

深圳大学实验报告

课程名称： 计算机网络

实验项目名称： Socket 网络编程

学院： 计算机与软件学院

专业： 计算机科学与技术

指导教师： 李雪亮

报告人： 林宪亮 学号： 2022150130 班级： 国际班

实验时间： 2024 年 4 月 10 日

实验报告提交时间： 2024 年 4 月 10 日

教务处制

实验目的：

掌握通过 TCP 实现网络文件传输的方法，掌握 TCP 连接是基于字符流。

实验环境：

使用 Windows 操作系统；Internet 连接
Eclipse+Java 开发环境

实验内容：

通过 Socket 和 ServerSocket 类实现 TCP 文件传输。

实验步骤：

（用文字描述实验过程，可用截图辅助说明）

利用 Socket 类和 ServerSocket 类，编写一个 C/S 程序，实现网络文件传输。

• 服务器端：

```
1. ServerSocket serverSocket = new ServerSocket(12345); // 创建服务器Socket，端口号12345
2. System.out.println("服务器启动，等待客户端连接...");
3.
4. Socket clientSocket = serverSocket.accept(); // 等待客户端连接
5. System.out.println("客户端连接成功！");
6.
7. // 获取输入输出流
8. BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));
9. PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);
10.
11. // 读取客户端发送的文件名
12. String fileName = in.readLine();
13. System.out.println("客户端请求文件：" + fileName);
14.
15. // 发送文件
16. File fileToSend = new File(fileName);
17. if (fileToSend.exists()) {
18. System.out.println("开始传输文件");
19. out.println(fileToSend.length()); // 发送文件长度给客户端
20. FileInputStream fileInputStream = new FileInputStream(fileToSe
```

```

        nd);
21. byte[] buffer = new byte[1024];
22. int bytesRead;
23. while ((bytesRead = fileInputStream.read(buffer)) != -1) {
24.     clientSocket.getOutputStream().write(buffer, 0, bytesRead);
25. }
26. fileInputStream.close();
27. System.out.println("文件发送完毕!");
28. } else {
29.     out.println(-1); // 文件不存在
30.     System.out.println("文件不存在!");
31. }
32.
33. // 关闭连接
34. in.close();
35. out.close();
36. clientSocket.close();
37. serverSocket.close();

```

以上为服务器端代码，首先创建一个端口号为 040311 的服务器，之后输出服务器启动提示信息，紧接着等待客户端连接，待客户端连接后输出提示信息，创建输入输出流用来接收目标文件信息以及输出文件的内容。等待读取客户发送的目标文件名之后，打印客户请求的文件名称，然后判断文件是否存在，如果文件存在，则发送文件的长度给客户用来告知客户文件是否存在，如果长度为-1 则目标文件不存在。之后就可以开始发送文件的内容了。最后输出提示信息文件发送完毕，然后关闭输入输出流以及服务器端，客户端。

• 客户端:

```

1. Socket socket = new Socket("localhost", 12345); // 连接服务器
2. System.out.println("成功连接服务器!");
3.
4. // 获取输入输出流
5. BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
6. PrintWriter out = new PrintWriter(socket.getOutputStream(), true);
7.
8. // 发送文件名
9. String fileName = "D:\\实验报告\\计算机网络\\CODE\\EX4\\src\\lx1.txt"; // 假设要请求的文件名为 test.txt
10. System.out.println("需要的文件为: "+fileName);
11. out.println(fileName);
12.

```

```
13. // 接收服务器发送的文件长度
14. int fileLength = Integer.parseInt(in.readLine());
15. if (fileLength != -1) {
16.     FileOutputStream fileOutputStream = new FileOutputStream("received_lxl.txt"); // 接收到的文件保存为received_file.txt
17.     byte[] buffer = new byte[1024];
18.     int bytesRead;
19.     while (fileLength > 0 && (bytesRead = socket.getInputStream().
        read(buffer, 0, Math.min(buffer.length, fileLength))) != -1) {
20.         fileOutputStream.write(buffer, 0, bytesRead);
21.         fileLength -= bytesRead;
22.     }
23.     fileOutputStream.close();
24.     System.out.println("文件接收完毕! ");
25.     System.out.println("文件保存为: received_lxl.txt");
26.     } else {
27.         System.out.println("服务器端文件不存在! ");
28.     }
29.
30. // 关闭连接
31. in.close();
32. out.close();
33. socket.close();
```

以上为客户端的代码，首先新建一个客户端连接到服务器，输出提示信息后，创建输入输出流，接着指定需要的文件名然后输出名称。之后开始等待服务器端传输的文件长度信息，如果长度为-1则文件不存在，程序结束。如果长度不为-1，则开始接收文件，最后输出文件保存信息。

实验结果：

（此页附完成的实验结果、并给出个人对结果的分析、结论）

• 客户端：

```
成功连接服务器！  
需要的文件为：D:\实验报告\计算机网络\CODE\EX4\src\lxl.txt  
文件接收完毕！  
文件保存为：received_lxl.txt
```

图 1 客户端输出结果

• 服务器端：

```
服务器启动，等待客户端连接...  
客户端连接成功！  
客户端请求文件：D:\实验报告\计算机网络\CODE\EX4\src\lxl.txt  
开始传输文件  
文件发送完毕！
```

图 2 服务器端输出结果

如图 1, 2, 客户端向服务器端请求文件，服务器端接收请求后，先传输文件名然后传输文件内容，客户端接收文件后，对文件进行改名保存。符合实验要求。

lxl.txt 文件内容：

```
Java is useful !!!
```

图 3 目标文件内容

Received_lxl.txt 文件内容；

```
Java is useful !!!
```

图 4 保存的文件内容

目标文件和保存的文件内容一致，传输成功。

实验小结:

(实验中出现问题解决方法, 实验心得体会等)

本次实验我掌握通过 TCP 实现网络文件传输的方法, 通过使用 java 代码实验客户端和服务端端的文件传输, 让我对网络文件传输有了更加深刻的理解。

让我直观的认识到了网络文件传输是如何工作的。

在实验中, 我出现过程序一直输出文件不存在的情况, 后面使用绝对文件路径就解决了这个问题, 相比相对文件路径, 绝对文件路径可以减少一些不在预期内的错误, 对文件路径使用不熟练使用绝对文件路径更好。

指导教师批阅意见:

成绩评定:

指导教师签字:

年 月 日

备注：