

温州大学计算机与人工智能学院

Java程序设计（17网工） 课程作业

实验名称	计算圆的面积（客户端）				
班 级	18电科2	姓 名	方涛涛	学 号	18211110208
实验地点		实验时间	2020-12-31,14:27:54	指导老师	

一、问题编号：

2212

地址：<http://10.132.254.54/problem/2212/>

二、问题描述：

给出一个客户端程序和一个服务器程序。

客户端向服务器发送数据。服务器接收数据，并用它来生成一个结果，然后，将这个结果返回给客户端。

*在本例中，客户端发送的数据是圆的半径，服务器生成的结果是圆的面积，如下图所示。

三、输入说明：

四、输出说明：

五、输入样例：

六、输出样例：

七、解答内容：

所用语言：

源代码：

```
001. import java.awt.BorderLayout;
002. import java.awt.Color;
003. import java.awt.event.ActionEvent;
004. import java.awt.event.ActionListener;
005. import java.io.BufferedReader;
006. import java.io.IOException;
007. import java.io.InputStreamReader;
008. import java.io.PrintWriter;
009. import java.net.Socket;
010.
011. import javax.swing.BorderFactory;
012. import javax.swing.JButton;
013. import javax.swing.JFrame;
014. import javax.swing.JPanel;
015. import javax.swing.JScrollPane;
016. import javax.swing.JTextArea;
017.
018. public class Main extends JFrame {
019.     Socket s = null;
020.     Client client = new Client();
021.     final JTextArea messageText = null;
022.
023.     public Main() {
024.         client.connect("127.0.0.1",1237);
```

```

025. JButton BtnSend = new JButton("Enter radius");
026. JPanel jp1 = new JPanel();
027. jp1.setBackground(Color.WHITE);
028. final JTextArea messageText= new JTextArea();
029. final JTextArea sendText = new JTextArea();
030. sendText.setBorder( BorderFactory.createLineBorder(Color.BLACK));
031. sendText.setLineWrap(true);
032. JScrollPane jsp = new JScrollPane(messageText);
033. jsp.setBounds(13, 10, 350, 340);
034. jsp.setVerticalScrollBarPolicy( JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
035.
036. jp1.add(BtnSend);
037. jp1.add(sendText);
038.
039. BtnSend.addActionListener(new ActionListener() {
040.     @Override
041.     public void actionPerformed(ActionEvent e) {
042.         messageText.append("radius:" + Double.parseDouble(sendText.getText()));
043.         messageText.append("\r\n");
044.         new Thread(new Runnable() {
045.             @Override
046.             public void run() {
047.                 client.sendMessage(sendText.getText());
048.             }
049.         }).start();
050.     }
051. });
052. setLayout(new BorderLayout());
053. add(jp1,BorderLayout.NORTH);
054. add(jsp, BorderLayout.CENTER);
055. setTitle("Client");
056. setSize(400, 400);
057. setLocation(300, 300);
058. setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
059. setVisible(true);
060.
061. new Thread(new Runnable() {
062.     @Override
063.     public void run() {
064.         while (true) {
065.             try {
066.                 String a = client.receiveMessage();
067.                 if (a != null) {
068.                     messageText.append("area from server:" +a);
069.                     messageText.append("\r\n");
070.                 }
071.             } catch (Exception e) {
072.             }
073.         }
074.     }
075. }).start();
076.
077. }
078.
079.
080.
081.
082. public static void main(String[] args) {
083.     new Main();
084. }
085.
086. }
087.
088. class Client {
089.     private Socket socket;
090.
091.     Client(){
092.     }
093.
094.     public void connect(String ip, int port) {
095.         try {
096.             socket = new Socket(ip, port);
097.         } catch (IOException e) {
098.             e.printStackTrace();
099.         }
100.     }
101.
102.
103.     public void close() {
104.         try {
105.             socket.close();
106.

```

```
107.         } catch (IOException e) {
108.             e.printStackTrace();
109.         }
110.     }
111.
112.     public void sendMessage(String message) {
113.         try {
114.             PrintWriter pw = new PrintWriter(socket.getOutputStream());
115.             pw.println(message);
116.             pw.flush();
117.         } catch (IOException e) {
118.             e.printStackTrace();
119.         }
120.     }
121.
122.     public String receiveMessage() {
123.         try {
124.             BufferedReader b = new BufferedReader(
125.                 new InputStreamReader(socket.getInputStream()));
126.             return b.readLine();
127.         } catch (IOException e) {
128.             e.printStackTrace();
129.         }
130.         return null;
131.     }
132. }
133.
134. }
```

八、判题结果

AC - 正确

判题结果补充说明: