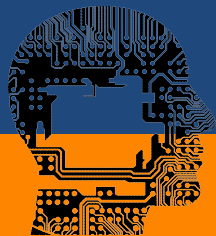




한국IT진흥부설

정보보호교육학원

아이섹



# 자바 프로그래밍 기초

## 추상 클래스 (abstract class)

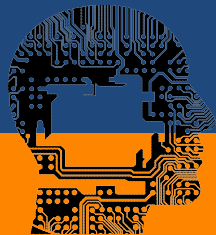
By SoonGu Hong(Kokono)



한국IT진흥부설

정보보호교육학원

아이섹



# 자바 프로그래밍 기초

## 1. 추상클래스의 개념

## 1-1. 추상적의 의미

- 사전적 의미로 **추상(abstract)**은 어떤 사물이 실체가 없이 일정한 형태나 성질이 없고 막연하고 일반적인 것을 뜻합니다.
- 예를 들면 차량, 동물, 회사 등은 구체적인 대상이 아닌 많은 카테고리들을 모아둔 추상적인 집합적 개념이라는 것입니다.
- 호랑이, 고양이, 개와 같은 구체적인 대상들을 하나로 모아 서 동물이라고 부르는 것과 같이 '동물'이라는 것은 추상적 개념이 됩니다.
- 클래스에도 추상적 개념이 들어간 추상 클래스가 존재합니다. 추상적인 것은 실체가 없기 때문에 **추상클래스는 객체를 생성할 수 없습니다.**
- 단지 구체적인 실체클래스에게 **속성과 기능을 전달하는 매개체**가 될 뿐이죠.

## 1-2. 추상클래스의 용도

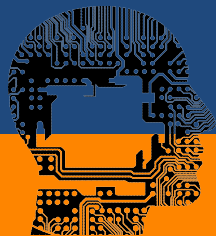
- 객체지향 프로그래밍에서는 객체간 공통된 속성과 기능을 상속이라는 개념을 통해 처리합니다.
- 상속을 통해 생성된 부모클래스를 추상 클래스로 만들어 사용하는 데 그 이유가 뭘까요?

- # 실체 클래스들의 **공통된 필드와 메서드의 이름을 통일**하기 위해서
- 개발 프로젝트에서 설계자와 코더는 일반적으로 다른사람이고 코더는 여러명일 가능성이 큼니다.
  - A라는 코더는 객체의 움직임을 처리하는 기능의 이름을 `move()`라고 짓고, B라는 코더는 `walk()`라고 짓는다면 같은 기능임에도 불구하고 객체마다 사용방법이 달라집니다.
  - 이를 해결하기 위해 설계자는 추상클래스로 추상메서드 `move()`를 선언하고 코더들에게 이 추상클래스를 상속받게함으로써 이문제를 해결합니다.



한국IT진흥부설

정보보호교육학원 아이섹



# 자바 프로그래밍 기초

## 2. 추상클래스의 선언

## 2-1. 추상적 의미를 부여하는 키워드 abstract

```
class abstract Person {  
    //필드  
    String name;  
  
    //생성자  
    Person(String name) {  
        this.name = name;  
    }  
  
    //메서드  
    void move() {}  
}
```

```
class Student extends Person {  
    int grade; //학년필드  
  
    Student(String name) {  
        super(name);  
    }  
}
```

Person kim = new Person(); //(X) 불가능!

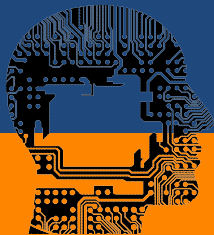
Person kim = new Student(); //(O)

- 추상적 개념인 사람은 실제로 객체로 만들 필요가 없다. 구체적인 개념인 학생, 직원, 교사, 의사같은 객체의 규격(다형성) 역할만 하면 된다!



한국IT진흥부설

정보보호교육학원 아이섹



# 자바 프로그래밍 기초

## 3. 추상메서드

## 3-1. 추상 메서드

- 추상 클래스는 실제 클래스가 공통적으로 가져야 할 필드와 메서드들을 정의해 놓은 추상적인 클래스로 실제 클래스의 **멤버를 통일**화하는데 목적이 있습니다.
- 만약 실제 객체들이 모두 동일한 실행 내용을 가지고 있다면 추상 클래스에서 그 내용들을 정의하여 상속해주면 되지만, 이름은 같아도 실제 객체들의 실행내용이 다를 경우에는 추상 클래스에서 **추상 메서드**를 만들어 **오버라이딩을 강제**할 수 있습니다.

```
public abstract class Car {  
  
    public abstract void run();  
    public abstract void break();  
}
```

- 추상 메서드는 일반 메서드와 달리 메서드 블록을 열어 실행내용을 작성하지 않고, 세미콜론으로 닫아 선언만 하여 추상적인 규격만 만들어 둡니다.



## 3-2 추상메서드는 실체클래스에게 오버라이딩을 강요

```
class Avante extends Car {  
  
    //Car클래스의 run(), break()메서드의 오버라이딩이 강제됨.  
  
}
```

### - 추상 메서드 선언시 주의사항

1. 추상 메서드는 반드시 추상 클래스에서만 선언 가능.
2. 추상 메서드를 실체 클래스가 오버라이딩 하지 않을 경우 컴파일 에러 발생!
3. 추상 클래스에는 추상 메서드 이외에 일반적인 인스턴스 메서드, 정적 메서드도 선언 가능.

**감사합니다**  
**THANK YOU**