package messenger\_server**;**import messenger\_network.TCPCOnnectionListener**;**import messenger\_network.TCPConnection**;**import com.google.gson.Gson**;**import java.io.IOException**;**import java.net.ServerSocket**;**import java.util.ArrayList**;**import java.util.HashMap**;**import java.util.Map**;**class Message {  
 private String TYPE**;** private String USERNAME**;** private String USERNAME\_TO**;** private String TEXT**;** private ArrayList<String> NAMELIST**;** public Message(String TYPE**,** String USERNAME**,** String USERNAME\_TO**,**String TEXT**,** ArrayList<String> NAMELIST) {  
 this.TYPE = TYPE**;** this.USERNAME = USERNAME**;** this.USERNAME\_TO = USERNAME\_TO**;** this.TEXT = TEXT**;** this.NAMELIST = NAMELIST**;** }  
  
  
 public String getTYPE() {  
 return TYPE**;** }  
 public String getUSERNAME() {  
 return USERNAME**;** }  
 public String getUSERNAME\_TO() {  
 return USERNAME\_TO**;** }  
 public String getTEXT() {  
 return TEXT**;** }  
  
 public ArrayList<String> getNAMELIST() {  
 return NAMELIST**;** }  
}  
  
public class ChatServer implements TCPCOnnectionListener {  
  
 Integer Connection\_Send\_Index = null**;** String UserName\_To = null**;** String UserName = null**;** String Text = null**;** private String Type = null**;** int cnt = **0;** public static void main(String[] args) {  
  
 new ChatServer()**;** }  
  
 private final ArrayList<TCPConnection> connections = new ArrayList<>()**;** private final Map<String**,** Integer> connection\_list = new HashMap<String**,** Integer>()**;** private ArrayList<String> name\_list = new ArrayList<String>()**;** private ChatServer(){  
 System.*out*.println("Server running")**;** try (ServerSocket serverSocket = new ServerSocket(**8882**)) {  
 while(true) {  
 try {  
 new TCPConnection(serverSocket.accept()**,** this)**;** } catch (IOException e) {  
 System.*out*.print("TCPConnection exeption" + e)**;** }  
  
 }  
 } catch (IOException e) {  
 throw new RuntimeException(e)**;** }  
 }  
  
 @Override  
 public void onConnectionReady(TCPConnection tcpConnection**,** String value) {  
  
 connections.add(tcpConnection)**;** Message message1 = new Gson().fromJson(value**,** Message.class)**;** UserName = message1.getUSERNAME()**;** name\_list.add(UserName)**;** System.*out*.println("nameList"+name\_list)**;** connection\_list.put(UserName**,**cnt )**;** System.*out*.println("connectionlist"+connection\_list)**;** System.*out*.println(name\_list)**;** System.*out*.println(connections)**;** System.*out*.println(name\_list.indexOf(UserName))**;** System.*out*.println(connections.indexOf(tcpConnection))**;** sendToAllConnections("user\_conected"**,**"Admin"**,** null**,** null**,** name\_list)**;** System.*out*.println(name\_list)**;** cnt = connections.size()**;** System.*out*.println("это просто индекс "+cnt)**;** // sendToAllConnections("Client connected" + tcpConnection);  
 }  
  
 @Override  
 public void onReceiveString(TCPConnection tcpConnection**,** String value) {  
  
  
  
  
 Message message1 = new Gson().fromJson(value**,** Message.class)**;** Type = message1.getTYPE()**;** UserName = message1.getUSERNAME()**;** UserName\_To = message1.getUSERNAME\_TO()**;** Text = message1.getTEXT()**;** System.*out*.println(Type)**;** sendToAllConnections( Type**,** UserName**,** UserName\_To**,** Text**,** name\_list)**;** }  
  
 @Override  
 public void onDisconnect(TCPConnection tcpConnection) {  
 int indexEl = connections.indexOf(tcpConnection)**;** int i = connections.size()**;** String name = name\_list.get(indexEl)**;** name\_list.remove(indexEl)**;** connection\_list.remove(name)**;** connections.remove(tcpConnection)**;** // System.out.println("======="+name);  
 // connection\_list.remove(name);  
  
  
 System.*out*.println("this anme list need to be "+name\_list)**;** //sendToAllConnections("Client disconnected" + tcpConnection);  
 sendToAllConnections("user\_disconected"**,**null**,** null**,** null**,** name\_list)**;** }  
  
 @Override  
 public void onException(TCPConnection tcpConnection**,** Exception e) {  
  
  
 System.*out*.println("TCPCOnection exeption " + e)**;** }  
 private void sendToAllConnections(String Type**,**String UserName**,**String UserName\_To**,**String Text**,** ArrayList<String> name\_list){  
  
 final int cnt = connections.size()**;** if(Type == "refresh"){  
 Connection\_Send\_Index =name\_list.indexOf(UserName)**;** Message message1 = new Message("name\_list"**,** null**,** null**,** null**,** name\_list)**;** String json1 = new Gson().toJson(message1)**;** connections.get(Connection\_Send\_Index).sedString(json1)**;** } else {  
  
 Message message = new Message(Type**,** UserName**,** null**,** Text**,** name\_list)**;** String json = new Gson().toJson(message)**;** if (UserName\_To == null) {  
  
 for (int i = **0;** i < cnt**;** i++) {  
 connections.get(i).sedString(json)**;** System.*out*.println(json)**;** }  
 }  
 if (UserName\_To != null) {  
  
  
 Connection\_Send\_Index = name\_list.indexOf(UserName\_To)**;** Message message1 = new Message(Type**,** UserName**,** UserName\_To**,** Text**,** name\_list)**;** String json1 = new Gson().toJson(message1)**;** connections.get(Connection\_Send\_Index).sedString(json1)**;** }  
  
  
 Message message2 = new Message("name\_list"**,** null**,** null**,** null**,** name\_list)**;** String json2 = new Gson().toJson(message2)**;** for (int i = **0;** i < cnt**;** i++) {  
 connections.get(i).sedString(json2)**;** System.*out*.println("json on connection " + json2)**;** }  
 }  
  
 }  
}