```java
import java.io.*;

import java.net.*;

import java.util.*;

public class Chat server{

Private static Set<ClientHandler>clientHandlers=Collection.synchronisedSet(new Hashset<>());

public static void main(String []args){

int port= 12345;

System.out.println("Server started on port"+port);

try(ServerSocket serversocket =new ServerSocket(port)){

while (true){

Socket client Socket = serverSocket.accept();

System.out.println("New client connected:" +clientSocket);

ClientHandler handler = new ClientHandler(clientSocket);

clientHandlers.add(handler);

new Thread(handler).start();

}

}

catch (IOException e){

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

}

}

public static void broadcast(String message, ClientHandler sender){

synchronised(clientHandlers){

for(ClientHandler client: ClientHandler){

if(client!=sender){

client.sendMessage(message);

}
```

```java
        }

    }

}

public static void removeClient(ClientHandler client){

clientHandlers.remove(client);

}

}

class ClientHandler implements Runnable{

private Socket socket;

private print writer out;

private BufferedReader in;

public ClientHandler (Socket socket){

this.socket = socket;

}

public void run() {

try{

in = new BufferedReader(new Input stream reader(socket.getInputStream()));

out = new Print writer(socket.getOutStream(),true);

System.out.println("Welcome! You can start typing message:");

String message;

while ((message = in read line())!=null){

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

Chat server.broadcast(message,this);

}

}

catch(IOEXCEPTION e) {

System.out.println("Client disconnected:" +e.getMessage());
```

```java
}
finally{
Chat server.removeClient(this);
try {
socket.close();
}
catch(IOEXCEPTION e) {}
}
}
public void send message(String message){
System.out.println(message);
}
}




import java.io.*;
import java.net.*;
public class Chat client {
public static void main(String [] args) {
String hostname = "localhost";
int port = 12345;
try(Socket socket = new Socket(hostname,port)) {
System.out.println("Connected to the chat server!");
BufferedReader reader = new BufferedReader (new Input stream reader(socket.getInuputStream()));
```

```java
Print writer writer = new print writer(socket.getOutputStream(),true);

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

new Thread (() -> {

String response;

try {

while ((response = reader.readLine())!=null) {

System.out.println(response);

}

}

catch(IOEXCEPTION e) {

System.out.println("Disconnected from server.");

}

}

Start ();

String input;

while ((input = user input.readLine()) != null) {

writer.println(input);

}

}

catch(IOException e) {

System.out.println("Error:" +e.getMessage());

}

}

}
```