

## 객체지향설계 #Week09

제출일 : 20.11.04

학번 이름 : 201902721 유찬희

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

Homework01.

```
12 //Unary operator
13 int operator++(int) { //postfix operation (indicated by dummy 'int')
14     int dummy = number;
15     number++;
16     return dummy;
17 }
18 int operator++(){ //prefix operation
19     number++;
20     return number;
21 }
22
```

<< 연산자 오버로딩 코드

```
→ week9 git:(master) x g++ homework_09_01.cpp
→ week9 git:(master) x ./a.out
one
1
1
3
This number is: one
```

<< 실행 결과

- 매개변수로 int가 들어가있는 함수는 후위 증가 연산자이고, 매개변수가 없는 함수는 전위 증가 연산자이다.
- 전위 연산일때는 number값을 증가시키고 그 값을 리턴해주면 된다.
- 후위 연산일때는 number값을 증가시켜야 하는것은 같지만 그 값이 아닌 증가하기 전의 값을 리턴해줘야하기 때문에 dummy를 선언하여 미리 값을 복사 시켜놓고 그 값을 리턴 해주었다.

## Homework02.

```
9 public:
10     Employee(std::string name, int age) : name(name), age(age) {}
11
12     virtual void showInfo() = 0;
13 };
14
15 class Manager : public Employee {
16     int managerBonus;
17 public:
18     Manager(int managerBonus, std::string name, int age) : managerBonus(managerBonus), Employee(name, age) {}
19
20     void showInfo() override { std::cout << "Manager Name:" << name << ", Age: " << age << ", managerBonus:" << managerB
21 };
22
23 class Intern : public Employee {
24     std::string majorName;
25 public:
26     Intern(std::string major, std::string name, int age) : majorName(major), Employee(name, age) {}
27
28     void showInfo() override { std::cout << "Intern Name:" << name << ", Age: " << age << ", Major:" << majorName << std:
29 };
30
31 class Janitor : public Employee {
32     int salary;
33 public:
34     Janitor(int salary, std::string name, int age) : salary(salary), Employee(name, age) {}
35
36     void showInfo() override { std::cout << "Janitor Name:" << name << ", Age: " << age << ", Salary:" << salary << std:
37 };
```

<< 수정한 코드

```
➔ week9 git:(master) x g++ homework_09_02.cpp -std=c++11
➔ week9 git:(master) x ./a.out
Manager Name:James, Age: 33, managerBonus:200
Manager Name:Chulsoo, Age: 50, managerBonus:150
Intern Name:Minsu, Age: 24, Major:security
Intern Name:Yong, Age: 19, Major:HCI
Janitor Name:Black, Age: 90, Salary:100
Janitor Name:White, Age: 100, Salary:200
```

<< 실행 결과

- 이전 코드에서는 main 함수에서 배열을 Employee로 선언을 해주었기 때문에 Employee의 showInfo가 다른 클래스의 showInfo를 override 해서 실행된것 같다.
- 위와 같은 문제를 해결하기 위해 Employee의 showInfo를 가상함수로 선언해주고, 각각의 클래스에서 showInfo를 override 해주는 방식으로 해결하였다.