**package** network;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**import** java.io.OutputStreamWriter;

**import** java.net.ServerSocket;

**import** java.net.Socket;

**public** **class** ProtocolServer {

**private** ServerSocket serverSocket;

**private** Socket socket;

**private** BufferedReader br;

**private** BufferedWriter bw;

**public** ProtocolServer() { // 항상 먼저 실행

**try** {

// 하염없이 9500번 방을 보면서 Client가 들어오길 기다린다

serverSocket = **new** ServerSocket(9500);

System.***out***.print("서버준비완료");

// 클라이언트가 들어오면 낚아채서 클라이언트와 연결할 소켓을 만들어준다

// ServerSocket의 역할. 그러곤 대화에 참여하지 않고 본인은 빠짐

socket = serverSocket.accept();

br = **new** BufferedReader(**new** InputStreamReader(socket.getInputStream()));

bw = **new** BufferedWriter(**new** OutputStreamWriter(socket.getOutputStream()));

} **catch** (IOException e) {

System.***out***.println("클라이언트와 연결할 수 없습니다");

e.printStackTrace();

System.*exit*(0);

}

String line = **null**;

**while** (**true**) {

**try** {

// 받는 쪽

line = br.readLine(); // 100:angel or 200:angel or 300:angel:안녕

// 보내는 쪽

String[] ar = line.split(":"); // 콜론을 기준으로 나눠주세요

**if** (ar[0].equals("100")) { // 0번째 방이 100으로 들어왔습니까

bw.write(ar[1] + "님 입장\n"); // 클라이언트에게 보내주자 여기서도 엔터값

bw.flush();

} **else** **if** (ar[0].equals("200")) {

bw.write(ar[1] + "님 퇴장\n");

bw.flush();

br.close();

bw.close();

socket.close();

System.*exit*(0);

} **else** **if** (ar[0].equals("300")) {

bw.write("[" + ar[1] + "]" + ar[2] + "\n");

bw.flush();

}

} **catch** (IOException e) {

e.printStackTrace();

}

} // while

} // ProtocolServer

**public** **static** **void** main(String[] args) {

**new** ProtocolServer();

}

}