

Gamelobby 游戏大厅(服务器自学作业)

作业内容

使用python 2.7，制作一个游戏大厅，内容要求如下：

1. 客户端可以使用账户名、密码登录进入游戏大厅
2. 可以注册新用户
3. 支持聊天，例如输入\$chat haha，则大厅里其他人可以看到haha
4. 账户有在线时长的属性，需要存盘，下线再上不会丢失数据
5. 进阶需求
 1. 有创建房间，进入房间，退出房间功能
 2. 支持多频道聊天，例如大厅、房间、私聊
 3. 小游戏：21点
 1. 每个房间，每逢半点(8电板，9点，9点半等)，会随机生成4个1~10内的数字，发布在房间内，所有人可以看到
 2. 玩家可以用+,-,*,/和(), 让4个数的计算结果尽量接近于21，但不能超过21
 3. 玩家可用聊天的方式，向服务器提供回答，例如：\$21game (1+2)*(3+5)。只接受任意一个玩家第一次的回答，且此回答其他人看不到
 4. 服务端发布问题之后，15秒内，如果有玩家的回答，刚好等于21，那么此玩家获胜，否则，计时结束时，计算结果最大的玩家获胜，如果有相同的答案，取第一个回答者

具体实现

项目简介

- 项目使用python 2.7进行开发，实现了基于terminal的游戏大厅系统，支持登陆、注册、创建房间、多频道聊天、在线时间存储、21点游戏、查询信息等多种功能
- 开发环境：
 - Mac OS X El Captain 10.11.6
 - Python 2.7.10

项目设计

- 用户信息存储：本项目使用json结构对用户信息进行存储，其结构为

```
username : (password, online_time)
```

其存储在文件users_data.db中，每次用户下线或关闭客户端时，都会调用 `save_user_data` 将在线时间进行更新并保存，以实现存盘功能

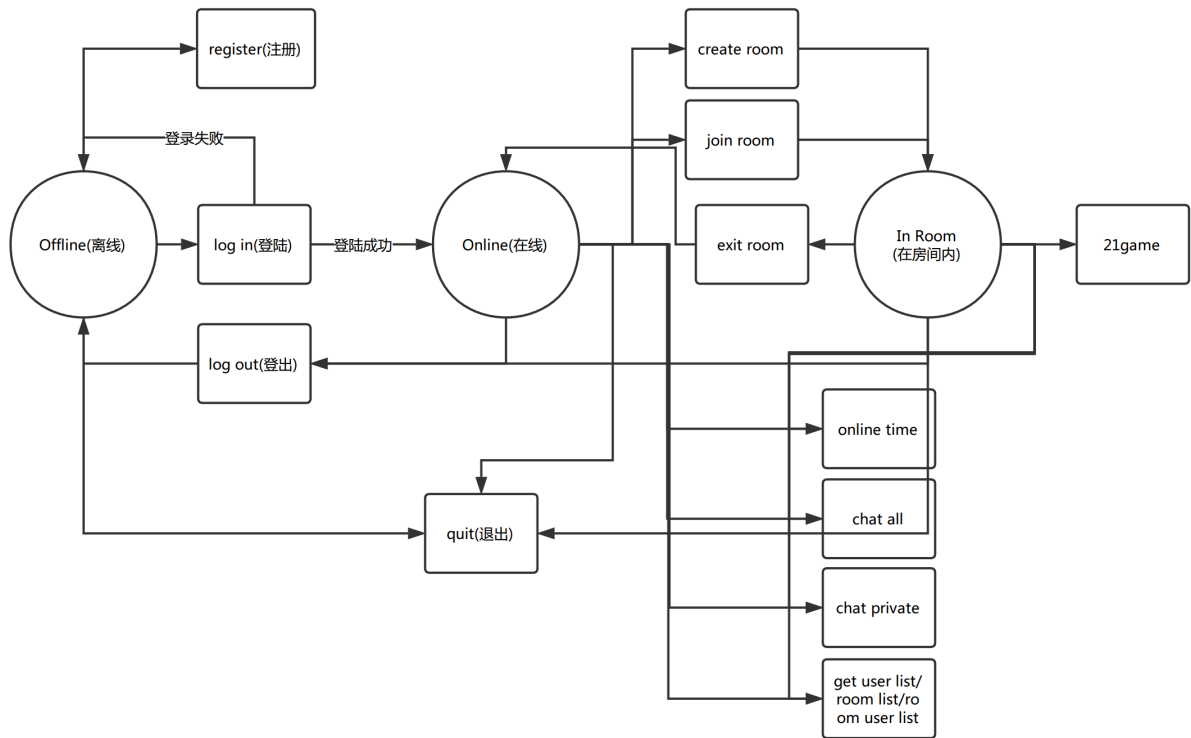
- 信息包结构：服务器和用户之间的信息交互，同样使用json结构对信息进行封装，即每一个信息由 `header` 和 `content` 组成，其中 `header` 用于标识信息的功能，`content` 为信息的内容
- 客户端设计：客户端设计非常简单，因为只是在terminal端实现，并没有增加GUI的内容，对于每个客户端，只需使用select函数同时监听std input和客户端socket（目前收到的客户端信息，其 `header` 仅有 `msg` 情况，因此简单的 `print` 即可实现对收到信息的处理）
- 服务器设计：服务器设计相对复杂一些，在每次服务器运行时，动态维持一个Server_data的全局变量，其中保存如下信息：

```
# Server Data: Store all info and data when server is running
class ServerCache:
    # 服务器句柄
    server = None
    # 用户名 密码 在线时长信息库
    users_db = {} # username : (password, onlinetime)
    # 已登录用户信息
    login_dict = {} # username : socket
    # 已连接（活跃）客户端信息
    act_client_dict = {} # socket : active_client
    # 房间信息
    room_dict = {} # roomname : [user1, user2, user3]
    # 21点游戏信息
    game_start = False
    game_nums = []
    answers_dict = {} # username : (answer, answer_time)
```

其中，“已连接客户端信息”较为复杂：对于每一个连接的客户端，需要先将该客户端添加到监听列表中，同时，生成一个 `ActiveClient` 实例，一个用于处理该客户端信息的收发的结构

与此同时，在服务器主线程运行时，还需一个后台线程来按时更新游戏信息

- 21点游戏：正如之前所说，游戏也是服务器端所运行，使用的是服务器端的一个后台线程进行控制，游戏具体信息如下：
 - 每到半点开始游戏（为了方便测试，这里每到整分钟开始游戏，可以在代码中修改）
 - 游戏持续15秒（游戏时长可以在utilities.py中修改）
 - 所有已登录用户，均可以收到游戏信息的广播，但只有进入房间的人可以参与游戏，当以正确的格式并使用正确的数字输入之后，客户端可以得到计算结果及答题时间
 - 游戏结束时，系统会通知所有房间内的用户游戏结果
- 客户端流程图：



使用说明

- 服务器：在terminal输入 `python server.py`，即可运行，出现如下状态：

```
server — python server.py — 80x24
[wangxiaoming:server User$ python server.py
Server start at: 127.0.0.1:8888
Waiting for connections...
Client (127.0.0.1:51626) connected
```

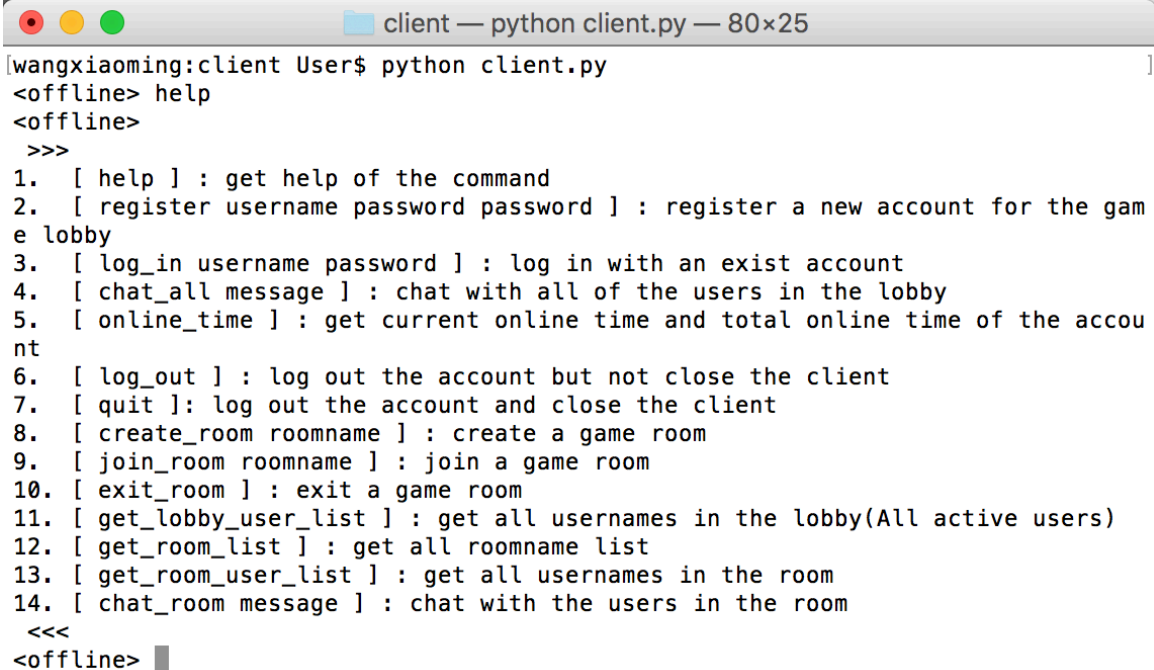
- 客户端：在terminal输入 `python client.py`

接下来具体指令如下

```
"1. [ help ] : get help of the command \n" \
"2. [ register username password password ] : register a new account for the
game lobby \n" \
"3. [ log_in username password ] : log in with an exist account \n" \
"4. [ chat_all message ] : chat with all of the users in the lobby\n" \
"5. [ online_time ] : get current online time and total online time of the
account\n" \
"6. [ log_out ] : log out the account but not close the client\n" \
"7. [ quit ]: log out the account and close the client \n" \
"8. [ create_room roomname ] : create a game room \n" \
"9. [ join_room roomname ] : join a game room \n" \
"10. [ exit_room ] : exit a game room \n" \
"11. [ get_lobby_user_list ] : get all usernames in the lobby(All active users)
\n" \
"12. [ get_room_list ] : get all roomname list \n" \
"13. [ get_room_user_list ] : get all usernames in the room \n" \
"14. [ chat_room message ] : chat with the users in the room \n" \
"15. [ 21game expression ] : play 21 game\n"
```

- 运行样例:

帮助菜单



```
wangxiaoming:client User$ python client.py
<offline> help
<offline>
>>>
1. [ help ] : get help of the command
2. [ register username password password ] : register a new account for the gam
e lobby
3. [ log_in username password ] : log in with an exist account
4. [ chat_all message ] : chat with all of the users in the lobby
5. [ online_time ] : get current online time and total online time of the accou
nt
6. [ log_out ] : log out the account but not close the client
7. [ quit ]: log out the account and close the client
8. [ create_room roomname ] : create a game room
9. [ join_room roomname ] : join a game room
10. [ exit_room ] : exit a game room
11. [ get_lobby_user_list ] : get all usernames in the lobby(All active users)
12. [ get_room_list ] : get all roomname list
13. [ get_room_user_list ] : get all usernames in the room
14. [ chat_room message ] : chat with the users in the room
<<<
<offline> █
```

登陆/注册/登出

```
client — python client.py — 80x28

[wangxiaoming:client User$ python client.py
<offline> chat_all Hello World
<offline>
>>> please log in first! <<<
<offline> log_in netease1 123456
<offline>
>>> account doesn't exist <<<
<offline> register netease1 123456 654321
<offline>
>>> password and repassword are not match! <<<
<offline> register netease1 123456 123456
<offline>
>>> create account successfully! <<<
<offline> log_in netease 12345678
<offline>
>>> account doesn't exist <<<
<offline> log_in netease1 12345678
<offline>
>>> password is incorrect <<<
<offline> log_in netease1 123456
<offline>
>>> login successfully <<<
<netease1>
>>> [netease1] log in <<<
<netease1> log_out
<netease1>
>>> [netease1] log out <<<
<offline> █
```

同时登陆

```
client — -bash — 80x25

[wangxiaoming:client User$ python client.py
<offline> log_in jiaww 123456
<offline>
>>> login successfully <<<
<jiaww>
>>> [jiaww] log in <<<
<jiaww>
>>> you kick by (127.0.0.1, 51767) <<<
<jiaww> disconnected
wangxiaoming:client User$ █
```

```
client — python client.py — 80x25

[wangxiaoming:client User$ python client.py
<offline> log_in jiaww 123456
<offline>
>>> login successfully <<<
<jiaww>
>>> [jiaww] log in <<<
<jiaww> █
```

多频道聊天/创建、进入房间

```
client — python client.py — 80x25
<netease1>
>>> [netease1] log in <<<
<netease1>
>>> [netease2] log in <<<
<netease1>
>>> [netease3] log in <<<
<netease1> create_room chatting
<netease1>
>>> Your room 'chatting' is created successfully <<<
<netease1>
>>> Room 'chatting' has been created <<<
<netease1>
>>> [netease1] join the room <<<
<netease1>
>>> [netease2] join the room <<<
<netease1> chat_all Everybody Saw
<netease1>
>>> (Lobby Message)[netease1]: Everybody Saw <<<
<netease1> chat_room Room Saw
<netease1>
>>> (Room Message)[netease1]: Room Saw <<<
<netease1> chat_private netease3 Private Saw
<netease1>
>>> (Private Message)[netease1]: Private Saw <<<
<netease1> []

client — python client.py — 80x25
wangxiaoming:client User$ python client.py
<offline> log_in netease3 123456
<offline>
>>> login successfully <<<
<netease3>
>>> [netease3] log in <<<
<netease3>
>>> Room 'chatting' has been created <<<
<netease3>
>>> (Lobby Message)[netease1]: Everybody Saw <<<
<netease3>
>>> (Private Message)[netease1]: Private Saw <<<
<netease3> []

server — python server.py — 80x25
wangxiaoming:server User$ python server.py
Server start at: 127.0.0.1:8888
Waiting for connections...
Client (127.0.0.1:51798) connected
Client (127.0.0.1:51800) connected
Client (127.0.0.1:51801) connected
Client (127.0.0.1:51798) disconnected
Client (127.0.0.1:51803) connected
Client (127.0.0.1:51800) disconnected
Client (127.0.0.1:51804) connected
[]
```

21点游戏

```
client — python client.py — 80x25
<netease1> create_room play
<netease1>
>>> Your room 'play' is created successfully <<<
<netease1>
>>> Room 'play' has been created <<<
<netease1>
>>> [netease1] join the room <<<
<netease1>
>>> [netease2] join the room <<<
<netease1>
>>> [netease3] join the room <<<
<netease1>
>>> No winner this time, good luck next round <<<
>>> 21 Game Start!: the 4 numbers are 2 1 5 3
Join the room and Hurry up !! <<<
<netease1> 21game 2+5+5*3
<netease1>
>>> please use the 4 given numbers: 1 2 3 5 <<<
<netease1> 21game 2+1+5*3
<netease1>
>>> Good for you! you got 18, your answer time is 23 <<<
<netease1>
>>> The winner is netease1, the answer equals: 18 <<<
<netease1> []

client — python client.py — 80x25
wangxiaoming:client User$ python client.py
<offline> log_in netease3 123456
<offline>
>>> login successfully <<<
<netease3>
>>> 21 Game Start!: the 4 numbers are 4 8 1 10
Join the room and Hurry up !! <<<
<netease3>
>>> [netease3] log in <<<
<netease3>
>>> Room 'play' has been created <<<
<netease3> join_room play
<netease3>
>>> [netease3] join the room <<<
<netease3>
>>> No winner this time, good luck next round <<<
>>> 21 Game Start!: the 4 numbers are 2 1 5 3
Join the room and Hurry up !! <<<
<netease3> 21game 2+1+5*3
<netease3>
>>> Good for you! you got 11, your answer time is 37 <<<
<netease3>
>>> The winner is netease1, the answer equals: 18 <<<
<netease3> []

client — python client.py — 80x25
<netease2>
>>> [netease2] log in <<<
<netease2>
>>> 21 Game Start!: the 4 numbers are 4 8 1 10
Join the room and Hurry up !! <<<
<netease2>
>>> [netease3] log in <<<
<netease2>
>>> Room 'play' has been created <<<
<netease2> join_room play
<netease2>
>>> [netease2] join the room <<<
<netease2>
>>> [netease3] join the room <<<
<netease2>
>>> No winner this time, good luck next round <<<
>>> 21 Game Start!: the 4 numbers are 2 1 5 3
Join the room and Hurry up !! <<<
<netease2> 21game 2+1+5*3
<netease2>
>>> Good for you! you got 18, your answer time is 31 <<<
<netease2>
>>> The winner is netease1, the answer equals: 18 <<<
<netease2> []

server — python server.py — 80x25
wangxiaoming:server User$ python server.py
Server start at: 127.0.0.1:8888
Waiting for connections...
Client (127.0.0.1:52621) connected
Client (127.0.0.1:52622) connected
Client (127.0.0.1:52623) connected
21 Game Start!: the 4 numbers are 4 8 1 10
No winner this time
21 Game Start!: the 4 numbers are 2 1 5 3
netease1 18 23
netease2 18 31
netease3 11 37
The winner is netease1, the answer equals: 18
[]
```

