UNIVERSITY OF
TEXAS
ARLINGTON

COMPUTER SCIENCE AND ENGINEERING

DISTRIBUTED SYSTEMS – CSE 5306 – 003

PROJECT-1

Team Members

Name	Student Id
Shreya Malraju	1002046338
Jyothi Bhavani Patibandla	1001960841

I have neither given nor received unauthorized assistance on this work. I will not post the project description and the solution online.

Sign: Shreya Malraju Date: September 11, 2023

Sign: Jyothi Bhavani Patibandla Date: September 11, 2023

Part 1 : Multi-threaded file server

Contribution – Upload and Download operations – Shreya

Rename and Delete operations – Jyothi

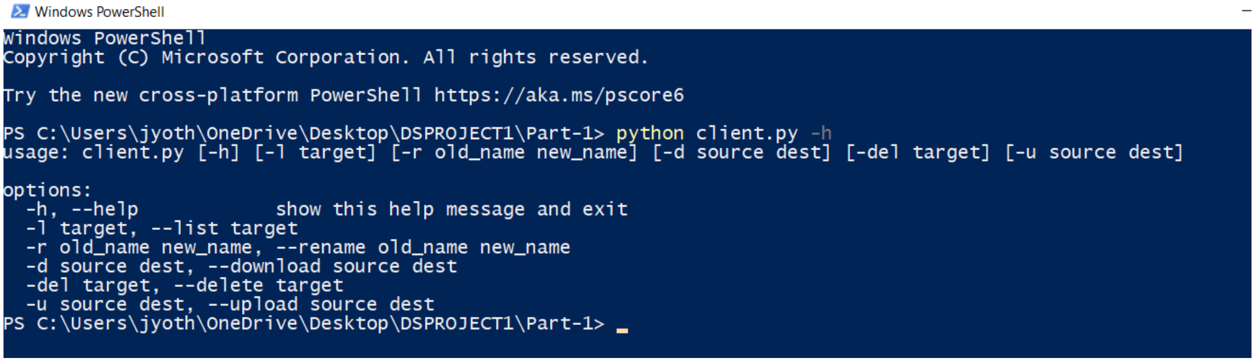
XML-RPC is a remote procedure call that uses XML, transported via HTTP. By using this procedure, a client can make request to a server and can obtain the response in return.

We perform 4 different operations i.e.,

- UPLOAD
- DOWNLOAD
- RENAME
- DELETE

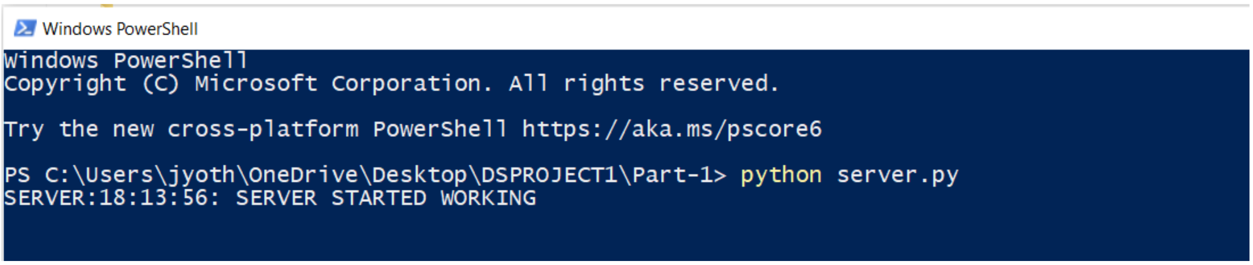
To find out what operations the fileserver supports, enter the following command.

python client.py -h



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1> python client.py -h
usage: client.py [-h] [-l target] [-r old_name new_name] [-d source dest] [-del target] [-u source dest]

options:
  -h, --help            show this help message and exit
  -l target, --list target
  -r old_name new_name, --rename old_name new_name
  -d source dest, --download source dest
  -del target, --delete target
  -u source dest, --upload source dest
PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1>
```

Start to run the server using the command: Python server.py

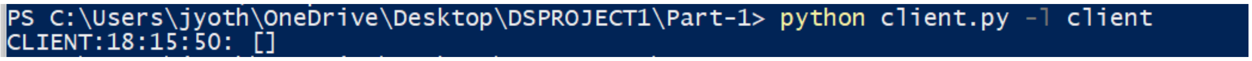


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1> python server.py
SERVER:18:13:56: SERVER STARTED WORKING
```

we see that the Client and server directories having no files in it.

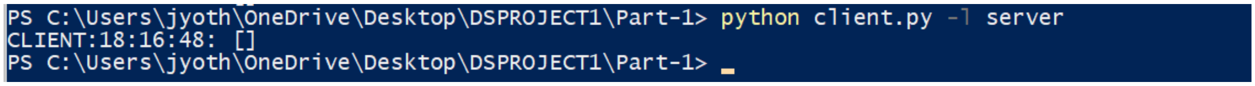
By running the following commands

python client.py -l client



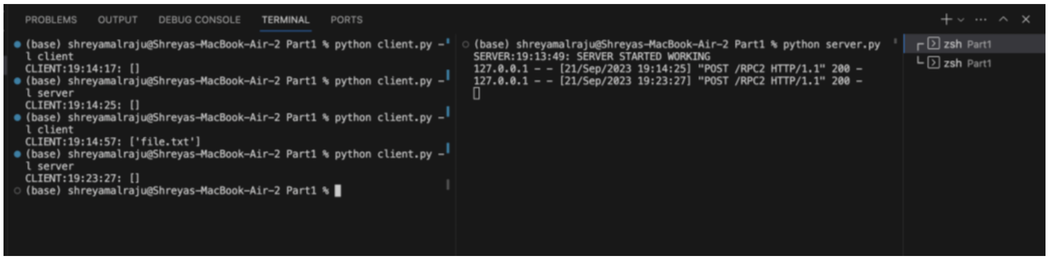
```
PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1> python client.py -l client
CLIENT:18:15:50: []
```

python client.py -l server



```
PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1> python client.py -l server
CLIENT:18:16:48: []
PS C:\Users\jyoth\OneDrive\Desktop\DSPROJECT1\Part-1> _
```

Adding File to the client directory and checking both client and server



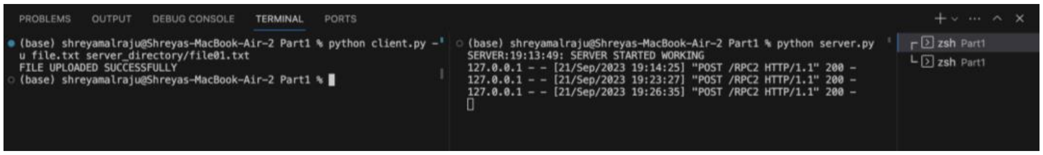
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l client
CLIENT:19:14:17: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:14:25: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l client
CLIENT:19:14:57: ['file.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:23:27: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
```

Performing Upload Operation:

Run the following command to upload the file to the server:

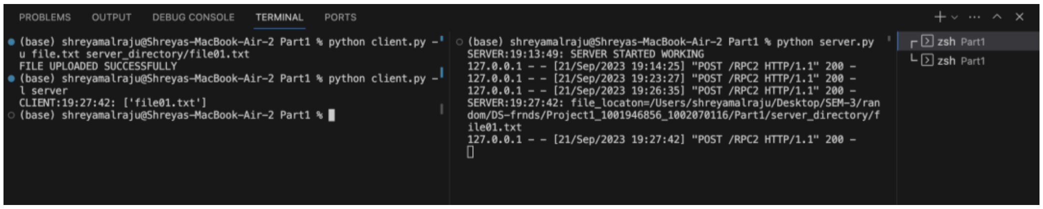
python client.py -u client_filename server_directory/filename



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
u file.txt server_directory/file01.txt
FILE UPLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
```

By listing the files on the server, we can verify that a file has been uploaded to the server by using following command: python client.py -l server

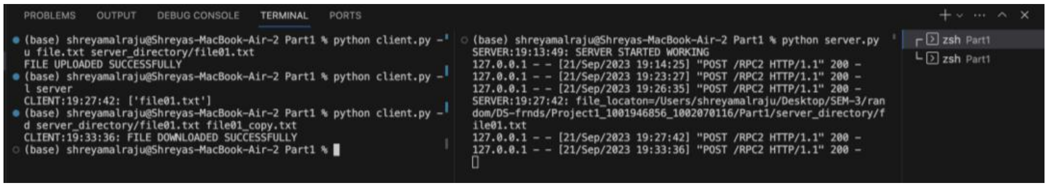


```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
u file.txt server_directory/file01.txt
FILE UPLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:27:42: ['file01.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:27:42: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile01.txt
127.0.0.1 - - [21/Sep/2023 19:27:42] "POST /RPC2 HTTP/1.1" 200 -
```

Performing Download Operation:

Download the file from server by running the following command: python client.py -d server_directory/filename client_filename

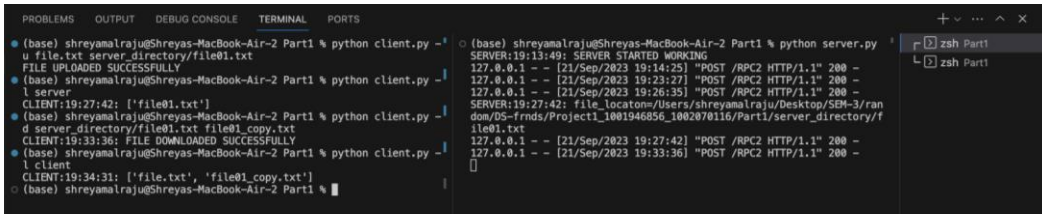


```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
u file.txt server_directory/file01.txt
FILE UPLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:27:42: ['file01.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
d server_directory/file01.txt file01_copy.txt
CLIENT:19:33:36: FILE DOWNLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:27:42: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile01.txt
127.0.0.1 - - [21/Sep/2023 19:27:42] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:33:36] "POST /RPC2 HTTP/1.1" 200 -
```

By Listing the files on the client allows us to verify that a file was downloaded from the server:

client.py in Python, -l client



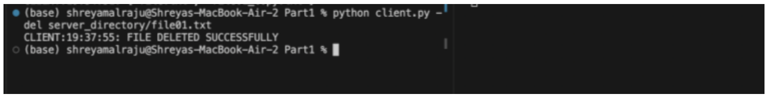
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
u file.txt server_directory/file01.txt
FILE UPLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:27:42: ['file01.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
d server_directory/file01.txt file01_copy.txt
CLIENT:19:33:36: FILE DOWNLOADED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l client
CLIENT:19:34:31: ['file.txt', 'file01_copy.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:27:42: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile01.txt
127.0.0.1 - - [21/Sep/2023 19:27:42] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:33:36] "POST /RPC2 HTTP/1.1" 200 -
```

Performing the Delete Operation:

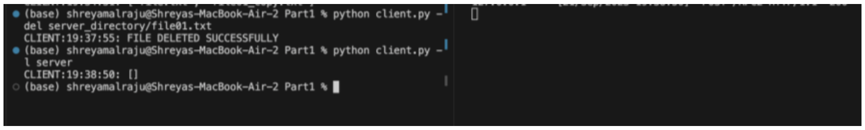
Executing the following command will delete the specified file from the file server:

python client.py -del server_directory/filename



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
del server_directory/file01.txt
CLIENT:19:37:55: FILE DELETED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

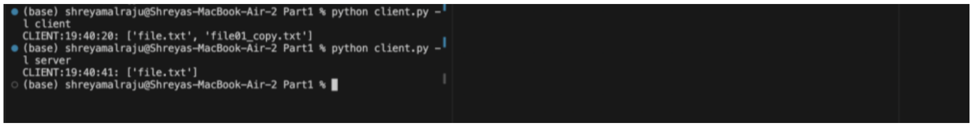
After deleting a file from the server, we can see the file was deleted on server by using the below command: python client.py -l server



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
del server_directory/file01.txt
CLIENT:19:37:55: FILE DELETED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:38:50: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

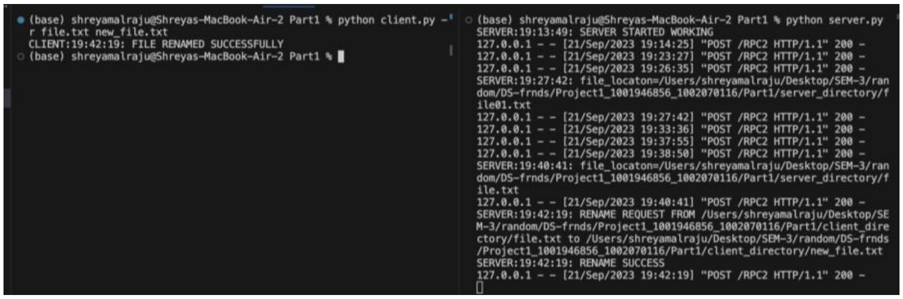
Renaming the File:

Verifying whether the file has the same name on the client and server directories.



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l client
CLIENT:19:40:20: ['file.txt', 'file01_copy.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:40:41: ['file.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

We can rename the file on the server by running the following command: python client.py -r filename new_filename

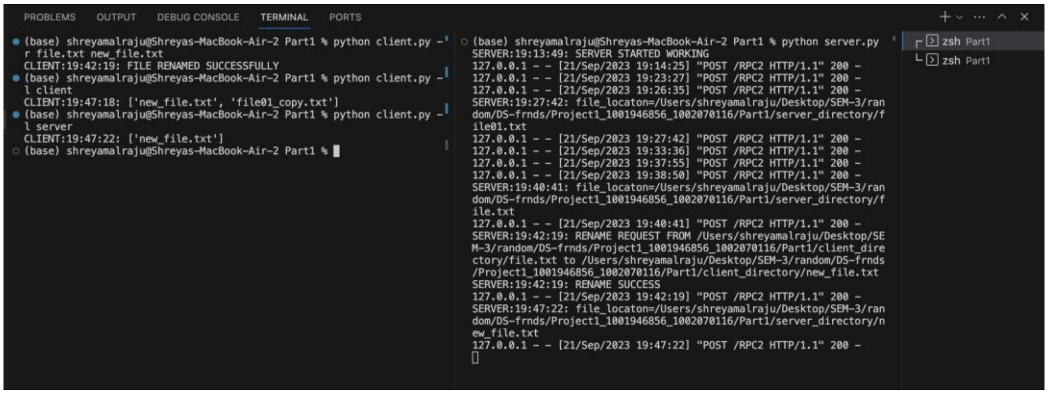


```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
r file.txt new_file.txt
CLIENT:19:42:19: FILE RENAMED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:27:42: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile01.txt
127.0.0.1 - - [21/Sep/2023 19:27:42] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:33:36] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:37:55] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:38:50] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:40:41: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile.txt
127.0.0.1 - - [21/Sep/2023 19:40:41] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:42:19: RENAME REQUEST FROM /Users/shreyamalraju/Desktop/SE
M-3/random/DS-frnds/Project1_1001946856_1002070116/Part1/client_dire
ctory/file.txt to /Users/shreyamalraju/Desktop/SEM-3/random/DS-frnds
/Project1_1001946856_1002070116/Part1/client_directory/new_file.txt
SERVER:19:42:19: RENAME SUCCESS
127.0.0.1 - - [21/Sep/2023 19:42:19] "POST /RPC2 HTTP/1.1" 200 -
```

After renaming the file on the server, we can verify the files are renamed on the server and client by the following commands: python client.py -l client

python client.py -l server



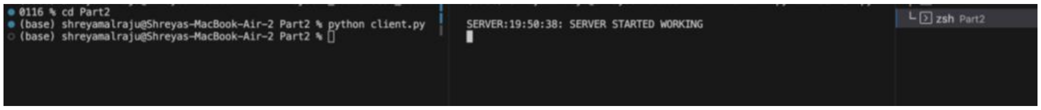
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
r file.txt new_file.txt
CLIENT:19:42:19: FILE RENAMED SUCCESSFULLY
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l client
CLIENT:19:47:18: ['new_file.txt', 'file01_copy.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python client.py -
l server
CLIENT:19:47:22: ['new_file.txt']
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part1 % python server.py
SERVER:19:13:49: SERVER STARTED WORKING
127.0.0.1 - - [21/Sep/2023 19:14:25] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:23:27] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:26:35] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:27:42: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile01.txt
127.0.0.1 - - [21/Sep/2023 19:27:42] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:33:36] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:37:55] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [21/Sep/2023 19:38:50] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:40:41: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/f
ile.txt
127.0.0.1 - - [21/Sep/2023 19:40:41] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:42:19: RENAME REQUEST FROM /Users/shreyamalraju/Desktop/SE
M-3/random/DS-frnds/Project1_1001946856_1002070116/Part1/client_dire
ctory/file.txt to /Users/shreyamalraju/Desktop/SEM-3/random/DS-frnds
/Project1_1001946856_1002070116/Part1/client_directory/new_file.txt
SERVER:19:42:19: RENAME SUCCESS
127.0.0.1 - - [21/Sep/2023 19:42:19] "POST /RPC2 HTTP/1.1" 200 -
SERVER:19:47:22: file_locaton=/Users/shreyamalraju/Desktop/SEM-3/ran
dom/DS-frnds/Project1_1001946856_1002070116/Part1/server_directory/n
ew_file.txt
127.0.0.1 - - [21/Sep/2023 19:47:22] "POST /RPC2 HTTP/1.1" 200 -
```

Part 2 : Synchronized Storage Service

Contribution – By Shreya and Jyothi

Start to run the server



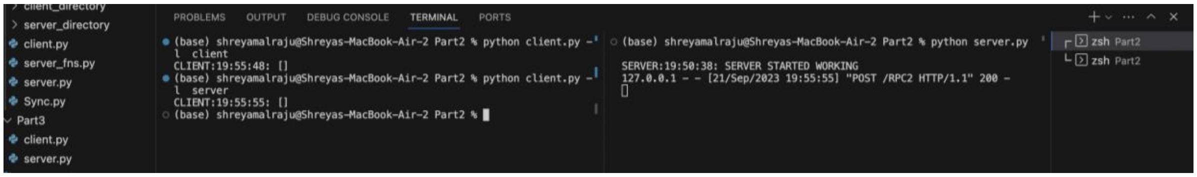
```
0116 % cd Part2
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 % python client.py
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 %
```

```
SERVER:19:50:38: SERVER STARTED WORKING
```

Firstly, let's check whether the files are available in both the client and server

python client.py -l client

python client.py -l server



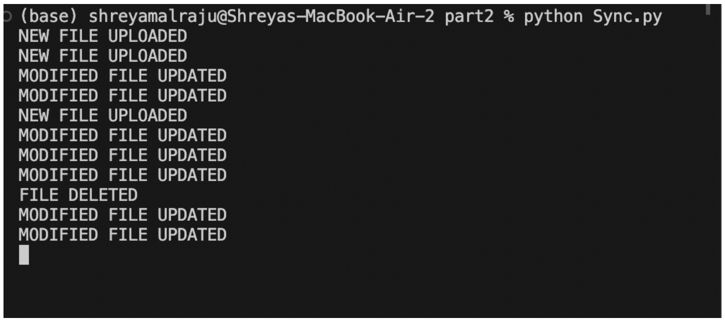
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 % python client.py -
l  client
CLIENT:19:55:48: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 % python client.py -
l  server
CLIENT:19:55:55: []
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part2 % python server.py

 SERVER:19:50:38: SERVER STARTED WORKING
 127.0.0.1 - - [21/Sep/2023 19:55:55] "POST /RPC2 HTTP/1.1" 200 -
```

Once the server is started, start to run the Sync.py by using the following command:

Python Sync.py

Perform Upload, modification and deletion of files, we can see changes after 10 seconds on the terminal.



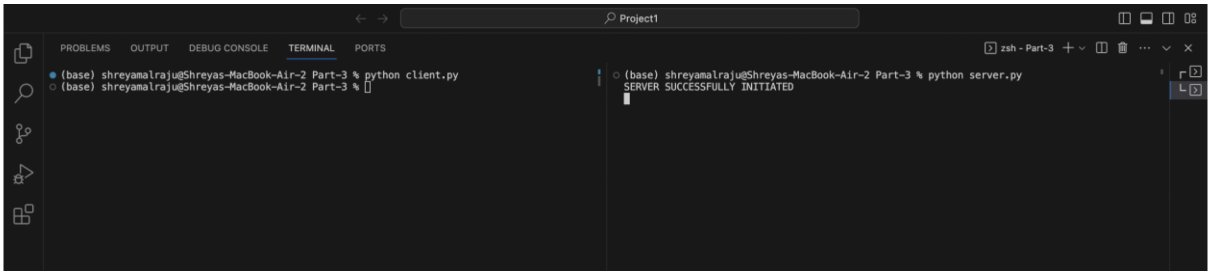
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 part2 % python Sync.py
NEW FILE UPLOADED
NEW FILE UPLOADED
MODIFIED FILE UPDATED
MODIFIED FILE UPDATED
NEW FILE UPLOADED
MODIFIED FILE UPDATED
MODIFIED FILE UPDATED
MODIFIED FILE UPDATED
FILE DELETED
MODIFIED FILE UPDATED
MODIFIED FILE UPDATED
```

Part 3 : Synchronous and Asynchronous add() and sort()

Contribution – Synchronous Operations – Shreya

Asynchronous Operations – Jyothi

Open 2 terminals and start the server on one terminal using the command python server.py and start the client module on the other terminal using the command python client.py



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python server.py
SERVER SUCCESSFULLY INITIATED
```

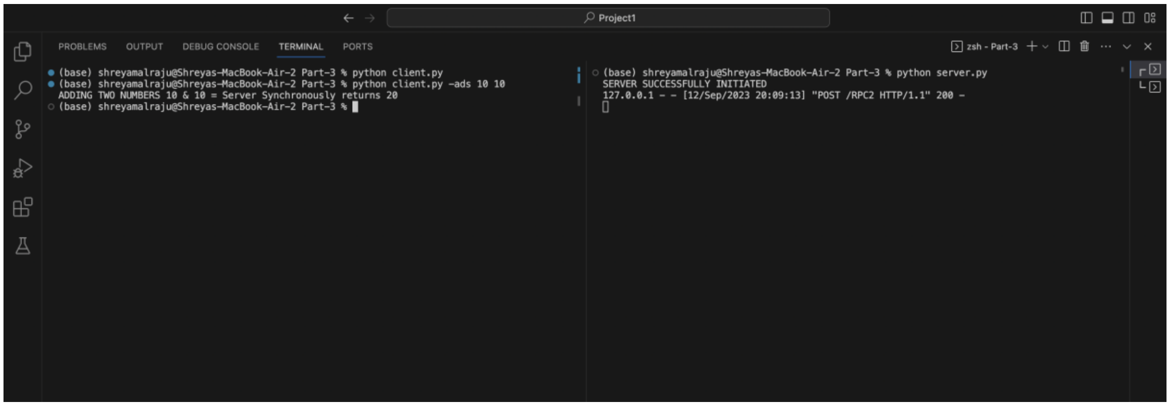
the computation So what supports 2 functions addition and sorting in both synchronous and asynchronous manner.

When we perform synchronous addition or sorting we make a request directly to the server and server directly sends the result to the client there is no delay in the response.

If we perform addition and sorting in asynchronous manner, the client requests the server for the response and first server sends an acknowledgement to the client and then the client uses this acknowledgement number to access the result when the computation is ready.

Synchronous Addition

To perform Synchronous addition, use the command python client.py -ads 10 10. We can observe that the response is obtained instantly.

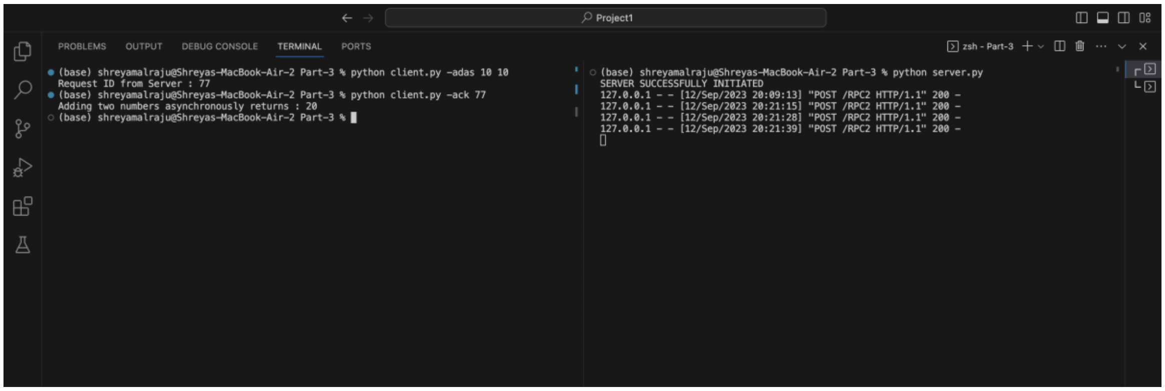


```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -ads 10 10
ADDING TWO NUMBERS 10 & 10 = Server Synchronously returns 20
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python server.py
SERVER SUCCESSFULLY INITIATED
127.0.0.1 - - [12/Sep/2023 20:09:13] "POST /RPC2 HTTP/1.1" 200 -
```

Asynchronous Addition

Addition is performed by using the command shown below Python client -asad 10 10. After executing this command we get an output from the server which is the acknowledgement number . We use this acknowledgement number in the below command to get the output of the original command. Python client.py -ack num.



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -adas 10 10
Request ID from Server : 77
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -ack 77
Adding two numbers asynchronously returns : 20
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 %
```

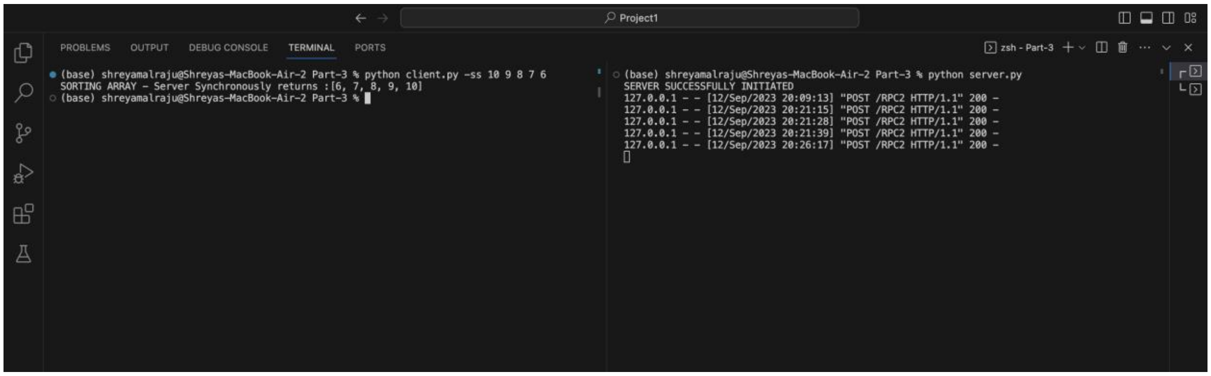
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python server.py
SERVER SUCCESSFULLY INITIATED
127.0.0.1 - - [12/Sep/2023 20:09:13] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:15] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:28] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:39] "POST /RPC2 HTTP/1.1" 200 -
```

Synchronous Sorting:

Synchronous sorting of 5 different numbers is performed using the below command.

Python client.py -ss n1 n2 n3 n4 n5

We see that the result is a sorted array in the increasing order which is obtained directly from the server as soon as the request is received.



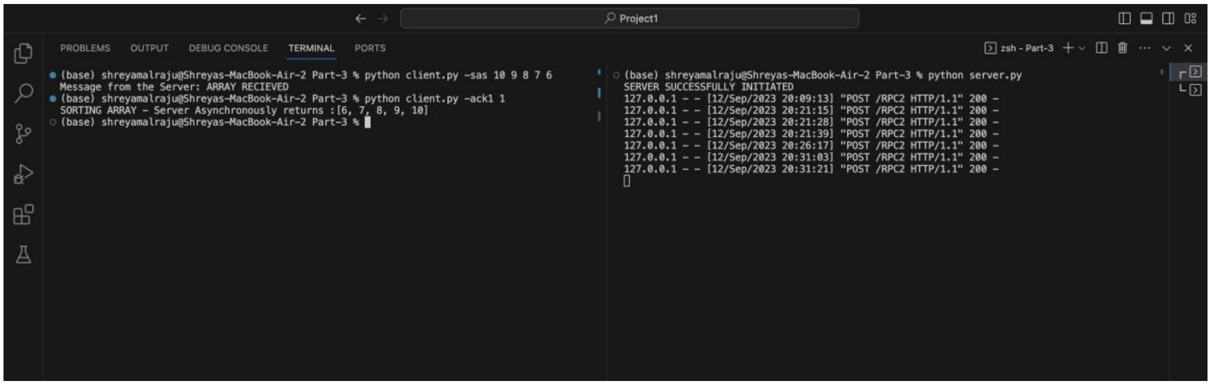
```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -ss 10 9 8 7 6
SORTING ARRAY - Server Synchronously returns :[6, 7, 8, 9, 10]
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python server.py
SERVER SUCCESSFULLY INITIATED
127.0.0.1 - - [12/Sep/2023 20:09:13] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:15] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:28] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:39] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:26:17] "POST /RPC2 HTTP/1.1" 200 -
```

Asynchronous Sorting

The same sorting can be performed using asynchronous method using the below command Python client.py -sas n1 n2 n3 n4 n5

As soon as this command is executed we receive an acknowledgement number from the server end whether result array is stored we use the below command to access the result at that particular location. Python client.py -ack1 1



```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -sas 10 9 8 7 6
Message from the Server: ARRAY RECIEVED
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python client.py -ack1 1
SORTING ARRAY - Server Asynchronously returns :[6, 7, 8, 9, 10]
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 %
```

```
(base) shreyamalraju@Shreyas-MacBook-Air-2 Part-3 % python server.py
SERVER SUCCESSFULLY INITIATED
127.0.0.1 - - [12/Sep/2023 20:09:13] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:15] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:28] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:21:39] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:26:17] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:31:03] "POST /RPC2 HTTP/1.1" 200 -
127.0.0.1 - - [12/Sep/2023 20:31:21] "POST /RPC2 HTTP/1.1" 200 -
```