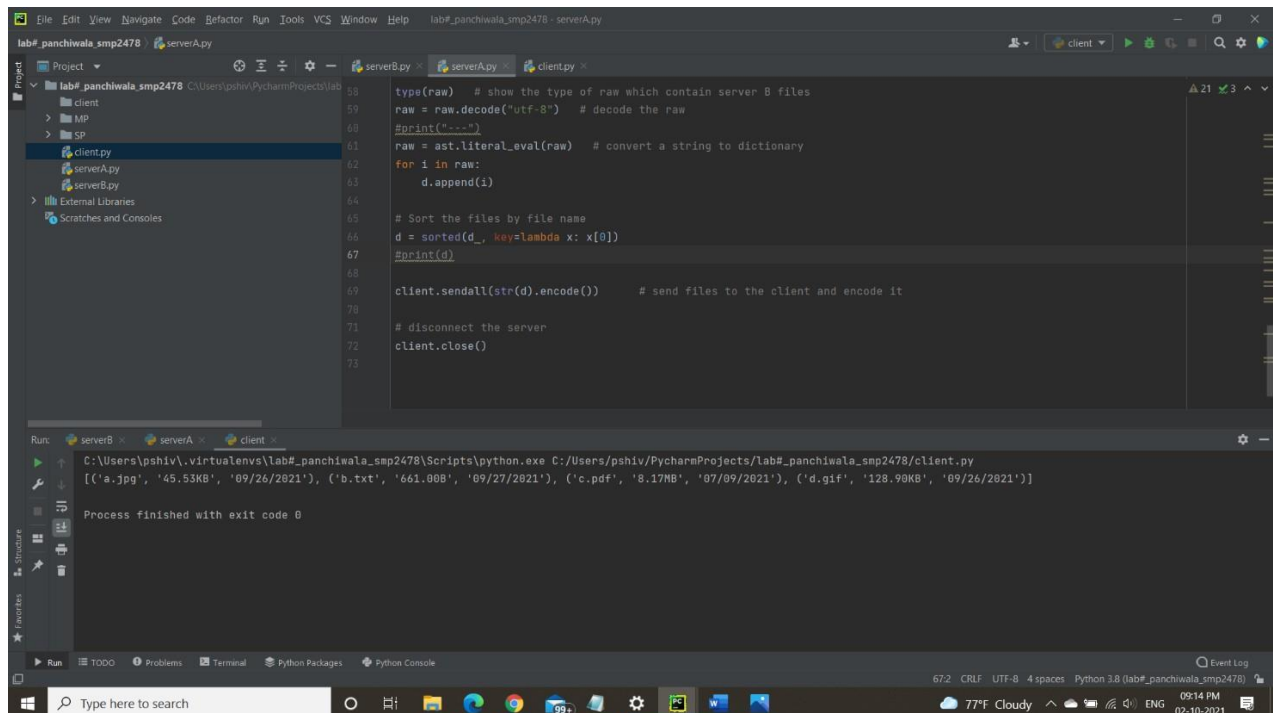
The screenshot shows the PyCharm IDE with a project named 'lab#_panchiwala_smp2478'. The file explorer on the left shows the project structure. The main editor displays the code for 'serverA.py'. The code defines a server that receives data from a client, decodes it, evaluates it as a dictionary, sorts it by file name, and sends it back to the client. The Run console at the bottom shows the output of the server, indicating a connection from '127.0.0.1' and the receipt of a list of file names and sizes.

```
lab#_panchiwala_smp2478 - serverA.py
58 type(raw) # show the type of raw which contain server B files
59 raw = raw.decode("utf-8") # decode the raw
60 print("----")
61 raw = ast.literal_eval(raw) # convert a string to dictionary
62 for i in raw:
63     d.append(i)
64
65 # Sort the files by file name
66 d = sorted(d_, key=lambda x: x[0])
67 print(d)
68
69 client.sendall(str(d).encode()) # send files to the client and encode it
70
71 # disconnect the server
72 client.close()
73
```

```
Run: serverB - serverA - client
C:\Users\pshiv\virtualenvs\lab#_panchiwala_smp2478\Scripts\python.exe C:\Users\pshiv\PycharmProjects\lab#_panchiwala_smp2478\serverA.py
CONNECTION FROM: ('127.0.0.1', 55594)
[('a.jpg', '45.53KB', '09/26/2021'), ('b.txt', '661.00B', '09/27/2021'), ('c.pdf', '8.17MB', '07/09/2021'), ('d.gif', '128.90KB', '09/26/2021')]
Process finished with exit code 0
```

After that I send all the data from Server A to Client and encode it.

```
client.sendall(str(d).encode()) # send files to the client and encode it
```



The screenshot shows the PyCharm IDE with the same project. The file explorer on the left shows the project structure. The main editor displays the code for 'client.py'. The code defines a client that connects to the server, receives data, and prints it. The Run console at the bottom shows the output of the client, indicating a connection to the server and the receipt of a list of file names and sizes.

```
lab#_panchiwala_smp2478 - serverA.py
58 type(raw) # show the type of raw which contain server B files
59 raw = raw.decode("utf-8") # decode the raw
60 print("----")
61 raw = ast.literal_eval(raw) # convert a string to dictionary
62 for i in raw:
63     d.append(i)
64
65 # Sort the files by file name
66 d = sorted(d_, key=lambda x: x[0])
67 print(d)
68
69 client.sendall(str(d).encode()) # send files to the client and encode it
70
71 # disconnect the server
72 client.close()
73
```

```
Run: serverB - serverA - client
C:\Users\pshiv\virtualenvs\lab#_panchiwala_smp2478\Scripts\python.exe C:\Users\pshiv\PycharmProjects\lab#_panchiwala_smp2478\client.py
[('a.jpg', '45.53KB', '09/26/2021'), ('b.txt', '661.00B', '09/27/2021'), ('c.pdf', '8.17MB', '07/09/2021'), ('d.gif', '128.90KB', '09/26/2021')]
Process finished with exit code 0
```

After sending files from Server A to the client, I got the all the files on Client.

In this Lab2, I modified the code based on lab 2. I need some functions which is required simultaneously. So, I created helper.py file and add that function in that file and given the reference in the code.

After that I used multithreading and create the function for sync the files from both server and list function for listing the files. Using the given reference.

<https://analyticsindiamag.com/how-to-run-python-code-concurrently-using-multithreading/>

<https://docs.python.org/2/library/socketserver.html#module-socketserver>

For Sync the files, I give the directory path and used lambda and set function for syncing the files from Server B to A and A to B. Then I create a for loop to find the common files from both server. Reference from <https://www.journaldev.com/37089/how-to-compare-two-lists-in-python>

Then I checked that which one is latest file from both server's file.

I also write the function for that both server is alive and continuously work and syncing the files until if users manually closed it.

Reference from <https://stackoverflow.com/questions/8627986/how-to-keep-a-socket-open-until-client-closes-it>

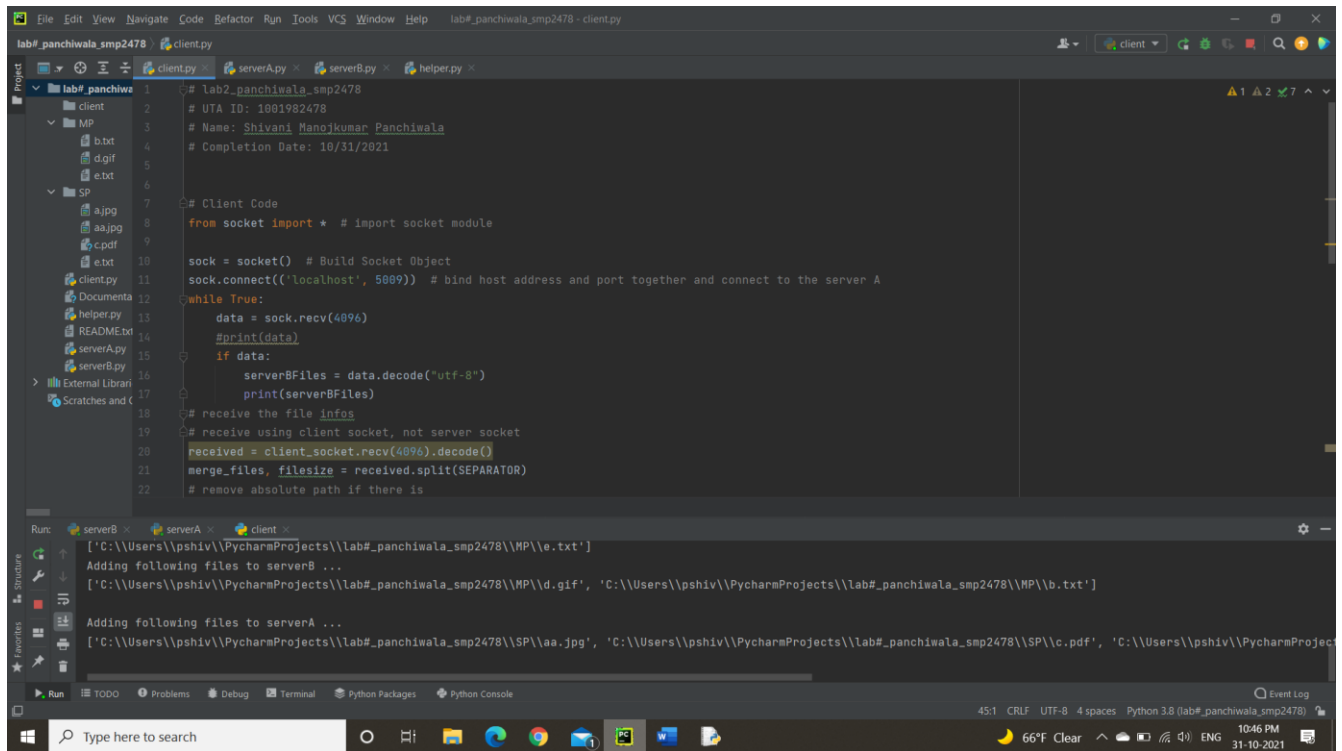
For keep the latest file to create function and check date and which one is latest file.

<https://stackoverflow.com/questions/41635547/convert-python-datetime-to-timestamp-in-milliseconds>

I also write the code for the read the data to client side and and write the data from server side to transfer the files. Using Reference <https://www.thepythoncode.com/article/send-receive-files-using-sockets-python>

After that Run the Server B then Server A and then Client. Both Server runs continuously and syncing the files and also give the notifications to any updates until the user kill the the both server manually.

So, It's continuously syncing the files. And show the updates.



HOW TO IMPLEMENT THE PROJECT

- 1) Open the Command Prompt and write the **serverB.py**
- 2) After that open another Command Prompt and write the **serverA.py**
- 3) And open again another Command Prompt and write **client.py**
- 4) Then you will get the output on the Client.

REFERENCES

- 1) <https://stackoverflow.com/questions/47539028/transfer-contents-of-a-folder-over-network-by-python>
- 2) <https://thispointer.com/python-get-list-of-files-in-directory-sorted-by-date-and-time/>
- 3) <https://stackoverflow.com/questions/40783029/os-stat-st-size-gives-me-incorrect-size-in-python>
- 4) <https://www.journaldev.com/32067/how-to-get-file-size-in-python>
- 5) <https://stackoverflow.com/questions/1094841/get-human-readable-version-of-file-size>
- 6) https://www.kite.com/python/docs/ast.literal_eval
- 7) <https://analyticsindiamag.com/how-to-run-python-code-concurrently-using-multithreading/>
- 8) <https://docs.python.org/2/library/socketserver.html#module-socketserver>
- 9) <https://www.journaldev.com/37089/how-to-compare-two-lists-in-python>
- 10) <https://stackoverflow.com/questions/8627986/how-to-keep-a-socket-open-until-client-closes-it>
- 11) <https://stackoverflow.com/questions/41635547/convert-python-datetime-to-timestamp-in-milliseconds>
- 12) <https://www.thepythoncode.com/article/send-receive-files-using-sockets-python>