

## 컴퓨터 네트워크 과제1 – 디자인 및 구현 문서

작성자: 2021038131 장준혁

### 1. 핵심 클래스 설명

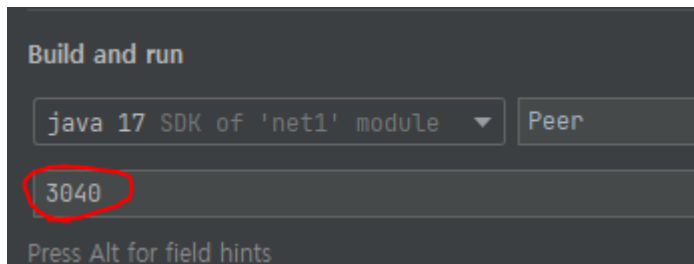
핵심이 되는 클래스는 ChatRoom 클래스이다. ChatRoom의 필드는 다음과 같다. 방의 이름, 멀티캐스트 주소, 포트번호, 멀티캐스트 소켓이다. 사용자가 #JOIN 명령어를 입력하면 ChatRoom 인스턴스가 생성되고 인스턴스가 생성될 때, ChatRoom의 이름을 키 값으로 SHA-256 암호화 방식을 통해 멀티캐스트 주소를 구한다. SHA-256 암호화를 거치면 32바이트의 문자열이 나오는데, 여기서 앞 3바이트에 각각 128 씩 더해서 225.x.y.z의 x, y, z 값으로 사용했다. getMulticastAddress는 멀티캐스트 주소를 구하는 static 함수이다. 그 다음에는 멀티캐스트 주소를 바탕으로 멀티캐스트 소켓을 생성한다.

그 다음으로는 Sender와 Receiver라는 클래스가 있다. 이 둘을 ChatRoom의 메소드로 구현하지 않고 분리한 가장 큰 이유는 이 둘을 IO작업을 기다려야 해야 하기 때문에 메인 스레드의 실행을 Blocking 하지 않도록 서브 스레드에서 실행하기 위해 별도의 클래스로 분리하였다. 이 둘은 ChatRoom에 있는 동일한 멀티캐스트 소켓을 사용한다. Sender는 사용자에게서 채팅을 입력한 후 채팅 내용의 앞에 사용자 이름을 추가하고 512 byte 단위로 잘라서 멀티캐스팅 방식으로 전송한다. 그리고 #EXIT을 입력 받으면 종료한다. Receiver에서는 멀티캐스트 채널에 있는 데이터를 받아서 사용자의 채팅방에 출력해준다. 그런데 교재에서는 Buffer 데이터그램의 값을 가져온 후 바로 출력하는데 이 경우 Buffer의 뒤쪽 빈 공간들이 모두 출력되는 문제가 생긴다. 따라서 버퍼에 입력 받은 후 패킷의 길이를 바탕으로 새로운 배열에 필요한 만큼의 길이만 다시 복사해서 출력한다. Main 함수의 앞부분은 포트, 방, 유저이름을 입력하고 이를 바탕으로 채팅방에 입력하는 부분이다. Main 문의 마지막에 sender.join() 함수를 넣은 이유는 Sender가 #EXIT을 입력 받고 스레드가 종료되었을 때, 메인 문도 종료되게 하기 위함이다.

### 2. 실행방법 및 실행 결과

젯브레인사의 인텔리제라는 IDE에서 코드를 작성했다. UTF-8로 인코딩되어 있으므로 이클립스처럼 EUC-KR을 기본 인코딩으로 사용하는 IDE에서는 인코딩이 깨질 경우 인코딩 방식을 수정해서 다시 불러와야 한다. 물론 코드에는 한글을 사용하지 않아서 큰 문제는 없을 것으로 예상된다.

Peer.java를 빌드해서 실행한다. (단일 파일로 구성되었으므로 실행 및 컴파일 방법은 어렵지 않아 생각한다.) 실행 시 인자로 포트 번호를 같이 입력해준다.



IDE를 사용할 경우에는 Run 설정에서 다음과 같이 인자 입력 옵션이 있다. 이는 이클립스에도 동일하게 있는 기능이다.

입력해주지 않을 경우에는 기본으로 3030포트가 설정된다. 그 후 #JOIN (room name) (user name)의 형식으로 커맨드를 입력해 채팅방에 입장한다. 약 2, 3개 정도의 프로세스를 만들어서 2, 3 명의 사용자가 채팅방에 입장하도록 한다. 채팅을 쳐보면 아래와 같이 통신이 되는 것을 알 수 있다.

```
port number: 3040
Welcome to Chat App!
command: #JOIN (room name) (user name)
Enter command: #join BattleNet Hong
Room address: /225.97.48.68
Hi!
Hong: Hi!
Lim: Hello!
GoGo
Hong: GoGo
|
```

```
port number: 3040
Welcome to Chat App!
command: #JOIN (room name) (user name)
Enter command: #join BattleNet Lim
Room address: /225.97.48.68
Hong: Hi!
Hello!
Lim: Hello!
Hong: GoGo
|
```

나가고 싶을 경우 #EXIT을 입력하면 아래와 같이 채팅방에서 나가진다.

```
port number: 3040
Welcome to Chat App!
command: #JOIN (room name) (user name)
Enter command: #join BattleNet Lim
Room address: /225.97.48.68
Hong: Hi!
Hello!
Lim: Hello!
Hong: GoGo
#EXIT
end
```