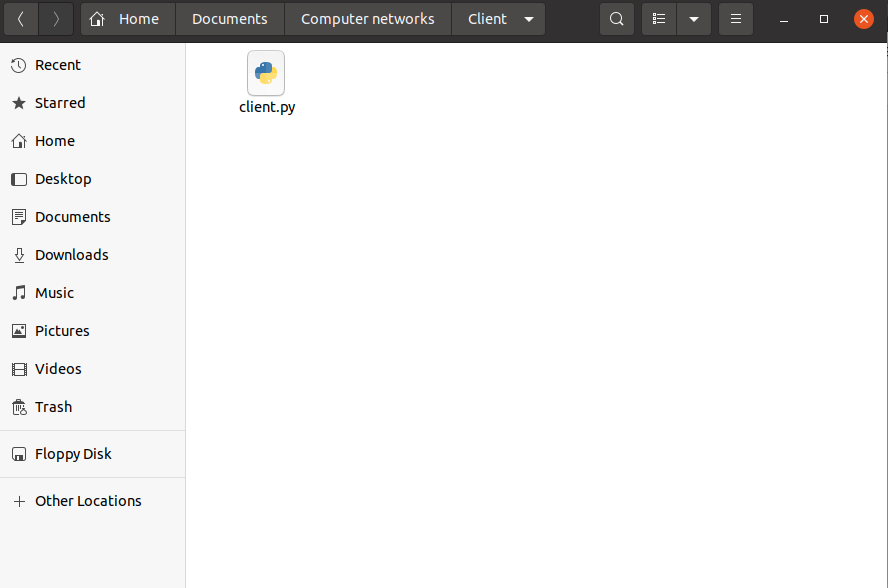
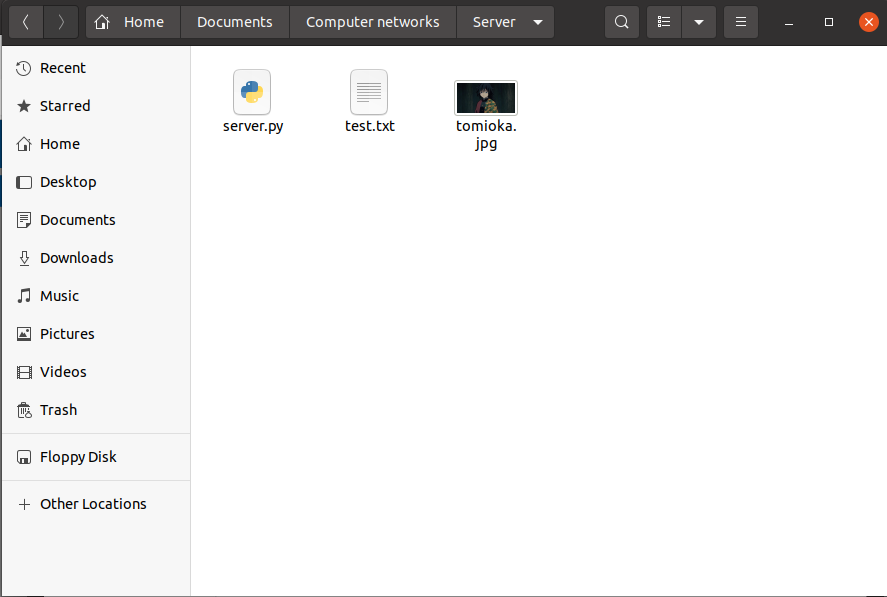
Assignment #2a: Reliable File Transfer

After reading the assignment, I decided to create two folders, one for the client and one for the server, and have the files that I would have liked to send in the server folder. I decided to start with copying the echo project that was given in the reference provided to us to get a better understanding of how to start the project. Once I had a better idea of how it worked, I then moved on to transfer files instead of just strings. Since I remembered that we did something similar in my Operating System class, I decided to use it as a reference. I was eventually able to successfully transfer a text file from the server to the client, and from there, decided that it would be best to start and try to transfer larger files. Using the lab from the Operating System class as a reference, I was able to eventually get it to transfer larger files.

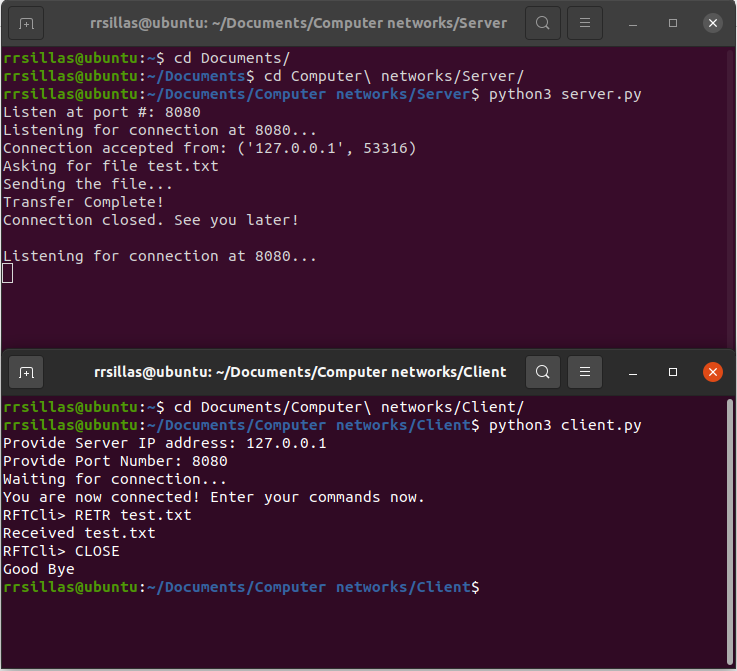
While starting off, the client folder would look like this:



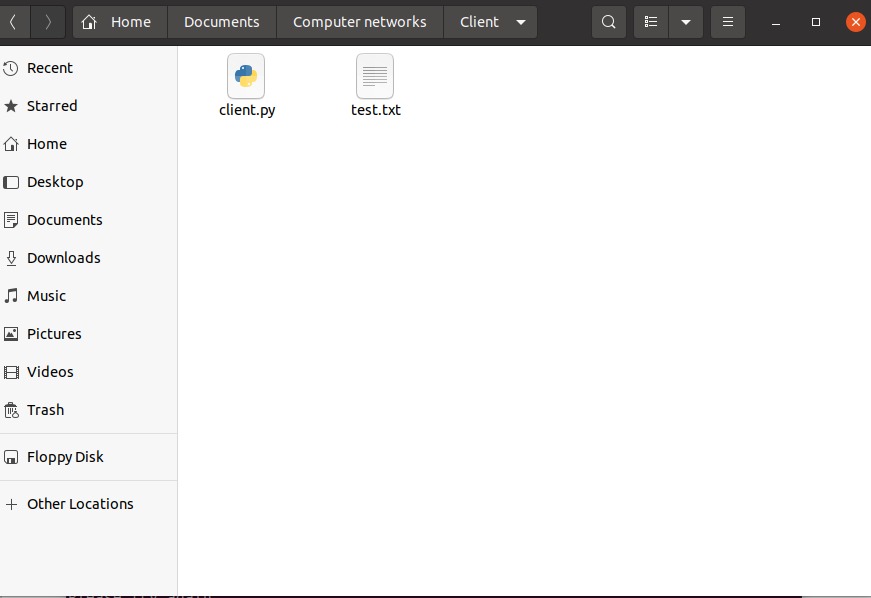
while the server folder would look like this:



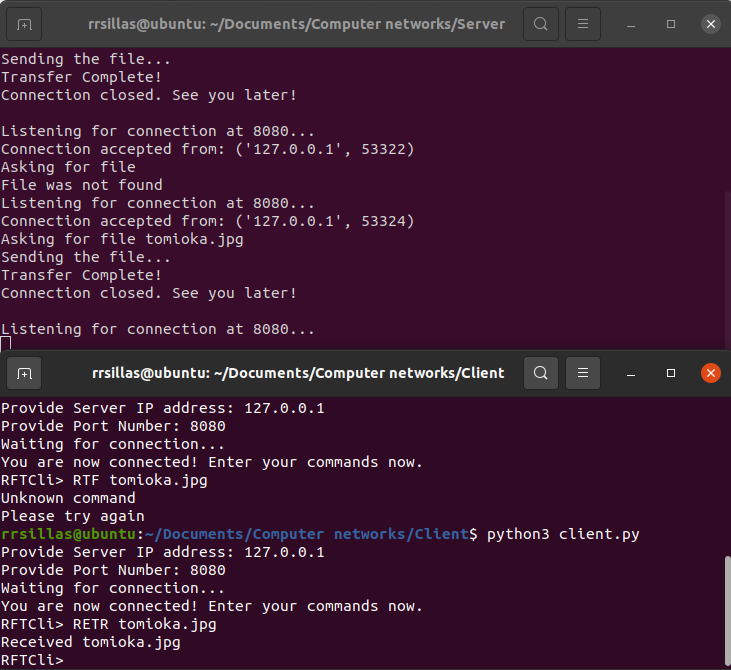
I decided to first transfer over the text file that was in the server file. This is what the terminals was showing:



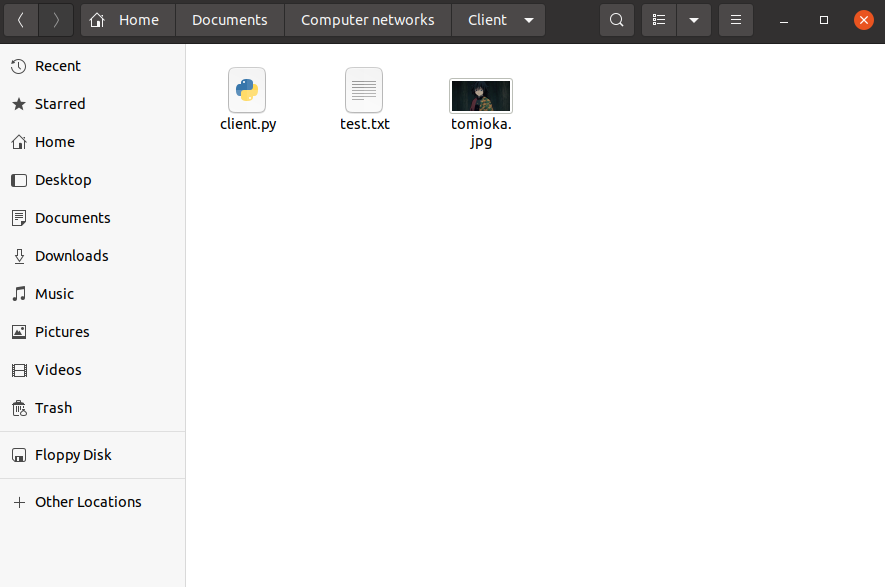
and this is what the client folder looked like:



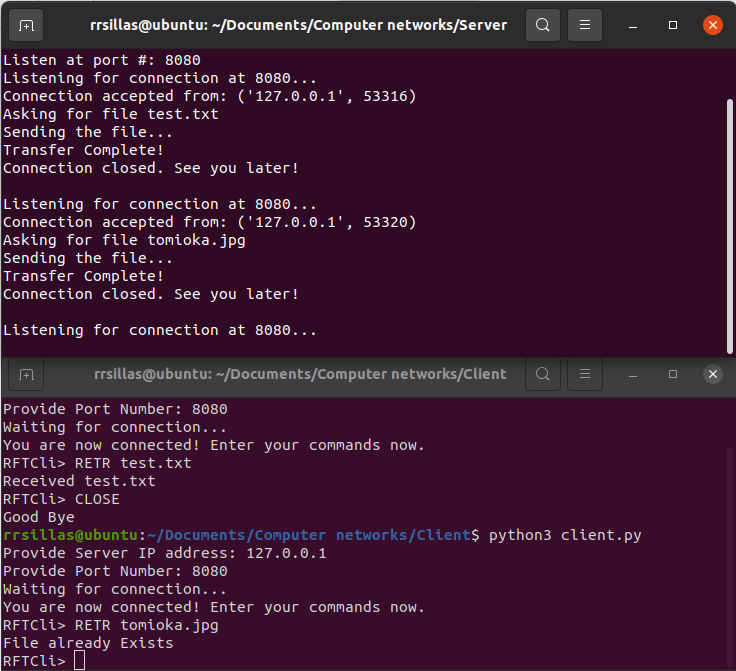
Then I decided to transfer over the jpg folder that was in the server folder. The terminals looked like this:



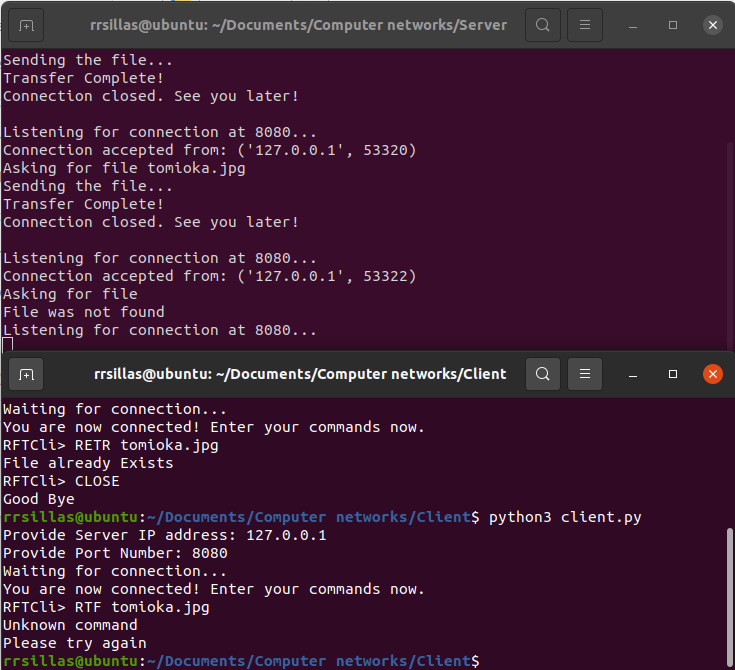
while the client folder looked like:



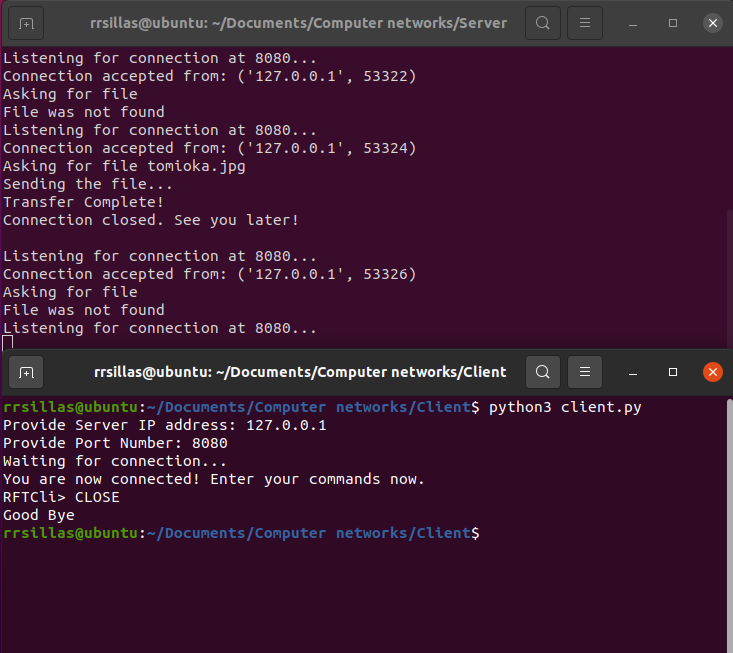
Then I decided to transfer the same picture over, which gave us this for the terminals:



As for if the command did not exist, the terminals would look like this:



Finally, this is what the terminals would display if the command close was to be inputted:



In order to run the code, you must have two terminals open, one for the client and one for the server. You must set up the server side first and then you should be able to set up the client side. In order to run the server side, you simply input “python3 server.py” into the terminal where server.py is located, and to set up the server, you must have “8080” as its port number. In order to run the client, you simply input “python3 client.py” into the terminal where client.py is located and to set up the client, you must input “127.0.0.1” as its IP address, and have “8080” as its port number. From there you are given the choice to input a command in the client side. The commands include “RETR” followed by the file you would like to transfer, and “CLOSE”. You must close and reopen the client side after every input to input something new.

References:

<https://realpython.com/python-sockets/>

<https://github.com/utep-cs-systems-courses/tcp-file-transfer-2-Ricardo-Sillas>