# Computer Networks I

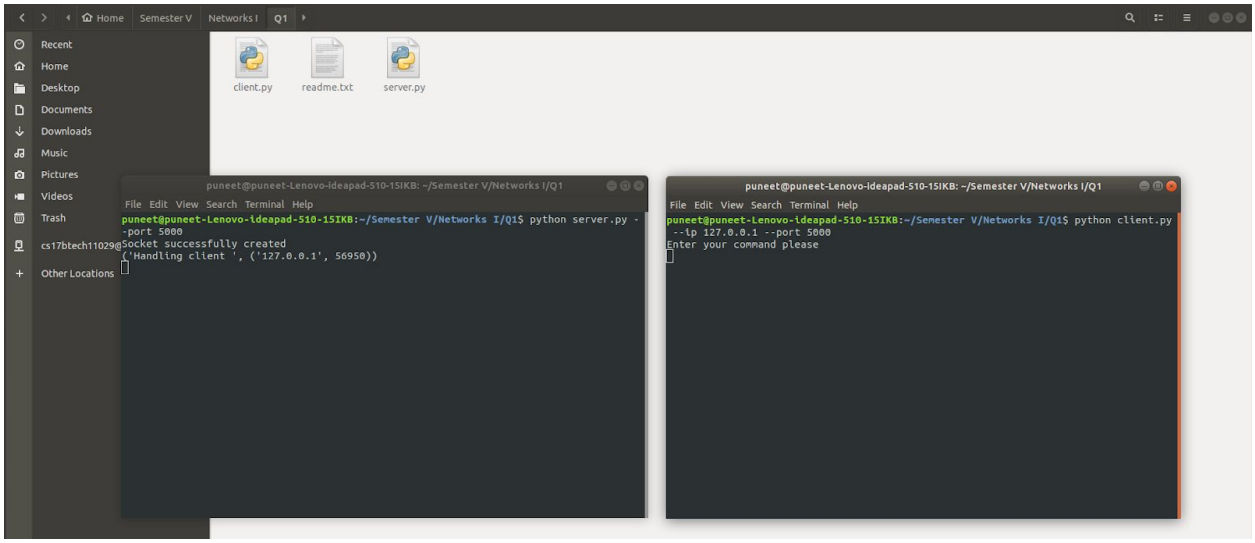
# Socket Programming Assignment

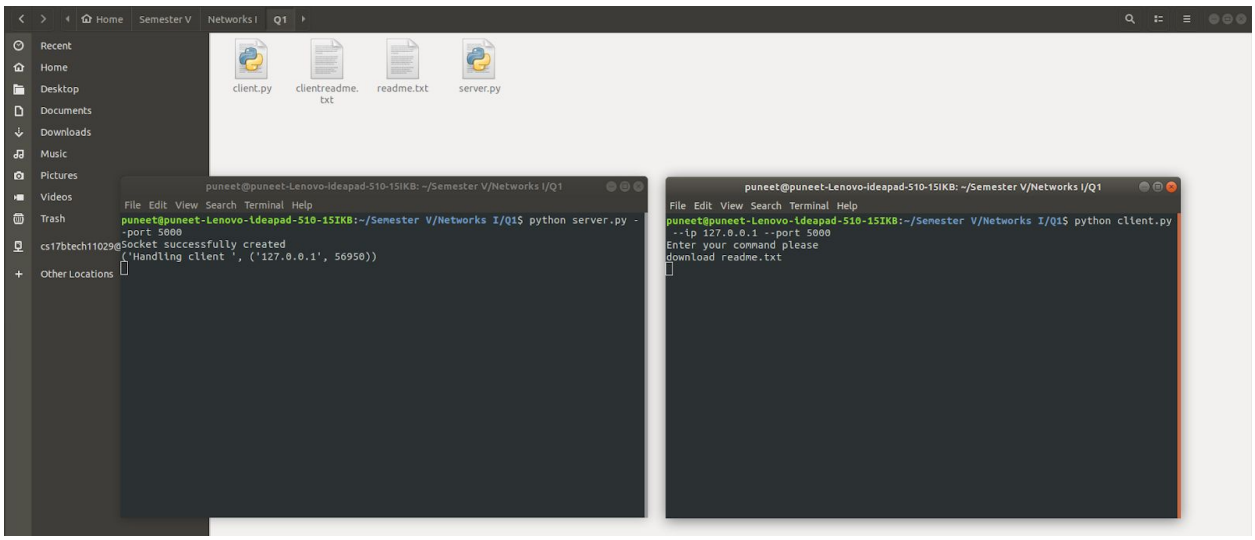
**Puneet Mangla (CS17BTECH11029)**

## Q1. Add two features to Echo Client /Server, and demonstrate them

***Feature 1*** *: Download file from server*

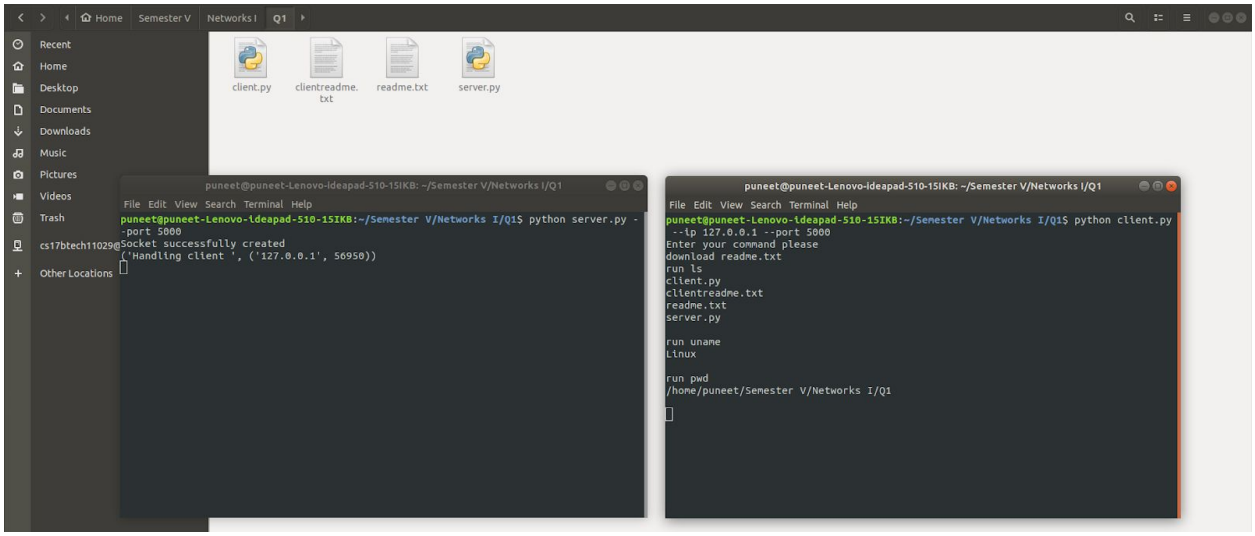
- Download any file present on server to client.
- Command : *`download < file_name >`*. Eg. *`download readme.txt`*
- Another file with prefix `client_` will be formed. Eg. `client_readme.txt`
- Significance : If one wants to create a local copy of a file, download a file to modify/use it.



***Feature 1** : Run and see output of terminal commands*

- Execute commands on the server and see output on client side.
- Command : `run < command >`. Eg. `run ls, run ps aux`
- Significance : Useful to check what process are currently running on server. Perform listing files, copying them to some other place etc.
- Will not be able to see the output of commands like `run cd` etc.




***References***

1. Socket Library : https://docs.python.org/2/library/socket.html
2. Sample socket programming codes in python : https://www.geeksforgeeks.org/socket-programming-python/
3. Argparse library for command line arguments : https://docs.python.org/2/howto/argparse.html

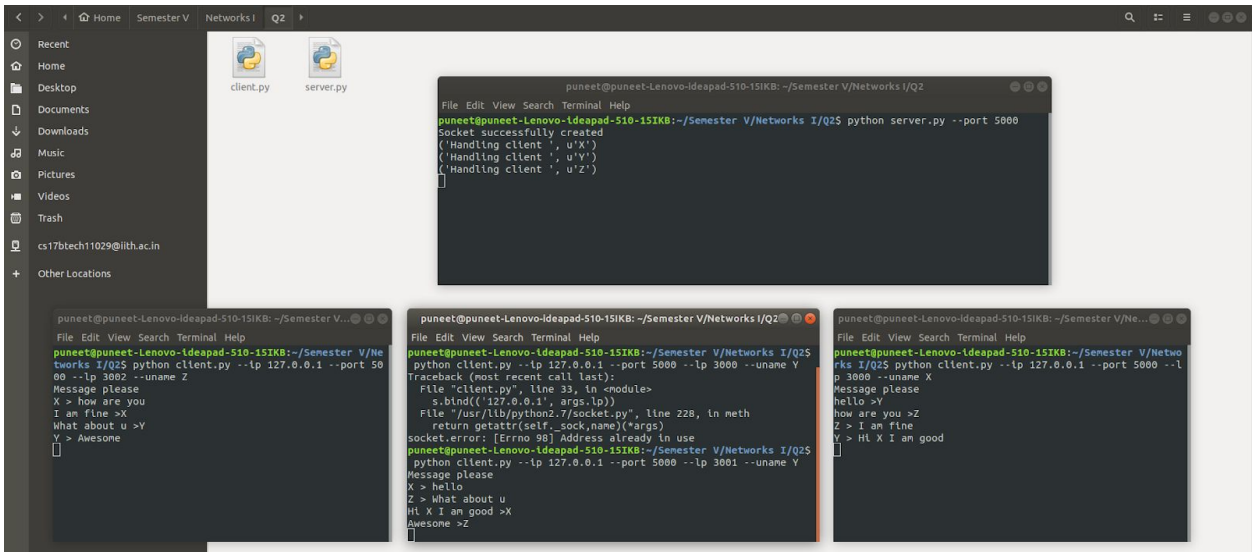
**Q2. HARD Mode: 1 server and N clients. Say N clients may connect to the server. Client 1 wants to talk with client m. How do you manage multiple clients and select the one you want to talk to?**

- Specify the listening port and your username to be used.
- Send a message as `<message> |<username_to_send>`.
- Message received from other user will look like : `<username_who_sent>| <message>`

- For each user, server maintains a map from its username to its corresponding socket. Server main thread spawns a thread for each user in the network that handles the message communication between two clients.

***References***

1. Socket Library : https://docs.python.org/2/library/socket.html
2. Sample socket programming codes in python : https://www.geeksforgeeks.org/socket-programming-python/
3. Argparse library for command line arguments : https://docs.python.org/2/howto/argparse.html
4. Threading library : https://docs.python.org/2/library/threading.html




**Q3. Revise echo client and server to be protocol independent (support both IPv4 and IPv6).**

- For server run as usual. Eg. `python server.py --port 5000`
- For client specify the corresponding IP address of server (ipv4/ipv6). Eg `python client.py --ip [::1/127.0.0.1] --port 5000`
- You can download and run shell commands likewise Q1.
- The server runs two threads. One thread accepts ipv6 connections whereas other thread accepts ipv4. The thread corresponding to client provided ip serves while other runs ideal and waits for connection.

***References***

1. Socket Library : https://docs.python.org/2/library/socket.html
2. Sample socket programming codes in python : https://www.geeksforgeeks.org/socket-programming-python/
3. Argparse library for command line arguments : https://docs.python.org/2/howto/argparse.html
4. Threading library : https://docs.python.org/2/library/threading.html