

```

import java.net.*;
import java.util.*;
import java.io.*;

public class StorageNode_Multicast extends Thread {
    Hashtable<Integer, String> nodeTable = new Hashtable<Integer, String>();
    String myIp;
    static int id = 0;
    int network;
    public StorageNode_Multicast (Hashtable<Integer, String> nodeTable, String myIp, int network) {
        this.nodeTable = nodeTable;
        this.myIp = myIp; //il mio indirizzo ip + porte socket UDP & TCP
        this.network = network; //indica se stiamo usando la versione locale o di rete
    }

    public Hashtable<Integer, String> getTable () {
        return nodeTable;
    }

    public void run() {
        try { //il mio identificatore
            //spedisco a tutti i nodi che fanno parte del multicast:
            //il mio indirizzo ip
            //porta UDP+porta TCP
            //tutto questo è contenuto in myIp
            String address;

            if (network == 0) //versione locale
                address = "224.0.0.1";
            else //versione di rete
                address = "226.226.226.226";

            InetAddress ia = InetAddress.getByName(address);

            MulticastSocket ms = new MulticastSocket(4000);

            ByteArrayOutputStream bout = new ByteArrayOutputStream();

            DataOutputStream dout = new DataOutputStream (bout);

            dout.writeUTF(myIp);

            byte []data = bout.toByteArray( );

            DatagramPacket dp = new DatagramPacket(data, data.length, ia, 4000);

            ms.send (dp);

            bout.reset();

            //entro a far parte del gruppo multicast
            ms.joinGroup (ia);

            while (true) {
                data = new byte[200];
                dp = new DatagramPacket (data, data.length);
                //qualcuno ha aggiornato la tabella
                ms.receive(dp);
                ByteArrayInputStream bin = new ByteArrayInputStream

(dp.getData(), 0, dp.getLength());

                DataInputStream ddis = new DataInputStream(bin);
                //ora myIp fa riferimento all'oggetto ricevuto da un nuovo

```

membro

```
        myIp = ddis.readUTF();
        int end = myIp.indexOf(" IP");
        String myIp_id = myIp.substring(3, end);
        int file_id = Integer.parseInt(myIp_id);
        nodeTable.put(file_id, myIp);
    }

} catch (IOException e) {
    e.printStackTrace();
}

}
```