目录

[一、 功能实现： 1](#_Toc135231876)

[1.1 基本设计思想： 1](#_Toc135231877)

[1.2 聊天室服务器端功能实现： 1](#_Toc135231878)

[1.2.1 载入python提供的socket库函数和多线程的库threading 1](#_Toc135231879)

[1.2.2 定义 ChatRoomServer 类，用于创建聊天室服务器。（以下函数均为类内函数） 1](#_Toc135231880)

[1.2.3 构造函数 \_\_init\_\_ 中初始化了服务器的 IP 地址、监听端口号、套接字、存储用户连接和群组信息的数据结构。 1](#_Toc135231881)

[1.2.4 定义 run 方法，该方法用于启动服务器并接受客户端连接 2](#_Toc135231882)

[1.2.5 定义handle\_user()方法用于处理客户端请求 2](#_Toc135231883)

[1.2.6 接下来，该方法（handle\_user()）通过一个无限循环来等待客户端的命令。 2](#_Toc135231884)

[1.2.7 定义主机地址和服务器端口，在主函数中启动程序 5](#_Toc135231885)

[1.3 聊天室客户端功能实现： 5](#_Toc135231886)

[1.3.1 载入python提供的socket库函数和多线程的库threading 5](#_Toc135231887)

[1.3.2 定义聊天室类（以下函数均为该类的类内函数） 5](#_Toc135231888)

[1.3.3 定义类的构造函数，接受主机地址和端口号作为参数，并初始化服务器的相关属性。 6](#_Toc135231889)

[1.3.4 定义连接函数，实现客户端连接到服务器并开始与之通信的功能。 6](#_Toc135231890)

[1.3.5 定义菜单，在每次指令结束的时候调用它 6](#_Toc135231891)

[1.3.6 定义方法实现用户在客户端的选择菜单，不断接收用户的输入，并将相应的消息发送给服务器。 6](#_Toc135231892)

[1.3.7 定义方法用于在客户端接收来自服务器的消息 8](#_Toc135231893)

[1.3.8 调用主函数，传入服务器地址和端口号，启动客户端并与服务器建立连接。 8](#_Toc135231894)

[二、 要求的功能（实验结果） 8](#_Toc135231895)

[2.1 启动服务器以及客户端： 8](#_Toc135231898)

[2.2 用户注册： 9](#_Toc135231899)

[2.3 两个用户服务器发送私密消息: 9](#_Toc135231900)

[2.4 用户创建一个或多个聊天群组 10](#_Toc135231901)

[2.4.1 通过开启多个聊天程序客户端，即可同时上线多个用户进行群聊： 10](#_Toc135231902)

[2.4.2 创建多个群组 11](#_Toc135231903)

[2.5 用户加入群组、在群组内发送消息、离开群组 12](#_Toc135231904)

[2.5.1 加入群组： 12](#_Toc135231905)

[2.5.2 在群组内发消息： 13](#_Toc135231906)

[2.5.3 离开群组： 14](#_Toc135231907)

[2.6 用户之间的聊天信息通过服务器端转发 14](#_Toc135231908)

# 功能实现：

## 基本设计思想：

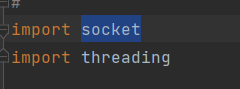
这份代码实现了一个基于 socket 通信的聊天室服务器与客户端。其设计思想是使用一个多线程的服务器，能够同时处理多个客户端的请求，并实现了以下功能：

1. 用户注册功能：客户端连接服务器后输入用户名，服务器将其存储在一个字典中，用于后续的私密消息和群组消息发送。
2. 两个用户服务器发送私密消息：客户端通过 private 命令向指定用户发送私密消息，服务器将该消息转发给指定用户。
3. 用户创建一个或多个聊天群组：客户端通过 create 命令创建聊天群组，服务器将其存储在一个字典中。一个用户可以创建多个聊天群组。
4. 用户加入群组、在群组内发送消息、离开群组：客户端可以通过 join 命令加入指定群组，group 命令向指定群组发送消息，leave 命令离开指定群组。服务器将用户加入或离开群组的信息存储在群组字典中。
5. 用户之间的聊天信息通过服务器端转发：服务器在收到客户端的私密消息或群组消息后，将其转发给指定的用户或群组中的所有成员。

此外，服务器在启动时会创建一个 socket，等待客户端连接。每当有一个新的客户端连接到服务器时，服务器会启动一个新的线程来处理该客户端的请求，以保证多个客户端可以同时连接和通信。

## 聊天室服务器端功能实现：

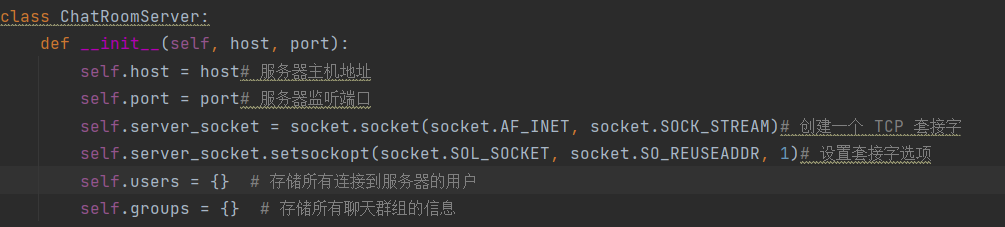
### 载入python提供的socket库函数和多线程的库threading



### 定义 ChatRoomServer 类，用于创建聊天室服务器。（以下函数均为类内函数）



### 构造函数 \_\_init\_\_ 中初始化了服务器的 IP 地址、监听端口号、套接字、存储用户连接和群组信息的数据结构。



### 定义 run 方法，该方法用于启动服务器并接受客户端连接



通过调用 socket 类的 bind 方法绑定主机地址和监听端口号，通过调用 socket 类的 listen 方法开启监听模式。使用无限循环不断接受新连接，每当 有新连接时，创建一个新线程去处理连接。user\_thread 线程调用 handle\_user 方法来处理连接。设置线程为守护线程，并启动线程。

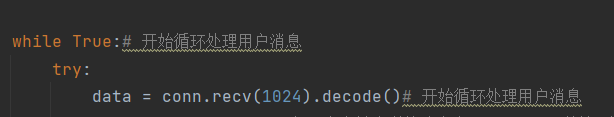
### 定义handle\_user()方法用于处理客户端请求



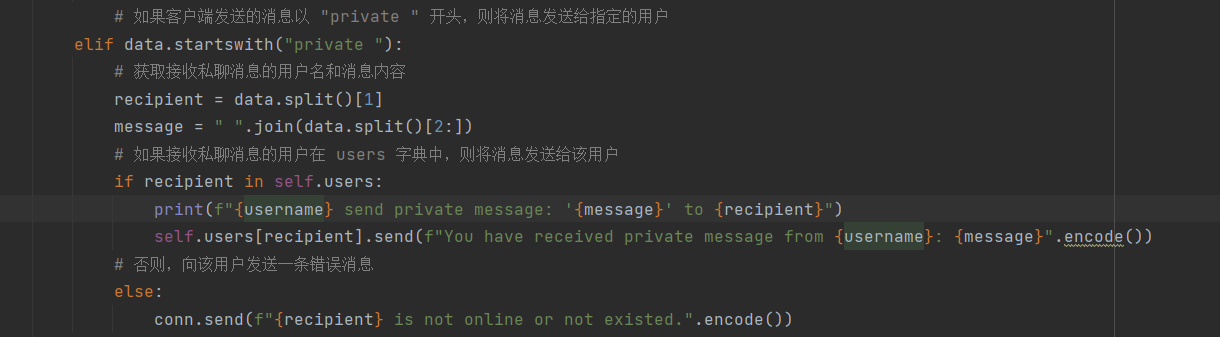
该方法首先从客户端接收用户名，然后根据用户名判断该用户是否已经 连接到服务器。如果该用户未连接到服务器，则将该用户添加到users字 典中，并向该用户发送欢迎信息，并且在服务器端的控制台中显示新加入的用户和目前在线用户的数量；否则向该用户发送用户名已被占用的提示，并关闭该用户的连接。

### 接下来，该方法（handle\_user()）通过一个无限循环来等待客户端的命令。

在while True循环中，服务器会根据接收到的数据进行不同的处理：



#### 1.2.6.1如果数据以"private "开头，表示是私聊信息，服务器会解析出私聊目标的用户名和消息内容，然后向该用户发送私聊消息。



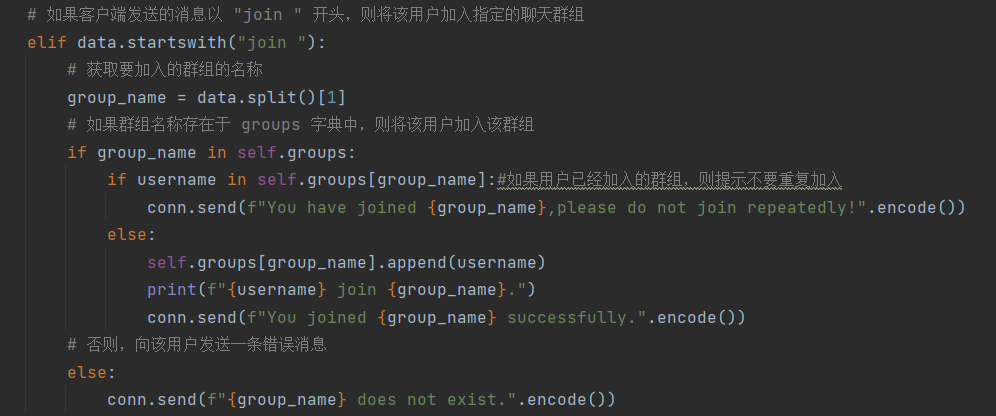
在服务器端中会显示发送方给接收方发送了私密消息。

#### 1.2.6.2如果数据以"create "开头，表示创建新的聊天群组。

服务器会解析出新群组的名称，如果该名称不存在，则创建新的群 组，并将发送该消息的用户添加到该群组中；否则向发送该消息的用户 发送群组已存在的提示。创建群组的消息会在服务器端中展示出来



#### 1.2.6.3如果数据以"join "开头，表示加入已有的聊天群组



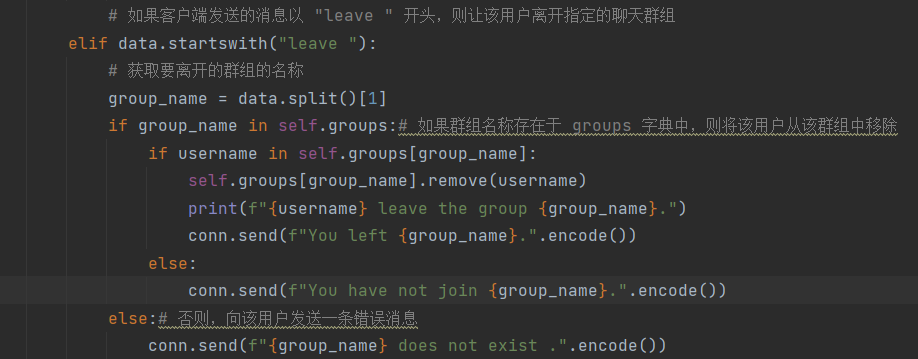
服务器会解析出群组名称，先判断群组是否存在，若群组存在，则接下来判断用户是否在群组中，若在该群组，则发送提示已经加入不要重复加入；

若用户不在群组，将该用户加入到群组中，并向该用户发送加入成功的提示；

若群组不存在向发送该消息的用户发送该群组不存在的提示。

用户加入群组成功的消息会在服务器端中显示。

#### 1.2.6.4如果数据以"leave "开头，表示离开已加入的聊天群组



服务器会解析出群组名称，如果该群组存在，并且该用户也在该群组中则将发送该消息的用户从该群组中移除，用户并向该用户发送离开成功的提示，服务器端中也显示用户离开群组的消息；

若群组存在但是用户不存在，则给用户发送提示消息你未加入群组，无法退出；

若群组不存在，则向发送该消息的用户发送该群组不存在的提示。

#### 1.2.6.5如果数据以"group "开头，表示在群组中发送消息

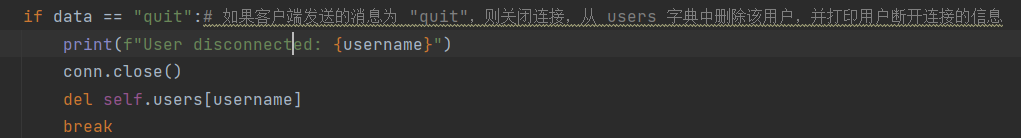


服务器会解析出群组名称和消息内容，如果该群组存在，并且该用户也在群组中，则向该群组中所有成员（除发送该消息的用户以外）发送群组消息；

若用户没有加入群组，则提示用户没有加入群组；

若群组不存在，则向用户发送该消息的用户发送该群组不存在的提示。

#### 1.2.6.6如果接收到的数据以"quit"开头，则表示用户想要退出聊天室

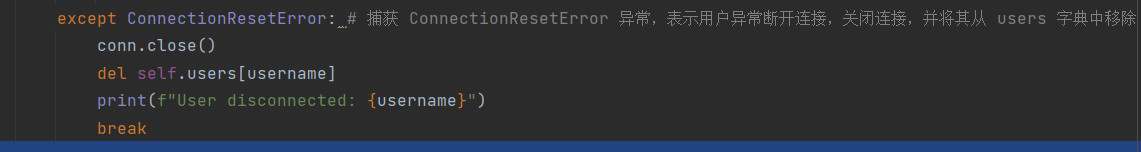


服务器会关闭该用户的连接，并将其从用户列表中移除，并且在服务器中也会显示。

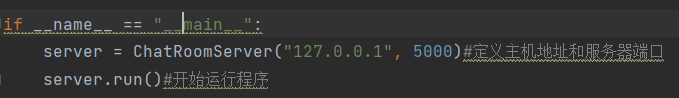
#### 1.2.6.7如果数据不属于以上任何一种类型，则向发送该消息的用户发送无效命令的提示。



#### 1.2.6.8如果在处理过程中捕获到ConnectionResetError异常，则表示用户异常断开连接，服务器也会关闭该用户的连接，并将其从用户列表中移除，在服务器端中显示该用户已经断开连接。



### 定义主机地址和服务器端口，在主函数中启动程序



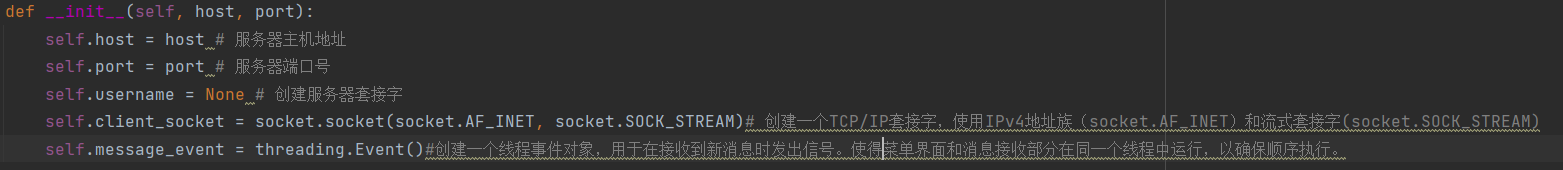
## 聊天室客户端功能实现：

### 载入python提供的socket库函数和多线程的库threading

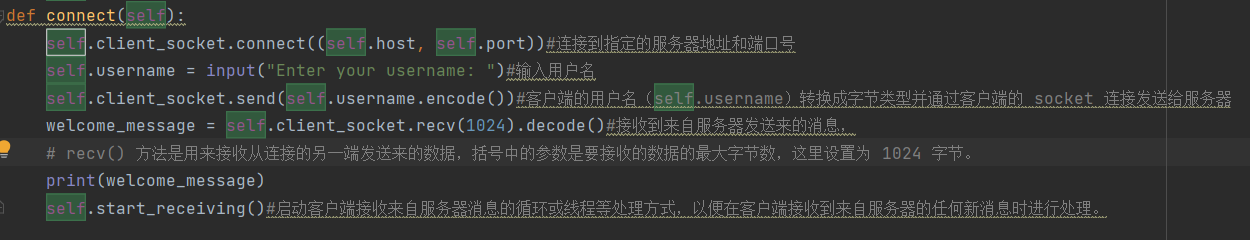
### 定义聊天室类（以下函数均为该类的类内函数）



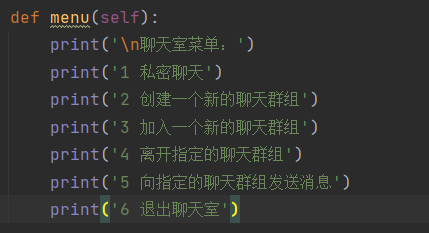
### 定义类的构造函数，接受主机地址和端口号作为参数，并初始化服务器的相关属性同时创建一个线程对象，使得菜单界面和消息接收部分在同一个线程中运行，以确保顺序执行。



### 定义连接函数，实现客户端连接到服务器并开始与之通信的功能。



### 定义菜单，在每次指令结束的时候调用它



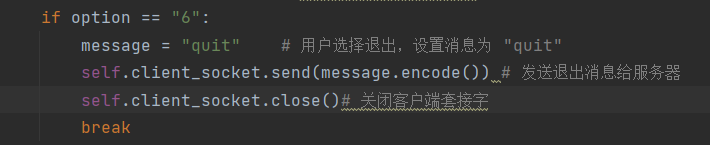
### 定义方法实现用户在客户端的选择菜单，不断接收用户的输入，并将相应的消息发送给服务器。



while循环表示不断循环,接受用户的请求。Option表示用户输入的选项；

利用wait属性函数和clear函数使得客户端等待新消息的到来，并在接收到新消息时先处理消息再显示菜单

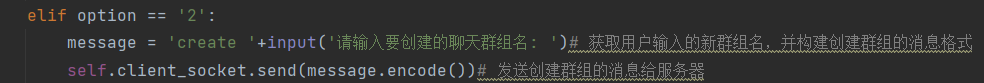
#### 1.3.6.1 如果用户选择 "6"，表示退出聊天室，将消息设置为 "quit"，发送给服务器，并关闭客户端套接字，退出循环。



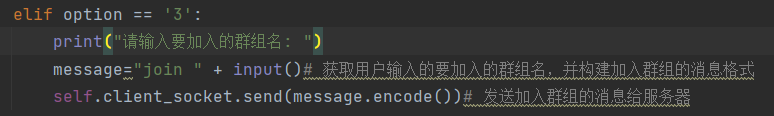
#### 1.3.6.2 如果用户选择 "1"，表示发送私密消息，获取接收者用户名和消息内容，构建私密消息格式的消息，并发送给服务器。



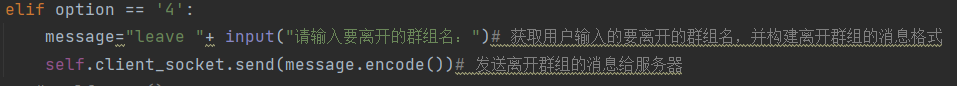
#### 1.3.6.3如果用户选择 "2"，表示创建群组，获取新群组名，构建创建群组的消息格式，并发送给服务器。



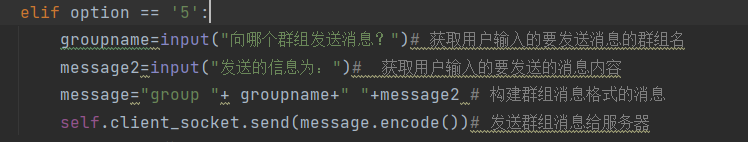
#### 1.3.6.4 如果用户选择 "3"，表示加入群组，获取要加入的群组名，构建加入群组的消息格式，并发送给服务器。



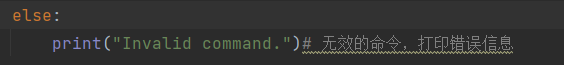
#### 1.3.6.5如果用户选择 "4"，表示离开群组，获取要离开的群组名，构建离开群组的消息格式，并发送给服务器。



#### 1.3.6.6如果用户选择 "5"，表示发送群组消息，获取要发送消息的群组名和消息内容，构建群组消息格式的消息，并发送给服务器。



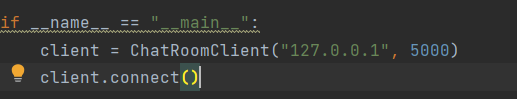
#### 1.3.6.7如果用户输入了无效的选项，打印错误信息提示用户。



### 定义方法用于在客户端接收来自服务器的消息



### 调用主函数，传入服务器地址和端口号，启动客户端并与服务器建立连接。

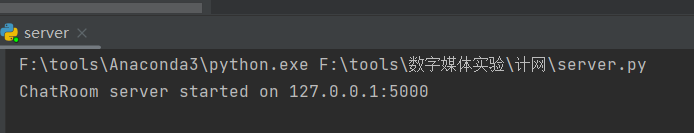


# 要求的功能（实验结果）

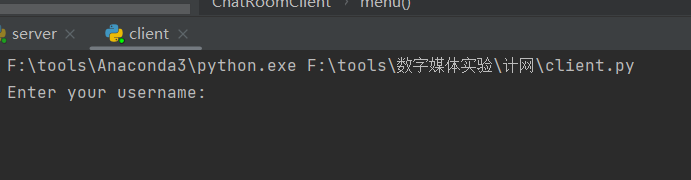


## 启动服务器以及客户端：

先启动服务器，确定该聊天室的主机地址和服务器端口。然后依次启动 3个客户端，定义与服务器相同的主机地址和端口号，启动聊天室。



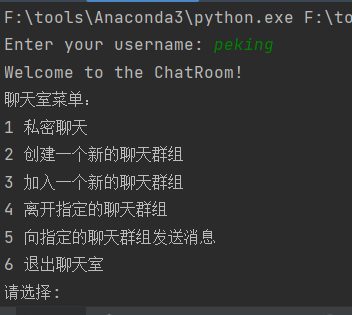
启动服务器



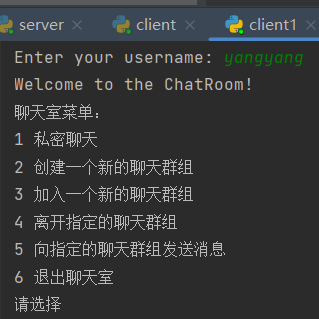
启动客户端

## 用户注册：

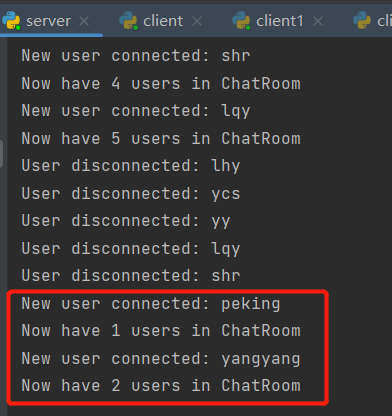
在用户端输入用户的自定义的聊天室昵称，实现用户的注册。之后会显 示菜单，提示用户接下来进行什么操作。



输入用户的昵称为peking



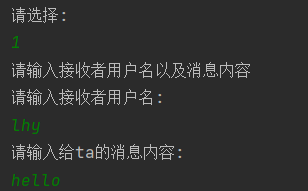
在另一个客户端输入昵称yangyang



服务器端显示新加入的俩个用户

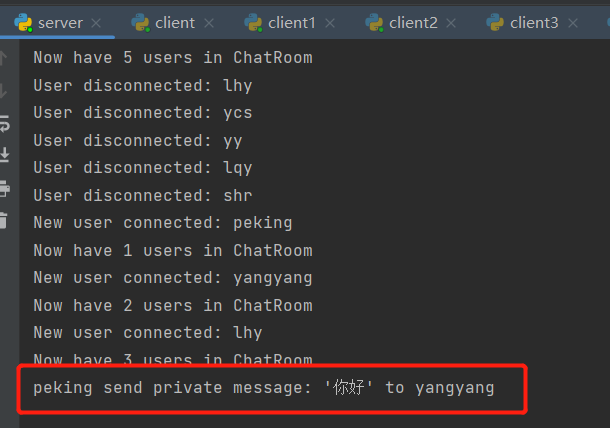
## 两个用户服务器发送私密消息:

用户输入1，实现私密聊天：然后输入接收者的用户名，接下来输入 要发送的内容。



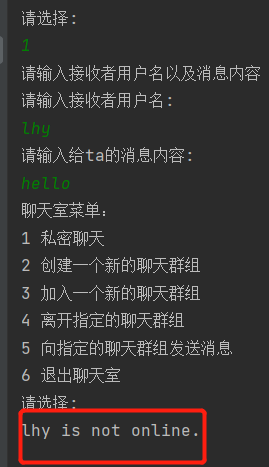


接收方yangyang收到的来自peking的私密消息



服务器端显示发送私密消息

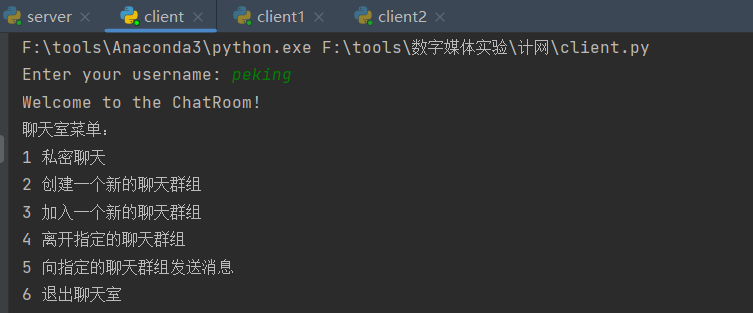
若接收方不在线，没有连接到服务器，那么客户端会提示发送对象不在线。



向lhy发送消息，昵称为lhy的用户未连接到服务器，因此无法向其发送消息，所以提示lhy不在线

## 用户创建一个或多个聊天群组

### 通过开启多个聊天程序客户端，即可同时上线多个用户进行群聊：



启动用户名为peking的客户端



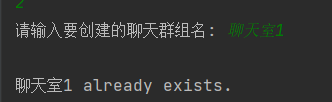
启动用户名为yangyang的客户端



启动用户名为luka的客户端

### 创建多个群组

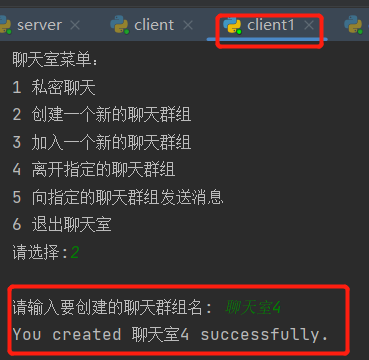
通过peking创建群组聊天室1，聊天室2：



因为之前已经创建过聊天室1，因此提示该群组已经存在，则会提示。



成功创建聊天室2



通过yangyang创建聊天群组“聊天室4”：

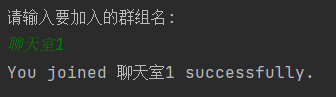
 

服务器端显示已经创建俩个聊天室成功

## 用户加入群组、在群组内发送消息、离开群组

### 加入群组：

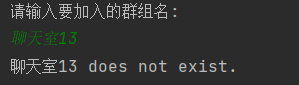
peking和yangyang加入群组“聊天室1”





服务器端显示peking和yangyang加入聊天室

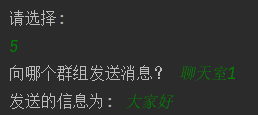
若群组不存在，则会提示群组不存在：

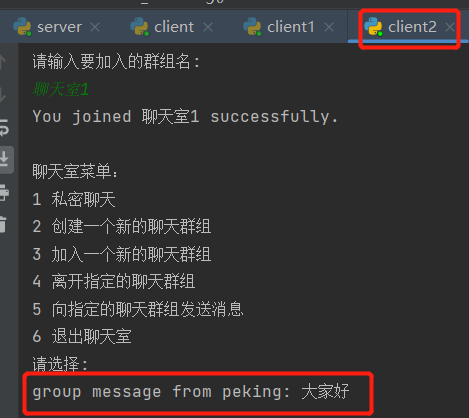
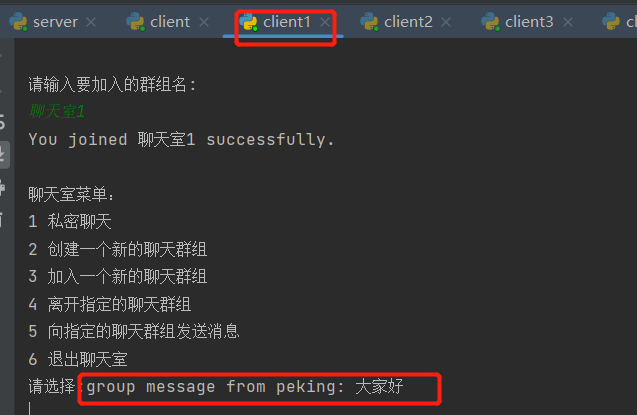


### 在群组内发消息：

peking在“聊天室1”中发送消息，yangyang和luka收到群消息

在群组中发送消息的时候，先选择5，然后输入群组名称和发送的消息



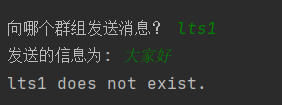


yangyang和luka收到了群组消息



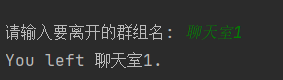
服务器端显示发送群组消息

若输入的群组名不存在，则会提示群组不存在：



### 离开群组：

用户选择4离开指定的群组，然后输入群组名称，之后会有提示成功离开：

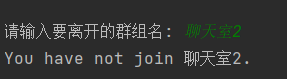


Peking离开“聊天室1”

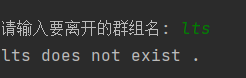


服务器端显示用户的离开

若未加入该群组，则会提示没有加入，就无法离开



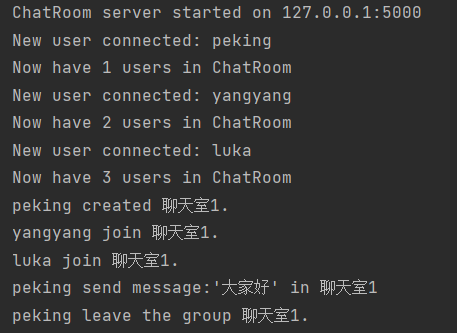
若输入群组名称错误，则输入报错信息群组不存在：



## 用户之间的聊天信息通过服务器端转发

实现这个功能是依靠服务器端线程函数，该线程不断监听各个服务器端发生的时间，负责处理各个客户端发生的事件，用于给所有在线用户发送信息。

在服务器端中会显示一些处理成功的事件并将发送的消息显示在服务器中，具体的内容见上面各个功能实现中所展示的。



部分运行截图