Test Cases: P2P Server

Below are test cases for P2PClient.java as this is the only program that allows user input. Test cases are separated into two sections. The first section contains test cases where one client is used. The second section is test cases where two clients are used. Results are provided below

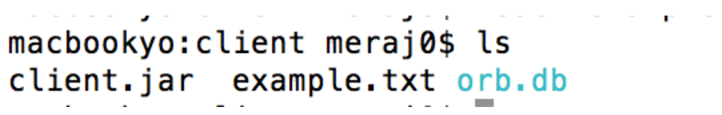
Section One *– 1 Client on P2P Server*

Test Items	Inputs	Expected Outputs	Actual Outputs	Results/Remarks
1) Upload existing file in directory	- User enters 1 to upload file -User enters a filename in current directory	- File successfully uploads	-File Successfully uploads	-Uploads working for client file
2) View uploaded file (upload file initially)	-User enters 2 to view uploaded files	-File name uploaded will be displayed	- File name is correctly displayed	-View uploaded files is working
3) Remove file on server (file must be uploaded)	-User enters 3 to remove file -User enters correct file name to remove	-File successfully removed message displays	-File successfully removed message displays	-Removal of file is working
4) Search file to download(file must be uploaded)	-User enters 4 to search file on server -User types correct file name on server	- message should display file is on server	-Message successfully displays file is on server	- Searching for files is working.
5) View all files available to download(file must be uploaded)	-User enters 5 to view all files available to download	- All files should appear on server	-All file names successfully appear	- View all files successfully works
6) Download file (file must be uploaded)	-User enters 6 to download file	- File downloads to user's current	-File successfully appears on	- Downloading file component successful

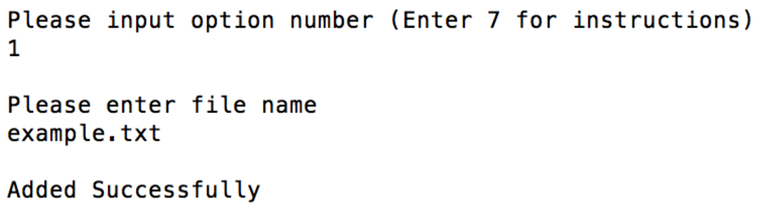
	-User types in correct name of file to download	directory. Message should also appear indicating file downloaded	users directory and message was displayed	
7) Upload non existent file	- User enters 1 to upload file -User enters a non existent filename	- File does not upload, error message thrown indicating file not found	- File does not upload, error message is shown	-Uploads working for client file
8) Remove non existent file	User enters 3 to remove file -User enters incorrect file name to remove	-Error message is thrown indicating no file found	- Appropriate error message is shown	-Removal of file is working
9) Search non existent file to download(file must be uploaded)	- User enters 4 to search file on server -User types incorrect file name on server	- message should display file not found	-Message displays file is not found	- Searching for files is working.
10) Download non existent file (file must be uploaded)	-User enters 6 to download file -User types in incorrect name of file to download	-Message displays file not found on server	-Message displays file not found on server	- Downloading file component successful
11) Exiting	-User enters 0 to exit	- All client uploaded files are erased	- Uploaded client files erased	- Exiting works
12) User enters correct port number	-User correct port number	- program proceeds to output further instructions	- program proceeds to output further instructions	- Port number works
13) User enters incorrect port number	-User incorrect port number	- message output indicating invalid port number.	- message output indicating invalid port number.	- Port number works

		- program terminates	- program terminates	

1) Initial directory. File to upload is example.txt

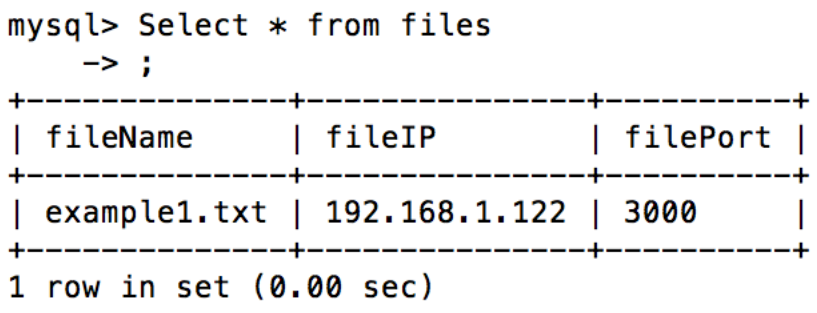


```
macbookyo:client meraj0$ ls
client.jar  example.txt orb.db
```



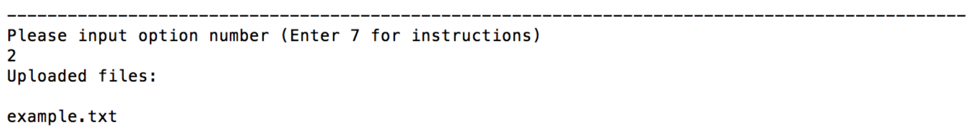
```
Please input option number (Enter 7 for instructions)
1

Please enter file name
example.txt

Added Successfully
```

MySQL Table:



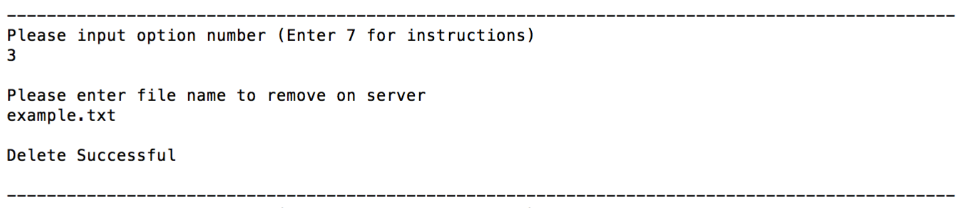
```
mysql> Select * from files
    -> ;
+--------------+---------------+----------+
| fileName     | fileIP        | filePort |
+--------------+---------------+----------+
| example1.txt | 192.168.1.122 | 3000     |
+--------------+---------------+----------+
1 row in set (0.00 sec)
```

2)



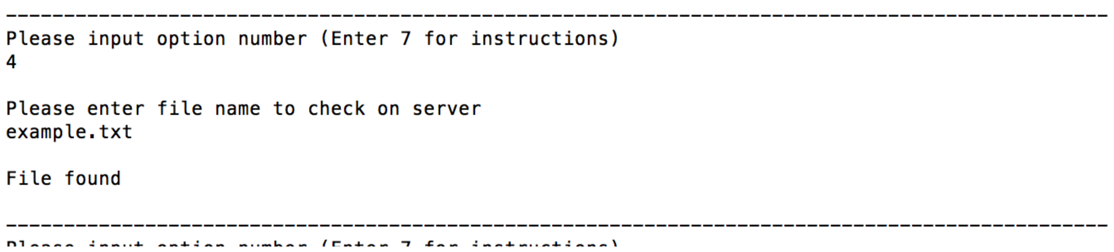
```
--------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
2
Uploaded files:

example.txt
```

3)



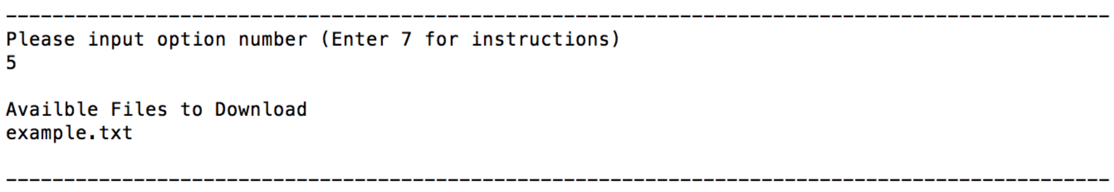
```
--------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
example.txt

Delete Successful

--------------------------------------------------------------------------------------------
```

4)



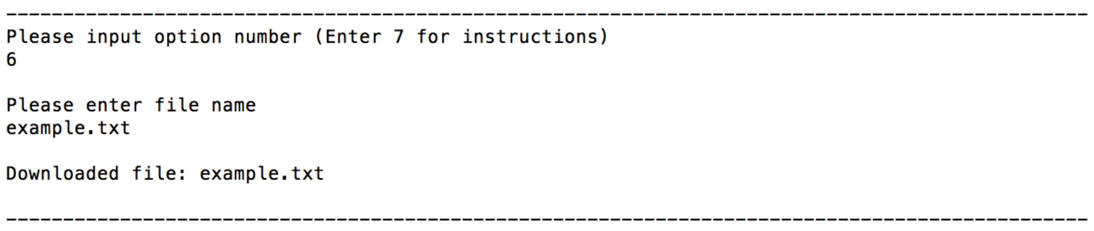
```
--------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
4

Please enter file name to check on server
example.txt

File found

--------------------------------------------------------------------------------------------------
```

5)



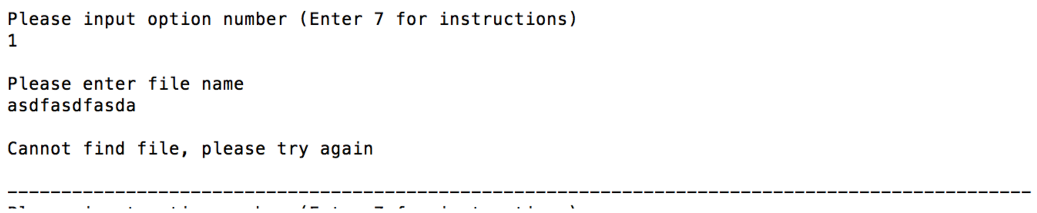
```
--------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
5

Availble Files to Download
example.txt

--------------------------------------------------------------------------------------------------
```

6)



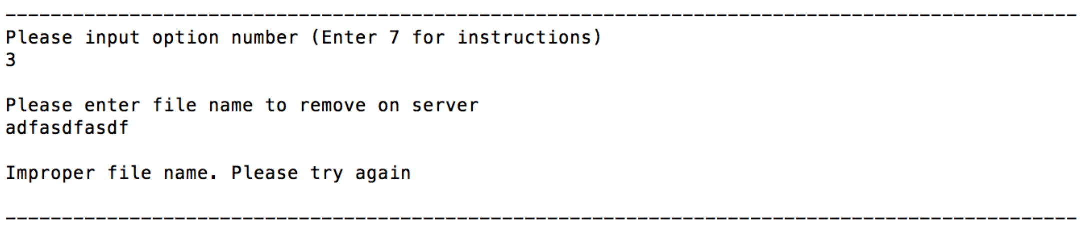
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
6

Please enter file name
example.txt

Downloaded file: example.txt

------------------------------------------------------------------------------------------------
```

7)



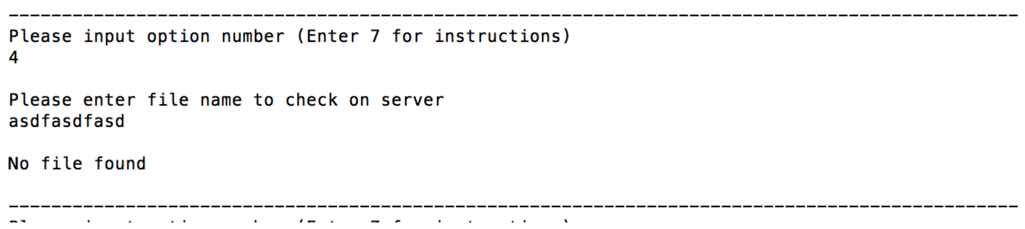
```
Please input option number (Enter 7 for instructions)
1

Please enter file name
asdfasdfasda

Cannot find file, please try again

--------------------------------------------------------------------------------------------
```

8)



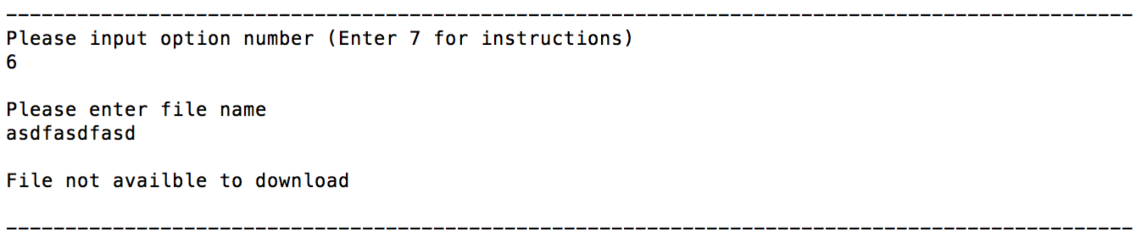
```
-----------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
adfasdfasdf

Improper file name. Please try again

-----------------------------------------------------------------------------------------------
```

9)



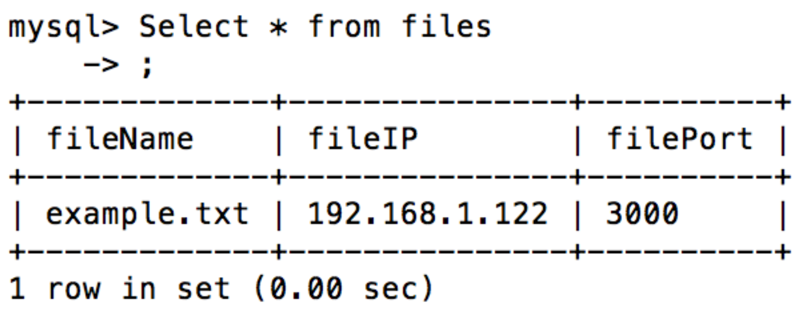
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
4

Please enter file name to check on server
asdfasdfasd

No file found

------------------------------------------------------------------------------------------------
```

10)



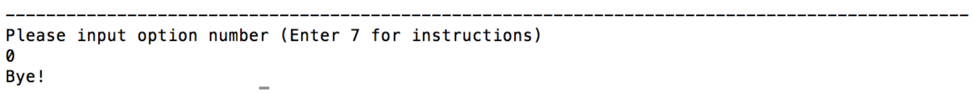
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
6

Please enter file name
asdfasdfasd

File not availble to download

------------------------------------------------------------------------------------------------
```

11) Current Mysql table (when file uploaded)



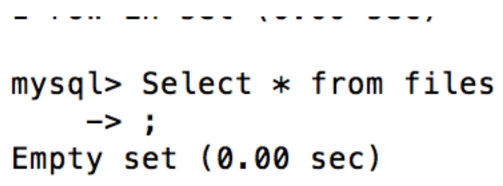
```
mysql> Select * from files
    -> ;
+-------------+---------------+----------+
| fileName    | fileIP        | filePort |
+-------------+---------------+----------+
| example.txt | 192.168.1.122 | 3000     |
+-------------+---------------+----------+
1 row in set (0.00 sec)
```

Exit:



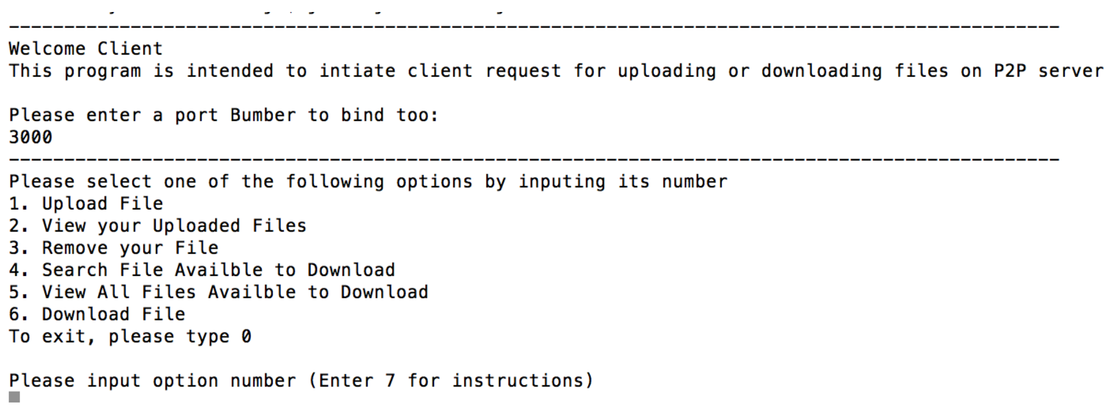
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
0
Bye!
```

Mysql table after exit:



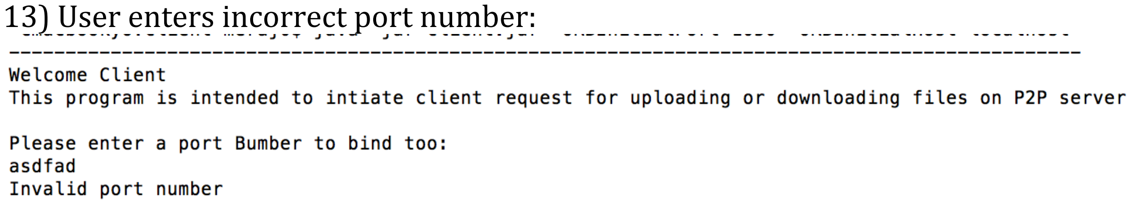
```
mysql> Select * from files
    -> ;
Empty set (0.00 sec)
```

12) User enters correct port number:



```
-----------------------------------------------------------------------------------------------
Welcome Client
This program is intended to intiate client request for uploading or downloading files on P2P server

Please enter a port Bumber to bind too:
3000
-----------------------------------------------------------------------------------------------
Please select one of the following options by inputing its number
1. Upload File
2. View your Uploaded Files
3. Remove your File
4. Search File Availble to Download
5. View All Files Availble to Download
6. Download File
To exit, please type 0

Please input option number (Enter 7 for instructions)
```

13) User enters incorrect port number:



```
-----------------------------------------------------------------------------------------------
Welcome Client
This program is intended to intiate client request for uploading or downloading files on P2P server

Please enter a port Bumber to bind too:
asdfad
Invalid port number
```

Section Two *– 2 Clients on P2P Server*

Test Items	Inputs	Expected Outputs	Actual Outputs	Results/Remarks
1) Both clients upload files	- Both clients enters 1 to upload file - Both clients enters a filename in current directory	- File successfully uploads	-File Successfully uploads	-Uploads working for client file
2) Both clients view uploaded file (upload file initially)	- Both clients enters 2 to view uploaded files	- Client 1 file name uploaded will be displayed to client 1 -Client 2 file name uploaded will be displayed	- File name is correctly displayed	-View uploaded files is working

		to client 2		
3) Both clients remove file on server (file must be uploaded)	- Both clients enters 3 to remove file - Both clients enters correct file name to remove	-File successfully removed message displays	-File successfully removed message displays	-Removal of file is working
4) Both clients view all files available to download(file must be uploaded)	- Both clients enters 5 to view all files available to download	- All files should appear on server	-All file names successfully appear	- View all files successfully works
5) Client 1 downloads Client 2 file (file must be uploaded)	-Client 1 enters 6 to download file - Client 1 types in correct name of file to download	- File downloads to Client 1 current directory. Message should also appear indicating file downloaded	-File successfully appears on users directory and message was displayed	- Downloading file component successful
6) Client 2 downloads Client 1 file (file must be uploaded)	- Client 2 enters 4 to search file on server - Client 2 types incorrect file name on server	- File downloads to Client 2 current directory. Message should also appear indicating file downloaded	-File successfully appears on users directory and message was displayed	- Downloading file component successful
7) Client 1 exits, client 2 views all files available to download	-Client 1 exists. - Client 2 enters 5 to view all files available to download	-Client 2 should only see his uploaded files	-Client 2 sees his uploaded files only	- Exiting, and view all file uploads works
8) Client 2 removes file, client 1 check	- Client 2 enters 3 to remove file.	- Client 1 should not see the file client	- Client does not see file	- Removal of files is working

available files to download	- Client 1 enters 5 to view all available files to download	2 recently removed		
9) Client 1 searches for file client 2 uploads	- Client 1 enters 4 to search file. - Client 1 types in Client 2 filename	- Client 1 should see message saying file is available to download	- Client 1 sees message indicating file is available to download	- Searching of files is working
10) Client 1 tries to remove client 2 file	- Client 1 enters 3 to remove file. - Client 1 types in client 2 filename (example2.txt)	- Client 1 should see message saying file cannot be removed	- Client 1 sees message indicating file cannot be removed	- removal is working

1) Client 1

```
Please input option number (Enter 7 for instructions)
1

Please enter file name
example1.txt

Added Successfully

-------------------------------------------------------------------------------------------------
```

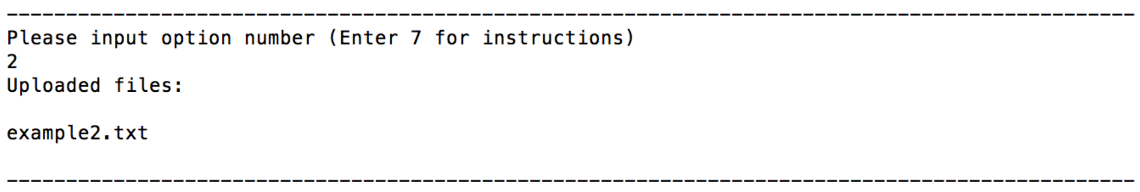
Client 2

```
Please input option number (Enter 7 for instructions)
1

Please enter file name
example2.txt

Added Successfully

----------------------------------------------------------------------------------------------
```

2) Client 1

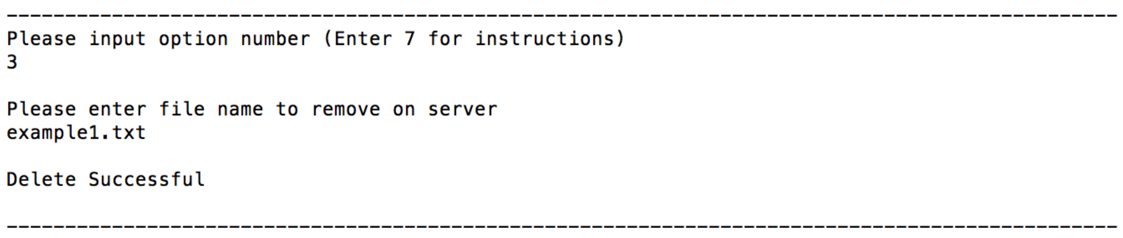
```
-----------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
2
Uploaded files:

example1.txt

-----------------------------------------------------------------------------------------------
```

Client 2



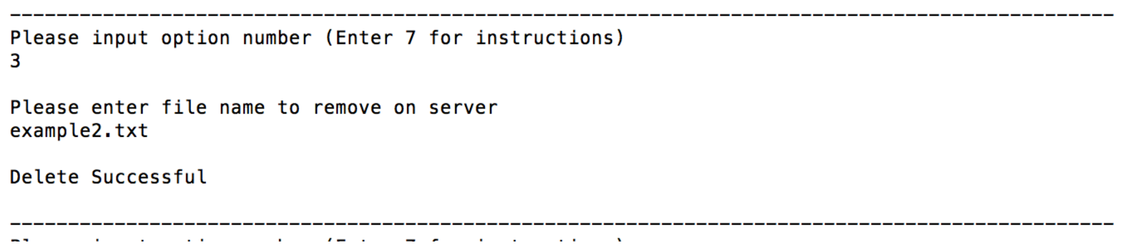
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
2
Uploaded files:

example2.txt

------------------------------------------------------------------------------------------------
```

3) Client 1



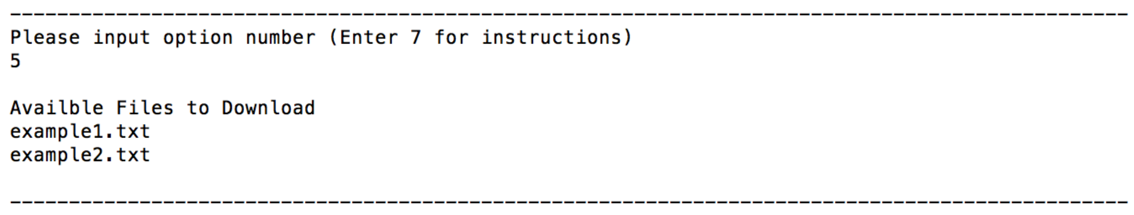
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
example1.txt

Delete Successful

------------------------------------------------------------------------------------------------
```

Client 2



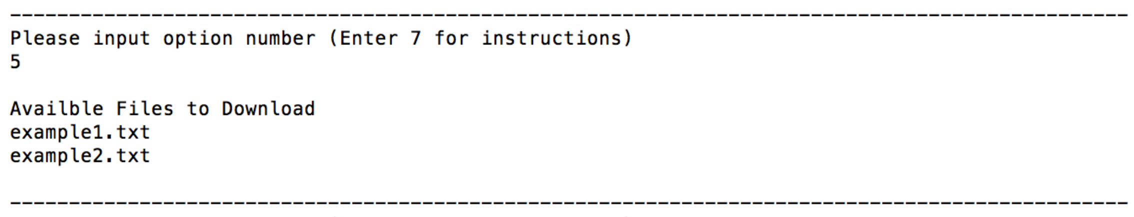
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
example2.txt

Delete Successful

------------------------------------------------------------------------------------------------
```

4) Client 1



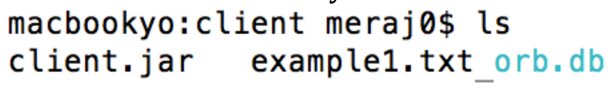
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
5

Availble Files to Download
example1.txt
example2.txt

------------------------------------------------------------------------------------------------
```

Client 2



```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
5

Availble Files to Download
example1.txt
example2.txt

------------------------------------------------------------------------------------------------
```

5) Client 1 current directory:



```
macbookyo:client meraj0$ ls
client.jar    example1.txt_orb.db
```

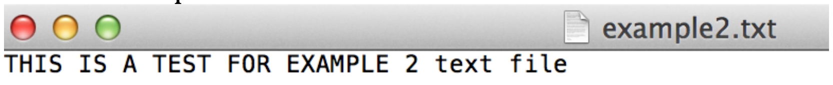
Client 1 Download example2.txt:

```
Please input option number (Enter 7 for instructions)
6

Please enter file name
example2.txt

Downloaded file: example2.txt

-----------------------------------------------------------------------------------------
```

Client 1 directory after download:

```
macbookyo:client meraj0$ ls
client.jar   example1.txt example2.txt orb.db
```

example2.txt file content:



```
THIS IS A TEST FOR EXAMPLE 2 text file
```

6) Client 2 current directory:

```
macbookyo:client2 meraj0$ ls
client2.jar  example2.txt server.jar
```

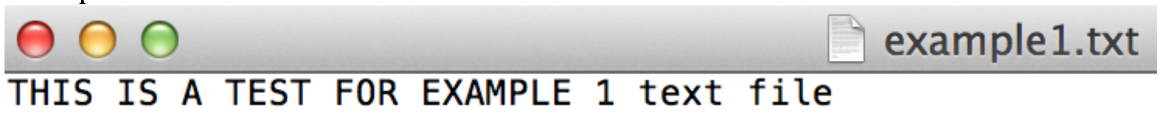
Client 2 Download example1.txt

```
Please input option number (Enter 7 for instructions)
6

Please enter file name
example1.txt

Downloaded file: example1.txt

-----------------------------------------------------------------------------------------
```

Client 2 Directory after download:

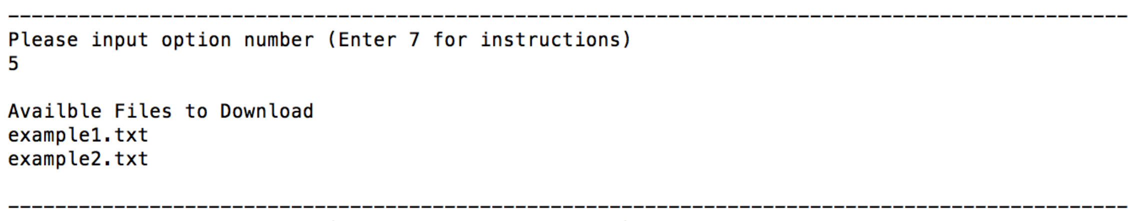
```
^Clmacbookyo:client2 meraj0$ ls
client2.jar  example1.txt example2.txt server.jar
```

example1.txt file contents:



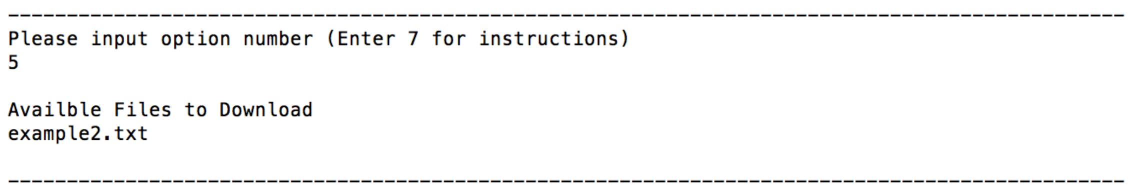
```
THIS IS A TEST FOR EXAMPLE 1 text file
```

7) Client 2 view all files available to download:



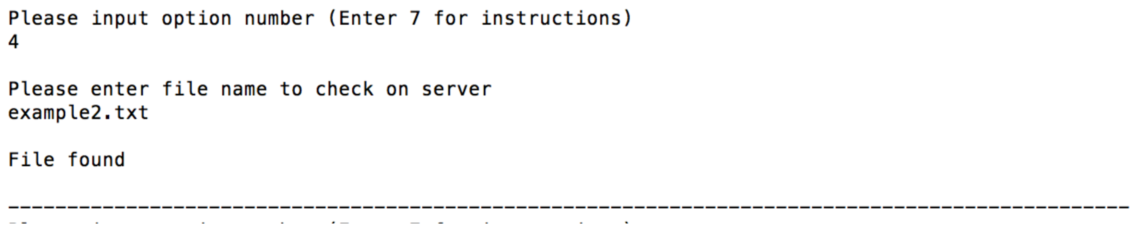
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
5

Availble Files to Download
example1.txt
example2.txt

------------------------------------------------------------------------------------------------
```

Client 1 Exits, Client 2 view all files available to download:



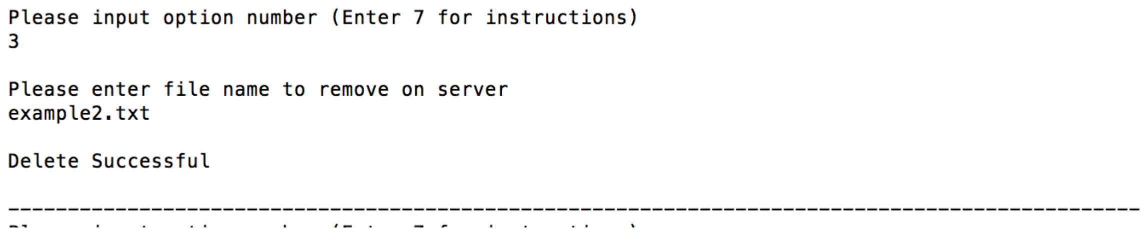
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
5

Availble Files to Download
example2.txt

------------------------------------------------------------------------------------------------
```

8) Client 2 uploads example2.txt. Client 1 searches for example2.txt:



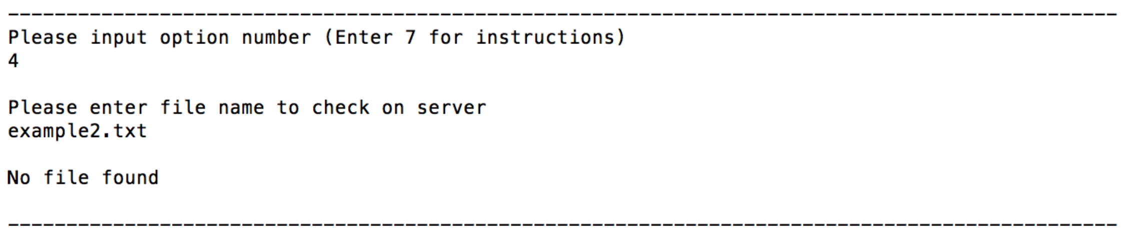
```
Please input option number (Enter 7 for instructions)
4

Please enter file name to check on server
example2.txt

File found

------------------------------------------------------------------------------------------------
```

Client 2 removes example2.txt



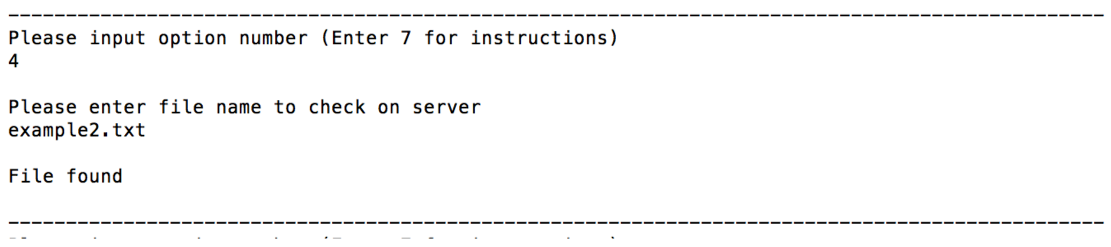
```
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
example2.txt

Delete Successful

------------------------------------------------------------------------------------------------
```

Client 1 searches for example2.txt



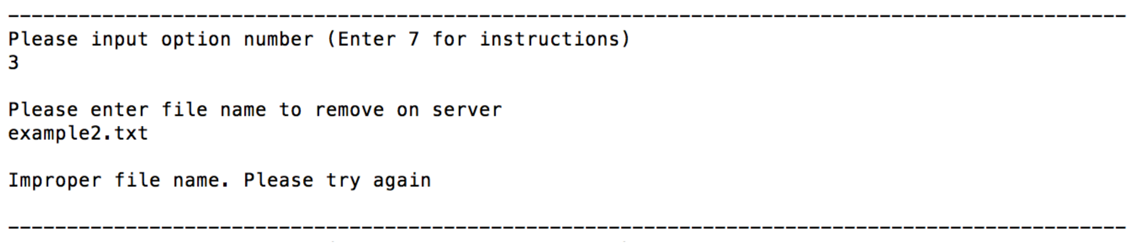
```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
4

Please enter file name to check on server
example2.txt

No file found

------------------------------------------------------------------------------------------------
```

9) Client 1 searches for example2.txt



```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
4

Please enter file name to check on server
example2.txt

File found

------------------------------------------------------------------------------------------------
```

10)Client 1 tries to remove client 2 file (example2.txt)



```
------------------------------------------------------------------------------------------------
Please input option number (Enter 7 for instructions)
3

Please enter file name to remove on server
example2.txt

Improper file name. Please try again

------------------------------------------------------------------------------------------------
```