

JAVA 12. 금요Project



| Background

- ✓ Thread & Network
- ✓ 프로젝트 분석, 설계, 구현

| Goal

- ✓ 네트워크 프로젝트를 분석하고, 설계하여 구현할 수 있다.

| Problem : 채팅 Project

1. 간단한 채팅 프로그램을 구축하여 보자.

서버는 서버 소켓을 열어 클라이언트의 접속을 기다리고, 클라이언트는 서버를 접속하여, 메시지를 보내면 서버는 접속된 모든 클라이언트에게 메시지를 전달하는 채팅 프로그램을 설계하고, 작성하여 보자.

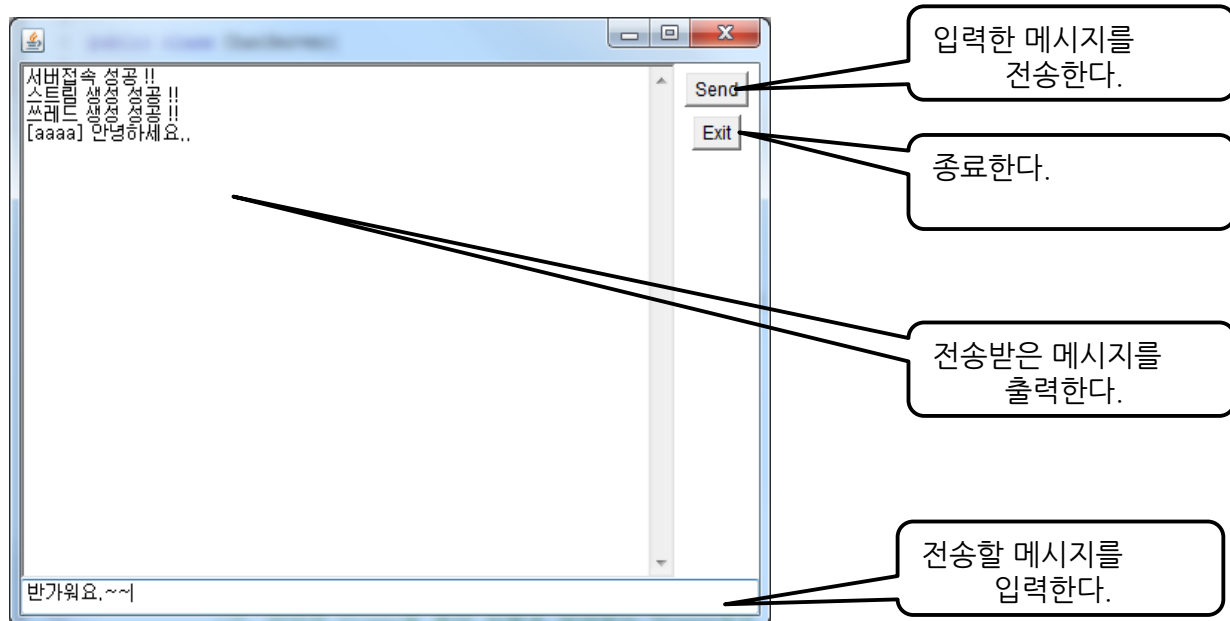
❖ 구현해야 할 내용

Server	Client
<ul style="list-style-type: none"> ➤ 서버는 서버소켓을 열어 클라이언트의 접속을 기다린다. ➤ 접속된 클라이언트와 소켓, 입출력 스트림을 생성한다. ➤ 사용자 정보를 저장하고, 접속된 클라이언트를 처리할 Thread를 생성한다. ➤ 클라이언트로부터 전달된 메시지를 접속된 모든 클라이언트에게 메시지를 보낸다. 	<ul style="list-style-type: none"> ➤ UI를 생성한다. ➤ ip, port 을 가지고 서버를 접속한다. ➤ 서버와의 입출력 스트림을 생성한다. ➤ 서버로부터 메시지를 처리할 Thread를 생성한다. ➤ 채팅을 수행한다.

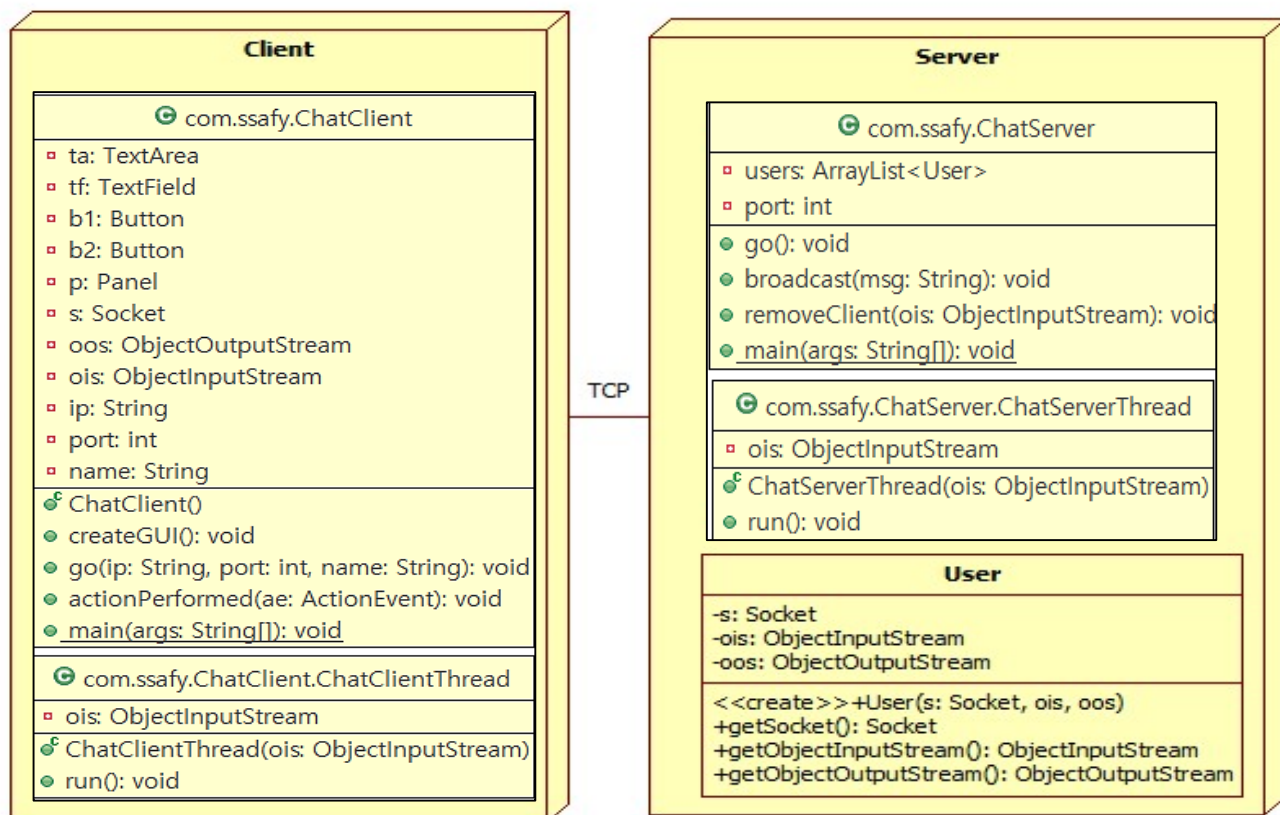
JAVA 12. 금요Project



➤ 클라이언트 화면 구성



➤ Class Diagram



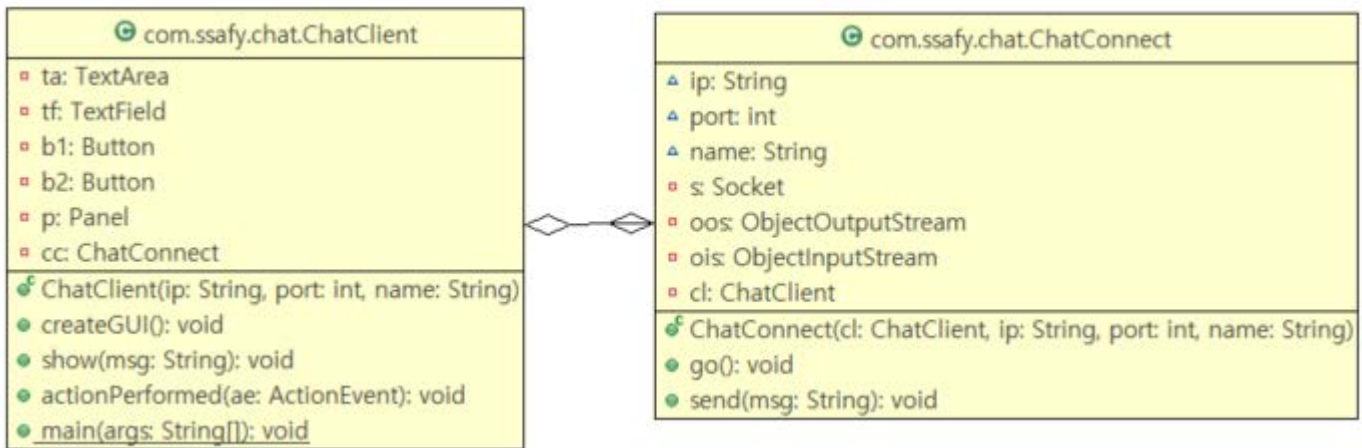
JAVA 12. 금요Project



- 클라이언트 프로그램의 ChatClient는 GUI와 통신 코드가 한꺼번에 구현되어 Cohesion을 떨어뜨려 좋은 설계라 할 수 없다.

GUI 기능과 통신 기능을 나누어 Refactoring 하여 보자.

- Client Class Diagram



- [선택] 로그인 기능을 넣어서 대화명을 입력 받아 보자
- [선택] 그룹을 생성할 수 있도록 하여 그룹별 채팅이 가능하도록 구현하여 보자

❖ 제출방법

- 프로젝트명 : "wsjava12_지역_반_성명" 으로 작성
- 완성 후 프로젝트를 wsjava12_지역_반_성명.zip으로 압축하여 제출