

객체지향설계 #Week06

제출일 : 20.10.11

학번 이름 : 201902721 유찬희

GitHub 주소 : <https://github.com/HanCiHu/OOP>

homework1.

```
36  My_cat& My_cat::eat(int food){
37      this->weight += food;
38      return *this;
39  }
```

<< eat함수 코드

```
PS C:\Users\kinm1\OneDrive\바탕 화면\CNU\객체지향설계\OOP\week6> g++ .\homework_06_01.cpp
PS C:\Users\kinm1\OneDrive\바탕 화면\CNU\객체지향설계\OOP\week6> .\a.exe
Weight : 10
Weight : 14
Weight : 18
Weight : 33
```

<<실행결과

- 4번째 weight를 증가해주는 코드에서 cat1.eat(4).eat(5).eat(6)와 같이 코드를 작성하기 위해서는 eat함수의 리턴값이 My_cat 이어야한다.
- 이때, 그냥 this는 리턴 해줄수가 없기 때문에 this의 주소값을 리턴해줘야 한다.
- 따라서, 레퍼런스를 이용하여 this의 주소값을 반환하게 코드를 작성하였다.

homework2.

```
//operator+ 추가하세요
Animal operator+([const Animal &a]){
    Animal temp;
    temp.name = this->name + a.name;
    return temp;
}
```

<< training_02 operator 코드

```
16     Animal & operator+(const Animal &a) {
17         name += a.name;
18         return *this;
19     }
```

<< training_03 operator 코드

- main 함수에서 dog + cat 코드 부분은 dog.operator+(cat)과 같다.
- training03에서 operator+ 코드를 보면 this 객체의 name을 바꿔준 후 this객체를 반환하는 방식이기 때문에 dog의 name까지 바뀌게 된다.
- 반면 training02에서의 operator+코드는 this객체의 name을 바꿔주는게 아닌 새로운 Animal 객체를 만들고 name을 바꿔준 후 그 객체를 반환해주기 때문에 dog의 name은 바뀌지 않은 채로 출력이 되게 된다.