

Socket通信编程

程序功能

客户端

- 指定服务器的ip地址和端口号
- 实现TCP和UDP形式发送请求
- 使用用户名和密码登录账户
- 实现echo功能，发送的消息能被服务端返回

服务端

- 指定服务器监听的端口
- 实现同时以TCP和UDP形式监听请求，能同时与多个客户端进行通信
- 实现广播功能，能将客户端发送的消息广播到所有在线客户端
- 显示与客户端的通信情况
- 查询在线用户
- 使某个用户下线

编译方式

下载项目，导入idea，使用Build - Build Project

生成的jar包在 ./out/artifacts 下

开发环境

```
openjdk 17.0.5 2022-10-18 LTS
```

```
OpenJDK Runtime Environment Microsoft-6841604 (build 17.0.5+8-LTS)
```

```
OpenJDK 64-Bit Server VM Microsoft-6841604 (build 17.0.5+8-LTS, mixed mode)
```

使用方式

客户端

启动命令

```
java -jar client.jar <protocol> [-h host] [-p port]
```

参数介绍

protocol: 通信协议，必选， TCP或UDP

host: 服务器ip地址， 默认127.0.0.1

port: 服务器端口号， 默认8080

使用方法

登录： login

注册： register

发送消息： echo

退出： exit

服务端

启动命令

```
java -jar server.jar [-p port]
```

参数介绍

port: 服务器监听的端口号， 默认8080

使用方法

查看在线用户： online

踢用户下线： kick

退出： exit

项目结构

```
socket
├─ README.md -- README文件
├─ client -- 客户端源码
│   └─ src
│       ├── Main.java -- 程序入口
│       ├── Client.java
│       ├── UDPClient.java
│       ├── TCPClient.java
│       ├── DefaultClientProcessor.java
│       └─ ClientProcessor.java
├─ server -- 服务端源码
│   └─ src
│       ├── Main.java -- 程序入口
│       ├── Server.java
│       ├── AbstractServer.java
│       ├── UDPServer.java
│       ├── TCPServer.java
│       └─ DefaultServerProcessor.java
```

```
|      └─ ServerProcessor.java
└─ out  -- 编译输出
```

核心代码

客户端

```
/**
 * 客户端通信接口
 * 负责连接服务器，发送消息，接收消息
 * 分别实现TCP和UDP两种通信方式
 */
public interface Client;

/**
 * 客户端消息处理器接口
 * 定义了客户端消息处理器的基本功能
 * 通过加锁来保证先收到指令的结果后再发送指令
 */
public interface ClientProcessor;
```

服务端

```
/**
 * 服务端接口
 * 负责启动服务端，停止服务端，广播消息
 * TCP实现通过ServerSocket处理客户端连接，同时通过线程池处理多个客户端连接
 * UDP实现通过DatagramSocket处理客户端连接
 */
public interface Server;

/**
 * 服务端处理器
 * 负责处理客户端请求
 * 服务端处理器应该是线程安全的
 * 通过使用线程安全的集合类如ConcurrentHashMap来存储数据来保证线程安全
 */
public interface ServerProcessor;
```