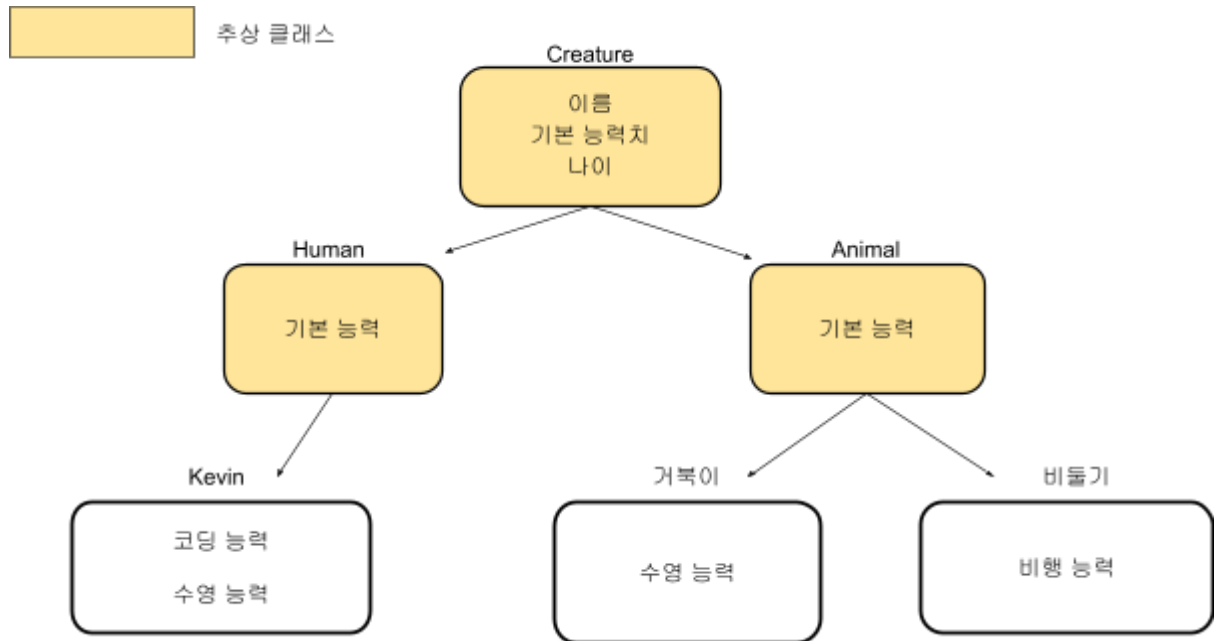


인터페이스 & 추상 클래스 & 상속 예제 => 파일 위치 IT_TA/src/Lv_2

기본 명세서

다음과 같은 관계를 가진 구조도입니다.



Kevin, 거북이, 비둘기의 능력은 모두 인터페이스로 생성하며 상속받아 작성하세요.

능력을 사용(메소드 호출) 할 때마다 능력치는 상승합니다.

최종 호출시에는

사진과 같이 최종 능력치와 나이가 함께 보여질 수 있어야 합니다.

- 1) 기본 능력은 능력치가 올라가지 않습니다.
- 2) 기본 능력을 제외한 능력을 사용(메소드 호출) 할 때마다 능력치는 상승합니다.
- 3) 명세서에 작성된 능력외에도 작성자가 새로 능력을 추가할 수 있습니다.
- 4) 나이도 증가시킬 수 있습니다.

** 이번 문제는 추상클래스와 인터페이스 간의 차이를 느껴보고자 함입니다.

상속을 받고 메소드를 어떻게 오버라이딩하는지 작성을 해보면서 아시길 바랍니다^^

```

비둘기능력치 : 10 나이 : 5
기본 공격 : 뚝뚝은 자신의 신체로 공격합니다.
날 수 있게 되었습니다. 능력치 10 상승
++특수능력 : 물에 다닙니다. 능력치 20 상승
비둘기능력치 : 40 나이 : 5
=====
거북이 능력치 : 10 나이 : 5
기본 공격 : 뚝뚝은 자신의 신체로 공격합니다.
수영을 할 수 있게 되었습니다. 능력치 20 상승
++ 특수능력 : 물에서 숨을 쉴 수 있습니다. 능력치 20 상승
거북이 능력치 : 50 나이 : 5
=====
Kevin 능력치 : 20 나이 : 10
인간은 도구를 사용하여 공격!
코딩을 할 수 있게 되었습니다. 능력치 30 상승
수영을 할 수 있게 되었습니다. 능력치 20 상승
Kevin 능력치 : 70 나이 : 20

```

<= 이 부분은 명세서에서 따로 추가한 메소드입니다. 이와 같이 따로 추가한 능력은 인터페이스가 아닌 일반 메소드로 진행 해도 됩니다.

```

// 능력치 , 나이
Pigeon p = new Pigeon("비둘기",10, 5);
p.printInfo();
p.attack();
p.fly();
p.move();
p.printInfo();

System.out.println("=====");
Turtle t = new Turtle("거북이",10,5);
t.printInfo();
t.attack();
t.swim();
t.breath();
t.printInfo();

System.out.println("=====");
Kevin k = new Kevin("Kevin",20,10);
k.printInfo();
k.attack();
k.coding();
k.oldAge(10);

k.swim();
k.printInfo();

```