Project Members:

Sakshi Sachdev

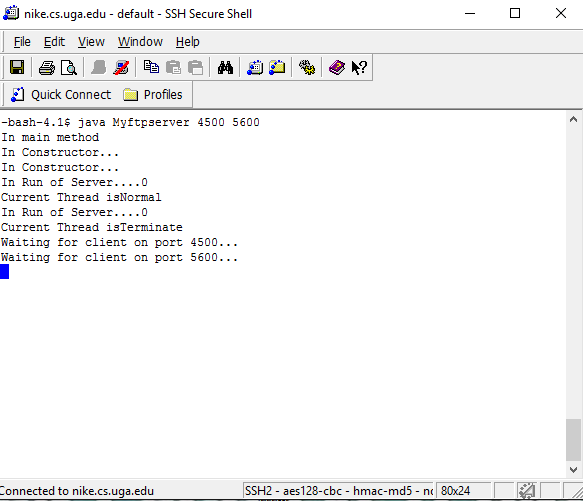
Anuja Jadhav

Steps:

1. Go to Server folder and open terminal in current directory. Type the following :

a. javac Myftpserver.java

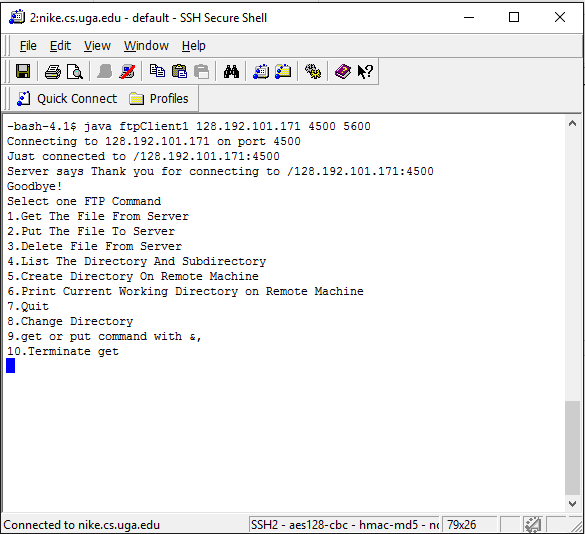
b. java Myftpserver 4500 5600



2. Go to Client1 folder and open new terminal in current directory. Type the following:

a. javac ftpClient1.java

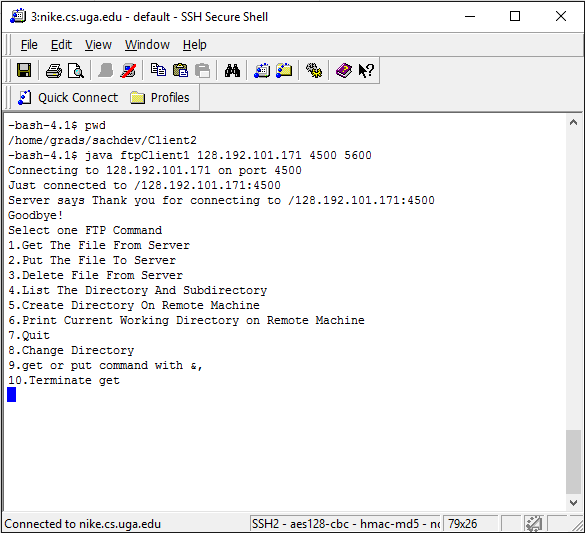
b. java ftpClient1 128.192.101.171 4500 5600



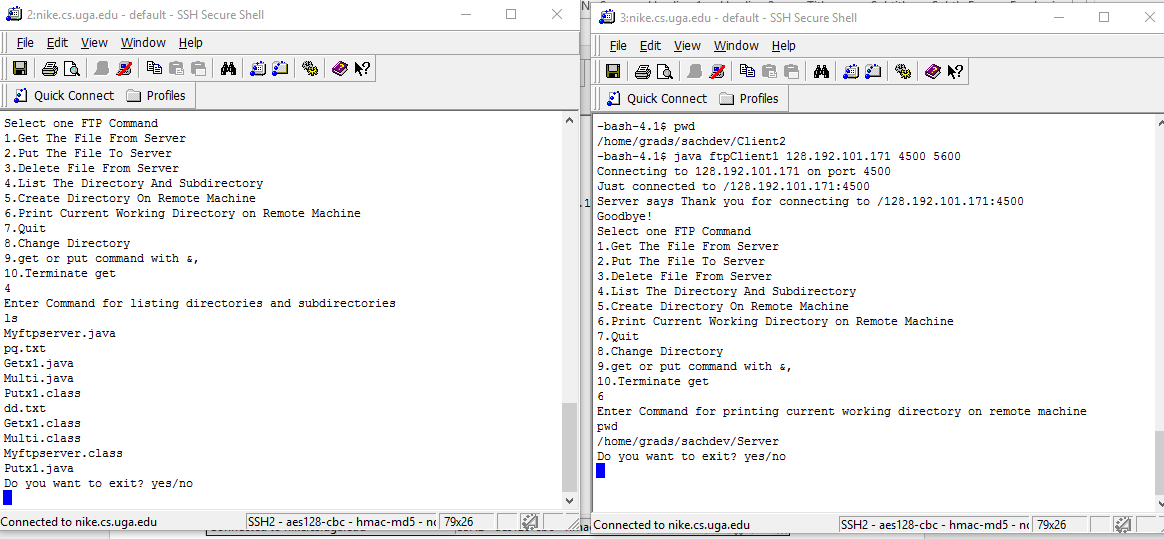
3. Go to Client2 folder and do the following:

a. javac ftpClient1.java

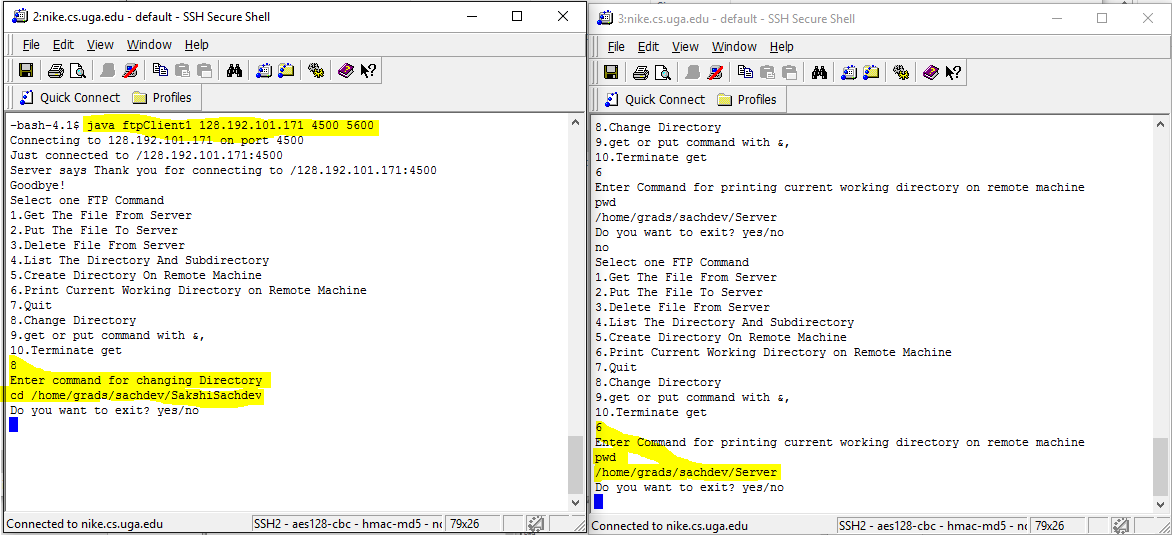
b. java ftpClient1 128.192.101.171 4500 5600



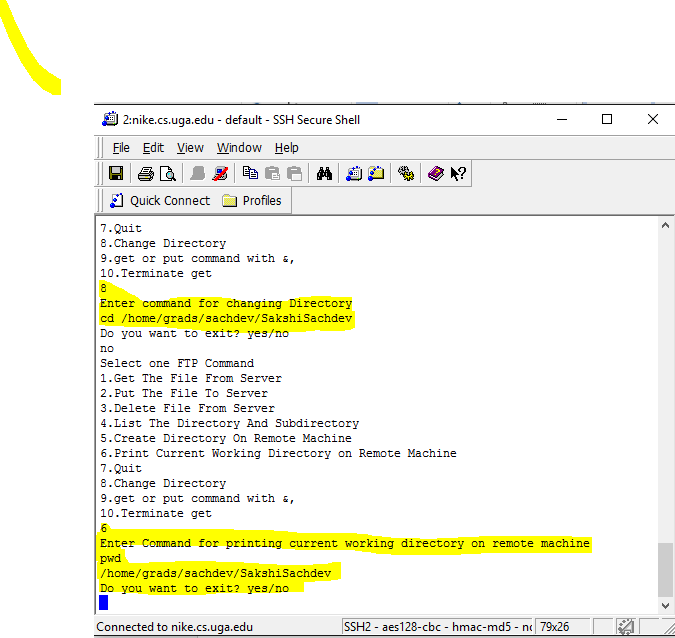
4. Multiple Clients running simultaneously:



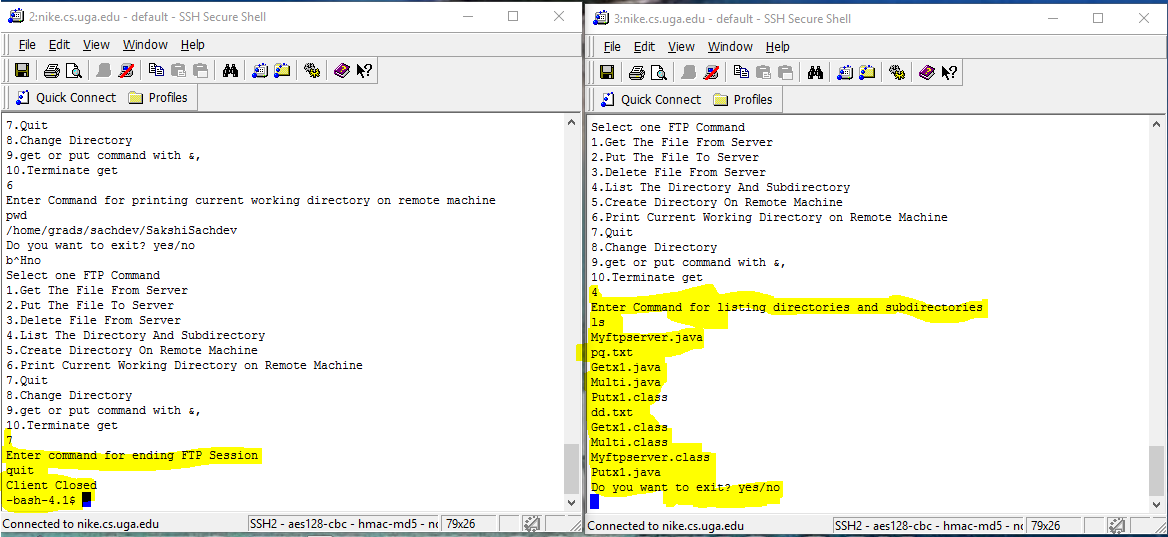
5.Change directory for client1 does not change directory for 2nd client



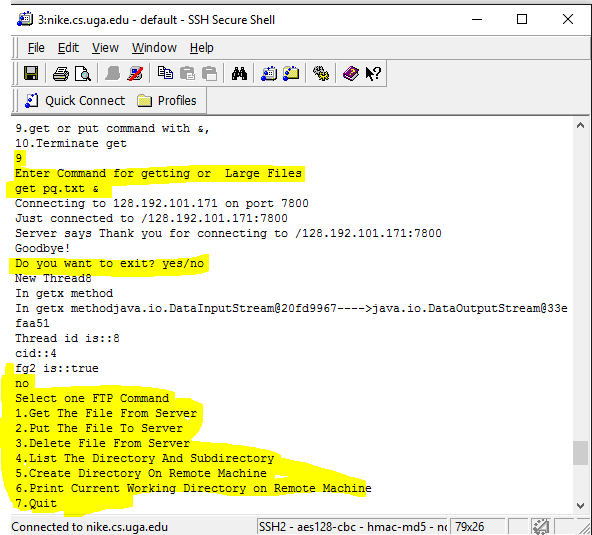
But Changes Directory for 1st Client



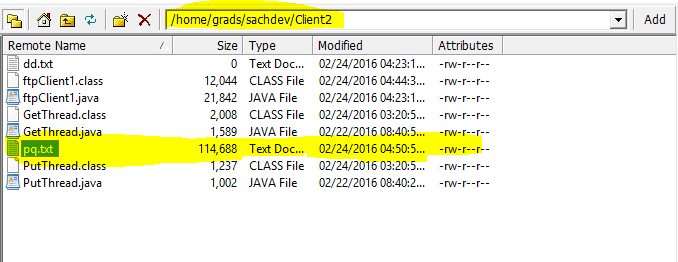
5. If 1st Client quits then 2nd will still Run



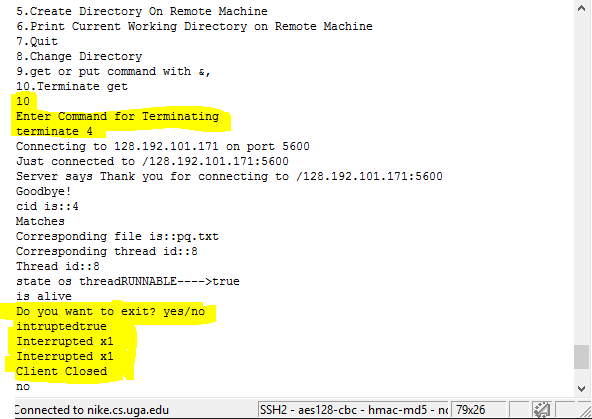
6. If Client does get pq.txt &,then large file gets reading, but simultaneously other commands can get executed.

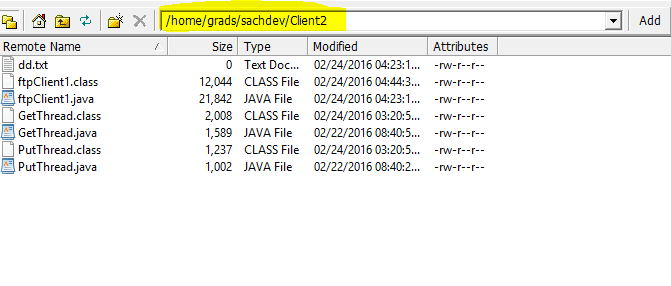


But, while file is being read, and many of the bytes are being read.File pq.txt at client side is as follows:



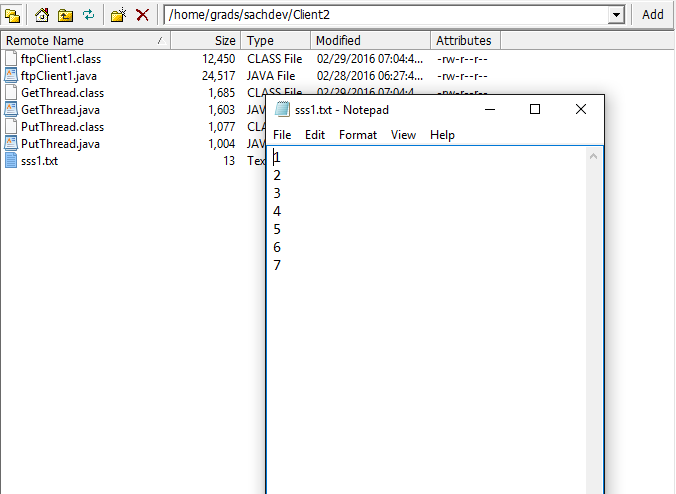
7.If in middle of reading a file client does terminate of that command id then,that file(pq.txt)(whatever bytes read ) will get deleted from client side.



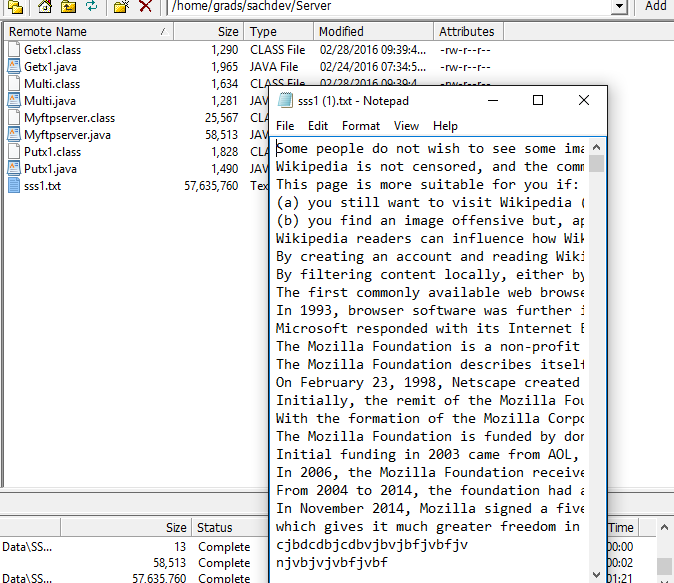
8. Read Write Synchronization:

If Client1 is reading something from file and Client2 write on same file

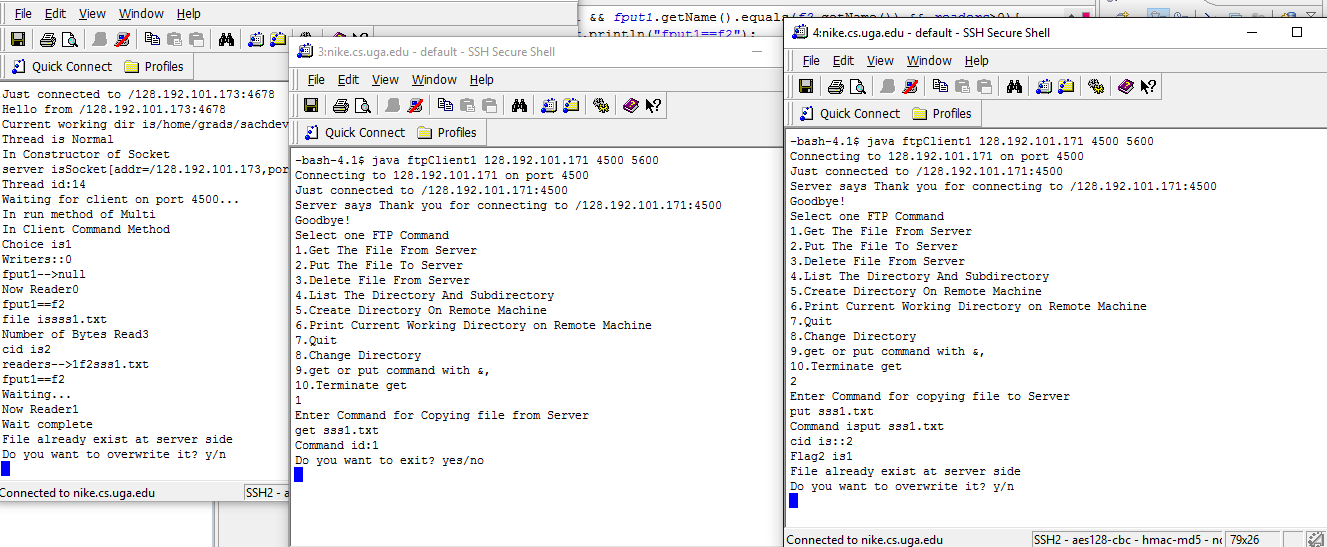
Then Client1 will get either 1st file or file written by 2nd Client , Client2 sss1.txt file is:



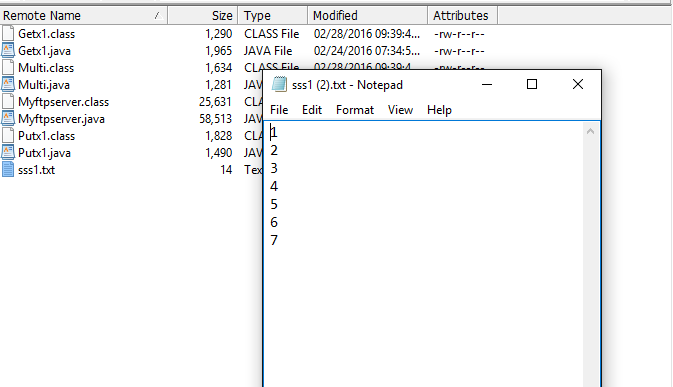
Server sss1.txt is:



When Client1 does get sss1.txt and Client2 put sss1.txt,then put (Client 2)will wait until get(Client1) is completed as below:

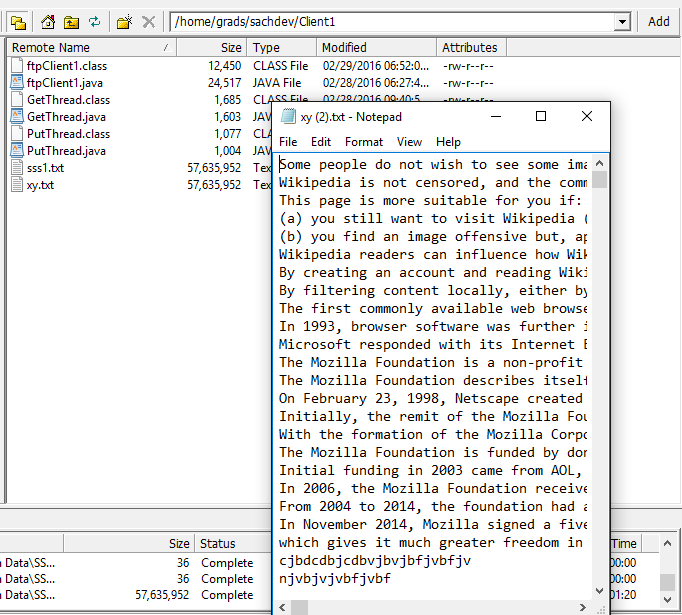


If Client2 does overwrite contents of Server then,server file is replaced by Client2 file as below:

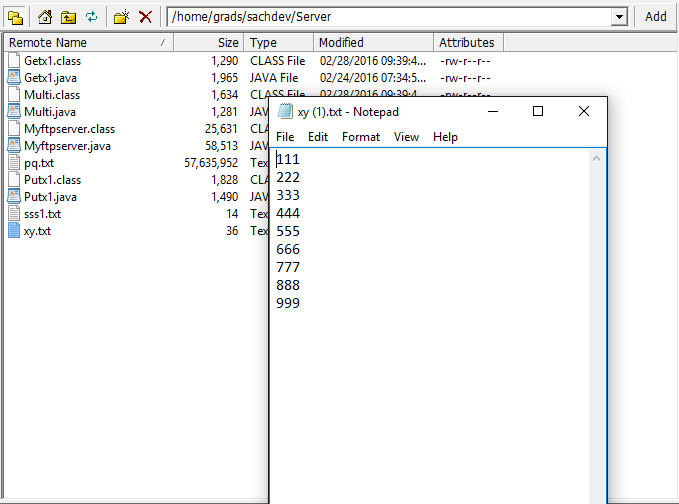


9. When Client1 is writing in Server and Client2 is reading contents of same file from server.Then Client2 waits until Client1 is finished:

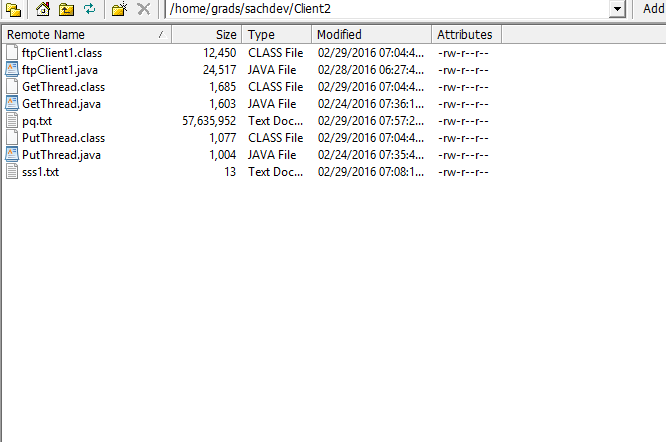
Client1 has xy.txt as:



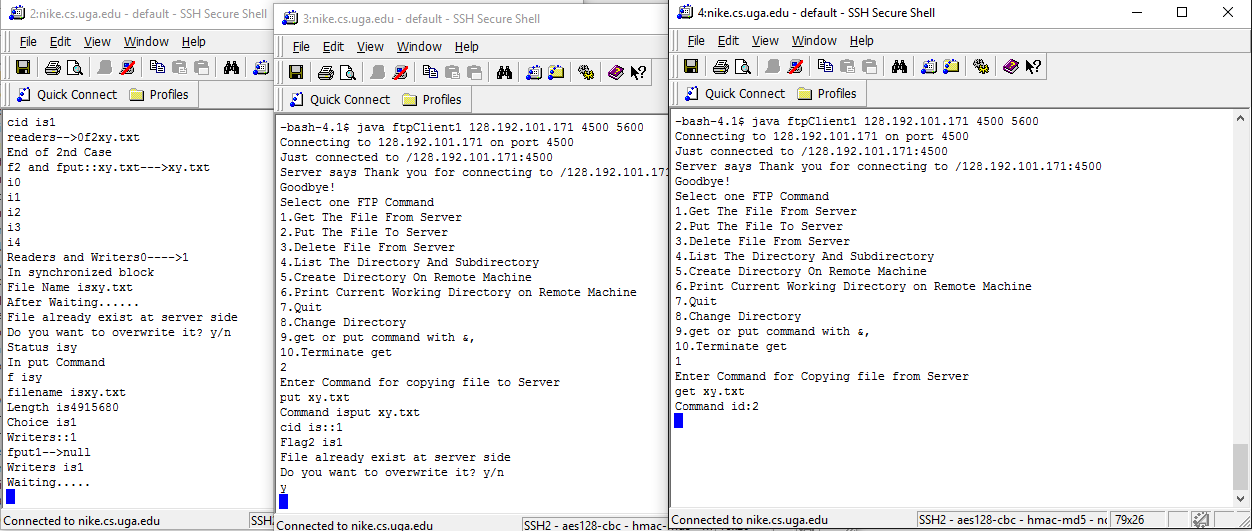
Server has xy.txt as:



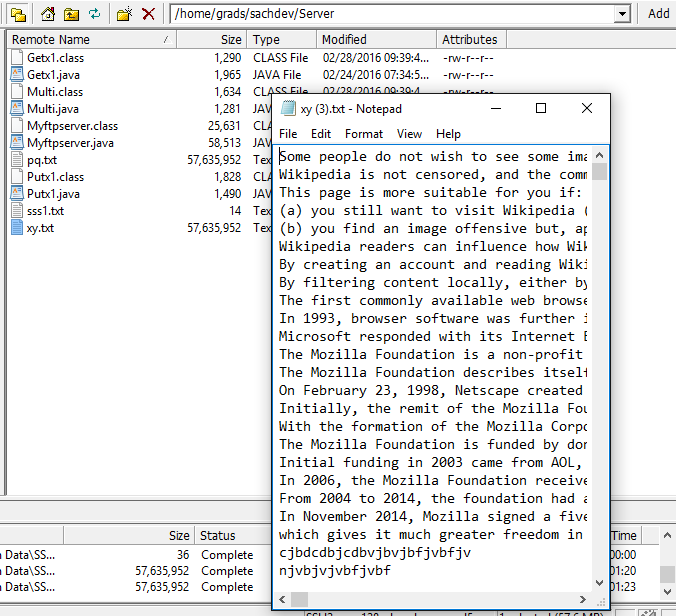
And Client2 does not has xy.txt

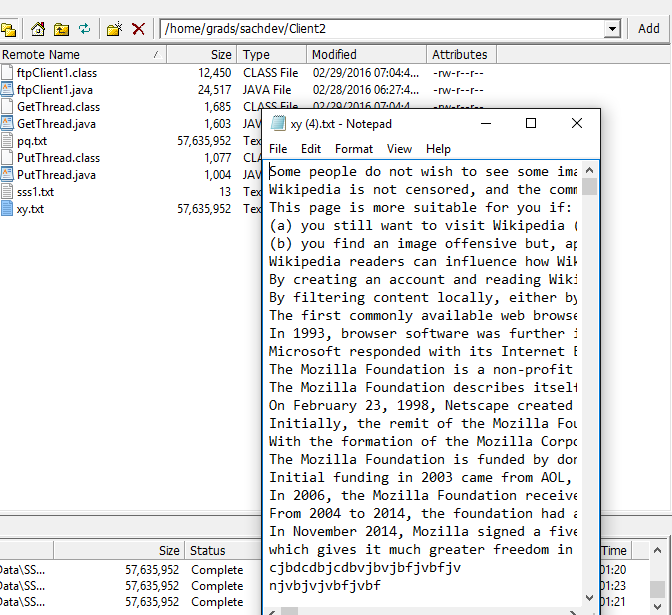


Suppose Client1 writes xy.txt to server and Client2 is trying to read it then Client2 has to wait till Client1 finishes writing as follows:



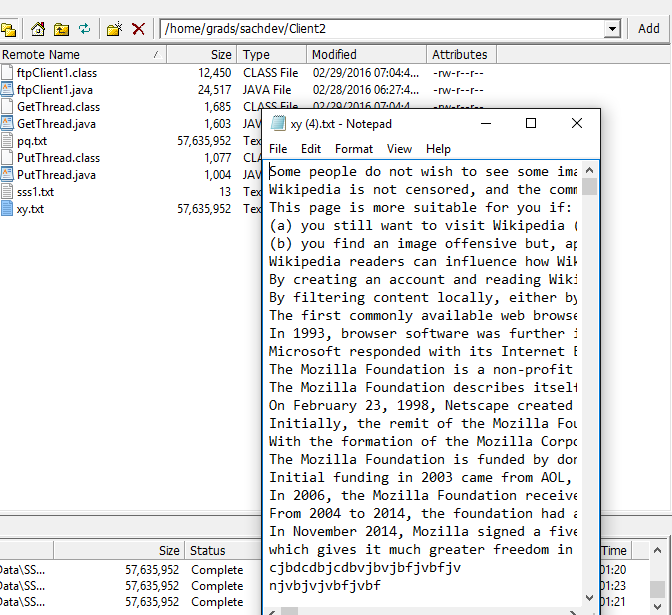
Once Writing by Client1 is finished then Client2 reads that file and final output is all Client2 and Server gets file written by Client1 as follows:

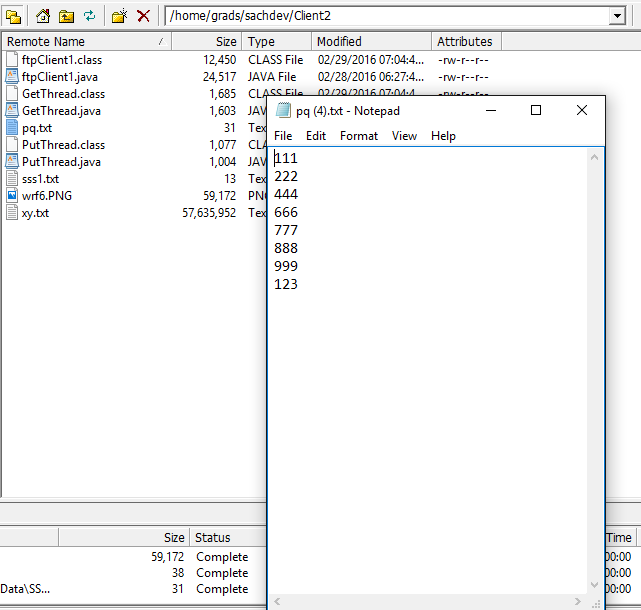




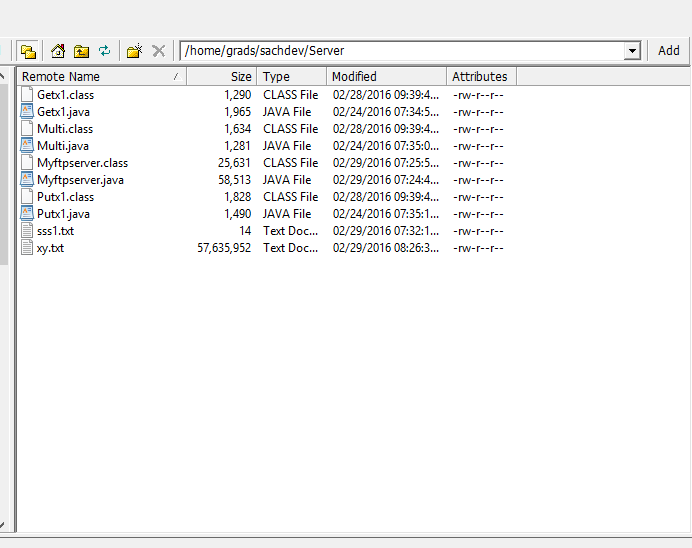
10.When Client2 and Client1 tries to write same file to Server.

Here pq.txt at Client1 and Client2 side as follows:





And server does not have pq.txt as follows:



When both Client1 and Client2 tries to put same file to server either Client2 file will be received by server or Client1’s file as follows:

