



链滴

Java 基础 3 多线程和 socket 结合 ---- 聊天室

作者: [syyQ](#)

原文链接: <https://ld246.com/article/1507648451454>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

思路

- 1、一个线程负责发送数据。一个负责接收数据。两个任务同时执行
- 2、收到over时结束聊天

```
//send
class Send implements Runnable{
    private DatagramSocket ds;

    public Send(DatagramSocket ds){
        super();
        this.ds = ds;
    }
    public void run() {
        // TODO Auto-generated method stub
        BufferedReader bufr = new BufferedReader(new InputStreamReader(System.in));

        String line = null;
        try{
            while((line = bufr.readLine())!=null){
                //数据封装到数据包中
                byte[] buf = line.getBytes();
                DatagramPacket dp =new DatagramPacket(buf,buf.length, InetAddress.getByName("
27.0.0.1"),10002);
                ds.send(dp);
                if("over".equals(line)){
                    break;
                }
            }
            ds.close();
        }catch(IOException e){
            e.printStackTrace();
        }
    }
}

// receive
```

```
class Receive implements Runnable{
    private DatagramSocket ds;

    public Receive(DatagramSocket ds){
        super();
        this.ds = ds;
    }
    public void run() {
        while(true){
```

```

// TODO Auto-generated method stub
byte[] buf = new byte[1024];

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

try {
    ds.receive(dp);
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}

String ip = dp.getAddress().getHostAddress();

String data = new String(dp.getData(),0,dp.getLength());
if("over".equals(data)){
    System.out.println(ip+"...离开聊天室!");

}
System.out.println(ip+": "+data);
}

}

```

//主函数

```

public class ChatTest {
public static void main(String [] args) throws SocketException{

    DatagramSocket send =new DatagramSocket(8888);
    DatagramSocket receive = new DatagramSocket(10002); //注意接收端端口号 与send中的
同

    new Thread(new Send(send)).start();
    new Thread(new Receive(receive)).start();
}
}

```