

1.运行环境

Server 端运行环境: Linux (Centos7 已测试)

Client 端运行环境: Window(Win10 已测试)/Linux(Centos7 已测试)

2.Server 启动

找到程序 server.py 在 server 目录下。

```
server:
__init__.py  game      server.py      sqlite_db      test
context      reactor  server_logger.py stacktracer.py thread_pool
control      server.log server_logger.pyc stacktracer.pyc trace.html
```

打开并修改 server.py 中的 host 和 port, 并运行程序。

```
host = '10.53.228.191'
port = 2003

class Server(object):
    def __init__(self, host, port):
```

3. Client 启动

找到程序 client.py 在目录 client 下。

```
client:
__init__.py action.py action.pyc client.log client.py client_logger.py
```

打开并修改 Client.py 中的 host 和 port, 并运行程序。

```
17
18 host = '10.53.228.191'
19 port = 2003
20
21 class Server(object):
22
```

4. Client 操作

```
*****
*                               *
*               Start Menu      *
*                               *
* register user_id passwords :   register user account      *
* login user_id passwords:       login server with account info *
* logout:                         user logout                  *
*                               *
*****
Login And Register Menu.
```

在上图注册于登录界面进行账户注册和登录。

1. 注册命令: register user password

2. 登录命令: login user password

登录成功后, 进入聊天界面:

```

*****
*                                     Chat Dialog                                     *
*                                                                              *
* chat message:                      send message to cur room *
* chat [user] message:               send message to user   *
* room create room_id:               create chat room       *
* room invite user1 user2:           invite users to room   *
* room enter room_id:               enter to room           *
* room leave:                       leave cur room          *
*                                                                              *
*****

input you chat command with the instruction above.

```

1. chat message: 向当前频道发送消息。
2. chat [user id] message: 向某个用户发送私信。
3. room create room_id: 创建名字为 room id 的房间，创建成功后，roomid 默认为用户名#room_id。
4. room enter room_id: 用户创建成功 room 后，需要进入 room，room id 可以是用户创建时使用的 id。
5. room invite user: 用户在房间后，通过 invite 使得其他的用户获得进入房间的权限。
6. room leave: 用户退出当前的房间。
7. 21game 1+2+3+4: 整点时间和半点时间，回答 21 点游戏的命令。

备注: 用户注册账号成功后，还需要进行登录; 用户创建 room 后，需要 room enter 后是在处在 room 中的状态; 在大厅中，room leave 无效; server 端可以检测到客户端的 socket 状态，可以接 ctrl c 结束 client 进程。