/\*

Name : Rohit Narayan Telgote

PRN : 1941054

Batch : B4

\*/

// Aim : Program for distributed chat server using TCP sockets

**Server.java**

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.io.PrintStream;

import java.net.ServerSocket;

import java.net.Socket;

import java.util.Scanner;

import java.util.concurrent.ExecutorService;

import java.util.concurrent.Executors;

import java.util.logging.Level;

import java.util.logging.Logger;

public class Server {

int port;

ServerSocket server = null;

Socket client = null;

ExecutorService pool = null;

int clientcount = 0;

public static void main(String[] args) throws IOException {

Server serverobj = new Server(5000);

serverobj.startServer();

}

Server(int port) {

this.port = port;

pool = Executors.newFixedThreadPool(5);

}

public void startServer() throws IOException {

server = new ServerSocket(5000);

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

System.out.println("Any client can stop the server by sending -1");

while (true) {

client = server.accept();

clientcount++;

ServerThread runnable = new ServerThread(client, clientcount, this);

pool.execute(runnable);

}

}

private static class ServerThread implements Runnable {

Server server = null;

Socket client = null;

BufferedReader cin;

PrintStream cout;

Scanner sc = new Scanner(System.in);

int id;

String s;

ServerThread(Socket client, int count, Server server) throws IOException {

this.client = client;

this.server = server;

this.id = count;

System.out.println("Connection " + id + "established with client " + client);

cin = new BufferedReader(new InputStreamReader(client.getInputStream()));

cout = new PrintStream(client.getOutputStream());

}

@Override

public void run() {

int x = 1;

try {

while (true) {

s = cin.readLine();

System.out.print("Client(" + id + ") :" + s + "\n");

System.out.print("Server : ");

// s=stdin.readLine();

s = sc.nextLine();

if (s.equalsIgnoreCase("bye")) {

cout.println("BYE");

x = 0;

System.out.println("Connection ended by server");

break;

}

cout.println(s);

}

cin.close();

client.close();

cout.close();

if (x == 0) {

System.out.println("Server cleaning up.");

System.exit(0);

}

} catch (IOException ex) {

System.out.println("Error : " + ex);

}

}

}

}

**Client.java**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.io.PrintStream;

import java.net.ServerSocket;

import java.net.Socket;

public class Client {

public static void main(String args[]) throws Exception {

Socket sk = new Socket("127.0.0.1", 5000);

BufferedReader sin = new BufferedReader(new InputStreamReader(sk.getInputStream()));

PrintStream sout = new PrintStream(sk.getOutputStream());

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

String s;

while (true) {

System.out.print("Client : ");

s = stdin.readLine();

sout.println(s);

if (s.equalsIgnoreCase("BYE")) {

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

break;

}

s = sin.readLine();

System.out.print("Server : " + s + "\n");

}

sk.close();

sin.close();

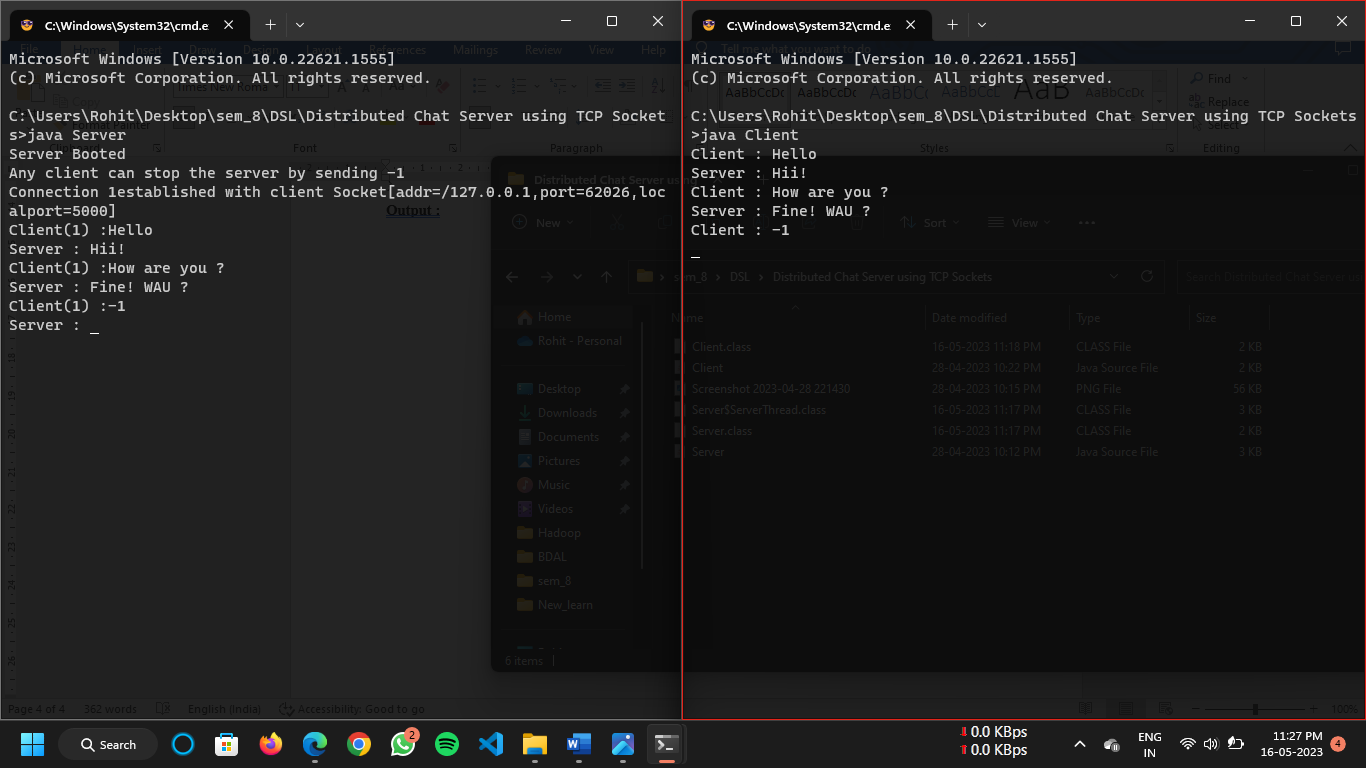
sout.close();

stdin.close();

}

}

**Output :**

****