Server.java

*// Write a Program to Demonstrate Group Communication using Java.*

*// Vishal Yadav - 211P042 - BE Computer*

*// READ OR WRITE DATA*

import **java**.**io**.**BufferedReader**;

import **java**.**io**.**InputStreamReader**;

import **java**.**io**.**PrintWriter**;

*// TO ESTABLISHED CONNECTION*

import **java**.**net**.**ServerSocket**;

import **java**.**net**.**Socket**;

*// FOR STORE NAMES OF CLIENT IN A VECTOR USED FOR MULTICAST*

import **java**.**util**.**Vector**;

*// ALL CLASS MUST BE PUBLIC TO COMMUNICATE ON NETWORK*

public class **Server**

{

*// WHAT TYPE OF DATA STORING IN VECTOR <PRINT WRITER>*

private static **Vector**<**PrintWriter**> writers=new **Vector**<**PrintWriter**>();

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

{

*// SERVER SOCKET CLASS IE SERVER PUBLISH THAT LISTENING ON THIS PORT*

**ServerSocket** listener=new **ServerSocket**(9001); *// PORT NO. 9001*

*// DISPLAYING MESSAGE*

**System**.out.**println**("The server is running at port 9001.");

*// CONTINUOUSLY LISTENING*

while(true) *// continuously accepts the cliect requests and start the thread for each client*

{

*//LISTENER.ACCEPT RETURN SOCKET OBJECT OF CLIENT*

*// START() EXECUTE RUN METHOD*

*// HANDLER CREATES OBJECTS*

new **Handler**(listener.**accept**()).**start**(); *//HANDLER CONSTRUCUTOR START THREAD*

}

}

*// INHERITING THREAD CLASS*

private static class **Handler** extends **Thread**

{

private **Socket** socket;

public **Handler**(**Socket** socket)

{

this.socket=socket;

}

public void **run**()

{

try

{ *// SOCKET.GETINPUTSTREAM-- READ DATA FROM PARTICULAR CLIENT*

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

*// WRITE TO EACH AND EVERY CLIENT*

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

out.**println**("SUBMITNAME");

*// SERVER READ AND STORE DATA*

**String** name=in.**readLine**();

**System**.out.**println**(name+ " joined the chat");

*// WRITER IS A VECTOR*

*// ADD METHOD TO STORE OUT IN VECTOR*

writers.**add**(out);

*// CONTINUOUSLY SEND AND RECIEVE*

while(true)

{

**String** input=in.**readLine**();

*//System.out.println("read at server:"+input);*

**System**.out.**println**("MESSAGE " +name+ ":" +input);

*//FROM WRITER VECTOR EACH ELEMENT PRINTING I.E FOR MULTICASTING*

for (**PrintWriter** x:writers)

x.**println**("MESSAGE " +name+ ":" +input); }

}

catch(**Exception** e)

{

**System**.err.**println**(e);

}

}

}

}

Master.java

import **java**.**io**.**BufferedReader**;

import **java**.**io**.**InputStreamReader**;

import **java**.**io**.**PrintWriter**;

import **java**.**net**.**Socket**;

import **java**.**util**.**Scanner**;

public class **master**

{

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

{

**Scanner** sc=new **Scanner**(**System**.in);

**Socket** socket=new **Socket**("localhost",9001);

*//READ DATA FROM SOCKET*

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

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

**System**.out.**print**("Enter your name:");

**String** name=sc.**nextLine**();

while(true)

{

*// READ MESSAGE OF A SERVER WHICH IS SUBMITNAME IN SERVER*

**String** line=in.**readLine**();

**String** msg="";

*//System.out.print("read at master: "+line);*

if(line.**startsWith**("SUBMITNAME"))

*// SEND NAME TO SERVER*

out.**println**(name);

*// MESSAGE COMING FROM SERVER FOR LOOP*

else if (line.**startsWith**("MESSAGE"))

**System**.out.**println**(line.**substring**(8));

*// MASTER GIVEN ON INPUT I.E ENTER YOUR NAME*

if(name.**startsWith**(""))

{

**System**.out.**print**("Enter a message:");

msg=sc.**nextLine**();

out.**println**(msg);

}

if (msg.**equals**("stop"))

{

**System**.**exit**(0);

}

}

}

}

Slave1.java

import **java**.**io**.**BufferedReader**;

import **java**.**io**.**InputStreamReader**;

import **java**.**io**.**PrintWriter**;

import **java**.**net**.**Socket**;

import **java**.**util**.**Scanner**;

public class **slave1**

{

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

{

**Scanner** sc= new **Scanner**(**System**.in);

**Socket** socket=new **Socket**("localhost",9001);

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

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

**System**.out.**print**("Enter your name: ");

**String** name =sc.**nextLine**();

while (true)

{

**String** msg="";

**String** line=in.**readLine**();

*//System.out.print("\nread at slave1: "+line);*

if(line.**startsWith**("SUBMITNAME"))

out.**println**(name);

else if (line.**startsWith**("MESSAGE"))

**System**.out.**println**("\n"+line.**substring**(8));

if(name.**startsWith**(""))

{

**System**.out.**print**("Enter a message: ");

msg=sc.**nextLine**();

out.**println**(msg);

}

if (msg.**equals**("stop"))

{

**System**.**exit**(0);

}

}

}

}

Slave2.java

import **java**.**io**.**BufferedReader**;

import **java**.**io**.**InputStreamReader**;

import **java**.**io**.**PrintWriter**;

import **java**.**net**.**Socket**;

import **java**.**util**.**Scanner**;

public class **slave2**

{

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

{

**Scanner** sc=new **Scanner**(**System**.in);

**Socket** socket=new **Socket**("localhost",9001);

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

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

**System**.out.**print**("Enter your name:");

**String** name=sc.**nextLine**();

while(true)

{

**String** msg="";

**String** line=in.**readLine**();

*//System.out.print("\n read at slave2: "+line);*

if(line.**startsWith**("SUBMITNAME"))

out.**println**(name);

else if (line.**startsWith**("MESSAGE"))

**System**.out.**println**("\n"+line.**substring**(8));

if(name.**startsWith**(""))

{

**System**.out.**print**("Enter a message: ");

msg=sc.**nextLine**();

out.**println**(msg);

}

if (msg.**equals**("stop"))

{

**System**.**exit**(0);

}

}

}

}

