* In your write-up explain why you need two new classes for threading
* There needs to be two runnables because it needs threading for both the Client and Server side.
* In your write-up, explain why there should be a separate class to receive data from the server and print it, and the client only gets data from the user and sends it to the server.Also, why is the class called a ‘listener’?
* There is a separate class for data so it can be printed and received no matter what, regardless of if the user is sending a message or not. Otherwise this isn't possilbe because read and send Data would only occur before or after receive and print Data. The class is called a listener because it basically 'istens' for data to receive from the server, only doing anything if there is data.
* In your write-up, explain why you need a separate thread for each client, and why you cannot handle all clients in the main server thread. Conceptually, why is the listener class ‘ClientSideServerListener’ different from the class ‘ServerSideClientIO’?
* Each client could send a message at the same time as receiving or broadcasting a message from another client. The server also handles the general sending aspect of the data, and the client handles receiving and printing data.
* In your write-up, explain why the broadcast() and remove() methods are synchronized. You may find it easier to answer this question after completing all programming.
* Broadcast goes through the entire list of serverSideClientIOList, but remove only removes on element of that list. If we removed during the broadcast the loop would go out of bounds of the array, and we would not be able to see the broadcast in other clients because they aren't synchronized.
* In your write-up, discuss all new methods and new code in existing methods that you wrote to handle LISTUSERS.
* We made a new class ListUsers that extends ClackData. This allows the request for the list of users to be sent just like any other message or file. Apart from MessageData and FileData, ListUsers doesn't have any data besides the name of the user that sent it and a type int with the value CONSTANT\_LISTUSERS. Also, several changes were made to ClackServer and ServerSideClientIO. ServerSideClientIOList also has a new updateUserName method which updates the userName whever the user sends data.