

JAVA 课程设计报告

题目：简易多人在线聊天室_____

姓名：李昊_____

学号：**2012239**_____

一：问题描述

1. 开发背景

当今社会，网络技术越来越发展，可以说，二十一世纪就是网络的世纪。网络迅猛发展，势不可挡。为了实现资源共享，各式各样的网站发展迅速，各种各样的聊天工具不断更新。本课程通过使用网络访问工具，数据库访问工具，多线程控制等方式制作简易多人聊天室。

网络编程的目的就是指直接或间接地通过网络协议与其他计算机进行通讯。两个主要的问题，一个是如何准确的定位网络上一台或多台主机，另一个就是找到主机后如何可靠高效的进行数据传输。在 TCP / IP 协议中 IP 层主要负责网络主机的定位，数据传输的路由，由 IP 地址可以唯一地确定 Internet 上的一台主机。而 TCP 层则提供面向应用的可靠的或非可靠的数据传输机制，这是网络编程的主要对象，一般不需要关心 IP 层是如何处理数据的。目前较为流行的网络编程模型是客户机 / 服务器 (C / S) 结构。即通信双方一方作为服务器等待客户提出请求并予以响应。客户则在需要服务时向服务器提出申请。服务器一般作为守护进程始终运行，监听网络端口，一旦有客户请求，就会启动一个服务进程来响应该客户，同时自己继续监听服务端口，使后来的客户也能及时得到服务。

为了实现一个更好的网络交互平台，各用户之间能够实时方便的传递信息，按照用户的要求不同，可以在聊天室里实现公共聊天，也可以邀几个私交好友私聊，同时提供了两个用户之间能够传送文件。本系统是基于方便用户聊天，传递信息，共享资源，所以系统的功能主要是从以上几个模块着手。

2. 需求分析

运用 java 程序编写聊天室，实现简单的聊天功能。它是图形界面，线程，流与文件系统等技术的综合应用。其界面主要采用了 java.awt 包，java.swing 包以及 java.util 包等。程序实现了聊天室的基本功能，其中有：

- 1) 用户登录
- 2) 用户注册
- 3) 用户列表板块
- 4) 用户私聊

- 5) 聊天室公聊
- 6) 发送表情
- 7) 服务端管理及生成日志
- 8) 粘贴，清屏等其他功能

二：聊天室系统设计

1. 系统目标

根据以上的需求分析，该系统应达到以下目标：

- 1) 界面设计友好，美观，清晰大方。
- 2) 采用套接字实现网络中的数据传输
- 3) 使用某个或某些应用层的协议实现系统的部分功能
- 4) 系统设计规范合理，操作流程方便简单

2. 基本设计理念

聊天室分为两大部分，分别是服务器端的实现和客户端的实现。服务器端主要负责侦听客户端发来的信息，进行处理后转发给对应的客户端。客户端在登录到服务器端之后，进行一系列操作，接收来自服务器端反馈后实现一系列功能

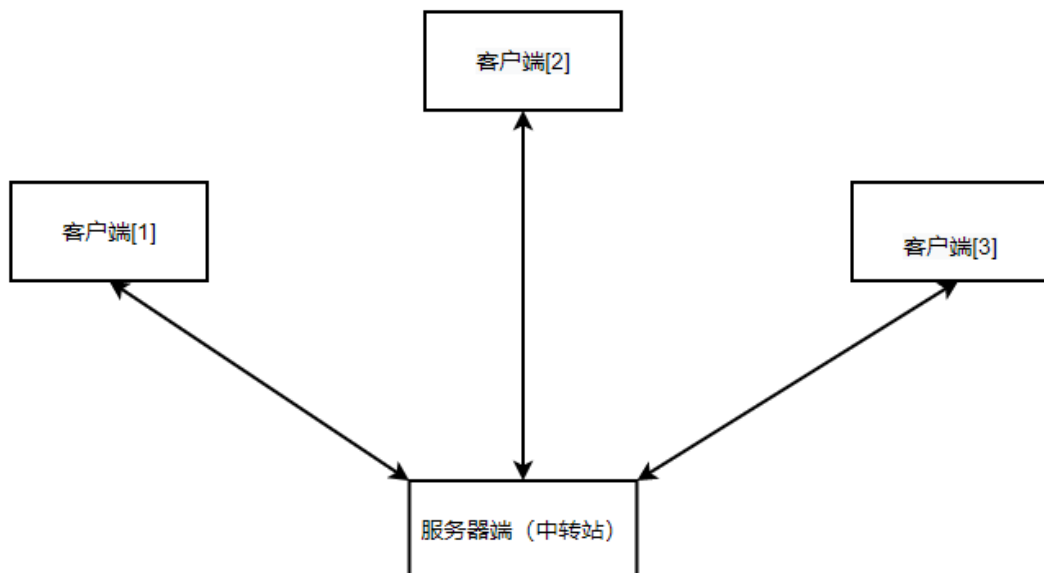


图 1 C/S 结构

其中服务器首先通过验证客户端输入的信息（账号密码）是否正确，如果正确才会进行连接

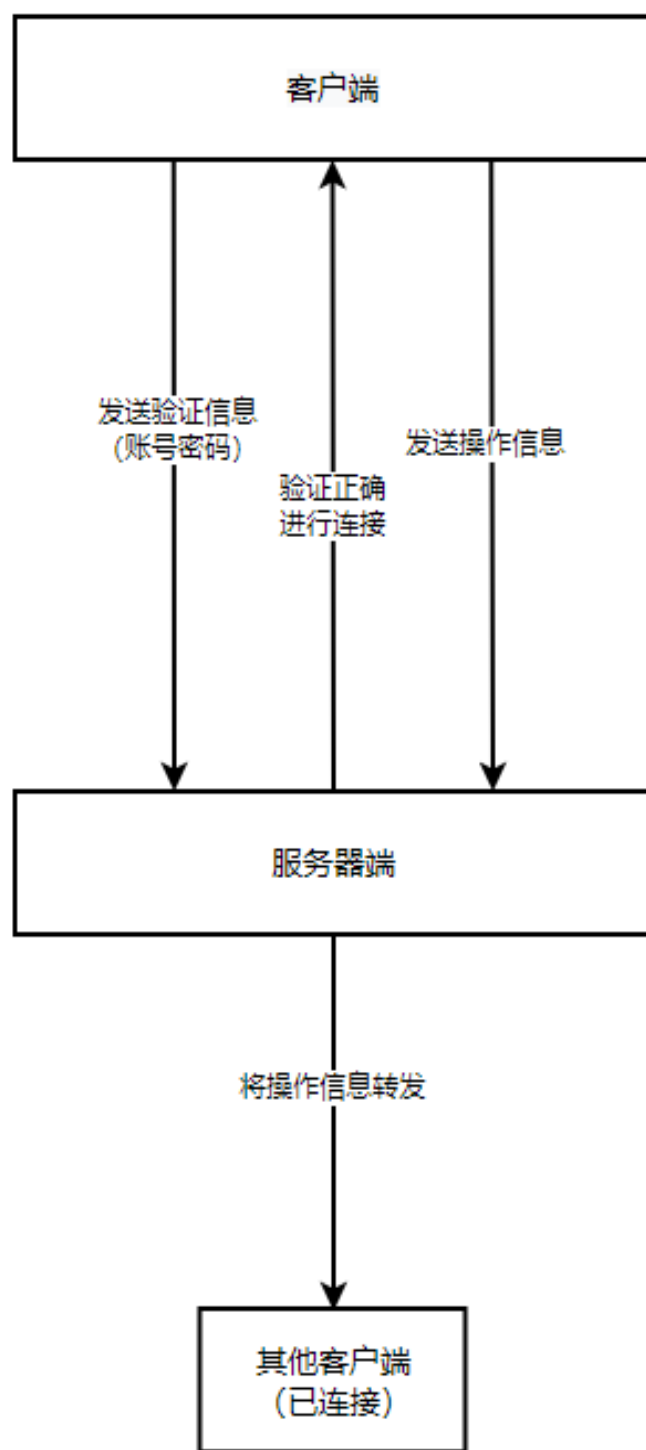


图 2 信息交互

服务器端和客户端分别实现自己的功能

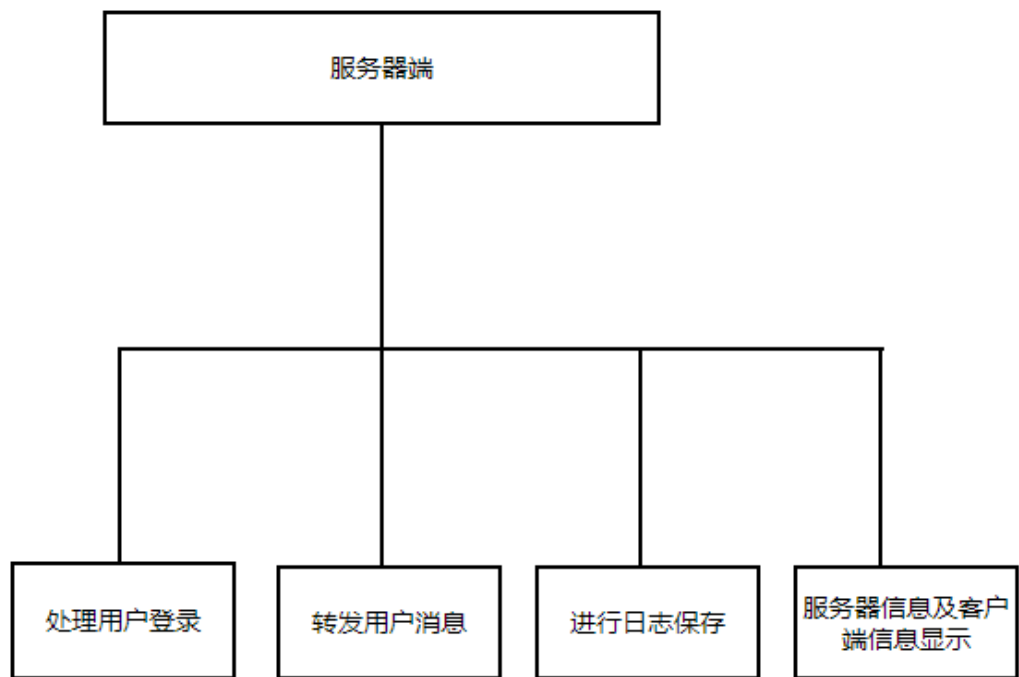


图 3 服务器端设计

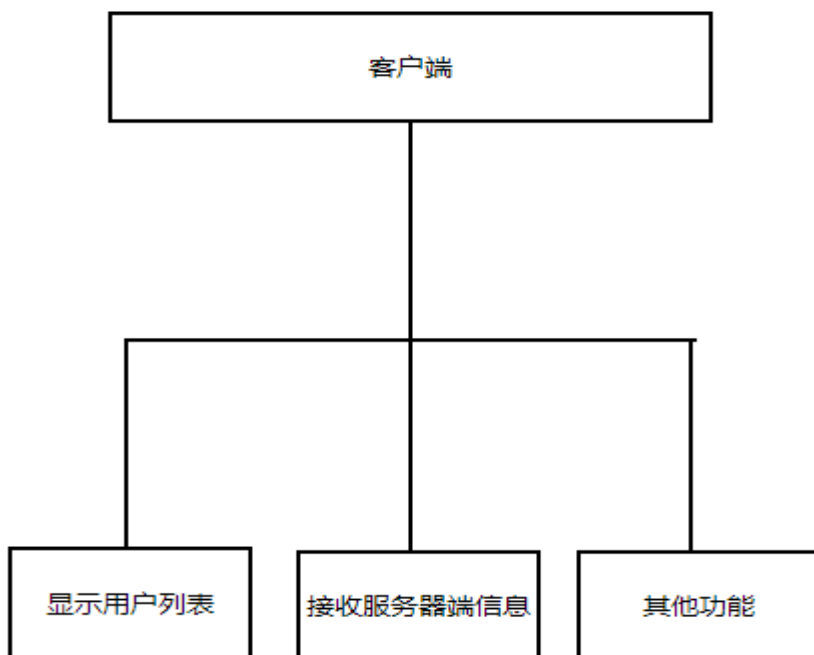


图 4 客户端设计

3. 设计方案

当客户程序需要与服务器程序通讯的时候，客户程序在客户机创建一个 socket 对象。创建了一个 Socket 对象后，那么它可以通过调用 Socket 的 `getInputStream0` 方法从服务程序获得输入流读传送来的信息，也可以通过调用 Socket 的 `getOutputStream0` 方法获得输出流来发送消息。

由于 Client 使用了流套接字，所以服务程序也要使用流套接字。这就要创建 `ServerSocket` 对象，接下来服务程序进入无限循环之中，无限循环从调用 `ServerSocket` 的 `accept0` 方法开始，在调用开始后 `accept` 方法将导致调用线程阻塞直到连接建立。在建立连接后，`accept` 返回一个最近创建的 Socket 对象，该 Socket 对象绑定了客户程序的 IP 地址或端口号。由于存在单个服务程序与多个客户程序通讯的可能，服务程序循环检测是否有客户连接到服务器上，如果有，则创建一个线程来服务这个客户，以此完成对客户端的监听。

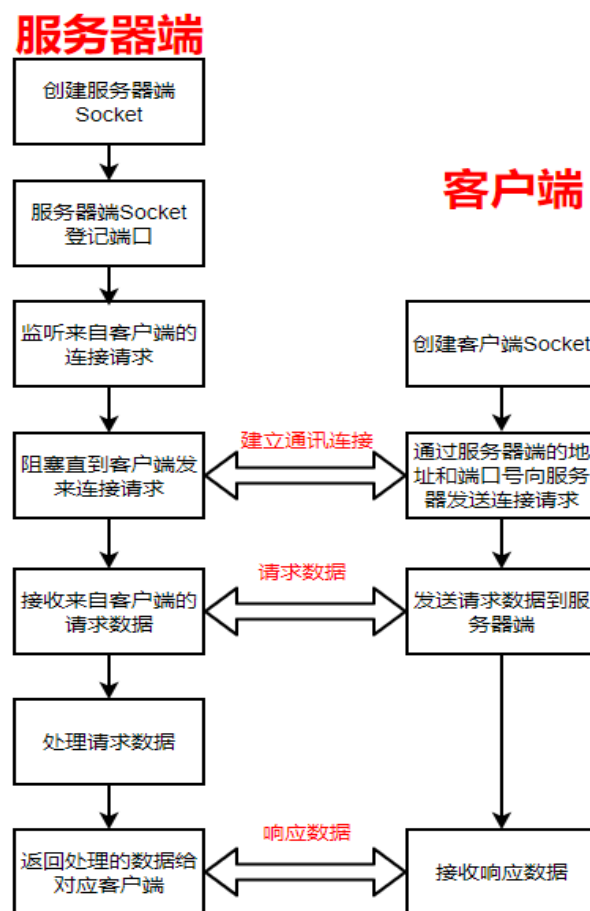


图 5 服务器端与客户端交互

4. 服务器端的实现

服务器端的实现分为三大部分，分别是服务端图形用户接口（GUI）的实现，服务器端处理信息内核的实现以及服务端启动的实现。分别为 `ServerFrame` 类，`ServerProcess` 类以及 `ChatroomServer` 类。

`ChatroomServer` 实现较简单，首先创建一个 `ServerSocket` 实例，定义端口号（默认为 8888），并启动服务。然后创建一个服务器框架 `ServerFrame`，并且得到服务器相关参数后将其显示在服务器框架中，之后创建无限循环监听客户端的连接请求，如果接到请求，则创建一个新线程（处理内核）`ServerProcess` 实例，将框架和套接字作为参数传入。

`ServerFrame` 的实现虽然麻烦，但是原理简单。首先通过 `JTabbedPane` 类实现选项卡界面，两大界面分别为服务器信息管理以及日志输出。服务器信息管理界面需要四个 `JLabel` 实例作为标签，和四个 `JTextField` 实例作为文本框展示服务器参数，这些实例传入 `JPanel` 的实例 `pnlServerInfo` 中作为第一部分。将一个 `JList` 实例传入 `JScrollPane` 实例中实现带滚动条的用户列表，加上若干标签传入 `JPanel` 实例 `pnlServer` 中作为第二部分，两者组成服务器信息管理界面。日志管理输出界面同理。

服务器

在线用户

服务器信息

软件信息

当前在线人数:

0 人

服务器名称:

LAPT0P-C4FJ9SRJ

服务器IP:

192.168.31.248

服务器端口:

8888

[在线用户列表]



ServerProcess 的实现最为复杂。首先定义一些私有成员变量。in 和 out 分别为 BufferedReader 输入流和 PrintWriter 输出流，分别用来接收客户端的信息，和转发客户端的信息。onlineUser 和 socketUser 用来存放客户端的账户字符串和客户端的套接字。st 为具有标记“|”的字符串（分隔符之前的内容用于判断条件，之后的内容为一些字符串或者提示等），用于分割并且判断条件。sFrame 为服务器框架。

首先是数据库的连接和账号密码的登录和注册。定义 select, selectyw 和 insert 三个成员方法用于实现使用 JDBC 连接数据库，并且进行登录注册。

成员函数 isExistUser 用于判断是否存在有该用户，逻辑为选择数据库中的所有账号并且逐一比对，返回 Boolean 值。

成员函数 isUserLogin 用于判断用户名密码是否正确，判断方法同理。

成员函数 register 用于实现注册功能，首先判断用户是否存在，如不存在，则注册成功，并且自动登录。

成员函数 `userLoginSuccess` 用于实现登录功能，登录后添加用户列表和相应客户端套接字。

成员函数 `login` 用于判断登录条件，首先判断在线列表该账户是否已登录，然后判断用户名和密码是否正确，如成功，则调用成员函数 `userLoginSuccess`。

以上为登录注册的内容。然后就是聊天信息的处理，也是本项目的难点之一。定义成员方法 `talk`，通过 `st` 得到聊天内容，发消息人，收消息人后判断收消息人是否为“ALL”（全体消息），如果是私聊消息，遍历用户列表判断收消息人在用户列表中的次序，取得相对应的套接字，调用 `getOutputStream0` 方法向其输出字符串“`talk|[私聊 TO me]`”+相对应的聊天消息，用于发送给接收私聊消息的相应客户端，同样的找到发送者获得套接字，发送“`talk|[私聊 TO +strReceiver+]`”实现发送者显示私聊消息。私聊或者公聊发送表情原理相同。

成员函数 `freshClientsOnline()`用于刷新用户列表。

成员函数 `closeSocket()`用于关闭客户端套接字和流，并且将其移除在线列表和套接字列表。

最后是重写的 `run` 方法，通过对 `in` 流输入的信息进行分割，“|”前分为五类：`Login`（登录），`talk`（聊天），`init`（刷新界面），`reg`（注册），`expression`（发送表情），分别调用不同的成员方法即可。如果用户退出，则捕捉到 `IOException`，调用 `freshClientsOnline()`刷新界面。

5. 客户端的实现

客户端的实现主要是客户端框架部分和客户端内核的实现。



客户端框架大部分部件与服务器端的部件相同，以下为不同部分的说明：

首先为私聊下拉列表的实现，使用了 JComboBox 类。

为实现每个客户端显示自己为[user],使用了 DefaultListModel 的实例作为 JList 的参数。

发表表情通过在 Button 上实现鼠标事件 butExpression，显示 JPanel 的实例，其中每个表情也都是一个 Button，分别实现鼠标事件向 PrintWriter 流中输出“expression|”+相应表情。

右键点击为菜单，可以选择退出菜单或者清屏，两者为 JMenuItem 的实例，并且添加到 JPopupMenu 中。

接下来是客户端内核的实现，客户端内核实现的难点主要在于鼠标事件的实现以及与服务器端的交互。

首先是注册 Button 事件的实现，通过调用 PrintWriter，向服务器端输出"reg|"+账号+密码，用于服务器端获取相应账号和密码进行注册。

然后是 actionPerformed 的重写，通过调用鼠标事件的 getSource()方法判断，如果点击登录按键，则发送"login|"+账号+密码用于服务器端判断登录条件。

如果点击注册按键，则显示注册界面。如果点击发送按键，则发送"talk|"+选中文

本+发信息者+收信息者，用于服务器端进行分割，实现私聊和公聊。

最后是 `run` 方法的重写，通过服务器的反馈，用 `BufferedReader` 实例接收信息，并且通过分割符判断以下几类：

- a) 如果是 `talk` 则发送相对应的消息。
- b) 如果是 `expression` 则发送表情。
- c) 如果是 `online` 则向用户列表以及私聊下拉列表中添加用户。
- d) 如果是 `remove` 则用户列表以及私聊下拉列表中删除用户。
- e) 如果是 `warning` 则弹出警告框，显示相应的警告。
- f) 如果是 `login` 则将登录框以及按钮隐藏，并且创建客户端线程，显示客户端。

三：技术难点和优缺点分析

1. 技术难点

本项目的技术难点分为四个部分：服务器端框架内核的实现，客户端框架内核的实现，服务器与客户端之间的交互，异常的处理。

服务器端内核实现主要通过分析客户端传来的信息，来调用不同的成员函数并且向客户端发送反馈，其中私聊和群聊的设计较困难，需要通过设计客户端发送来的消息进行分割，并且取得发送者和接受者套接字的客户端分别进行不同信息的发送（群发就是向所有客户端发一样的消息）。还有登录和注册的设计，需要通过客户端发送信息设计，分割出账号和密码，再通过调用数据库中的信息进行比对后判断登录注册的条件。

而客户端内核的实现主要为接收服务器端的反馈，然后通过分割字符串调用不同的方法（用户列表的分割和遍历同理），其中按钮事件的重写为难点之一，通过不同的按钮触发相对应的事件。

服务器和客户端的交互通过 `BufferedReader` 和 `PrintWriter` 进行信息交互，其中一种信息的设计为通过“|”符号的分割，前半部分为判断调用方法的类型，后半部分为输出至控制台，日志，或者聊天部分的文本区域。另一种为登录注册信息的设计，前半部分为登录或注册的标签，后半部分为账号+密码。

异常的处理例如登录或者注册账号密码错误的处理，客户端主动或者异常退出的处理，由于需要考虑的部分过多，很容易漏掉。

2. 优缺点分析

优点：该程序做到了实时更新数据，比如当有人退出或者有人进入的时候，会进行界面的刷新，进行用户列表的刷新，并且可以进行私聊或者群聊。

该程序还做到了进行日志的保存，对于后续的维护有很大的好处。并且因为程序连接了数据库，对于数据的保存以及安全性有很大保障。

缺点：该程序功能并不是很完善，如果要改进的话应从以下几个方面入手：

- 1) 功能的多样化，例如增加文件上传下载的功能，增加语音发送的功能等
- 2) 结构的复杂化，例如增加群的创建，私人好友列表的创建，同时可以同步数据库进行信息同步，将数据库的结构同样变得更加多样。
- 3) 性能优化，可以在一些地方进行资源释放等操作，并且如果聊天室人数增加，查找相对应用户功能可以选择一些时间复杂度低的算法。
- 4) 完善异常处理，比如一些用户账号密码长度的限制以及非法字符的输入的处理。
- 5) 权限管理分级，本程序没有区分管理员和一般用户，针对管理员可以增加一些高级权限操作，比如重启服务器，禁言用户，封禁用户等等，将这些权限分为不同的权限组，赋予不同权限级别的账号，实现更加严密的管理。

四：本学期收获

本学期学习了 java 相关基本语法知识，学习了 java 中面向对象的封装思想，使代码可读性更高，更加容易维护，然后了解了异常的处理和类的反射机制，反射是 java 提供的一个重要功能，可以在运行时检查类、接口、方法和变量等信息，无需知道类的名字，方法名等。学习了多线程基础，对于不同的线程进行加锁等操作。学习了 IO 流，程序中需要根据待传输数据的不同特性而使用不同的流。最后学习了网络编程和数据库编程，可以通过 Socket 使用 TCP 或者 UDP 协议来进行不同客户端的数据传输。

参考文献：

[1]胡晶晶,鲁漫红. 基于 Java Socket 的聊天室系统设计与实现[J]. 微处理机,2010,31(05):66-67+70.

[2]汪桃华,宋兵,万方. 基于 java 的局域网聊天系统[J]. 内江师范学院学报,2008,(S1):242-245.

[3]任琴,孔令慧,秦冰. 基于 JAVA 的多人聊天室的系统设计[J]. 时代农机,2018,45(01):142.

[4] Java 聊天室 (实现 群聊 、 私聊 功能) GUI 界面
https://blog.csdn.net/to10086/article/details/108739346?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522165654632316781647598766%2522%252C%2522scm%2522%253A%252220140713.130102334..%2522%257D&request_id=165654632316781647598766&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~top_click~default-2-108739346-null-null.142^v26^pc_rank_34,157^v15^new_3&utm_term=java%E8%81%8A%E5%A4%A9%E5%AE%A4&spm=1018.2226.3001.4187