# CSE PCC-CS692 L - COMPUTER NETWORKS LAB

**MULTICAST**

**Sender Code**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class Sender

{

public static void main(String[] args)

{

try

{

MulticastSocket ms = new MulticastSocket();

ms.joinGroup(InetAddress.getByName("225.0.1.1"));

while(true){

System.out.println();

System.out.println("Enter a message:");

Scanner sc = new Scanner(System.in);

String str = sc.nextLine();

if(str.equals("end")){

byte[] b2 = str.getBytes();

DatagramPacket dp2 = new DatagramPacket(b2,b2.length,InetAddress.getLocalHost(),4999);

ms.send(dp2);

break;

}

else{

byte[] b = str.getBytes();

DatagramPacket dp = new DatagramPacket(b,b.length,InetAddress.getByName("225.0.1.1"),4999);

ms.send(dp);

byte[] b1 = new byte[1024];

DatagramPacket dp1 = new DatagramPacket(b1,b1.length);

ms.receive(dp1);

String s = new String(dp1.getData(),0,dp1.getLength());

System.out.println();

System.out.println("RECEIVED MESSAGE: "+s);

if(s.equals("end")){

break;

}

}

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

**RECIEVERA**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class ReceiverA

{

public static void main(String[] args)

{

try

{

MulticastSocket ms = new MulticastSocket(4999);

ms.joinGroup(InetAddress.getByName("225.0.1.1"));

while(true){

byte[] b1 = new byte[1024];

DatagramPacket dp = new DatagramPacket(b1,b1.length);

ms.receive(dp);

InetAddress clientAddress = dp.getAddress();

int clientPort = dp.getPort();

String s = new String(dp.getData());

System.out.println();

System.out.println("RECEIVED MESSAGE: "+s);

if(s.equals("end")){

break;

}

else{

System.out.println();

System.out.println("Enter a message:");

Scanner sc = new Scanner(System.in);

String str = sc.nextLine();

if(str.equals("end")){

byte[] b2 = str.getBytes();

DatagramPacket dp2 = new DatagramPacket(b2,b2.length,clientAddress,clientPort);

ms.send(dp2);

break;

}

else{

byte[] b = str.getBytes();

DatagramPacket dp1 = new DatagramPacket(b,b.length,clientAddress,clientPort);

ms.send(dp1);

}

}

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

**RECIEVERB**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class ReceiverB

{

public static void main(String[] args)

{

try

{

MulticastSocket ms = new MulticastSocket(4999);

ms.joinGroup(InetAddress.getByName("225.0.1.1"));

while(true){

byte[] b1 = new byte[1024];

DatagramPacket dp = new DatagramPacket(b1,b1.length);

ms.receive(dp);

InetAddress clientAddress = dp.getAddress();

int clientPort = dp.getPort();

String s = new String(dp.getData());

System.out.println();

System.out.println("RECEIVED MESSAGE: "+s);

if(s.equals("end")){

break;

}

else{

System.out.println();

System.out.println("Enter a message:");

Scanner sc = new Scanner(System.in);

String str = sc.nextLine();

if(str.equals("end")){

byte[] b2 = str.getBytes();

DatagramPacket dp2 = new DatagramPacket(b2,b2.length,clientAddress,clientPort);

ms.send(dp2);

break;

}

else{

byte[] b = str.getBytes();

DatagramPacket dp1 = new DatagramPacket(b,b.length,clientAddress,clientPort);

ms.send(dp1);

}

}

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

**RECIEVERC**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class ReceiverC

{

public static void main(String[] args)

{

try

{

MulticastSocket ms = new MulticastSocket(4999);

ms.joinGroup(InetAddress.getByName("225.0.1.1"));

while(true){

byte[] b1 = new byte[1024];

DatagramPacket dp = new DatagramPacket(b1,b1.length);

ms.receive(dp);

InetAddress clientAddress = dp.getAddress();

int clientPort = dp.getPort();

String s = new String(dp.getData());

System.out.println();

System.out.println("RECEIVED MESSAGE: "+s);

if(s.equals("end")){

break;

}

else{

System.out.println();

System.out.println("Enter a message:");

Scanner sc = new Scanner(System.in);

String str = sc.nextLine();

if(str.equals("end")){

byte[] b2 = str.getBytes();

DatagramPacket dp2 = new DatagramPacket(b2,b2.length,clientAddress,clientPort);

ms.send(dp2);

break;

}

else{

byte[] b = str.getBytes();

DatagramPacket dp1 = new DatagramPacket(b,b.length,clientAddress,clientPort);

ms.send(dp1);

}

}

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

**OUTPUT**

