**重庆交通大学信息科学与工程学院**

**综合性设计性实验报告**

**班 级： 计科1703**

**姓名 学号： 黄富灵 631707060308**

**实验项目名称： 实验九、网络编程**

**实验项目性质： 综 合 性**

**实验所属课程： 《Java程序设计》**

**实验室(中心)：**

**指 导 教 师 ： 王桂平**

**实验完成时间： 2019年 4月 18日**

教师评阅意见：

签名： 年 月 日

实验成绩：

**一、实验题目**

用Java语言和相关技术编写一个网络应用程序。

**二、实验要求**

1) 分组要求：分组完成，自由组队，2人一组。

2) 程序要求：不少于150行代码，且具有较好的可读性（注释、正确缩进等）。

3) 文档要求：详细阐述该程序的设计思想、采用的Java技术或方法。

**三、设计思想、采用的Java技术或方法**

本程序用于计算长方形的面积，由客户端发送数据到服务器端，服务器端计算后再发送结果到客户端。

采用单线程服务器接收端，客户端采用Swing Gui 创建一个简单的操作界面用于连接服务器和接收数据。

客户端创建单一线程，先尝试连接服务器端，失败则报告异常。

**四、主要程序代码**

/\*

\* @Coding: utf-8 \n

\* @Despcription: \n

\* @Author: Suwen SP \n

\* @LastEditors: Suwen SP\n

\* @Date: 2019-04-17 19:32:59

\* @LastEditTime: 2019-04-18 19:39:44

\*/

**import** java.io.DataInputStream;

**import** java.io.DataOutputStream;

**import** java.io.IOException;

**import** java.net.ServerSocket;

**import** java.net.Socket;

**public** **class** Server {

**public** **static** **void** main(String args[]){

ServerSocket server = **null**;

//ServerThread thread;

Socket you = **null**;

**while**(**true**){

**try**{

server = **new** ServerSocket(4331);

}

**catch**(IOException e1){

System.***out***.println("正在监听");

}

**try**{

System.***out***.println("等待客户呼叫");

you = server.accept();

System.***out***.println("客户的地址："+you.getInetAddress());

}**catch**(IOException e){

System.***out***.println("正在等待客户");

}

**if**(you!=**null**){

**new** ServerThread(you).start();

}

}

}

}

**class** ServerThread **extends** Thread{

Socket socket;

DataOutputStream out;

DataInputStream in;

String s;

ServerThread(Socket t){

socket = t;

**try**{

out = **new** DataOutputStream(socket.getOutputStream());

in = **new** DataInputStream(socket.getInputStream());

}**catch**(IOException e){}

}

**public** **void** run()

{

**while**(**true**){

**try**{

**double** s1 = in.readDouble();

**double** s2 = in.readDouble();

**double** area = s1\*s2;

out.writeUTF("长"+s1+"宽"+s2+"的长方体的面积"+area);

}

**catch** (IOException e){

System.***out***.println("客户离开");

**return**;

}

}

}

}

/\*

\* @Coding: utf-8 \n

\* @Despcription: \n

\* @Author: Suwen SP \n

\* @LastEditors: Suwen SP\n

\* @Date: 2019-04-17 19:33:28

\* @LastEditTime: 2019-04-18 19:37:41

\*/

**import** java.awt.BorderLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.io.DataInputStream;

**import** java.io.DataOutputStream;

**import** java.io.IOException;

**import** java.net.InetAddress;

**import** java.net.InetSocketAddress;

**import** java.net.Socket;

**import** javax.swing.\*;

**public** **class** Window {

**public** **static** **void** main(String args[]) {

**new** Windows();

}

}

**class** Windows **extends** JFrame **implements** Runnable, ActionListener {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

JButton connection, send;

JTextField inputText1;

JTextField inputText2;

JTextArea showResult;

Socket socket;

DataInputStream in; // 数据输入流

DataOutputStream out;

Thread thread;

**public** Windows() {

socket = **new** Socket();

connection = **new** JButton("连接服务器");

send = **new** JButton("发送");

send.setEnabled(**false**);

inputText1 = **new** JTextField(6);

inputText2 = **new** JTextField(6);

showResult = **new** JTextArea();

add(connection, BorderLayout.***NORTH***);

JPanel p = **new** JPanel();

p.add(**new** JLabel("请输入长方形的长和宽："));

p.add(inputText1);

p.add(inputText2);

p.add(send);

add(**new** JScrollPane(showResult), BorderLayout.***CENTER***);

add(p, BorderLayout.***SOUTH***);

connection.addActionListener(**this**);

send.addActionListener(**this**);

thread = **new** Thread();

setBounds(10, 30, 460, 400);

setVisible(**true**);

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

**public** **void** actionPerformed(ActionEvent e) {

**if** (e.getSource() == connection) {

**try** {

**if** (socket.isConnected()) {

} **else** {

InetAddress address = InetAddress.*getByName*("localhost");

InetSocketAddress socketAddress = **new** InetSocketAddress(address, 4331);

socket.connect(socketAddress);

in = **new** DataInputStream(socket.getInputStream());

out = **new** DataOutputStream(socket.getOutputStream());

send.setEnabled(**true**);

**if** (!(thread.isAlive()))

thread = **new** Thread(**this**);

thread.start();

}

} **catch** (IOException ee) {

System.***out***.println(ee);

socket = **new** Socket();

}

}

**if** (e.getSource() == send) {

String s1 = inputText1.getText();

String s2 = inputText2.getText();

**double** r1 = Double.*parseDouble*(s1);

**double** r2 = Double.*parseDouble*(s2);

**try** {

out.writeDouble(r1);

out.writeDouble(r2);

}

**catch** (IOException e1) {

}

}

}

**public** **void** run() {

String s;

**while** (**true**) {

**try** {

s = in.readUTF();

showResult.append("\n" + s);

} **catch** (IOException e) {

showResult.setText("与服务器已断开" + e);

socket = **new** Socket();

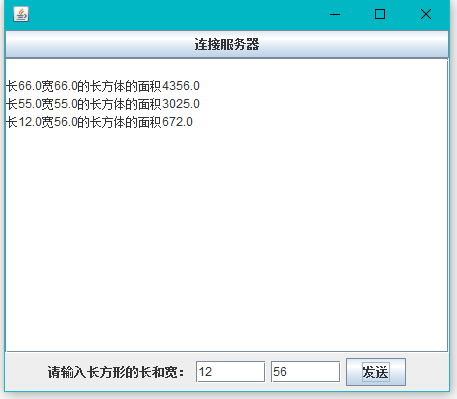
**break**;

}

}

}

}

**五、测试结果及说明**

**六、实验体会**

所谓 socket 通常也称作 "套接字"。应用程序通常通过 "套接字" 向网络发出请求或者应答网络请求。事实上网络编程简单的理解就是两台计算机相互通讯数据而已，对于 Java 而言，这些 Api 存在与 java.net 这个包里面。因此只要导入这个包就可以准备网络编程了。 网络编程的基本模型就是客户机到服务器模型。简单的说就是两个进程之间相互通讯，然后其中一个必须提供一个固定的位置，而另一个则只需要知道这个固定的位置。并去建立两者之间的联系.. 然后完成数据的通讯就可以了，也就是网络编程了。

因为端口是为了唯一标识每台计算机唯一服务的。另外端口号是从 0~65535 之间的，前 1024 个端口已经被 Tcp/Ip 作为保留端口。