

Java10

网络编程

题目中使用Channel的方式来进行网络编程 按照要求用Socket改写如下 这里分别是客户端和服务端两个类

- 服务器

```
package service;

import java.io.*;
import java.net.ServerSocket;
import java.net.Socket;

public class Server {
    public static void main(String[] args) {

        try {
            ServerSocket serverSocket = new ServerSocket(11001);
            Socket socket = serverSocket.accept();
            InputStream inputStream = socket.getInputStream();
            InputStreamReader inputStreamReader = new InputStreamReader(inputStream);
            BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
            String message = bufferedReader.readLine();
            System.out.println(message);
            bufferedReader.close();
            inputStreamReader.close();
            inputStream.close();
            socket.close();
        } catch (IOException e) {
            throw new RuntimeException(e);
        }
    }
}
```

- 客户端

```

package service;

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

public class Client {
    public static void main(String[] args) {

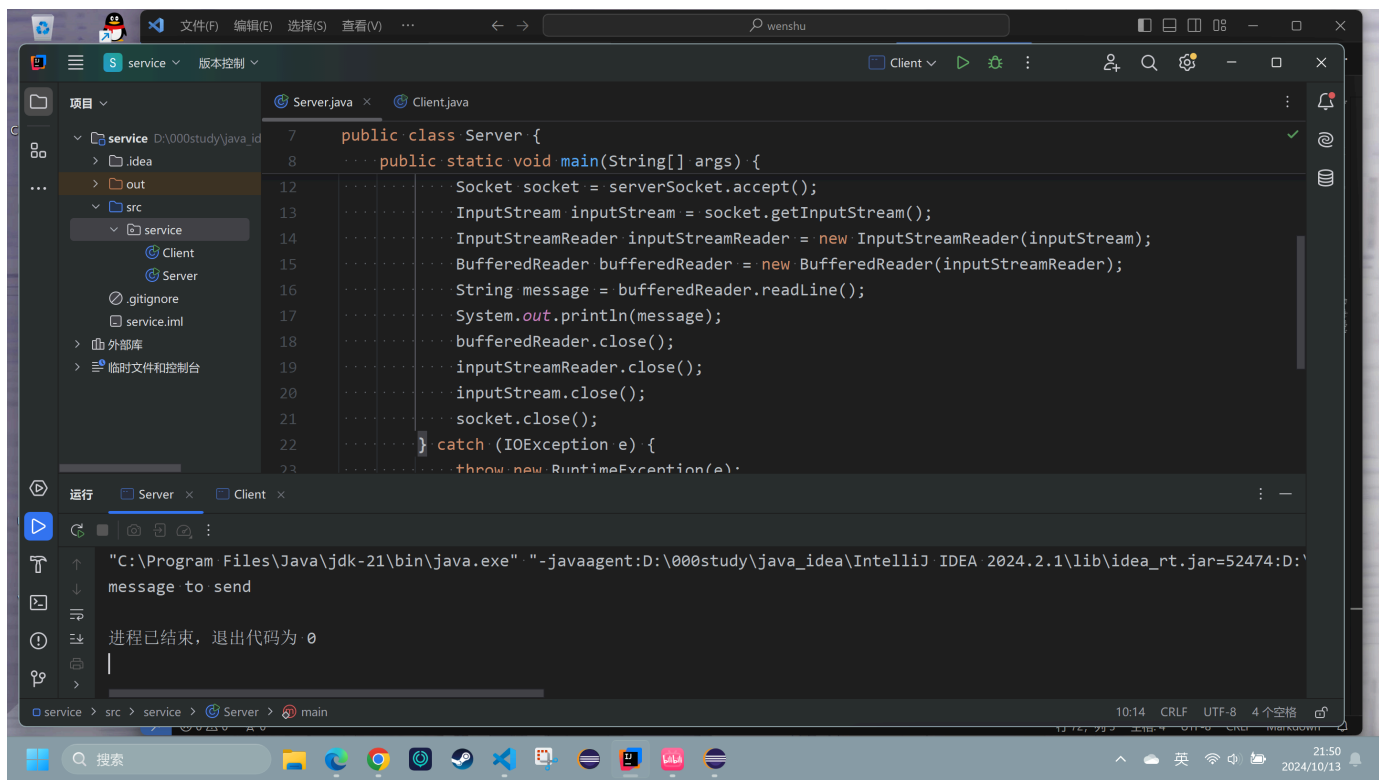
        try {
            Socket socket = new Socket("127.0.0.1", 11001);
            OutputStream outputStream = socket.getOutputStream();
            OutputStreamWriter outputStreamWriter = new OutputStreamWriter(outputStream);
            PrintWriter printWriter = new PrintWriter(outputStreamWriter);
            printWriter.println("message to send");
            printWriter.flush();
            printWriter.print("another message");
            printWriter.flush();
            printWriter.close();
            outputStreamWriter.close();
            outputStream.close();
            socket.close();
        } catch (IOException e) {
            throw new RuntimeException(e);
        }

    }
}

```

这里有点需要说明一下：

1. `readLine`是读取一行 当读到这一行后面的换行符时就会停止 题目中客户端发送了两个信息 由于第一条信息是使用的`println` 所以第二条信息服务器是收不到的
2. 貌似是因为`BufferedReader` 和 `PrintWriter` 都带有缓冲区 刚开始没加`flush` 服务器返回的结果一直为`null` 所以如果想要客户端每条信息发送后服务器立刻就能接收到 就在发送信息后面加上`flush` 把缓存区清空 强行把信息推过去
3. 虽然题目没有要求运行 但运行一下方便我找bug 我这里是建立的客户端与我本机服务器的连接 所以是可以运行的 不过要注意只能先跑服务器再跑客户端 `server`的`main`里有`accept` 当`client`的`socket`被创建时 就会和指定ip和接口建立连接了 `accept`要先等着这个连接 运行结果：



没有进阶挑战 感谢学长