import java.io.\*;

import java.net.\*;

import java.util.\*;

public class ChatServer {

private static final int PORT = 12345;

private static Set<PrintWriter> clientWriters = new HashSet<>();

public static void main(String[] args) {

System.out.println("Chat server started...");

try (ServerSocket serverSocket = new ServerSocket(PORT)) {

while (true) {

new ClientHandler(serverSocket.accept()).start();

}

} catch (IOException e) {

e.printStackTrace();

}

}

private static class ClientHandler extends Thread {

private Socket socket;

private PrintWriter out;

private BufferedReader in;

public ClientHandler(Socket socket) {

this.socket = socket;

}

public void run() {

try {

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

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

synchronized (clientWriters) {

clientWriters.add(out);

}

String message;

while ((message = in.readLine()) != null) {

System.out.println("Received: " + message);

synchronized (clientWriters) {

for (PrintWriter writer : clientWriters) {

writer.println(message);

}

}

}

} catch (IOException e) {

e.printStackTrace();

} finally {

try {

socket.close();

} catch (IOException e) {

e.printStackTrace();

}

synchronized (clientWriters) {

clientWriters.remove(out);

}

}

}

}

}

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class ChatClient {

private static final String SERVER\_ADDRESS = "localhost";

private static final int SERVER\_PORT = 12345;

public static void main(String[] args) {

try (Socket socket = new Socket(SERVER\_ADDRESS, SERVER\_PORT);

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

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

Scanner scanner = new Scanner(System.in)) {

System.out.println("Connected to chat server");

Thread messageReader = new Thread(() -> {

try {

String message;

while ((message = in.readLine()) != null) {

System.out.println(message);

}

} catch (IOException e) {

e.printStackTrace();

}

});

messageReader.start();

while (true) {

String message = scanner.nextLine();

if (message.equalsIgnoreCase("quit")) {

break;

}

out.println(message);

}

} catch (IOException e) {

e.printStackTrace();

}

}

}