

객체 지향 프로그래밍 (OOP)

-박지수

***클래스**: 가장 이상적인 객상(이데아) , 추상적 원형

↳ 속성(인스턴스 변수)와 행위(메서드)를 모아놓은 집합체

↳ 객체의 설계도

***인스턴스**: 현실의 객체 (이데아를 본땀지만 조금씩 차이가 있음) , 구체적 (클래스를 인스턴트화)

↳ 같은 클래스에서 나왔지만 서로가 구별되는 존재들. 별개임

-클래스 정의

```
class cat
{ //필드(데이터)
    private int height;
    private int weight;
    private Color color; //private: 클래스 내에서만 사용(외부에서 access 불가능)
}
```

***생성자**(constructor): 새로운 객체 만들때 호출되는, 클래스와 이름이 동일한 메서드
필드값을 초기화함

```
public cat() //public: 접근한정자 (외부에서 access가능)
{ weight=38
  height=20
  color=Color.Red;
}
```

- **상속:** 클래스의 재사용이 목적

부모 클래스 A: 상속을 하는 클래스

자식 클래스 B

```
class A:
```

```
{
```

```
}
```

```
class B:A
```

```
{
```

```
}
```