



스토리

통신 장비와 기본적인 연결은 성공했다. 하지만 단순히 내 메시지를 돌려 주기만 하는 에코 기능만 있을 뿐이다. 간단한 채팅이라도 하려고 보니 역시 많은 요소들이 필요했다. 채팅 기능의 맞은 편에 사람이 있었으면 좋겠지만 그래도 하나의 원가 화성 밖으로 전파가 나갈 수 있다는 희망이 있으니 그래도 계속 기능을 만들어 보아야겠다.

수행과제

- TCP/IP 소켓(Socket) 통신을 위한 서버쪽 코드를 작성한다.
- 여러 명과 동시에 통신을 하기 위해서 멀티 쓰레드를 이용한 소켓 통신을 작성해야 한다.
- 소켓에 접속하면 ‘~~님이 입장하셨습니다.’와 같은 메시지를 전체 접속자에게 전송해야 한다.
- 클라이언트에서 ‘/종료’라고 입력하면 연결이 끊어져야 합니다.
- 클라이언트에서 메시지가 입력하면 접속된 모든 클라이언트와 메시지가 공유 되어야 합니다.
- 출력되는 메시지에는 ‘사용자> 메시지’와 같이 누가 입력한 메시지인지 구별이 가능해야 합니다.
- 클라이언트를 3개 동시에 실행해서 채팅이 원활하게 가능한지 확인해 본다.

보너스 과제

- 특정 사람에게만 메시지를 보내기 위한 소위 말하는 귓속말 기능을 구현해 본다.

기본 평가 운영 원칙

시간	학생	학번 % 2 == 0	학번 % 2 == 1
19:00-19:30		평가자-참여신청	피평가자-평가요청
19:30-20:00		피평가자-평가요청	평가자-참여신청
20:00-20:30		평가-피평가 자율	
20:30-		우수 수행 과제 공개 발표	

★ 평가시간 30분 '가득' 채우면, 실력도 쑥쑥 !!!

★ 반드시 실행 결과 확인

평가 시 유의 사항 (Check list)

» 배경 지식 넓히기(TCP/IP)

- TCP 서버/클라이언트 절차 이해
 - ※ (server) socket()-bind()-listen()-accept()-sendall()/recv()-close()
 - ※ (client) socket()-connect()-sendall()/recv()-close()
- sendall()과 send() 차이 이해하고 설명하기
- TCP/UDP 통신 차이 이해하고 설명하기
- netstat 명령 이해하고 서버/클라이언트 동작 설명

» 배경 지식 넓히기(multi-thread)

- thread를 사용해야 하는 이유
- thread 정의 및 활용 방법
- thread.daemon = True 설명하기





