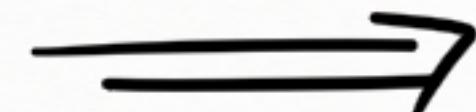
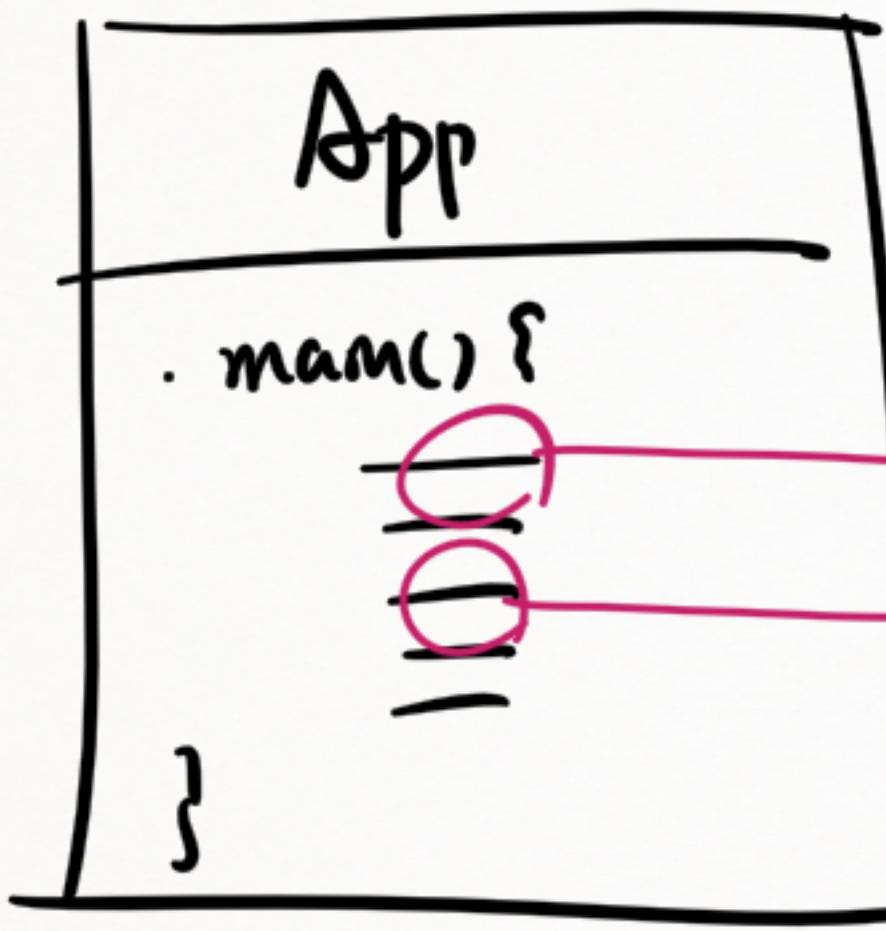
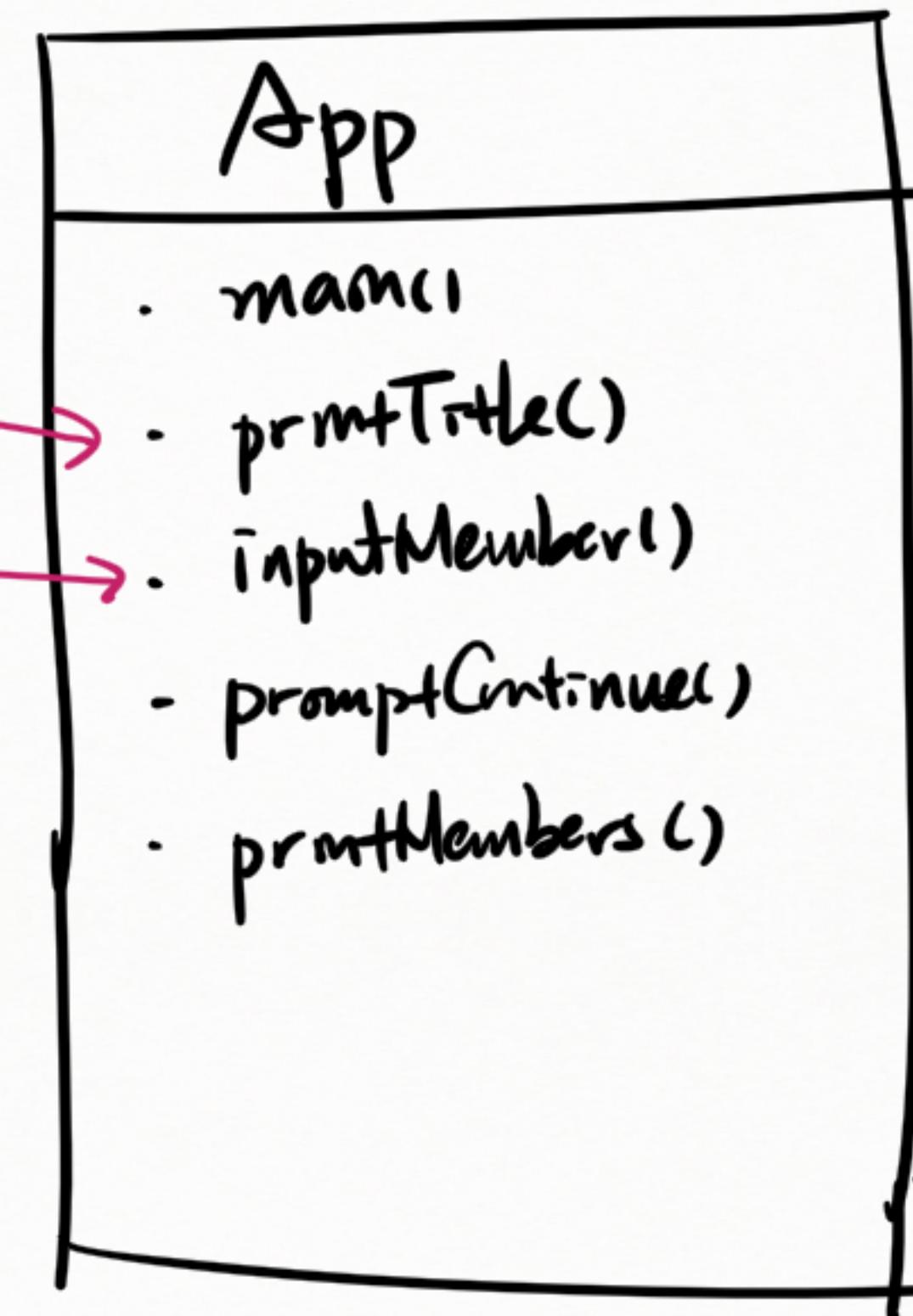


\* 1. 디렉트 사용법

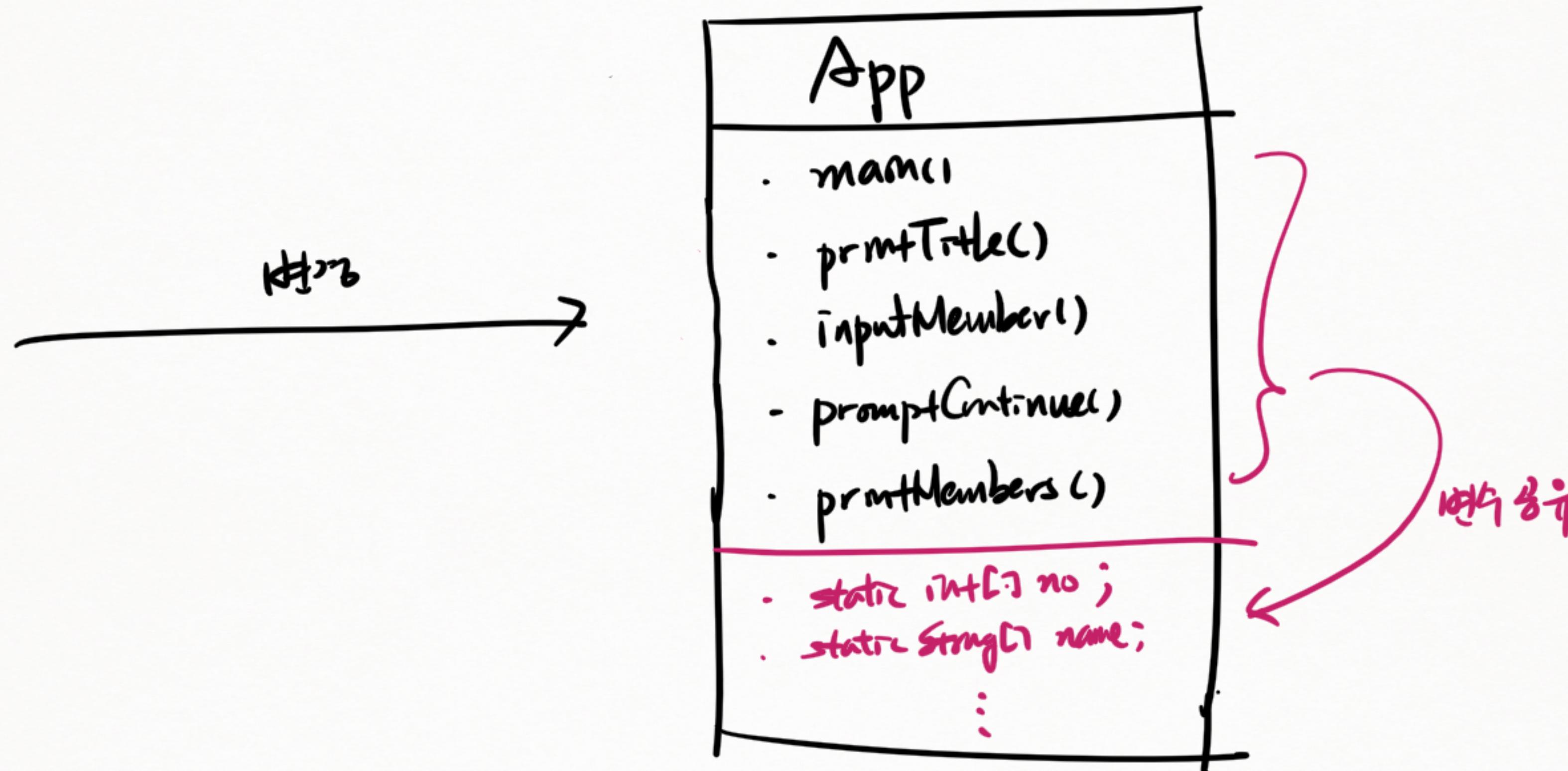
이전



변경

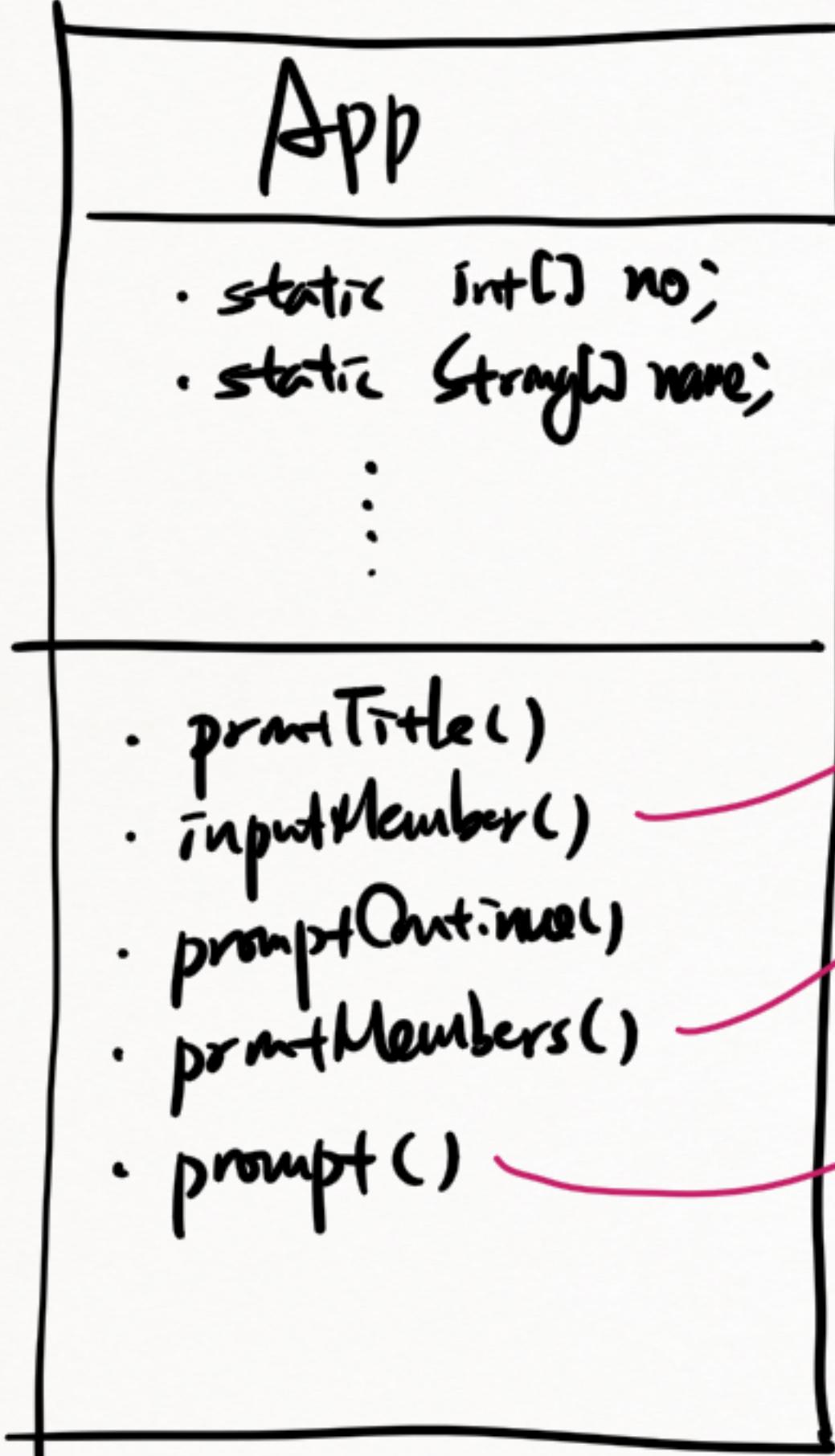


## \* Q. Lession with Array

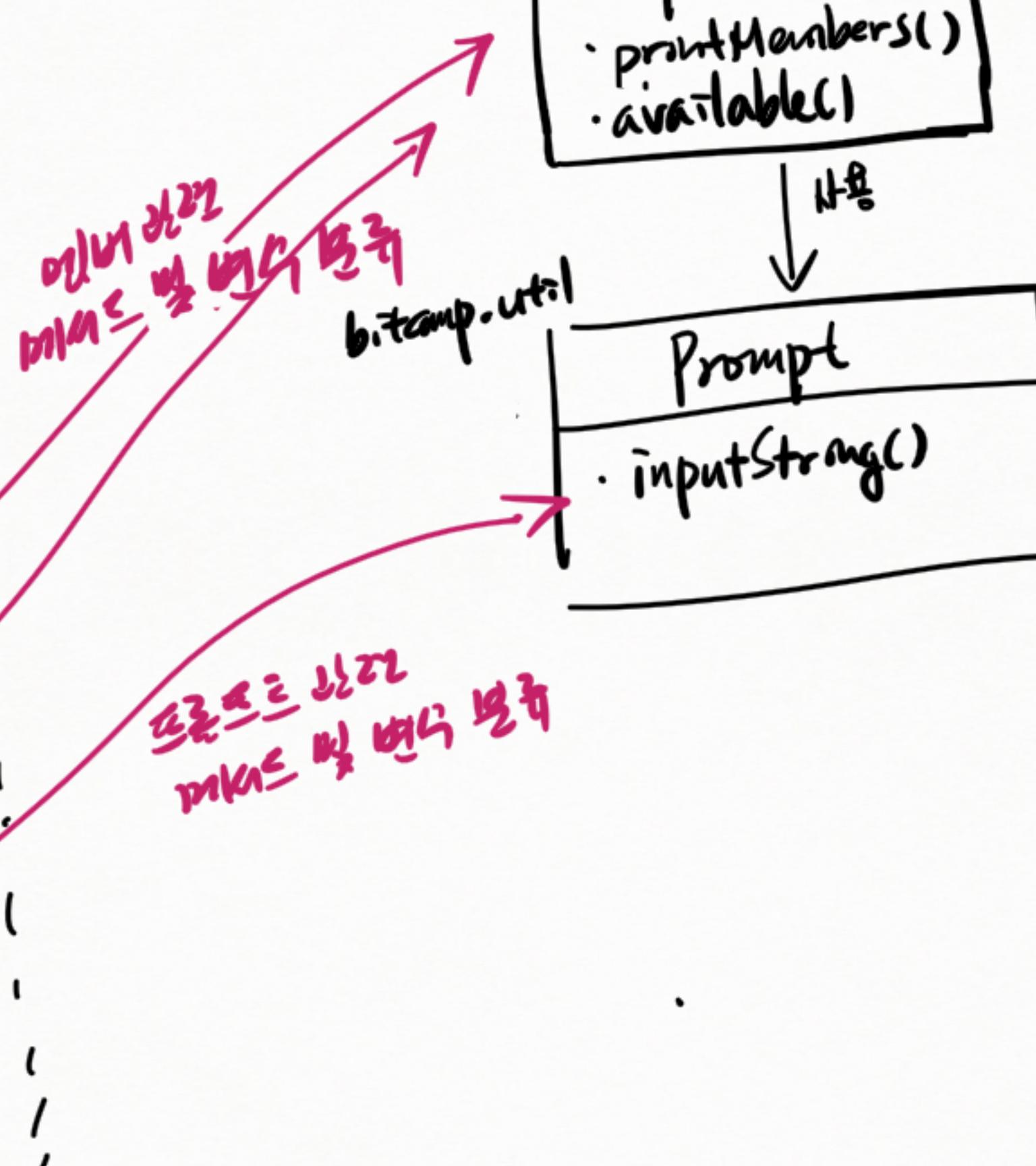
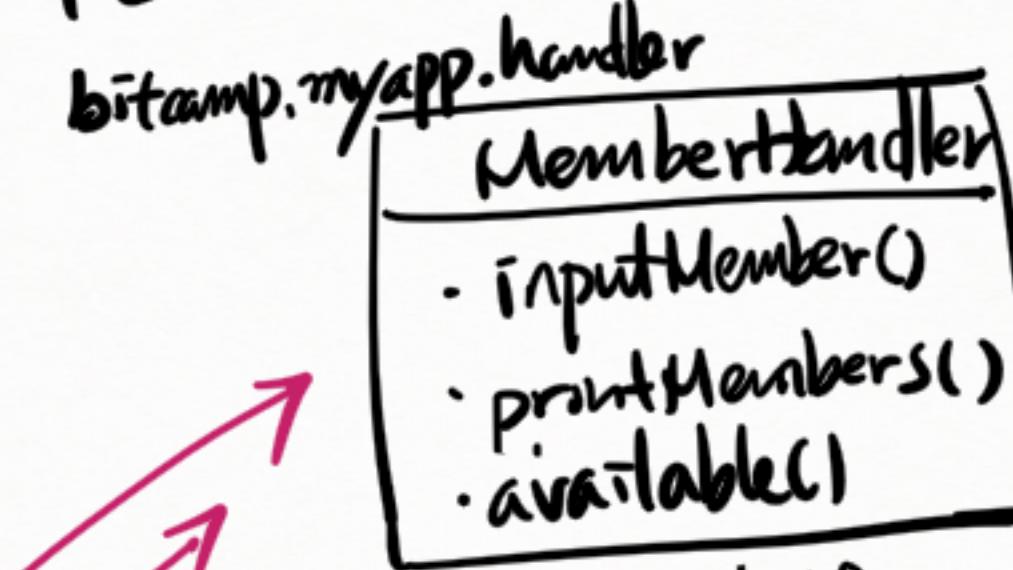


## \* 9. 클래스 및 패턴 학습

이전 구조  
~  
Architecture

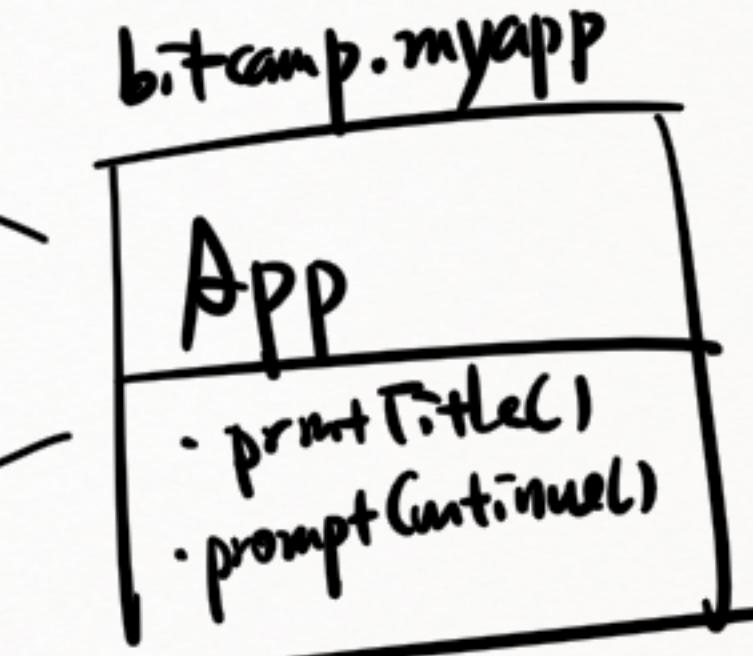


내 구조



MemberHandler를  
인스턴스에  
따라  
분류 →  
이유?  
유지보수를  
쉽게.

(  
다른  
속도를  
얻을  
수 있다.  
메모리  
节约.  
})

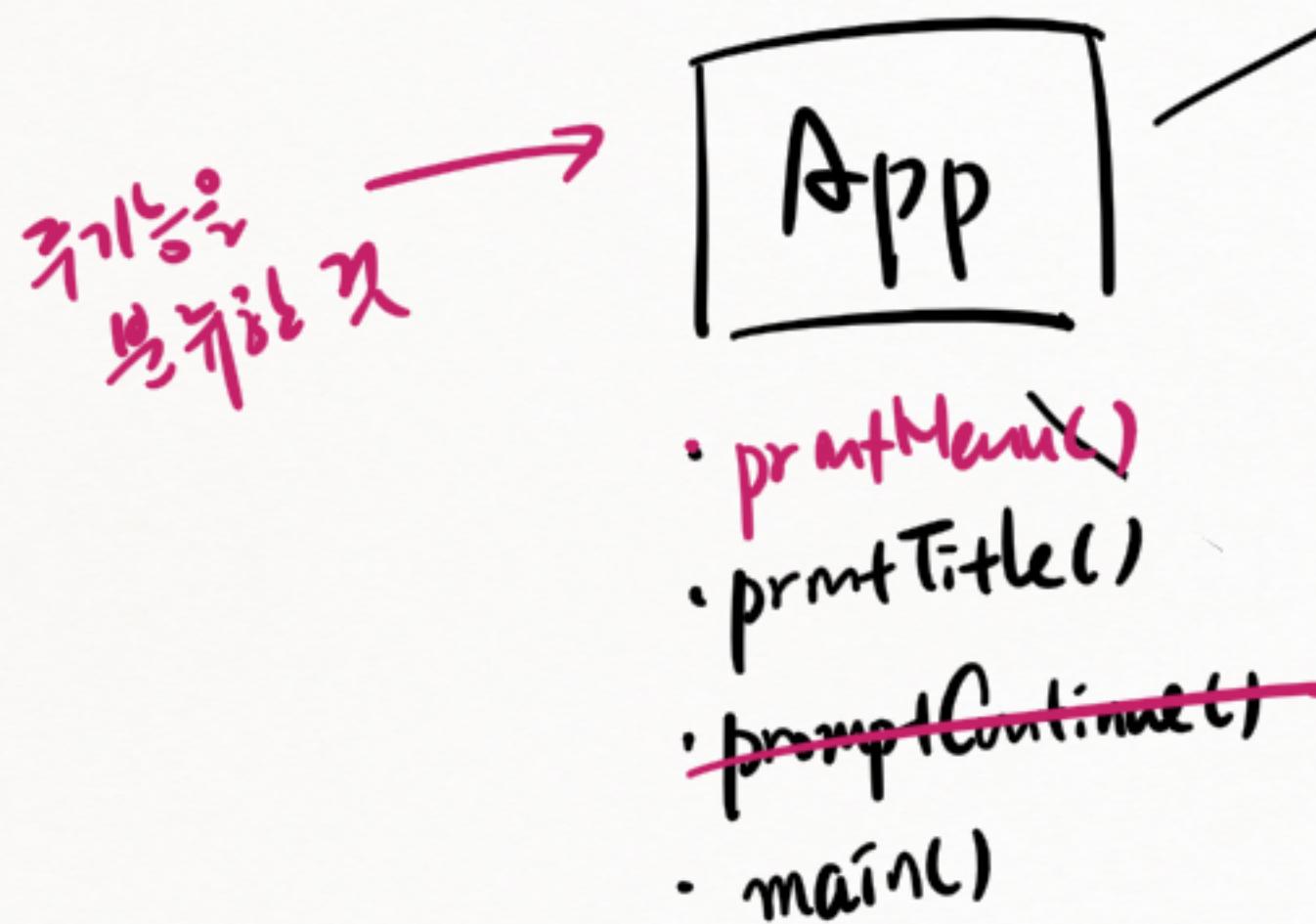


내부  
비용  
절감  
→  
H/W  
환경으로  
제작

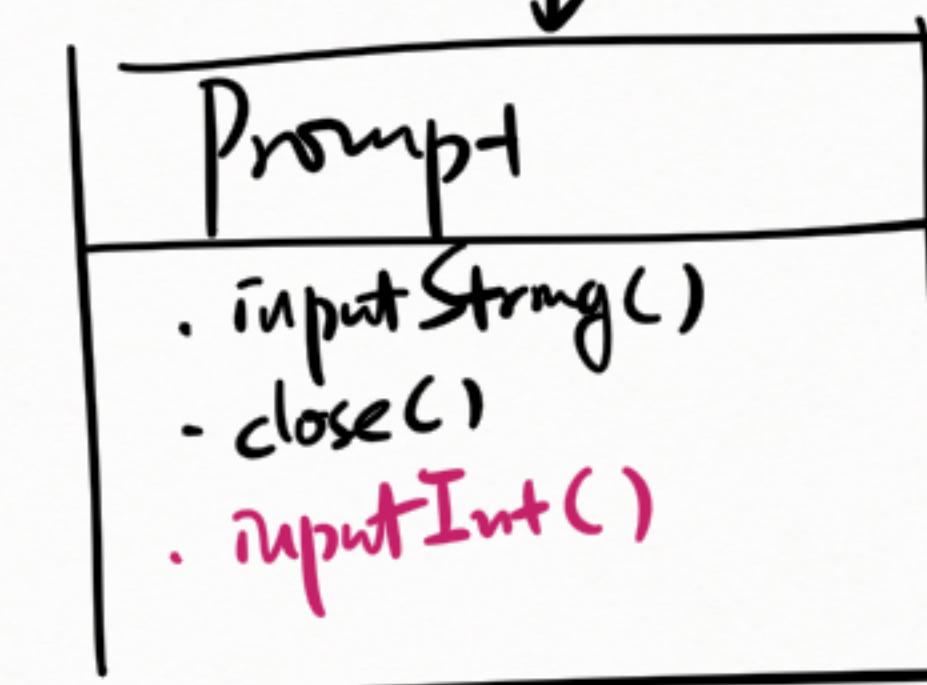
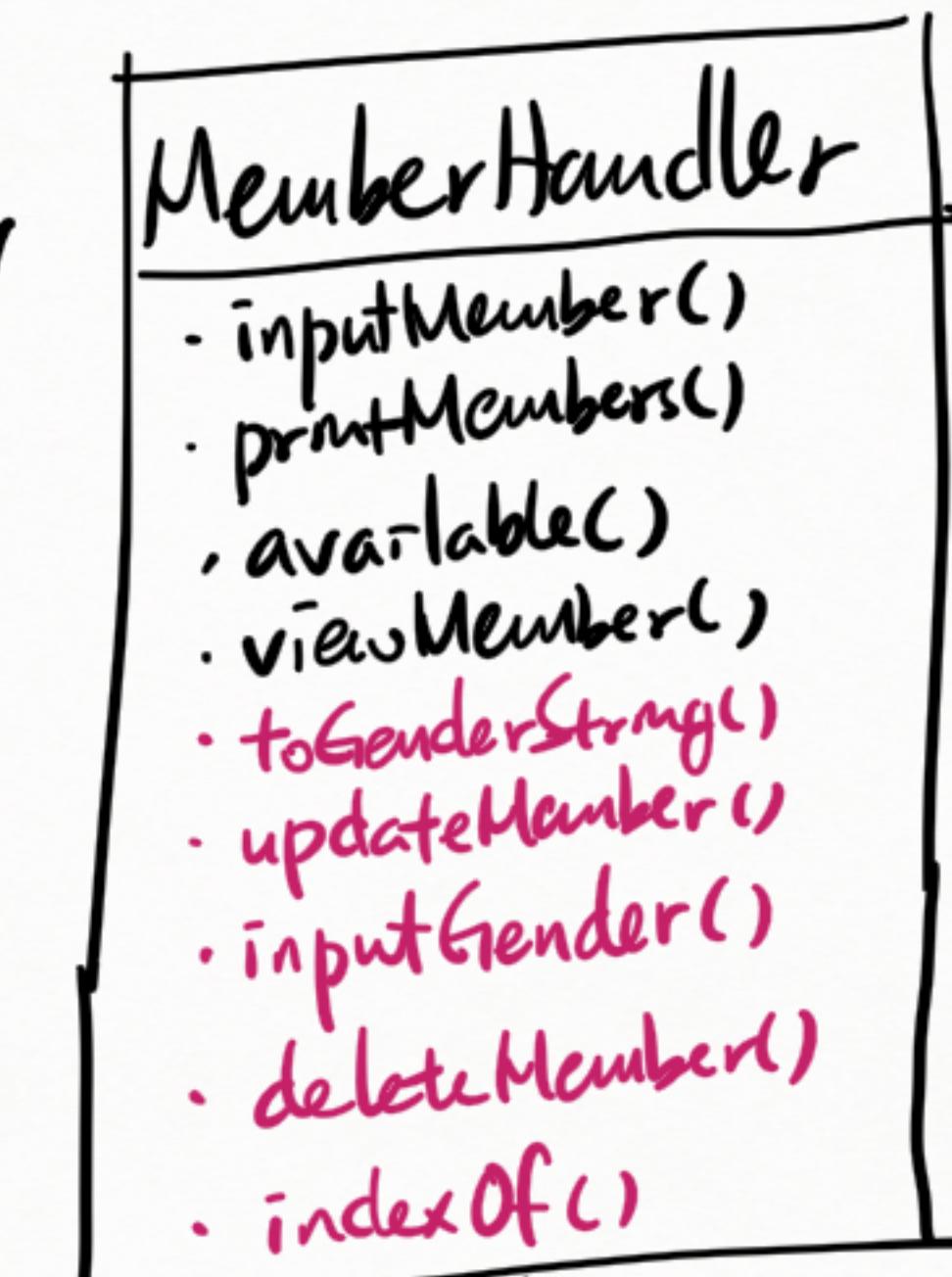
## \* 10. 멤버 및 CRUD 구현

\* 클래스  
↳ 애플리케이션 메서드를 분류할 것

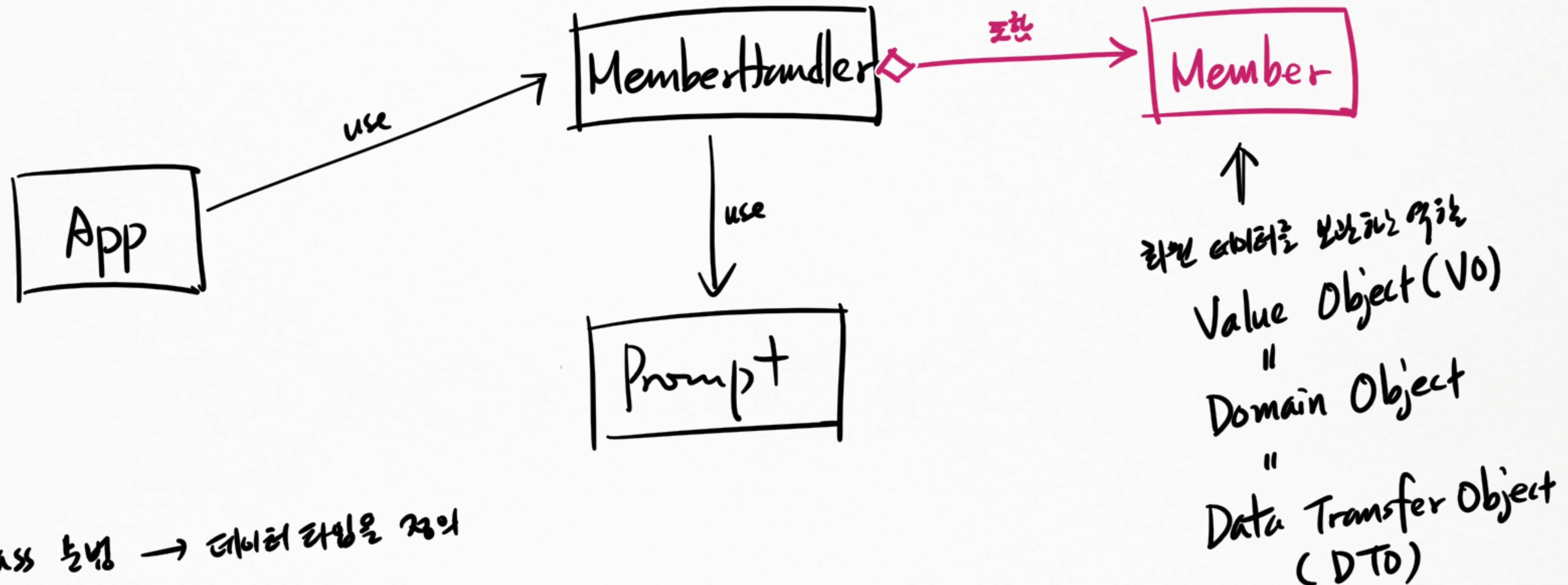
\* 패키지  
↳ 클래스를 분류할 것.



class 분류 → 메서드를 찾는 용도

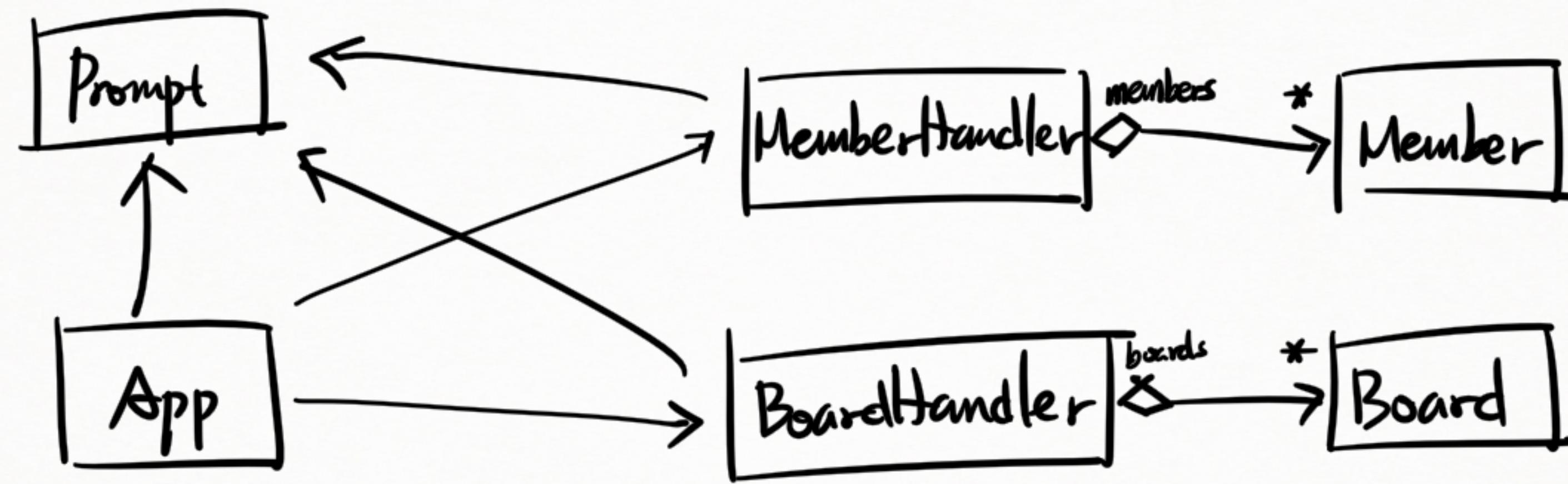


## 11. 사용자 정의 데이터 타입 만들기

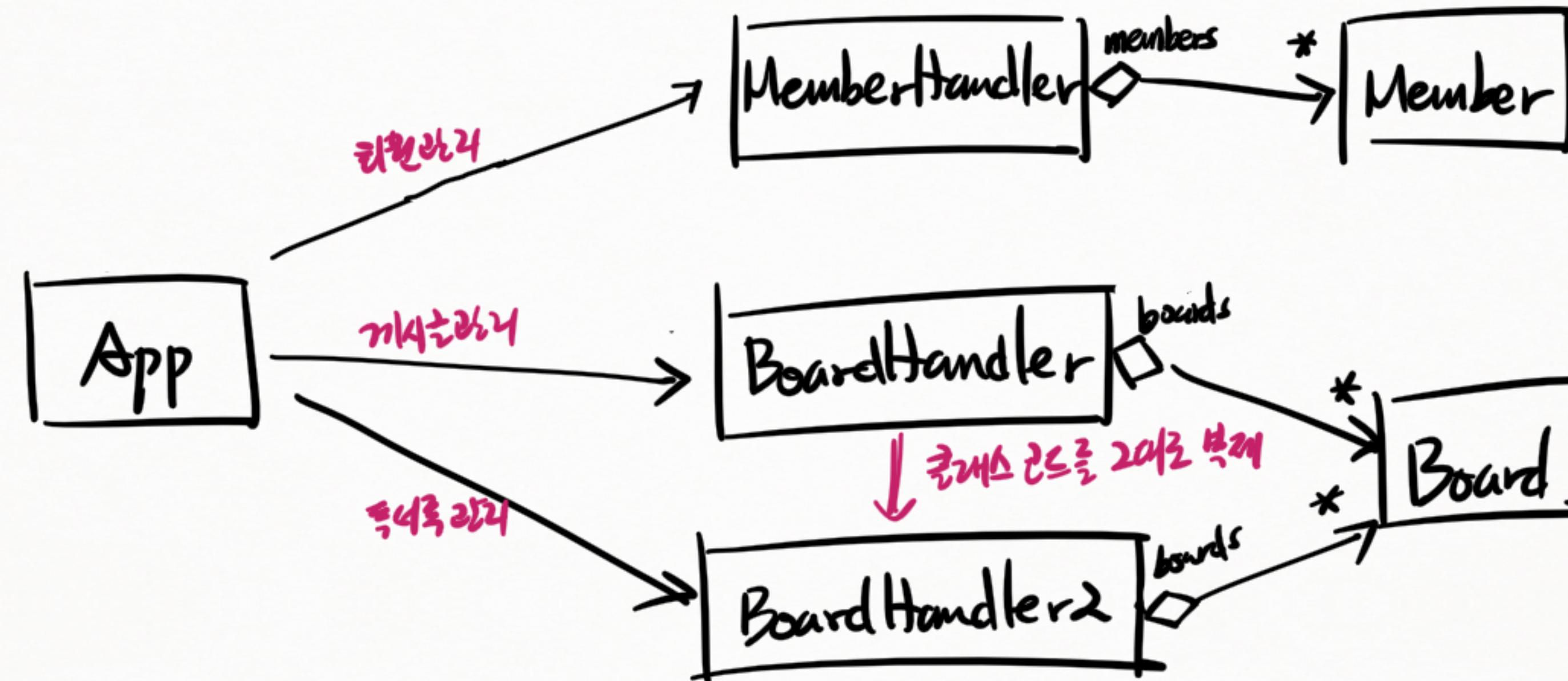


\* class 늘기 → 데이터 타입을 확장

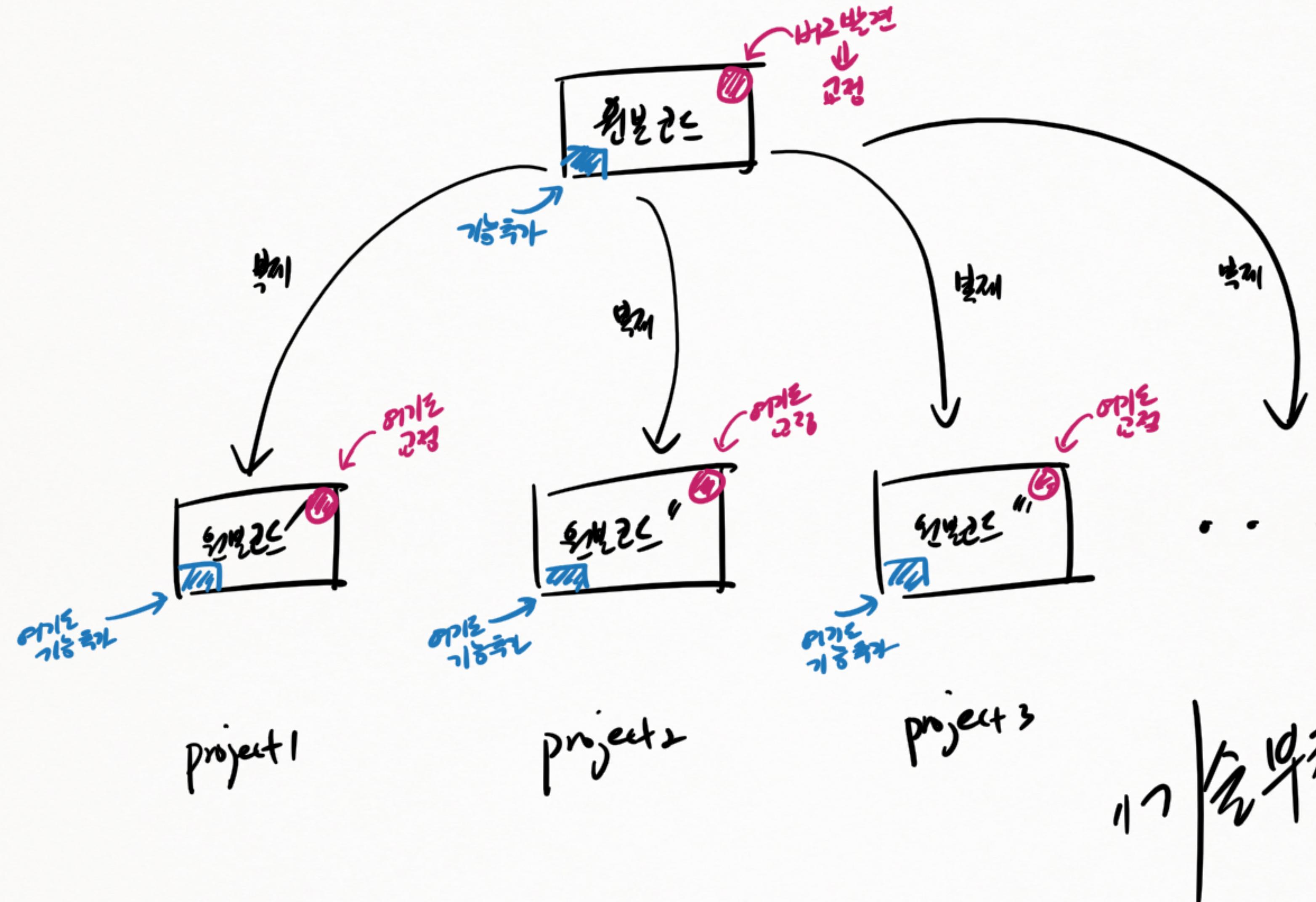
### 13. 깃허브 CRUD 추가



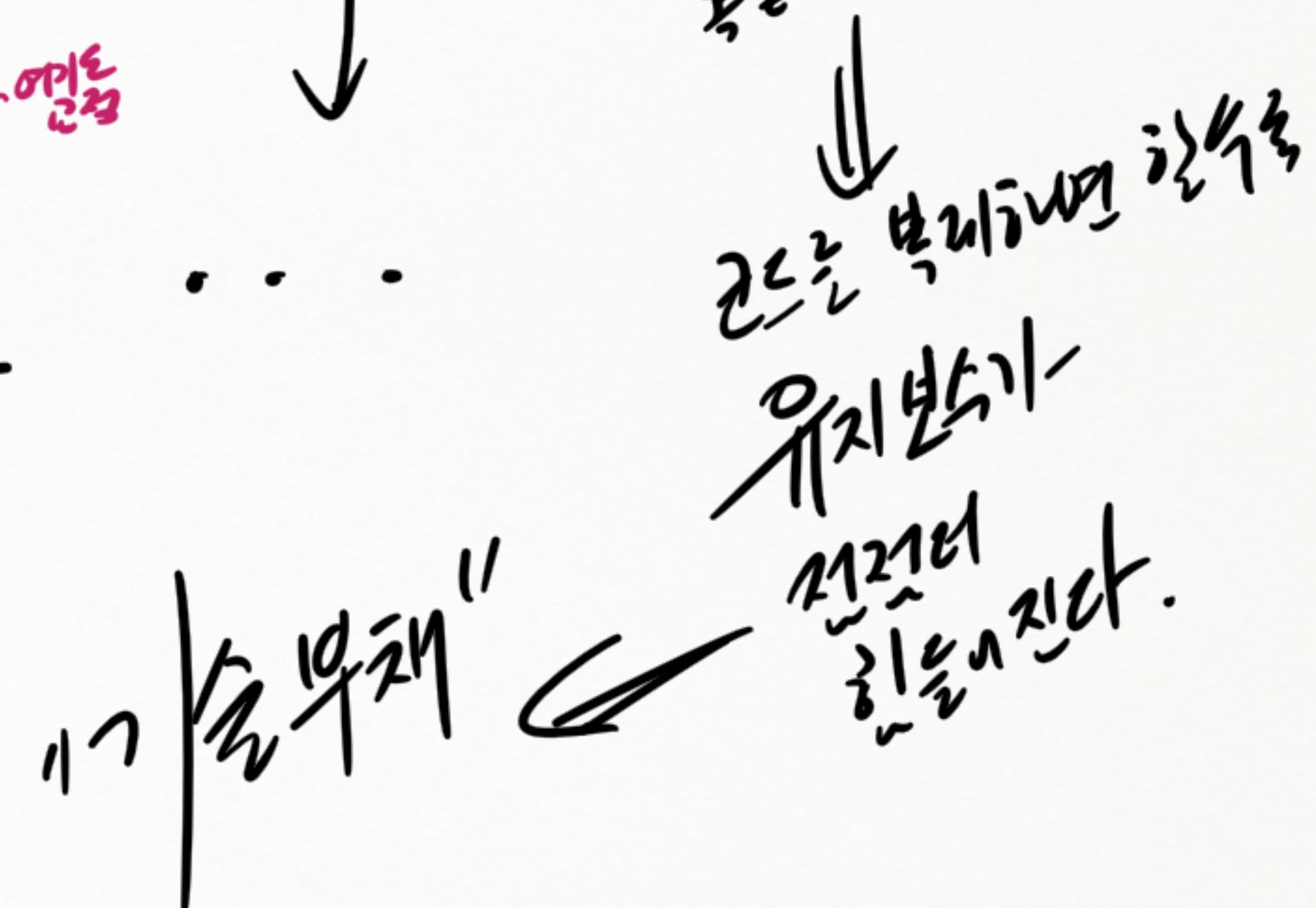
## 14. 토커를 CRUD 추가



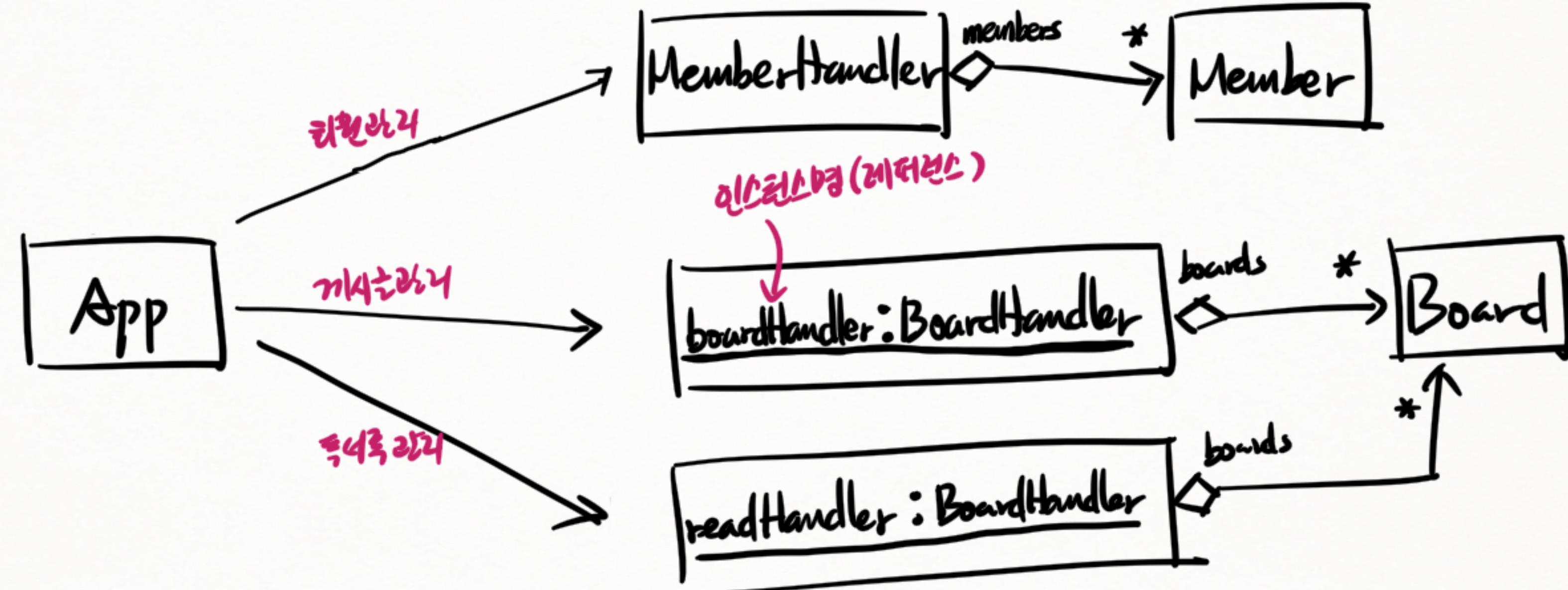
\* 가로관 복제로써 새기능을 주입하는 예



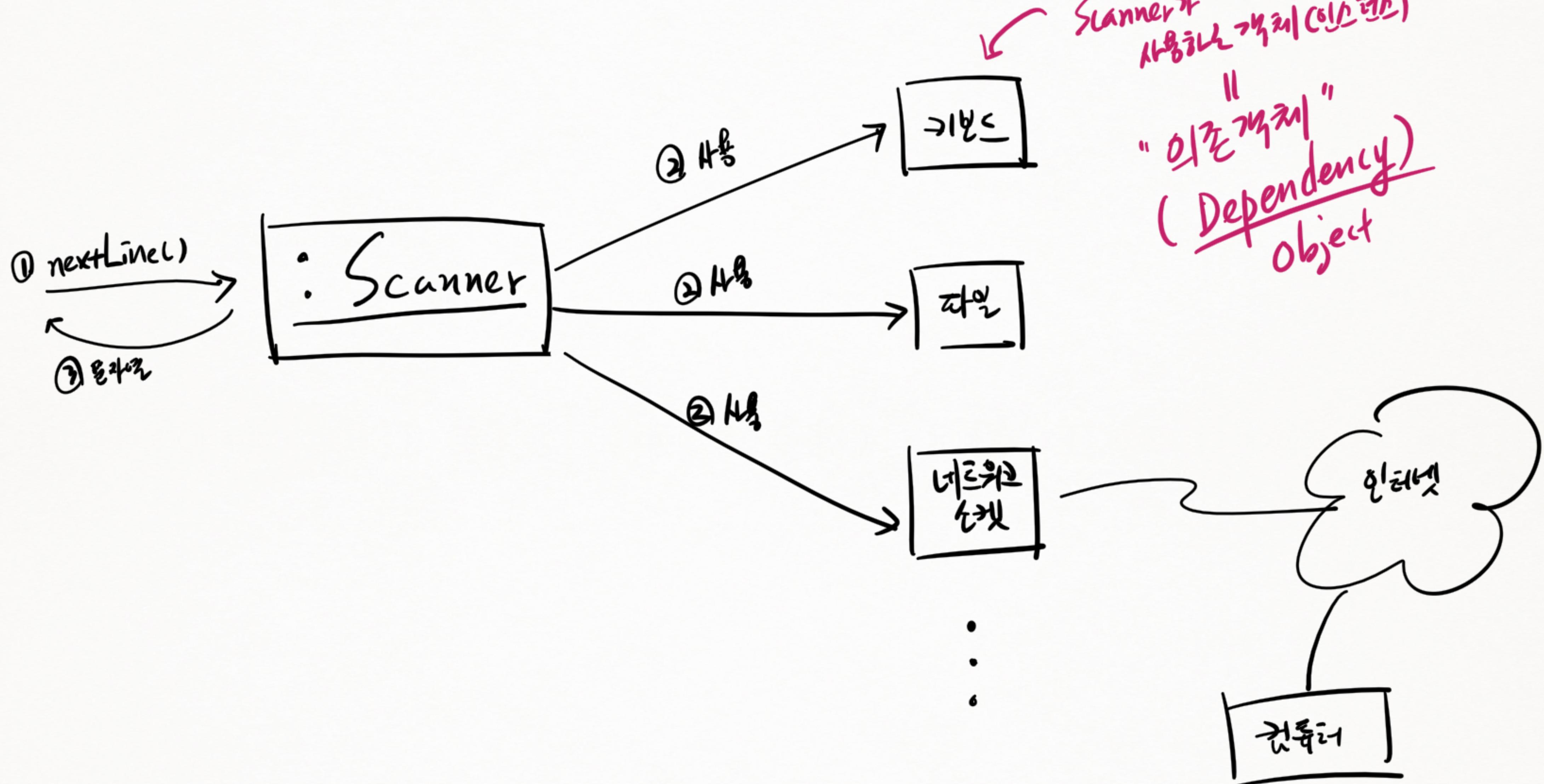
\* API를 고정하여 기능을 추가하여 API를 복제한 코드의 디자인은 동일한 일을 수행하는 코드다



## 15. 인스턴스 있는 B2M에서 학습



## Scanner 와 의존객체



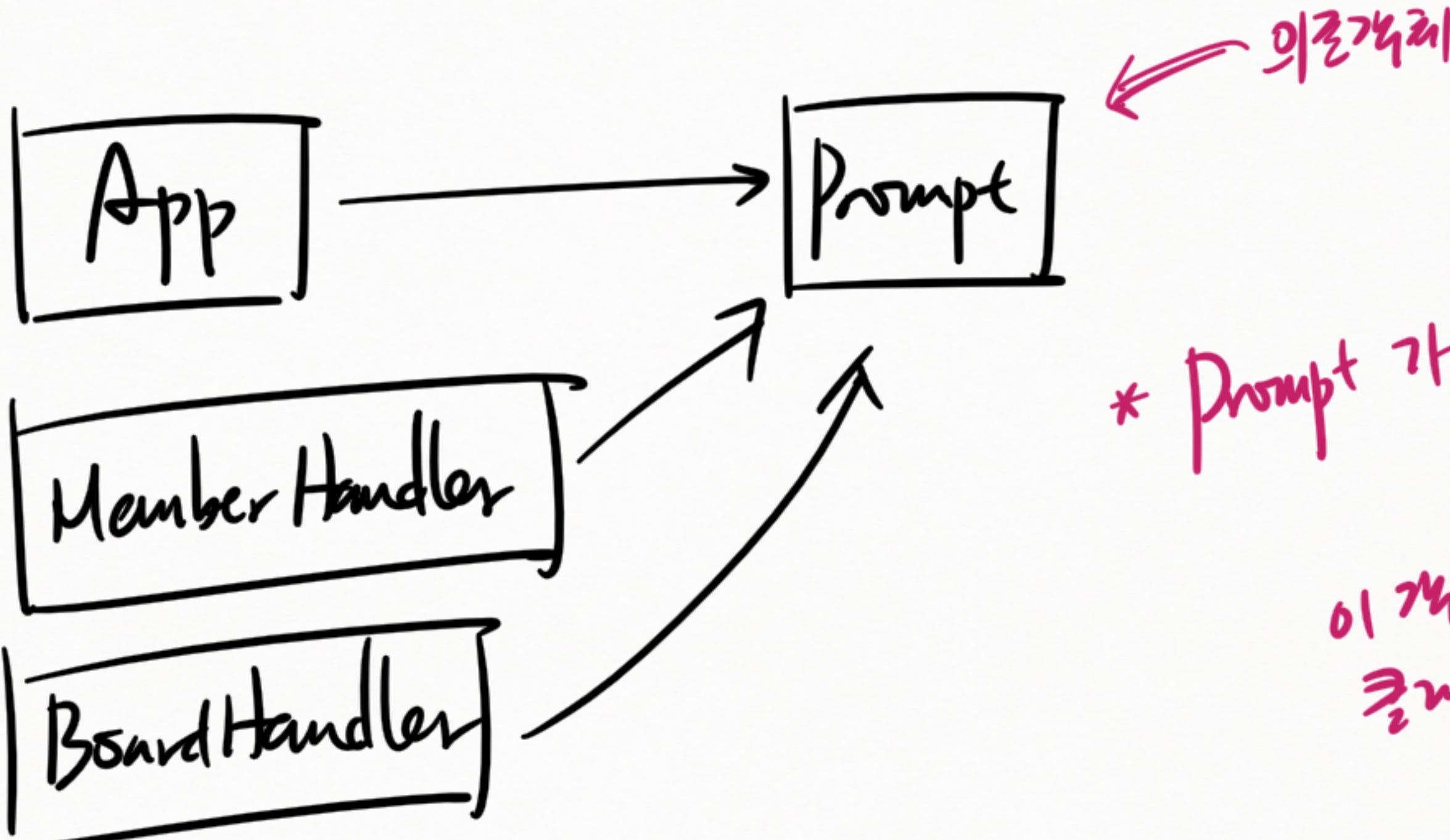
# App, MemberHandler, BoardHandler et prompt

생성자 주입!



부모에  
속해  
하는가?

"생성자 주입"  
Dependency Injection  
(DI)

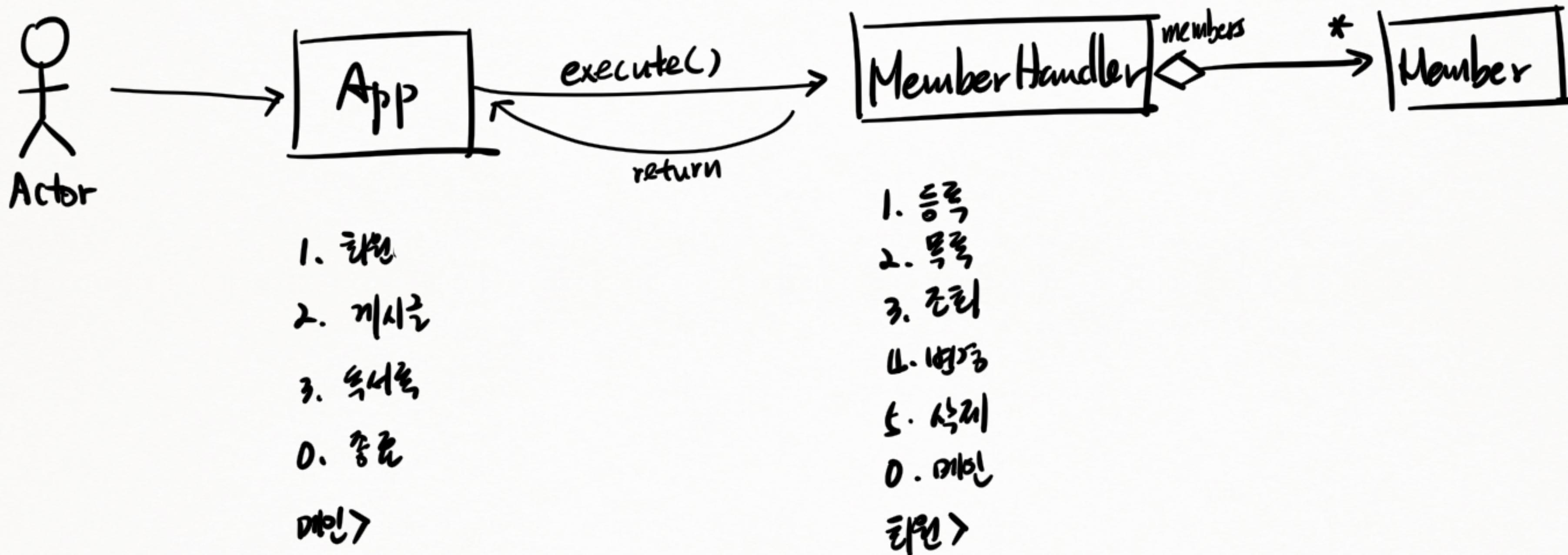


\* Prompt 가 인스턴스 별로 만들기  
의존주입

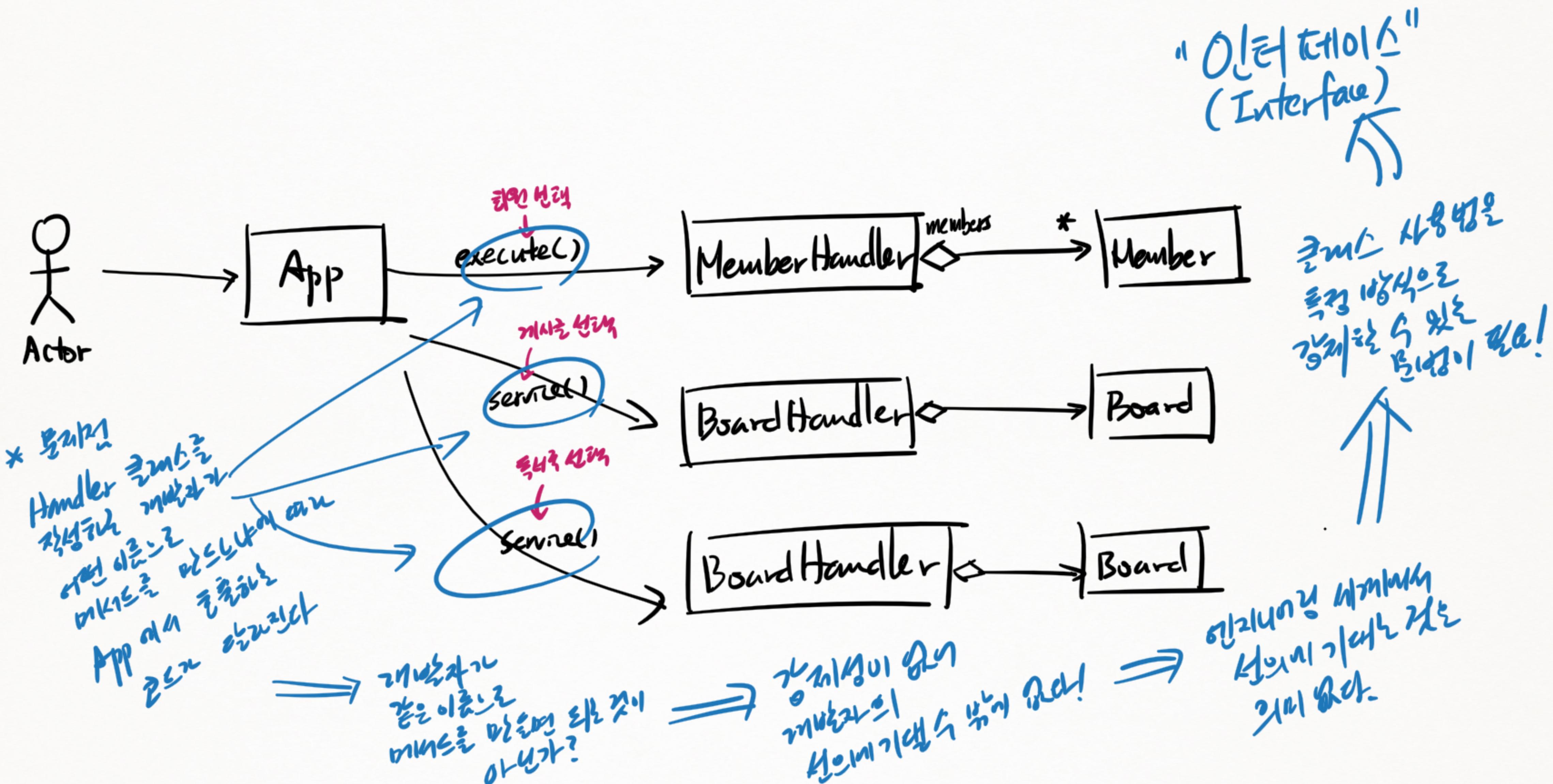
이 개체를 Algorithm  
으로 변경되었을 때  
↓

생성자를 쓸  
의존주입(Prompt)  
같이!  
Injection

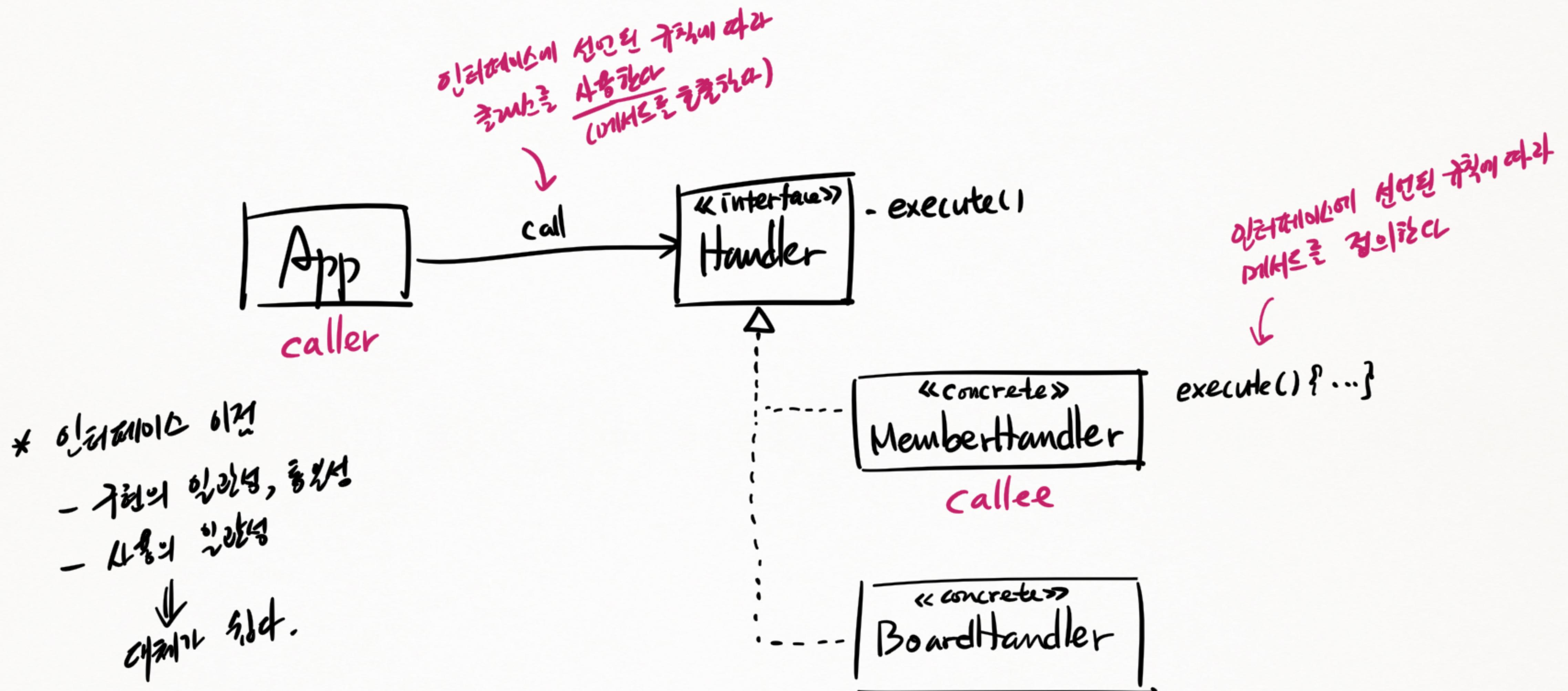
## 16. Handler에게 메뉴 기능을 위임



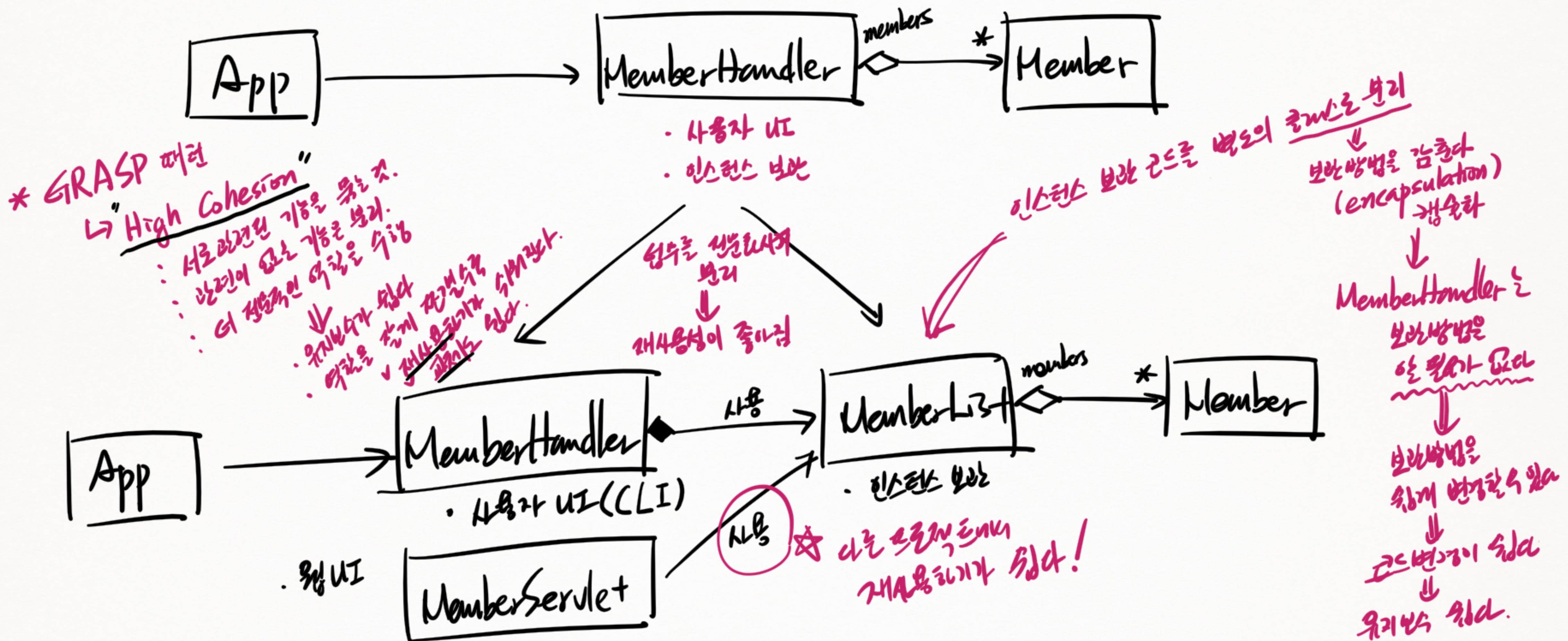
16. Handler에게 메뉴 기능을 위임



### III. Handler의 사용 구조를 인터페이스로 정의하기



18. 인스턴스 목록을 다루는 코드를 빙의 클래스로 묶기



## \* 배운 늘하기

$$\frac{5}{2} = \cancel{2} \cancel{\times}$$

Garbage  $\Rightarrow$ 

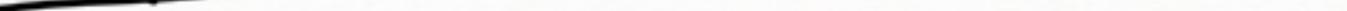
0	1	2	3	4
100	200	300	400	500

$$\underline{t+2} = 4$$

garbage →

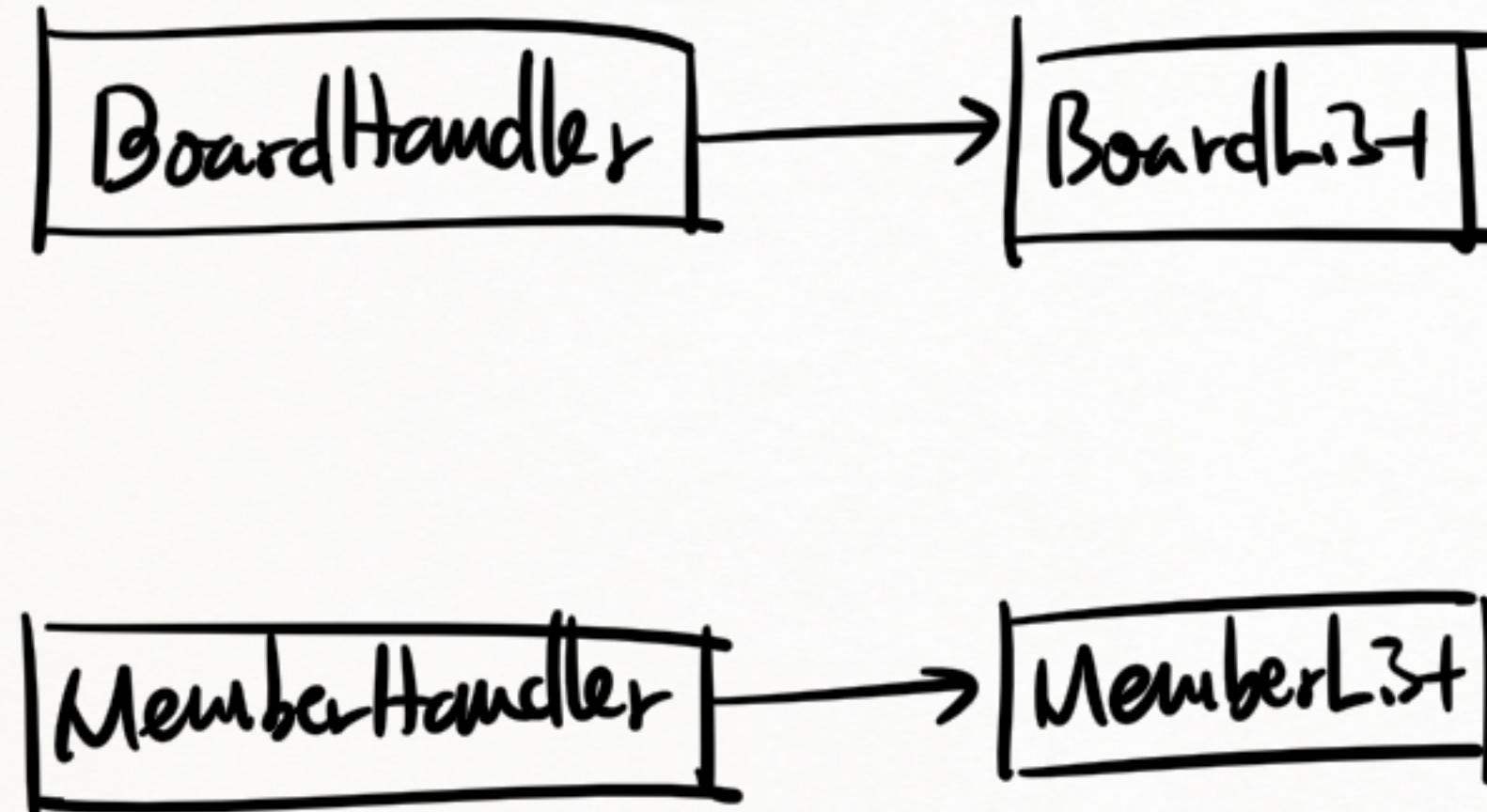
1	2	3	4	5
6	7	8	9	10

$$7 + \frac{1}{3} = 10$$

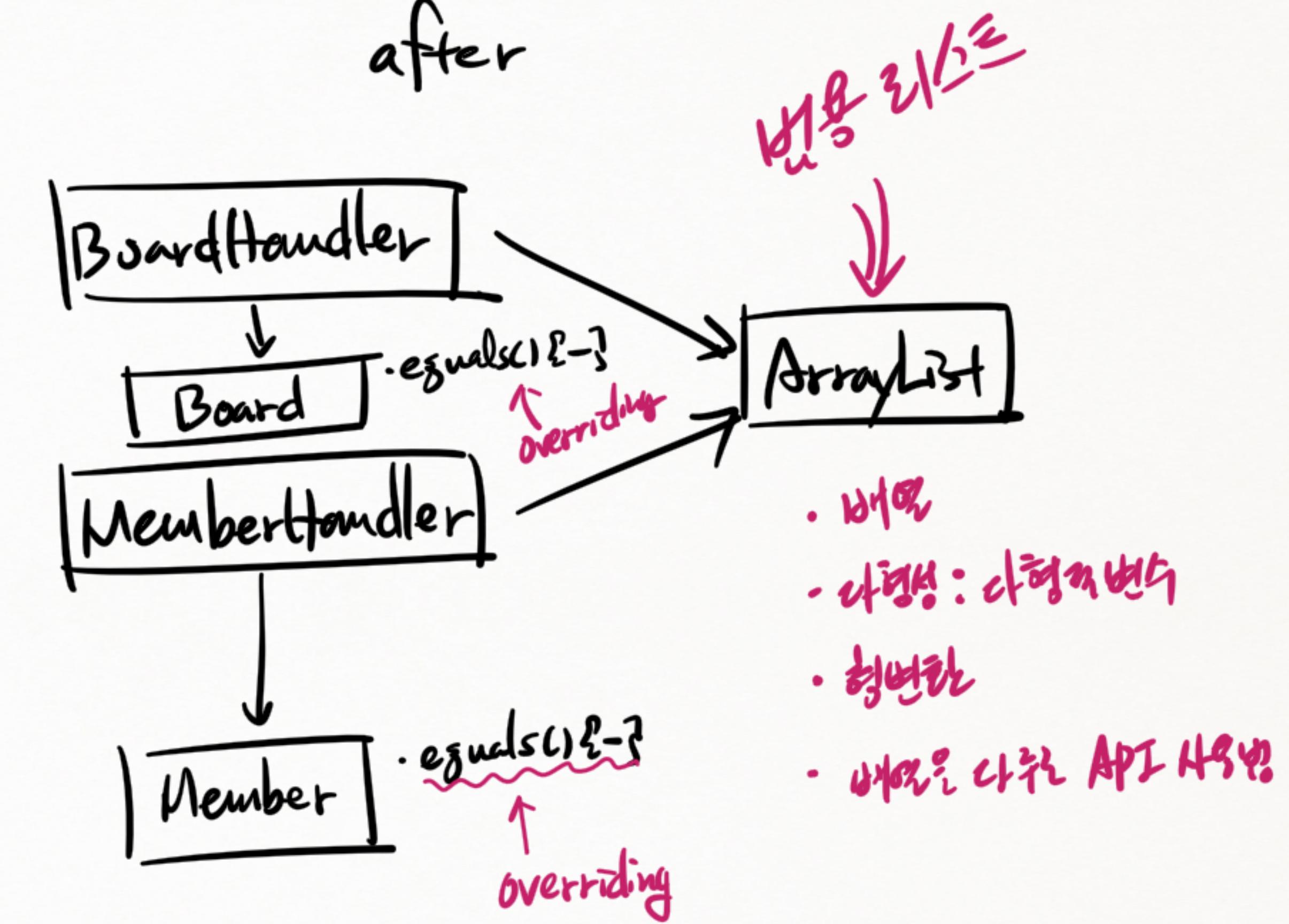
boards → 

## 19. 범용리스트 만들기

before



after

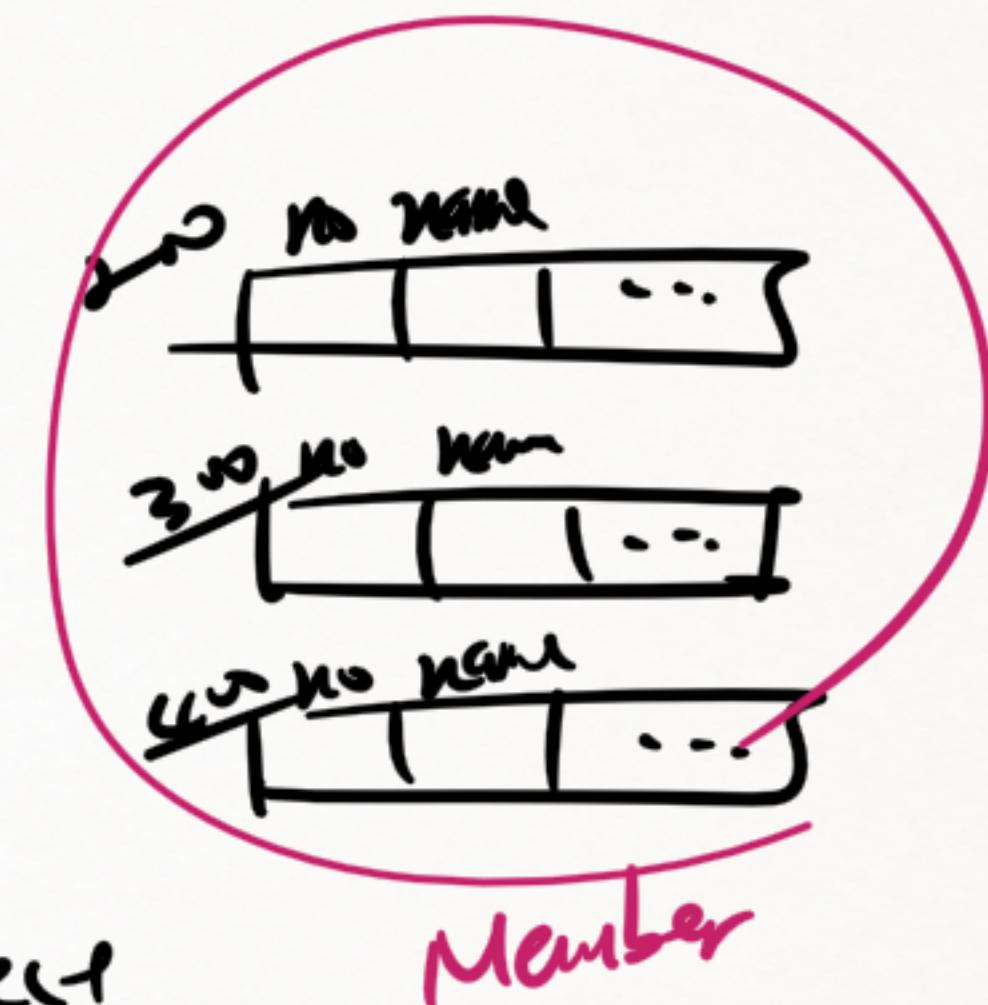
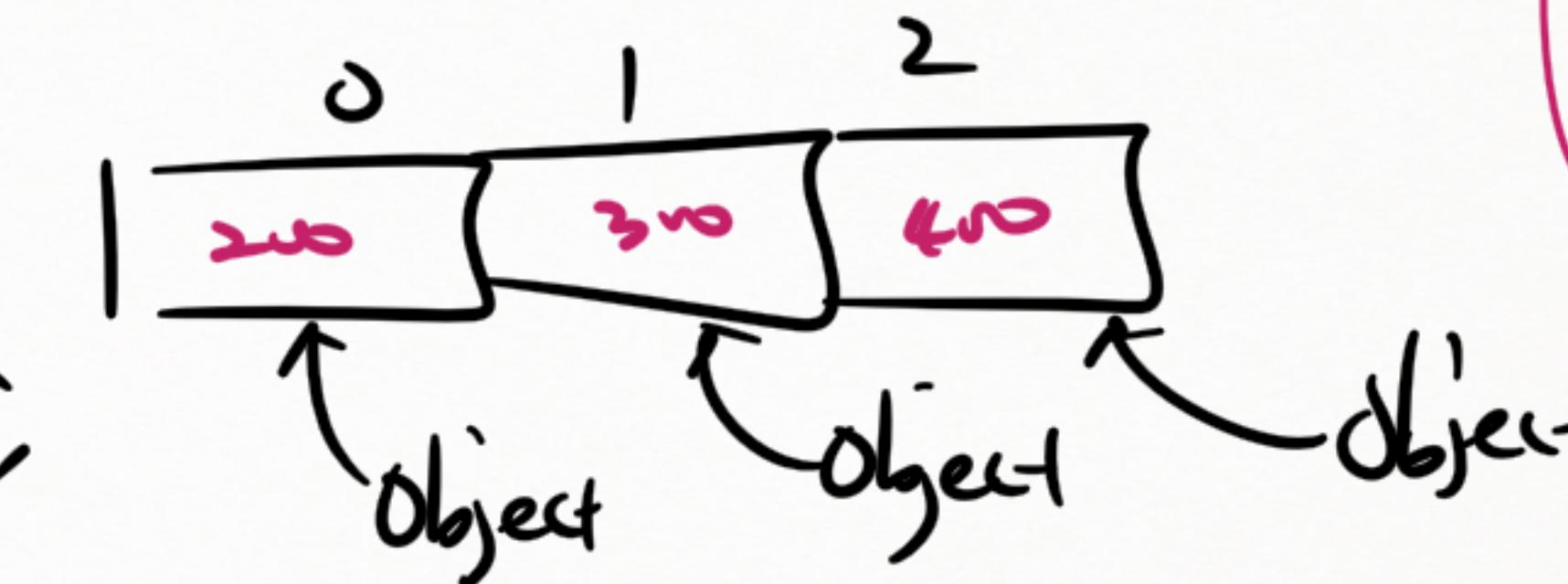


`Object[] arr = new Object[3];`

```
arr[0] = new Member();
```

obj[1] = new Member()

obj1[2] = new Member();



~~Member[] arr2 = (Member[]) arr1;~~

[1] arr1 ;  
↑  
arr1 이 가지는 것 ~  
Member 라는건이 배열의 원소.  
Object 라고는 하지 않지만 Object 라는  
과정은 있다.