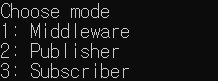
**미들웨어 구현 보고서**

2013011082 이상옥

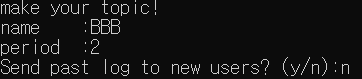
**1. 실행방법**

1.1. visual studio C#이 필요하다. ConsoleApp1.sln을 visual studio로 실행 후 릴리즈 모드로 여러 창을 동시에 run하면 된다.(Ctrl+F5를 여러 번 눌러 다수의 창을 띄울 수 있다)

1.2. 미들웨어, 퍼블리셔, 구독자의 구분 없이 구성했으며, 프로그램 작동 시 최초 선택지에서 어느 역할을 할 지 선택할 수 있다. 따라서 작동시키기 위해서는 최소 3개 이상의 창을 띄우고 실행해야 한다.

1.3. 미들웨어는 1개만 켤 수 있다.

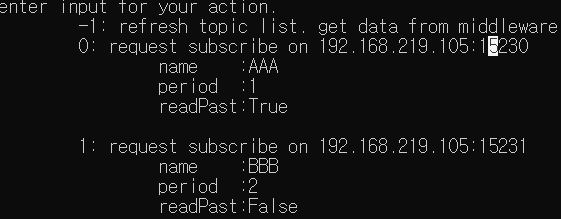
1.4. 미들웨어만 먼저 켜 지면 그 다음은 어떤 순서로 동작하든 상관없다. Subscriber가 먼저 연결되더라도 refresh를 선택해 topic 목록을 갱신할 수 있다.

1.5. publisher로 실행할 경우 입력해야 하는 값은 다음과 같다. 아래 항목을 모두 채워 넣으면 미들웨어에 토픽을 등록한다.

Name : subscriber에게 노출될 이름 입력

Period : integer입력. (현재 아무 기능 없음)

??? : 새로운 유저에게 과거 로그를 보여줄지 말지 여부를 결정.

1.6. subscriber로 실행할 경우 입력가능한 선택지는 다음과 같다. 아래 입력을 통해 subscriber는 여러 topic을 동시에 subscribe할 수 있다. Topic의 새 소식인 megazine을 받고 있는 중에도 아래의 입력은 여전히 가능하다.

[-1] : 미들웨어에 등록된 topic 목록을 refresh

[0 또는 자연수] : 받아온 topic 목록 중 해당되는 번호의 topic을 subscribe한다.

**2. 제한조건과 가정**

2.1. **한번에 하나의 동작만** 일어난다 (멀티 스레드를 사용하지만 lock은 하나도 안함)

2.2. 다수의 Client(pub 또는 sub) 가 접속할 수 있지만, **중간에 강제 종료하지 않는다**. (exception 처리 안 되어 있음)

2.3. **publisher 하나는 한 개의 topic**만 다룰 수 있다. 다수의 topic을 지원하려면 프로그램을 여러 개 켜야 한다.

2.4. publisher는 **영문 string data만**을 보낼 수 있다.

**3. 지원기능**

3.1. publisher들이 선언한 **topic은 middleware에 저장**된다.

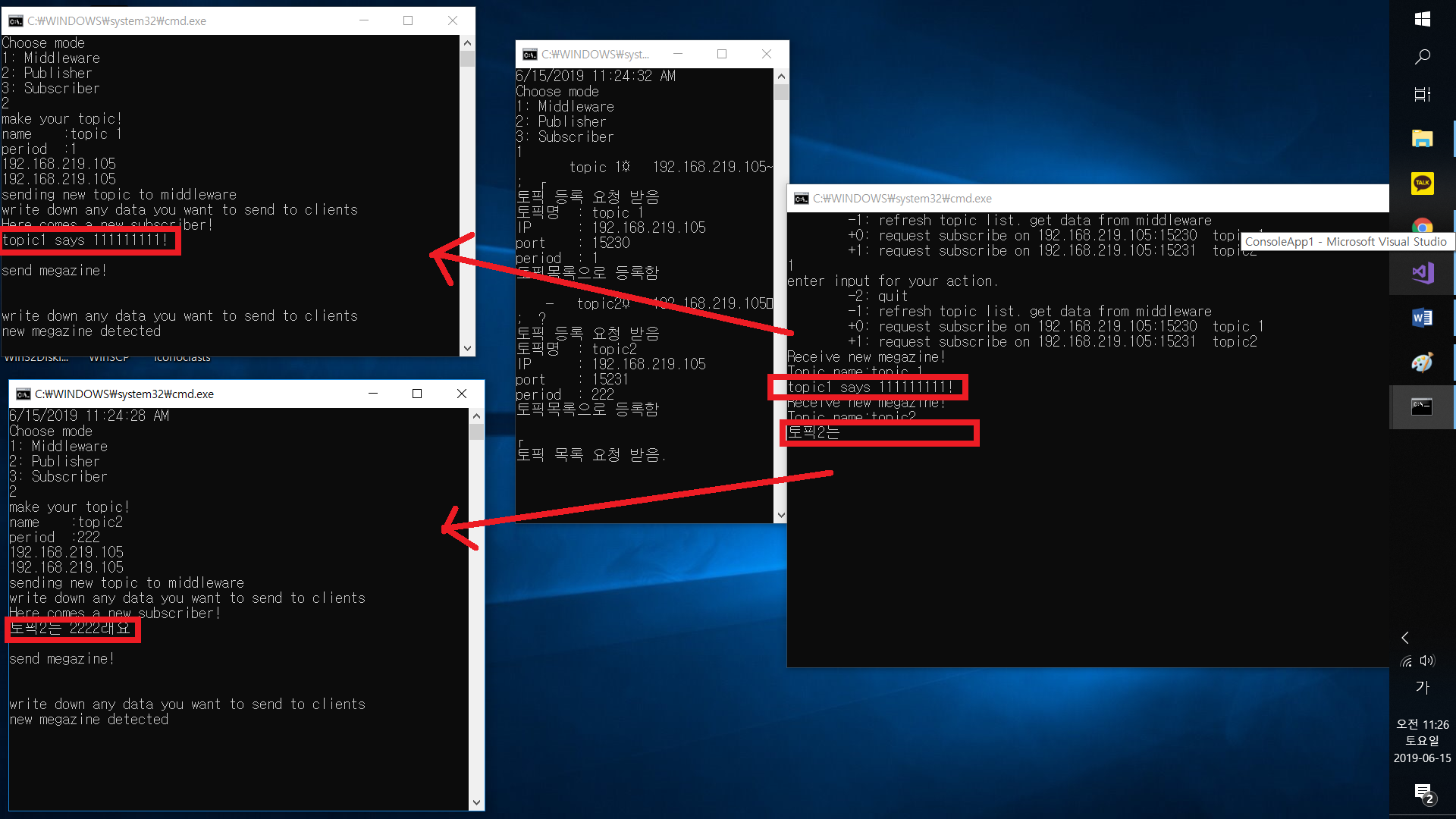
3.2. **동일한 이름의 topic**을 동일한 ip의 publisher들이 선언할 수 있다. (port로 구분)

3.3. **subscriber는 다수의 topic**을 subscribe 할 수 있다.

3.4 topic 최초 설정 시 new publisher에게 **최신정보가 아닌 과거 정보도 제공할지 정할** 수 있다.

**4. 작동 화면**

4.1. 다수의 topic을 subscribe 하는 예시 (한글입력 시 불완전한 전달 문제 발생)



4.2. 자신이 subscribing 하는 topic만 받아보는 동작 예시

