



12강_클래스-I

객체 지향 기법을 이용해서 프로그래밍하기 위한, 클래스와 객체에 대해서 학습합니다.

- 12-1 객체 지향 프로그래밍이란?
- 12-2 절차 지향 VS 객체 지향
- 12-3 클래스란?
- 12-4 클래스 만들기
- 12-5 중복 멤버 함수

12-1 : 객체 지향 프로그래밍이란?

인간의 세계와 동일하게 객체를 위주로 프로그래밍하는 방법

객체 (인간세계)

사람

속성 : 키, 몸무게
기능 : 의사

체중계

속성 : 바늘, 눈금
기능 : 몸무게 측정

자전거

속성 : 바퀴, 체인
기능 : 이동수단

TV

속성 : 채널, 사운드
기능 : 미디어 방출

승용차

속성 : 바퀴, 엔진
기능 : 이동수단

객체 (프로그램)

날씨 프로그램

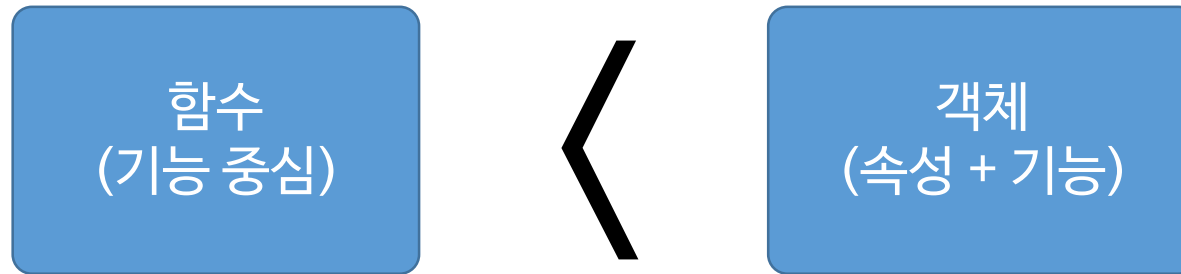
속성 : 온도, 미세먼지
기능 : 날씨 예보

사칙연산 프로그램

속성 : +, -, *, /
기능 : 연산기능

12-2 : 절차 지향 VS 객체 지향

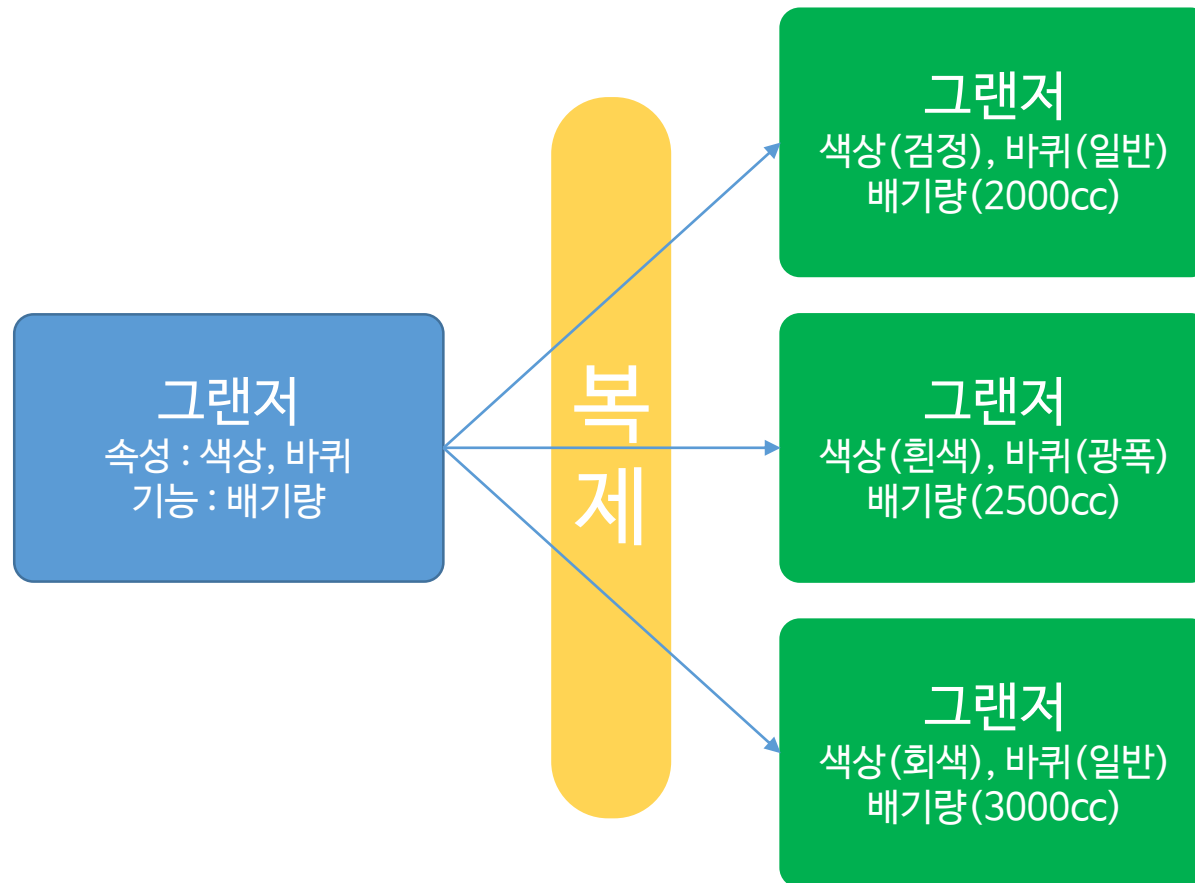
함수 위주의 절차 지향 프로그램 보다 객체 위주의 객체 지향 프로그램이 대세이다.



객체는 속성(데이터)를 가지고 있을 수 있다는 장점과 클래스로부터 동일한 객체를 계속해서 생성할 수 있다.

12-3 : 클래스란?

객체를 생성하기 위한 틀로 모든 객체는 클래스로부터 생성된다.



12-4 : 클래스 만들기

클래스는 멤버 변수와 멤버 함수로 구성된다.

클래스

멤버 변수 + 멤버 함수

class 선언 키워드

class 이름

접근자

```
class Calculator {
```

```
public:
```

```
int xNum;  
int yNum;
```

멤버 변수

```
int add() {  
    return xNum + yNum;  
}
```

멤버 함수

```
};
```

12-4 : 클래스 만들기

클래스는 멤버 변수와 멤버 함수로 구성된다.

```
class Calculator {  
public:  
    int xNum;  
    int yNum;  
  
    int add() {  
        return xNum + yNum;  
    }  
};
```

Calculator c0bj1; // 객체 생성

c0bj1.xNum = 10;

c0bj1.yNum = 20;

cout << "xNum + yNum -----> " << c0bj1.add() << endl;

Calculator c0bj2; // 객체 생성

c0bj2.xNum = 100;

c0bj2.yNum = 200;

cout << "xNum + yNum -----> " << c0bj2.add() << endl;

12-5: 중복 멤버 함수

멤버 함수도 일반 함수처럼 중복해서 여러 개를 이용할 수 있다.

```
class BMICal {  
public:  
    int getBmi(int h, int w) {  
        return h * w;  
    }  
  
    double getBmi(double h, double w) {  
        return h * w;  
    }  
};
```



```
bc1.getBmi(180, 50) --> 9000  
bc1.getBmi(180.5, 50.5) --> 9115.25
```

```
BMICal bc1;  
cout << "bc1.getBmi(180, 50) --> " << bc1.getBmi(180, 50) << endl;  
cout << "bc1.getBmi(180.5, 50.5) --> " << bc1.getBmi(180.5, 50.5) << endl;
```