

**ASSIGNMENT NO.1 REPORT:**

CS3001-COMPUTER NETWORKS

SECTION: BCS-6E

GROUP MEMBERS:

MUHAMMAD ANAS HASSAN -> 20k-1726

MUHAMMAD WARZAN -> 20k-1649

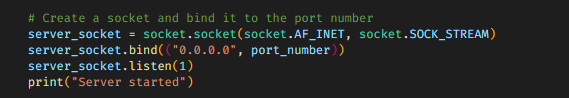
**SOCKET PROGRAMMING WITH GUI IN PYTHON:**

Socket programming with TCP (Transmission Control Protocol) is a method of establishing communication between two devices, end-points or nodes (such as a client and a server) over a network.

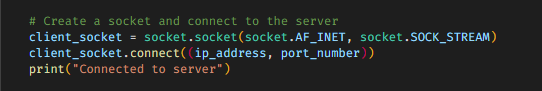
A socket is an end-point for sending or receiving data across a network. In TCP, data is sent as a stream of bytes, with the protocol providing a reliable, ordered and error-checked delivery of data.

Here is a brief overview of how to use socket programming with TCP in Python:

Create a server\_socket object on the server side, and bind it to a specific port.

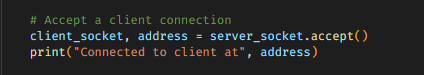


On the client side, create a Socket object, and connect it to the server's IP address and port number.



On the server side, use the accept() method of the server\_socket object to wait for an incoming connection from the client.

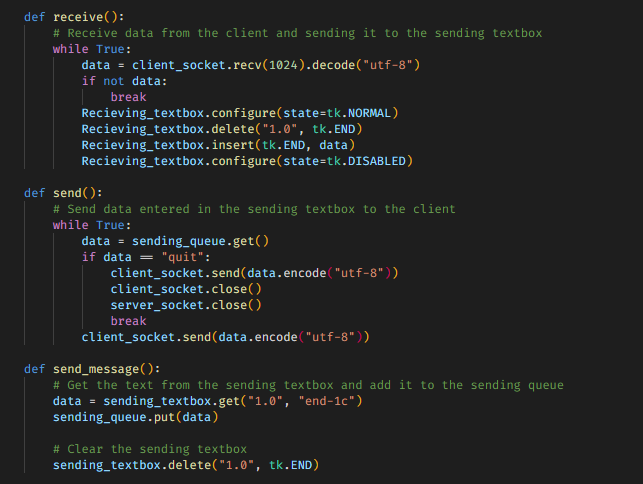
This method blocks the program until a connection is made.



In GUI part, we have created two text boxes for the sending and receiving messages and a send button.

Its function is to send the message from the sending queue to the other side.

On both the server and client side, some methods are implemented for the sending and receiving message purposes.

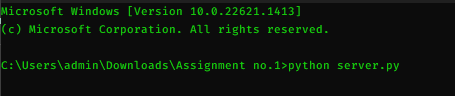


Now onwards, we are looking for the demonstration. We use python library named tkinter to create a simple GUI for the Client Server Application. The purpose of GUI is to make it easier for the user to understand the flow of the work.

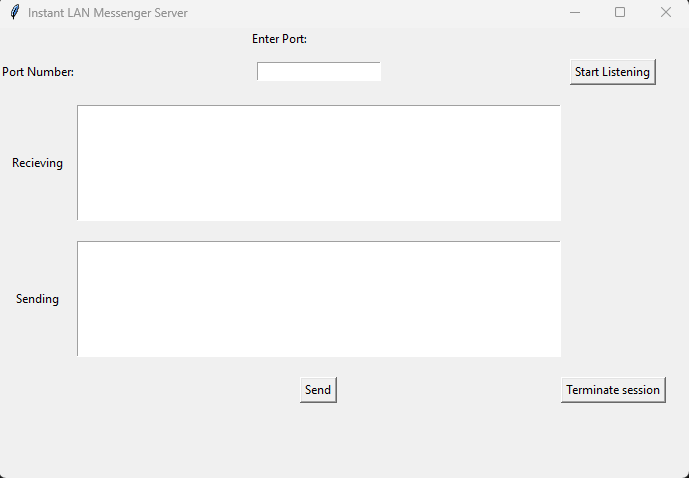
For implementing GUI in python, we have used tkinter library also the threading, socket, queue libraries.



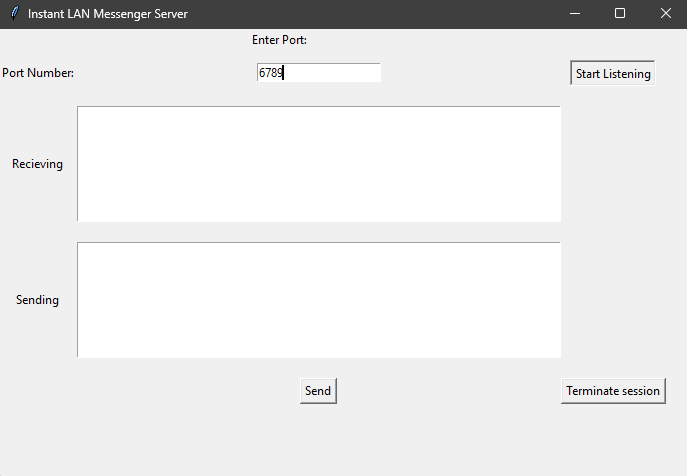
The first step that you have to do is to open the command prompt and run the server.py file.



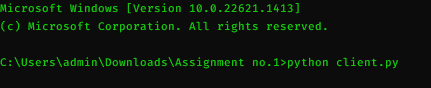
After running the server.py file, the Instant LAN Messenger Server dialog box appears as shown below:



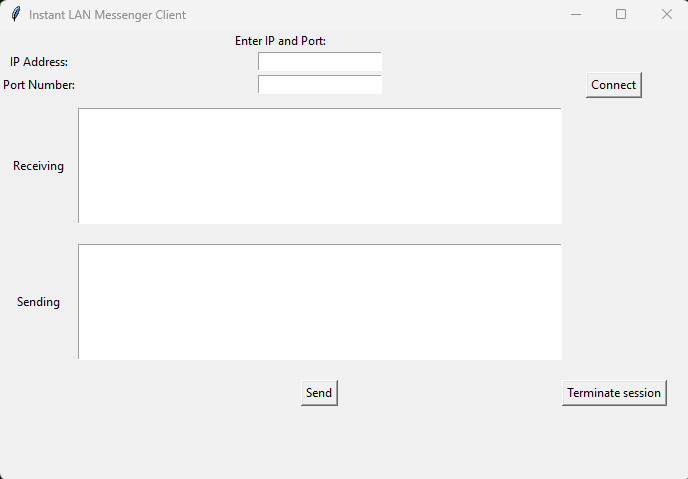
In this dialog box, firstly you will be asked for the port number, enter the port number and then click on start listening button, by clicking this button the program goes into the block state until any connection with a client has been made.



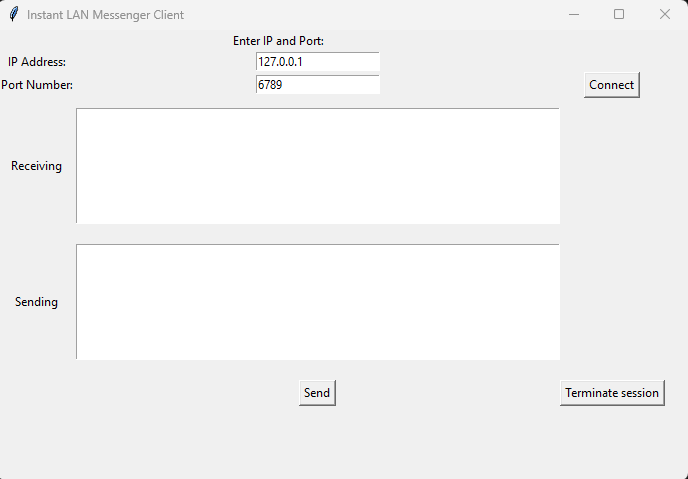
So, in order to establish a connection we have to run the client side (client.py file) now in command prompt as shown below:



After running the client.py file, the Instant LAN Messenger Client dialog box appears as shown below:

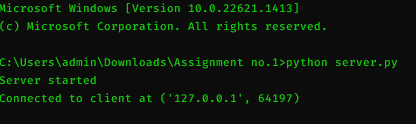


In this dialog box, firstly you will be asked for the IP address and the port number, enter the IP address and the port number and then click on connect button, by clicking this button a connection established successfully between the client and the server.

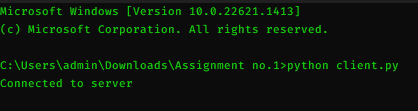


After clicking on the connect button on the client side, the connection between the client and the server established successfully. And the confirmation of that established connection has also been (printed) shown on each of their terminals as shown below:

On the server side;

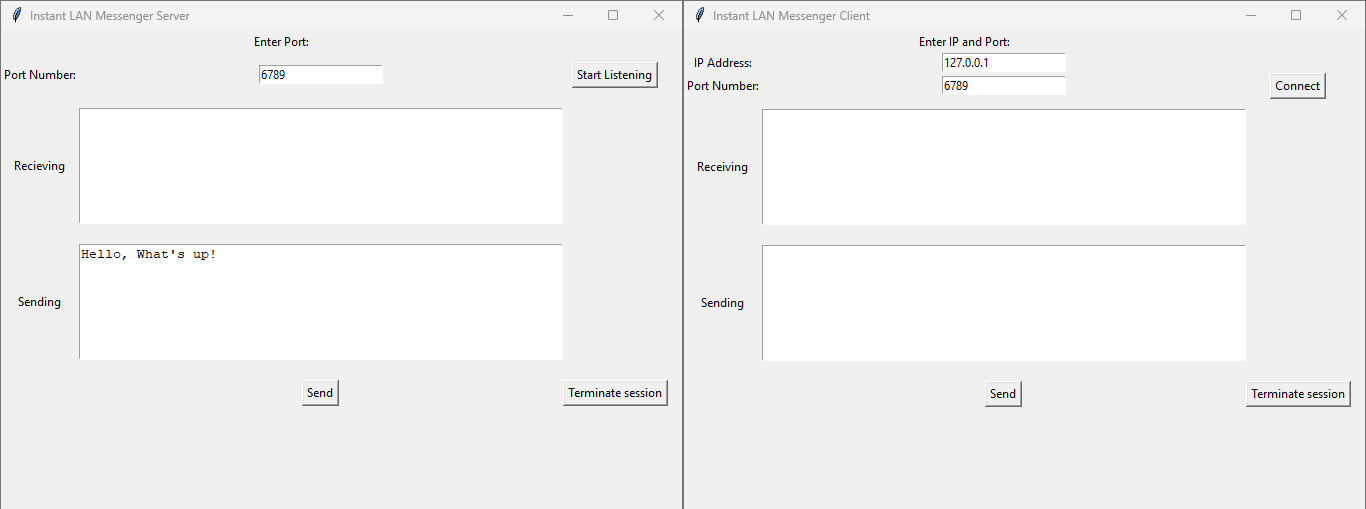


On the client side;

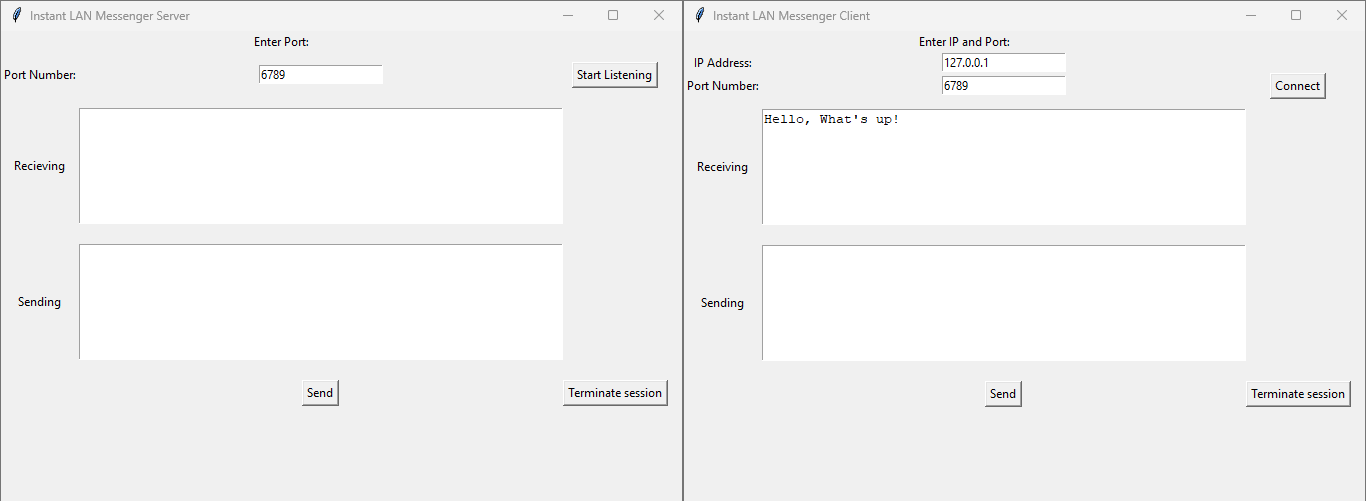


Now onwards as the connection is established between the client and the server, therefore they can communicate with each other easily by sending and receiving messages. In GUI part, we have created two text boxes for the sending and receiving messages and a send button. Its function is to send the message from the sending queue to the other side.

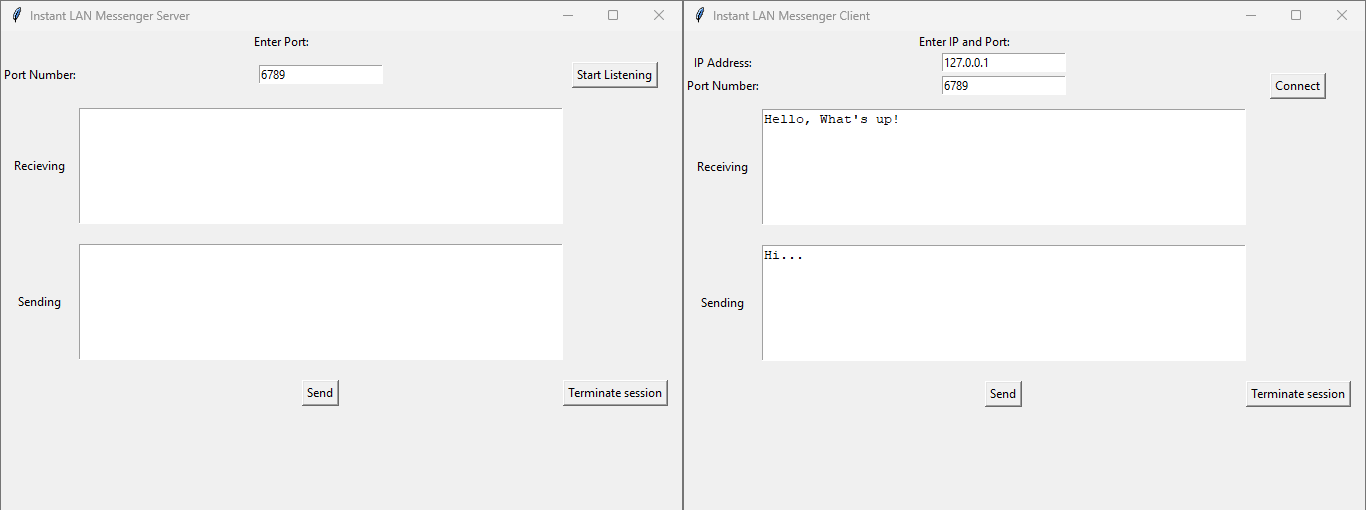
Firstly, anyone can send message to the other either the server or the client. In this case, the server is starting the conversation by sending a message to the client from the sending text box.

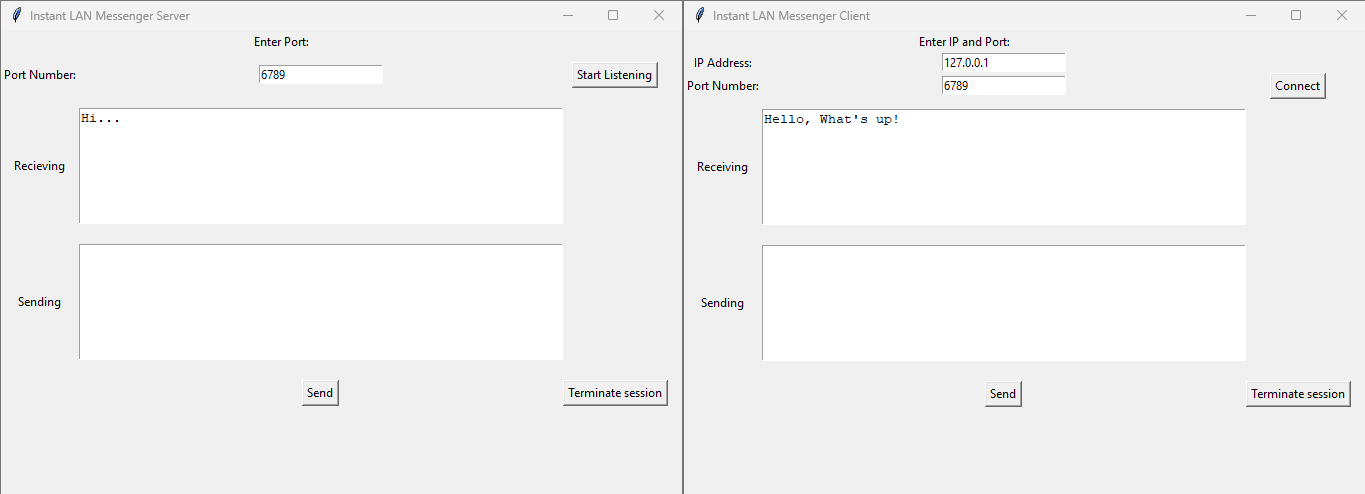


Now that message will be received by the client at the receiving text box.

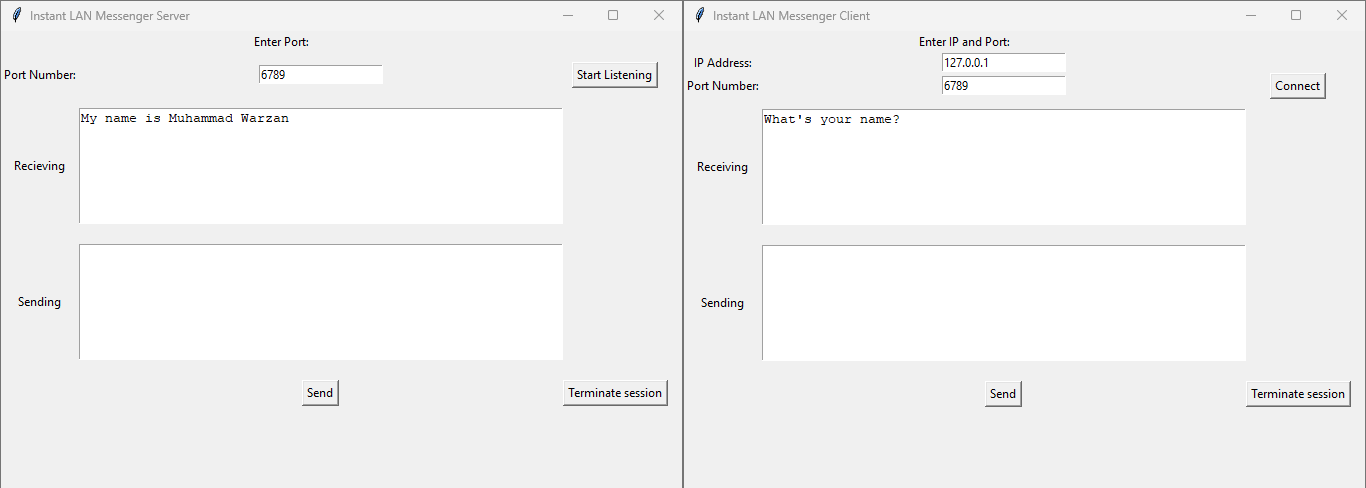


Now it’s the client’s turn, the client then sends a response/reply message to the server by writing the message in the sending text box and then clicking on the send button.





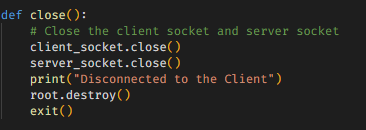
And this is how the conversation goes between the client and the server.



Now, after a long conversation between the client and the server, if any one of them wants to quit the conversation so they can quit the conversation by simply writing “quit” in the sending text box which then sends to the other side (means the member on the other side can see the “quit” message and then can’t send message back to you or if it does, it gives an error which is shown in command prompt).

They can also quit the conversation by simply clicking the terminate session button at the bottom right corner, by clicking on this button the connection between the client and the server is disconnected.

If server wants to terminate the session, the method called is;



If client wants to terminate the session, the method called is;

