

Glimmertask10

web socket

简单挑战——现代网络基础

1. 如何建立客户端与服务器的初始连接。在java中，建立客户端和服务端之间的初始连接通常使用Sockets一般做法是这样的：

服务器端：

```
ServerSocket serverSocket = new ServerSocket(1234); // 创建一个服务器端的端口，  
端口号为1234
```

客户端：

```
Socket socket = new Socket("localhost", 1234);  
// 这样客户端就会连接到本地的1234端口，与服务器相连
```

2. 如何从服务器接受消息：

在与客户端连接后，客户端接收服务器端的消息需要使用Reader或者Input方法，在上面创建服务器端口的基础上给出下面的例子：

```
BufferedReader in = new BufferedReader(new  
InputStreamReader(socket.getInputStream()));  
String input;  
input = in.readLine(); // 这样既可通过BufferedReader和InputStreamReader来实现对  
于客户端的消息的接收
```

注：除了BufferedReader以外，我们还可以使用DataInputStream（读取二进制或基本数据类型）、ObjectInputStream（前一个task所学过的Serializable接口的知识）或者直接使用InputStream直接读取（读取字节）

3. 如何向服务器发送消息：

通过客户端向服务器发送消息需要使用Writer或者Output方法，以下是实例：

```
PrintWriter out = new PrintWriter(socket.getOutputStream(), true); // 第二个true的  
含义是是否自动刷新缓冲区，如果是true，我们print进去的东西会直接写出，如果为false，我们需  
要另外调用flush()方法来刷新缓冲区  
out.println("hello");
```

注：除了PrintWriter，我们还可以使用PrintStream（用于字节流）、BufferedWriter（字符流）、DataOutputStream（读取二进制或基本数据类型）、ObjectOutputStream

注：上面的Socket类和下面的SocketChannel类均可以建立网络连接，但是还有一定区别，比如阻塞和非阻塞、SocketChannel可以与选择器（Selector）一起使用、SocketChannel使用缓冲区来处理数据，这提供了更多的控制和灵活性、SocketChannel提供了更多的方法来控制通道的状态

建立连接

TCP/IP包含的一系列协议

协议	作用
TCP（传输控制协议）	应用程序之间的通信
UDP（用户数据报协议）	应用程序之间的简单通信
IP(网络协议)	计算机之间的通信
ICMP(因特网消息控制协议)	针对错误和状态
DHCP(动态主机协议)	针对动态寻址

TCP/IP意味着 TCP 和 IP 在一起协同工作，TCP 负责应用软件（比如您的浏览器）和网络软件之间的通信，IP 负责计算机之间的通信。

```
// 通过IP地址和TCP端口号创建一个对象并打开通道
InetSocketAddress serverAddress = new InetSocketAddress("196.164.1.103", 5000); //
这里是举例
SocketChannel socketChannel = SocketChannel.open(serverAddress);
```

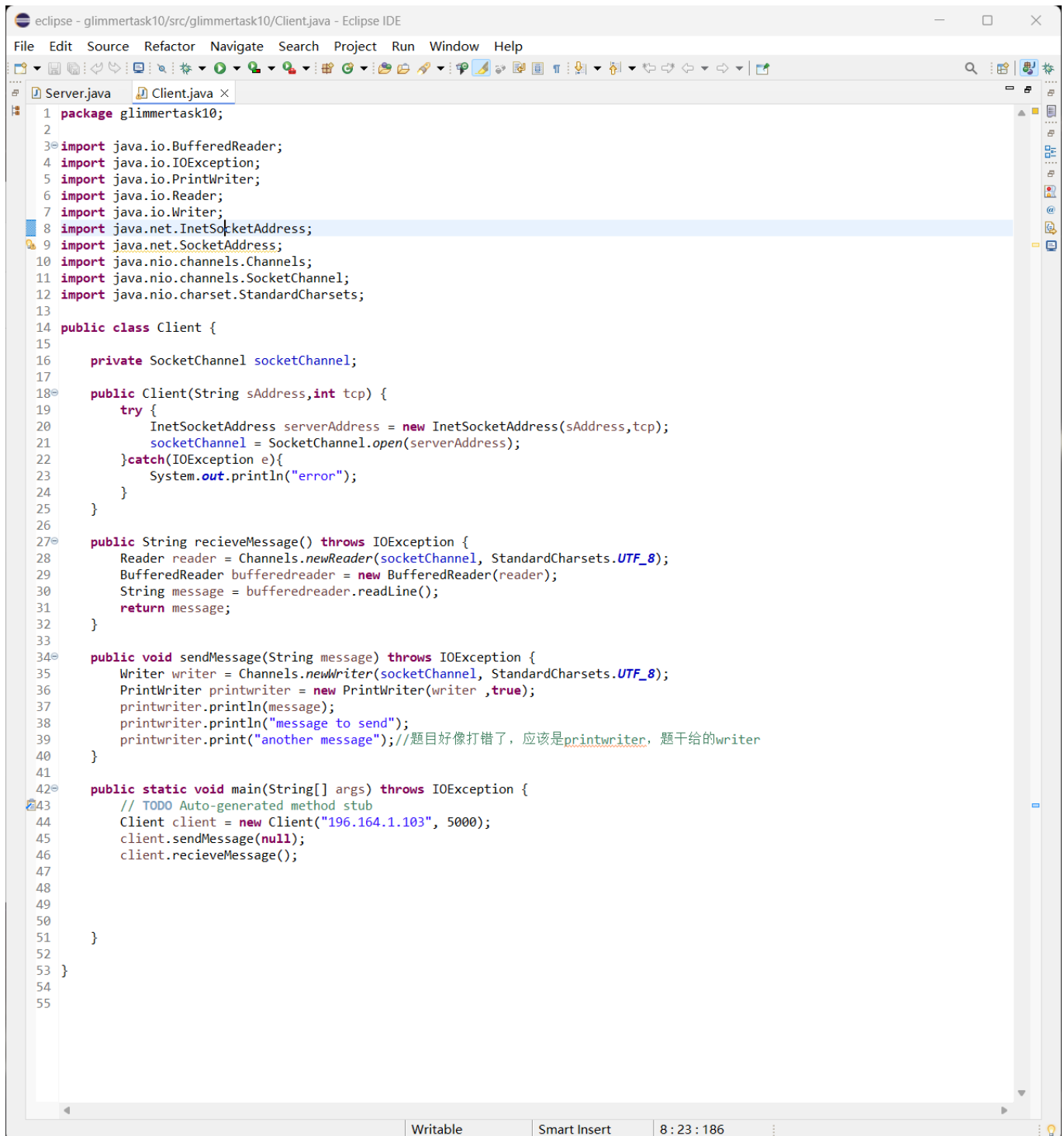
接收信息

```
//建立一个与服务器的链接
SocketAddress serverAddr = new InetSocketAddress("127.0.0.1", 5000);
SocketChannel socketChannel = SocketChannel.open(serverAddr);
//从这个链接创建并获取一个Reader
Reader reader = Channels.newReader(socketChannel, StandardCharsets.UTF_8); //reader
是底层字节流和高层字符流的桥梁
//创建或获取一个BufferedReader并读取
BufferedReader bufferedReader = new BufferedReader(reader); //将BufferedReader串联到
reader
String message = bufferedReader.readLine();
```

发送消息

```
//建立一个与服务器的链接
SocketAddress serverAddr = new InetSocketAddress("127.0.0.1", 5000);
SocketChannel socketChannel = SocketChannel.open(serverAddr);
//从这个链接创建并获取一个Writer
```

```
Writer writer = Channels.newWriter(socketChannel, StandardCharsets.UTF_8);
//创建一个PrintWriter并打印一些内容
PrintWriter printWriter = new PrintWriter(writer);
writer.println("message to send");
writer.print("another message");//这边题干好像出错了, 应该是printwriter
```



```
eclipse - glimmertask10/src/glimmertask10/Client.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Server.java Client.java x
1 package glimmertask10;
2
3 import java.io.BufferedReader;
4 import java.io.IOException;
5 import java.io.PrintWriter;
6 import java.io.Reader;
7 import java.io.Writer;
8 import java.net.InetSocketAddress;
9 import java.net.SocketAddress;
10 import java.nio.channels.Channels;
11 import java.nio.channels.SocketChannel;
12 import java.nio.charset.StandardCharsets;
13
14 public class Client {
15
16     private SocketChannel socketChannel;
17
18     public Client(String sAddress,int tcp) {
19         try {
20             InetSocketAddress serverAddress = new InetSocketAddress(sAddress,tcp);
21             socketChannel = SocketChannel.open(serverAddress);
22         } catch (IOException e) {
23             System.out.println("error");
24         }
25     }
26
27     public String recieveMessage() throws IOException {
28         Reader reader = Channels.newReader(socketChannel, StandardCharsets.UTF_8);
29         BufferedReader bufferedreader = new BufferedReader(reader);
30         String message = bufferedreader.readLine();
31         return message;
32     }
33
34     public void sendMessage(String message) throws IOException {
35         Writer writer = Channels.newWriter(socketChannel, StandardCharsets.UTF_8);
36         PrintWriter printwriter = new PrintWriter(writer,true);
37         printwriter.println(message);
38         printwriter.println("message to send");
39         printwriter.print("another message");//题目好像打错了, 应该是printwriter, 题干给的writer
40     }
41
42     public static void main(String[] args) throws IOException {
43         // TODO Auto-generated method stub
44         Client client = new Client("196.164.1.103", 5000);
45         client.sendMessage(null);
46         client.recieveMessage();
47
48
49
50
51     }
52 }
53
54
55
Writable Smart Insert 8:23:186
```