**Client**

import java.io.\*;

import java.net.\*;

public class Client{

private Socket socket=null;

private PrintWriter out=null;

private BufferedReader input=null;

private BufferedReader serverInput=null;

public Client(String address, int port){

try{

socket=new Socket(address, port);

System.out.println("Client connected");

serverInput=new BufferedReader(new InputStreamReader(socket.getInputStream()));

out=new PrintWriter(socket.getOutputStream(), true);

input=new BufferedReader(new InputStreamReader(System.in));

Thread readThread=new Thread(()->{

try{

String line;

while((line=serverInput.readLine())!=null)

{

System.out.println("Server says: "+line);

}

} catch(IOException e){

System.out.println("Server Disconnected");

}

});

readThread.start();

String line;

while(!(line=input.readLine()).equals("Over"))

{

System.out.println("Client Says: "+line);

out.println(line);

}

} catch(IOException i)

{

System.out.println("Client Disconnected");

}

}

public static void main(String args[])

{

new Client("127.0.0.1", 5000);

}

}

**Server**

import java.io.\*;

import java.net.\*;

public class Server {

    private Socket socket = null;

    private ServerSocket server = null;

    private BufferedReader clientInput = null;

    private PrintWriter out = null;

    private BufferedReader input = null;

    public Server(int port) {

        try {

            server = new ServerSocket(port);

            System.out.println("Server started");

            System.out.println("Waiting for a client ...");

            socket = server.accept();

            System.out.println("Client accepted");

            clientInput = new BufferedReader(new InputStreamReader(socket.getInputStream()));

                out = new PrintWriter(socket.getOutputStream(), true);

                input = new BufferedReader(new InputStreamReader(System.in));

            // Thread to handle client messages

            Thread readThread = new Thread(() -> {

                String line;

                try {

                    while ((line = clientInput.readLine()) != null) {

                        System.out.println("Client says: " + line);  // Debug line for server

                    }

                } catch (IOException e) {

                    System.out.println("Connection closed by client.");

                }

            });

            readThread.start();

            // Main thread for sending messages to client

            String line;

            while (!(line = input.readLine()).equals("Over")) {

                System.out.println("Server sending: " + line);  // Debug line for server

                out.println(line);

            }

            // Close everything

            socket.close();

            clientInput.close();

            out.close();

            input.close();

        } catch (IOException i) {

            System.out.println("Server error: " + i);

        }

    }

    public static void main(String args[]) {

        new Server(5000);

    }

}