Client.java

import java.io.\*;  
import java.net.\*;  
import java.util.Scanner;  
  
public class Client {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 Socket clientSocket = null;  
 DataInputStream intData = null;  
 DataOutputStream outData = null;  
 Thread thread;  
 Read read = null;  
 try {  
 clientSocket = new Socket();  
 read = new Read();  
 thread = new Thread(read);  
 System.*out*.print("输入服务器的IP: ");  
 String ip = scanner.nextLine();  
 System.*out*.print("输入端口号：");  
 int port = scanner.nextInt();  
 scanner.nextLine(); *// 消耗换行符* if (!clientSocket.isConnected()) { *// 检查是否已连接* InetAddress address = InetAddress.*getByName*(ip);  
 InetSocketAddress socketAddress = new InetSocketAddress(address, port);  
 clientSocket.connect(socketAddress);  
 InputStream in = clientSocket.getInputStream();  
 OutputStream out = clientSocket.getOutputStream();  
 intData = new DataInputStream(in);  
 outData = new DataOutputStream(out);  
 read.setDataInputStream(intData);  
 read.setDataOutputStream(outData);  
 thread.start();  
 }  
 } catch (Exception e) {  
 System.*out*.println("服务器已断开" + e);  
 }  
 }  
}  
  
class Read implements Runnable {  
 Scanner scanner = new Scanner(System.*in*);  
 DataInputStream in;  
 DataOutputStream out;  
  
 public void setDataInputStream(DataInputStream in) {  
 this.in = in;  
 }  
  
 public void setDataOutputStream(DataOutputStream out) {  
 this.out = out;  
 }  
  
 public void run() {  
 try {  
 while (true) {  
 System.*out*.println("输入账单：");  
 String content = scanner.nextLine();  
 out.writeUTF("账单" + content);  
 out.flush(); *// 刷新输出流* String str;  
 while ((str = in.readUTF()) != null) {  
 System.*out*.println(str);  
 }  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
}

Server.java

import java.io.\*;  
import java.net.\*;  
import java.util.\*;  
  
public class Server {  
 public static void main(String args[]) {  
 ServerSocket server = null;  
 Socket you = null;  
 try {  
 server = new ServerSocket(8080); *// 监听端口号为8080* System.*out*.println("正在监听");  
 while (true) {  
 you = server.accept(); *// 等待客户端连接* System.*out*.println("客户的地址：" + you.getInetAddress());  
 new ServerThread(you).start();  
 }  
 } catch (IOException e) {  
 System.*out*.println("" + e);  
 }  
 }  
  
 static class ServerThread extends Thread {  
 Socket socket;  
 DataInputStream in = null;  
 DataOutputStream out = null;  
  
 ServerThread(Socket t) {  
 socket = t;  
 try {  
 out = new DataOutputStream(socket.getOutputStream());  
 in = new DataInputStream(socket.getInputStream());  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
  
 public void run() {  
 try {  
 while (true) {  
 String item = in.readUTF();  
 Scanner scanner = new Scanner(item);  
 scanner.useDelimiter("[^0123456789.]+");  
 if (item.startsWith("账单")) {  
 double sum = 0;  
 while (scanner.hasNext()) {  
 try {  
 double price = scanner.nextDouble();  
 sum = sum + price;  
 System.*out*.println(price);  
 out.writeUTF("您的账单：");  
 out.writeUTF(item);  
 out.writeUTF("总额：" + sum + "元");  
 out.flush(); *// 刷新输出流* } catch (InputMismatchException exp) {  
 String t = scanner.next();  
 out.writeUTF("您的账单：");  
 out.writeUTF(item);  
 out.writeUTF("总额：" + sum + "元");  
 out.flush(); *// 刷新输出流* }  
 }  
 }  
 }  
 } catch (Exception exp) {  
 exp.printStackTrace();  
 }  
 }  
 }  
}

