

프로그래밍 언어 응용

chapter08

상속과 다형성

제공된 자료는 훈련생의 수업을 돕기 위한 것으로, 타인과 공유하시면 안됩니다.

Contents

part.1

상속

part.2

형변환

part.3

메소드 오버라이드

part.4

다형성

part.5

다운캐스팅과 instanceof

상속이란 부모가 자식에게 자산을 물려주는 것을 말한다.
자바에서 상속이란 클래스가 다른 클래스에게 속성을 전달하는 것이다.



상속관계가 맺어지면 자식은 부모의 특성을 물려받는다.

부모



국적
성

부모



다리가 있다
먹이를 먹는다
짖는다

자식



자식



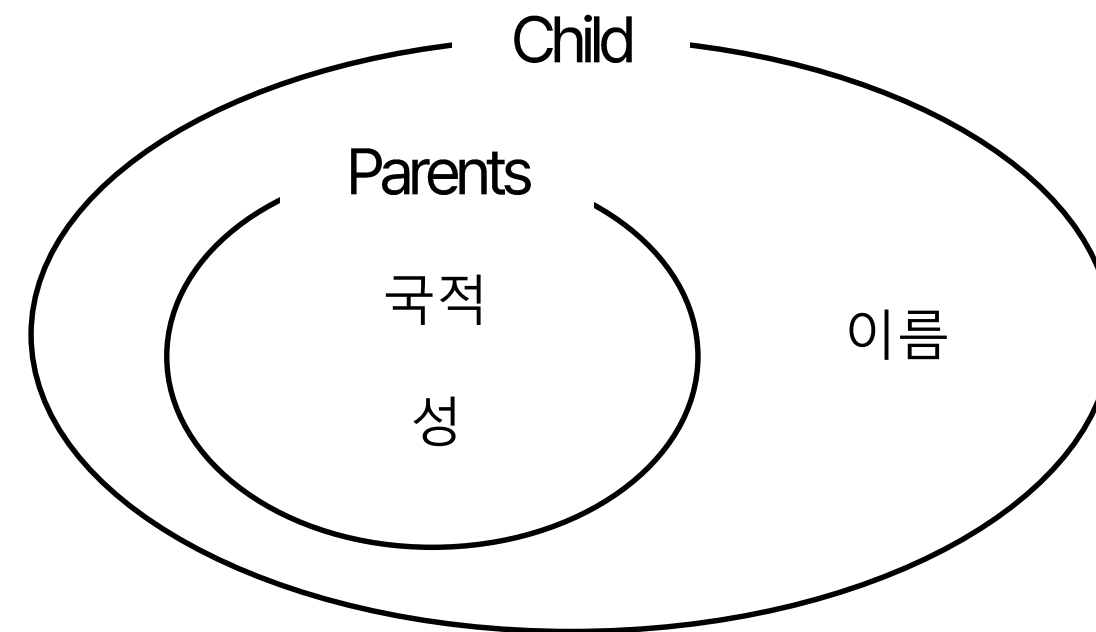
자식클래스는 'extends'를 사용하여 부모클래스를 상속받는다.

자식클래스는 부모의 필드와 메소드를 물려받는다.

자식클래스는 상속받은 속성 외에 더 필요한 속성과 기능을 추가할 수 있다.

```
Class Parents { //부모클래스  
    String nationality;  
    String lastName;  
}
```

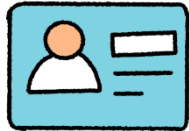
```
Class Child extends Parents { //자식클래스  
    String firstName;  
}
```




Q. 쇼핑몰 회원을 관리하는 프로그램을 구현한다.

일반회원은 물건을 구매하면 포인트 1퍼센트가 적립된다.

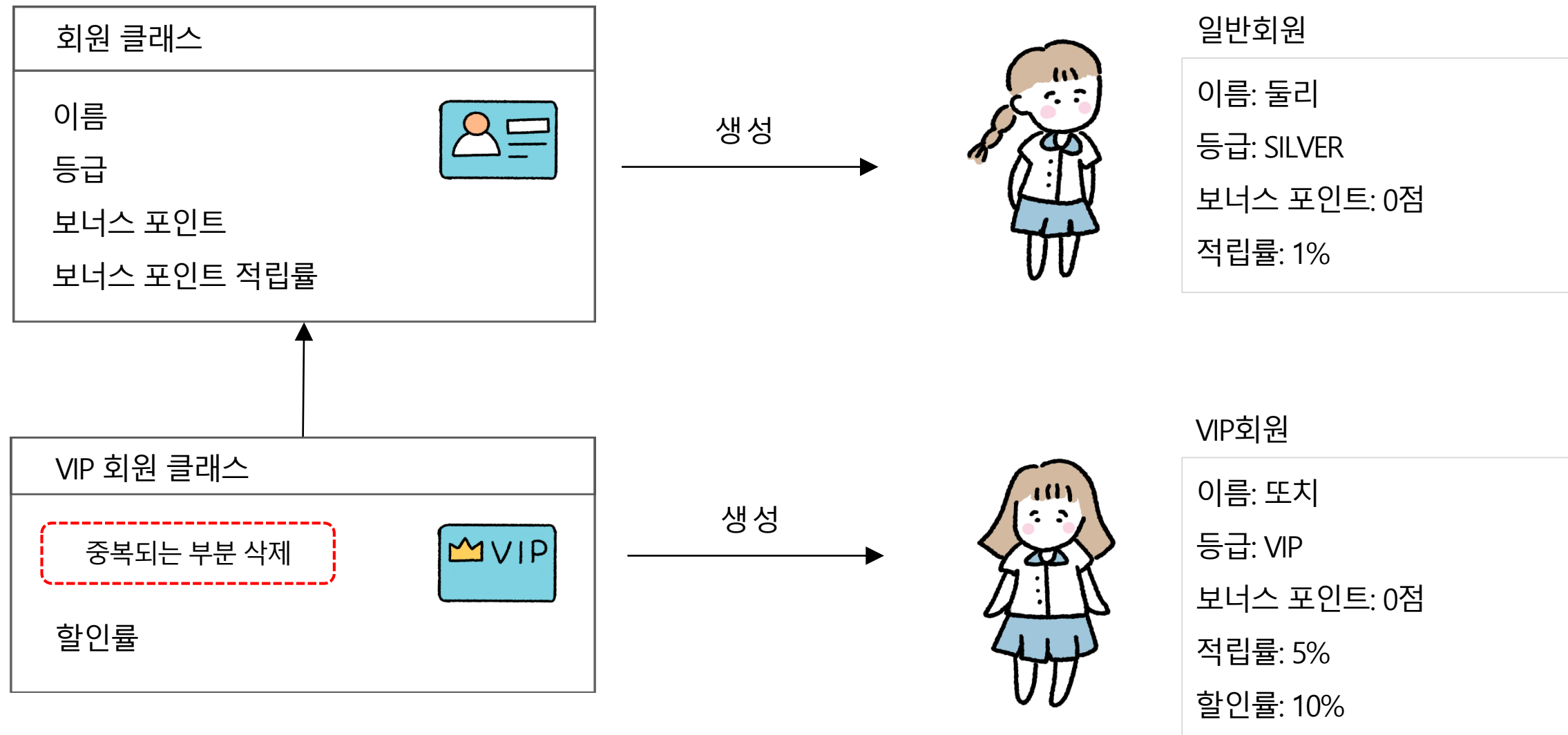
VIP회원은 물건을 구매하면 포인트 5퍼센트가 적립되고, 가격의 10퍼센트를 할인 받는다.

회원 클래스	
이름	
등급	
보너스 포인트	
적립률	

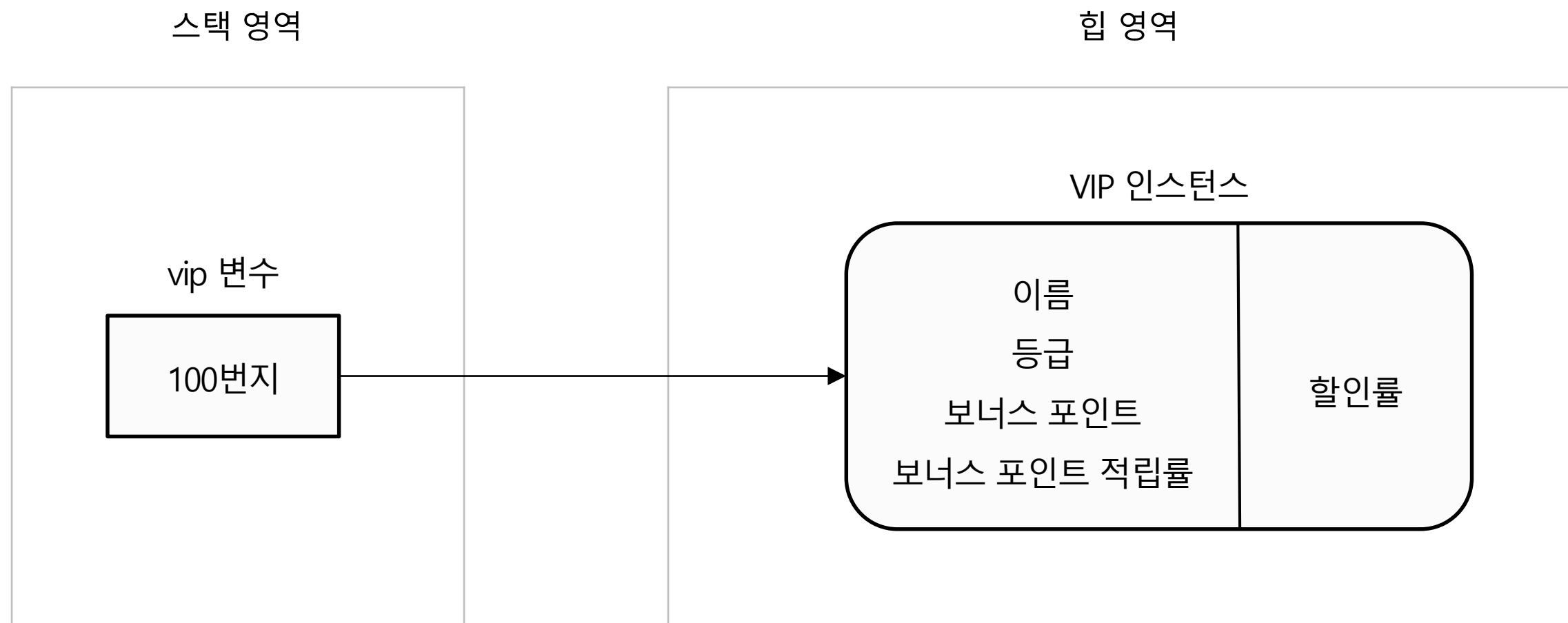
VIP 회원 클래스	
이름	
등급	
보너스 포인트	
적립률	
할인률	

Customer 클래스와 중복

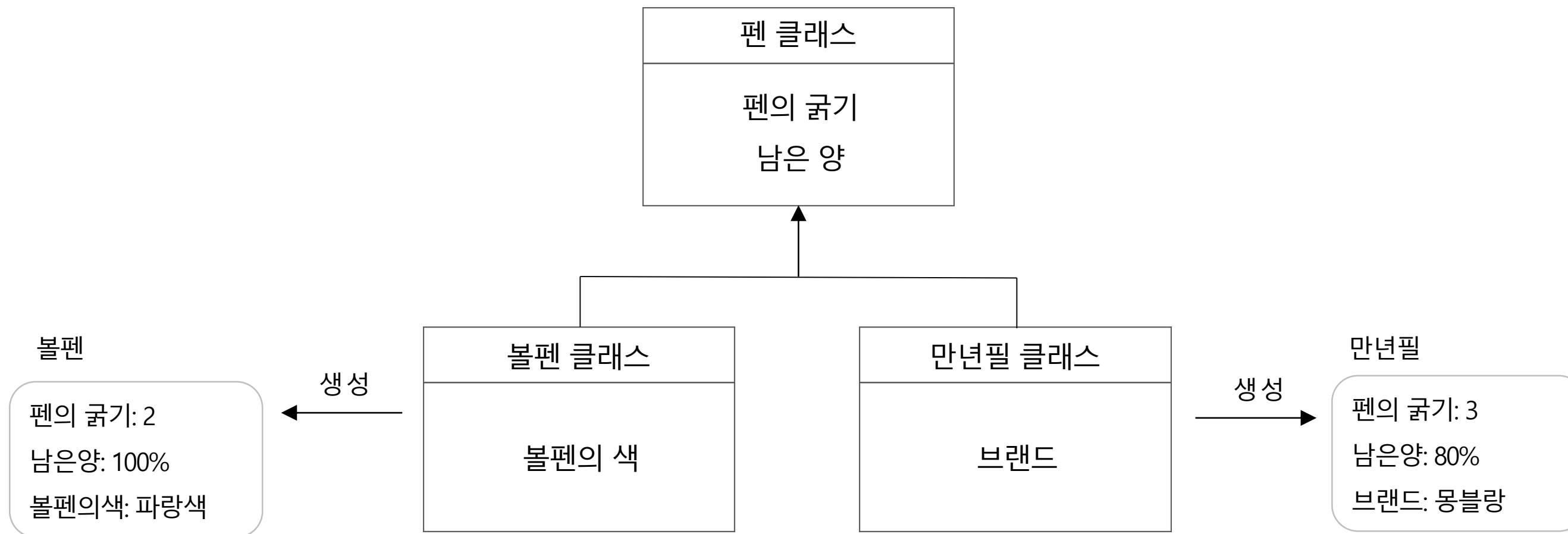
Customer클래스에 기본정보를 넣고, VIP클래스는 Customer클래스를 상속받는다.



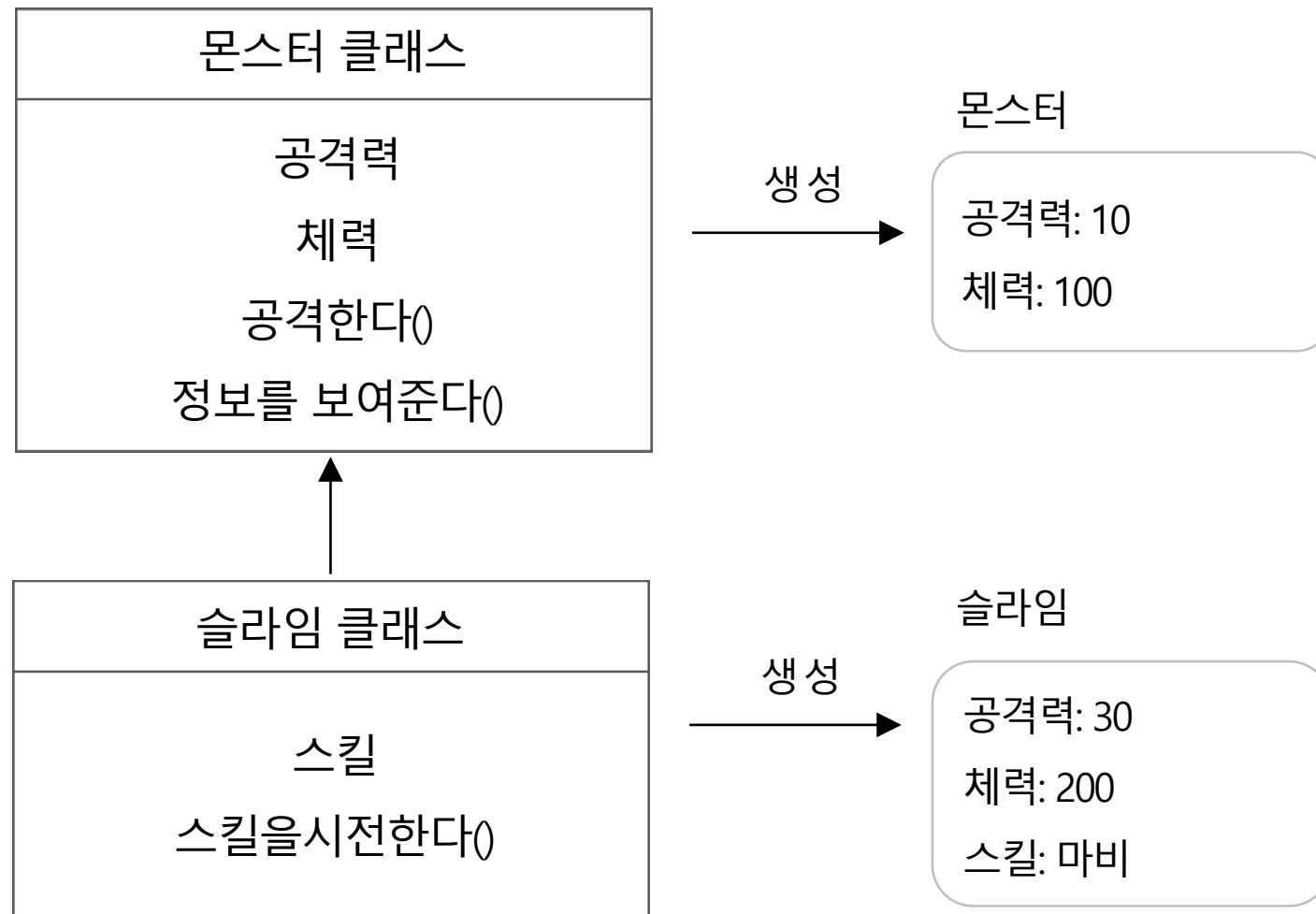
VIP 인스턴스는 물려받은 멤버와 VIP클래스에서 추가한 멤버를 모두 가지고 있다.



아래 그림은 펜과 볼펜, 만년필 클래스의 상속관계를 표현한 것입니다.
다음과 같이 클래스를 구현하고, 객체를 생성하세요.



아래 그림은 몬스터와 슬라임 클래스의 상속관계를 표현한 것입니다.
다음과 같이 클래스를 구현하고, 객체를 생성하세요.



테스트 결과

```
Problems @ Javadoc Declaration Console X History Coverage
<terminated> Quiz4 (14) [Java Application] C:\Users\wimjiyeon\p2\pool\plugins\org.e
몬스터의 공격력은 10이고 체력은 100이다.
몬스터가 10만큼 공격했다!

몬스터의 공격력은 30이고 체력은 200이다.
몬스터가 30만큼 공격했다!
슬라임이 마비올 시전했다!
```

super란?

- super는 자식클래스 내부에서 부모를 가리키는 참조변수이다.
- super()는 상위클래스의 생성자를 호출한다.
- 생성자 함수에 super() 호출 코드가 없으면 컴파일러가 자동으로 생성한다.

Class VIP extends Customer {

public VIP(){

super();

}

}

Class Customer {

public Customer(){

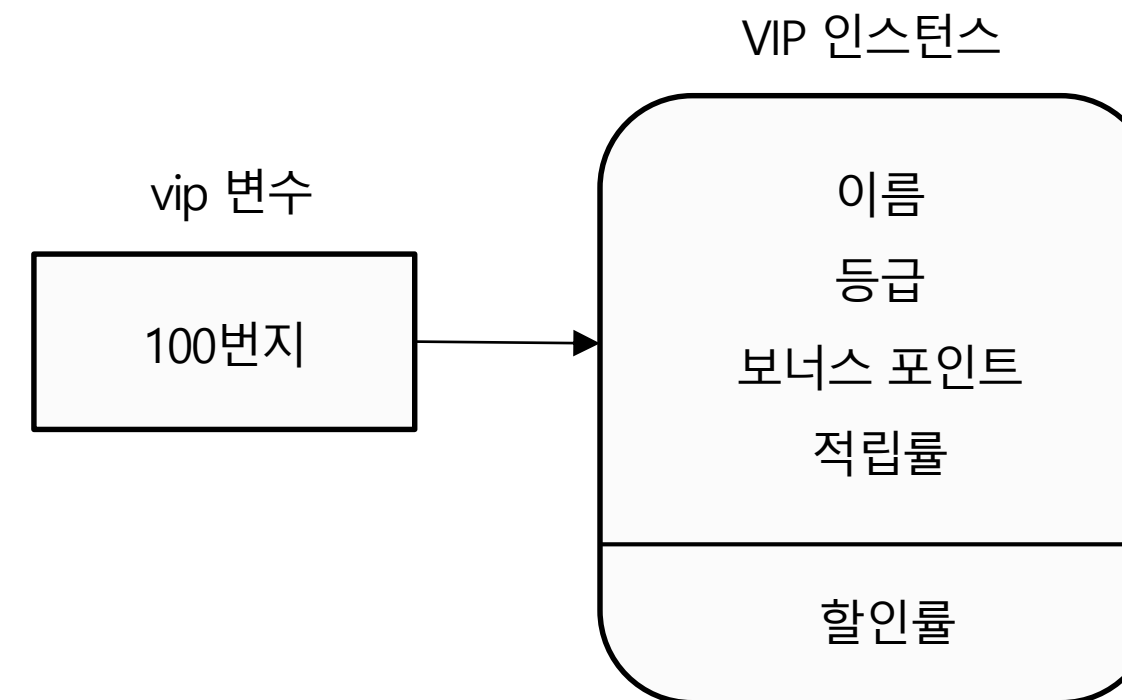
}

}

부모클래스의 생성자 호출

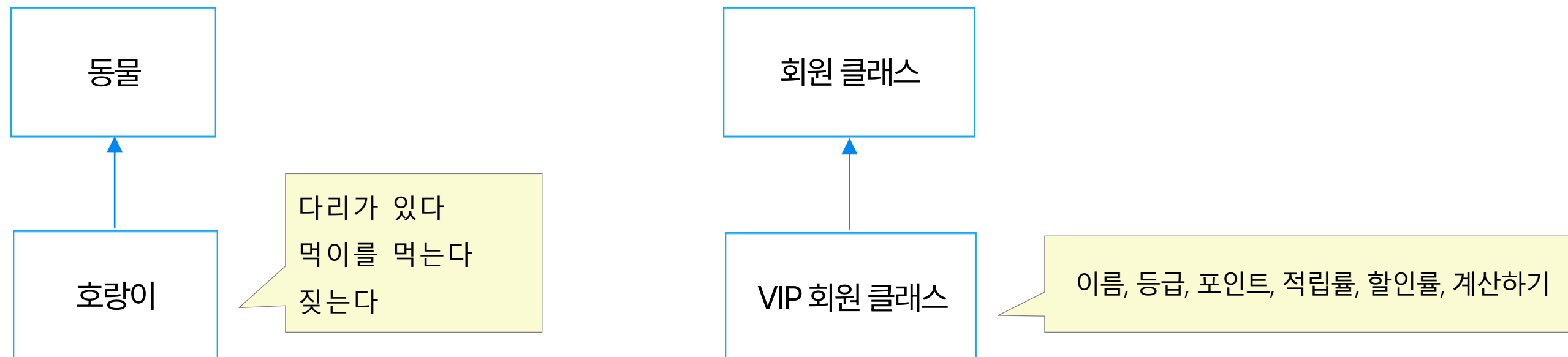
자식클래스의 인스턴스가 생성될 때, 부모클래스의 생성자가 먼저 호출된다.

```
Class Customer {  
    public Customer( ){  
    }  
}  
  
Class VIP extends Customer {  
    public VIP( ){  
        super( );  
    }  
}
```



클래스의 형변환이란?

- 클래스의 형변환은 클래스를 다른 클래스로 변환하는 것을 의미한다.
- 상속관계에서 'A는 B다' 라는 문장이 성립된다.
예를 들어, VIP회원은 회원의 기능을 모두 가지고 있기 때문에 'VIP는 회원이다' 라고 표현할 수 있다.
- 따라서 VIP클래스는 Customer클래스로 형변환할 수 있다.

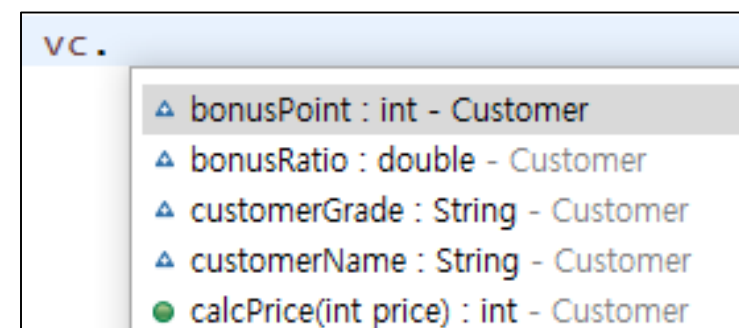


클래스의 형변환은 상속관계에서 성립되며, 자식클래스는 부모클래스로 형변환할 수 있다.
이는 상속관계에서 부모 클래스의 참조변수로 자식 클래스의 객체를 참조하는 것을 의미한다.

부모타입 변수

자식 인스턴스

```
Customer vc = new VIPCustomer();
```



vc변수

100번지

VIP 인스턴스

이름

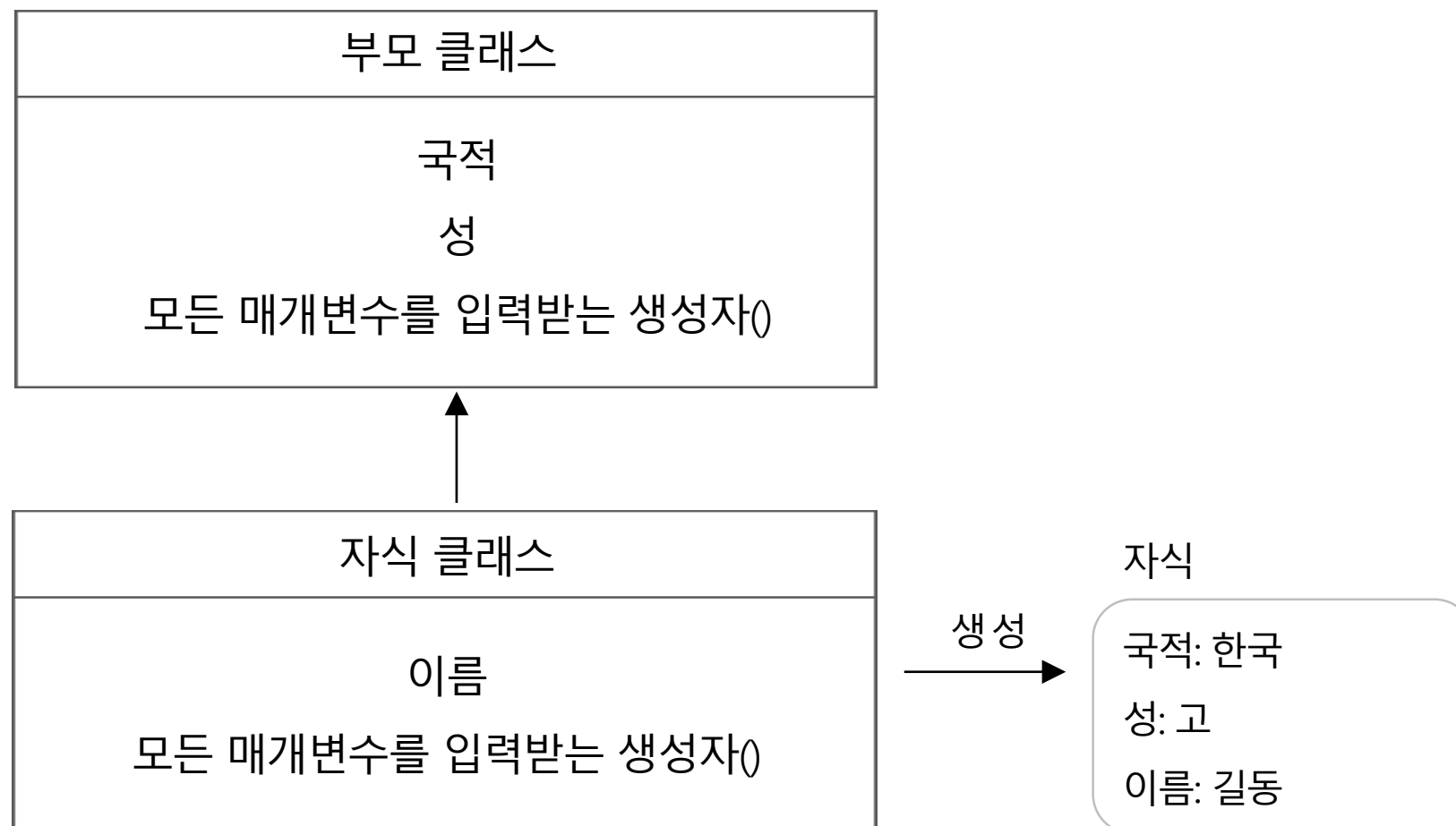
등급

보너스 포인트

적립률

할인률

아래 그림은 부모와 자식 클래스의 상속관계를 표현한 것입니다.
다음과 같이 클래스를 구현하고, 객체를 생성하세요.
단, 부모클래스 참조변수를 선언하고 자식 클래스로 생성한 인스턴스를 대입하세요.
그리고 자식클래스의 정보를 출력하세요.



오버라이딩이란?

- 상위클래스의 메소드가 하위클래스와 맞지 않는 경우에는 메소드를 재정의 할 수 있다.
- 메소드를 재정의하여, 동작을 바꾸는 것을 '메소드 오버라이딩'이라고 한다.
- 오버라이딩의 규칙은 반환타입, 메소드이름, 매개변수가 모두 같아야 한다.
- VIP 회원에게 할인을 적용하려면, Customer 클래스의 계산 메소드를 재정의해야 한다.

Customer

```
public void calcPrice(int price) {  
    print( price );  
}
```

물건의 가격을 그대로 지불한다

VIP

@Override

```
public void calcPrice(int price) {  
    print(price - (int)(price*saleRatio));  
}
```

할인을 적용하여 세일가를 지불한다

VIP객체를 생성하고 계산 메소드를 호출하면, 재정의된 메소드가 실행된다.
이제 모든 회원은 본인의 등급에 맞게 물건 가격을 지불할 수 있다.

```
Customer customer = new Customer();  
customer.calcPrice(10000);
```

```
VIPCustomer vip = new VIPCustomer();  
vip.calcPrice(10000)
```

[실행결과]

일반회원은 10000원을 지불했습니다.

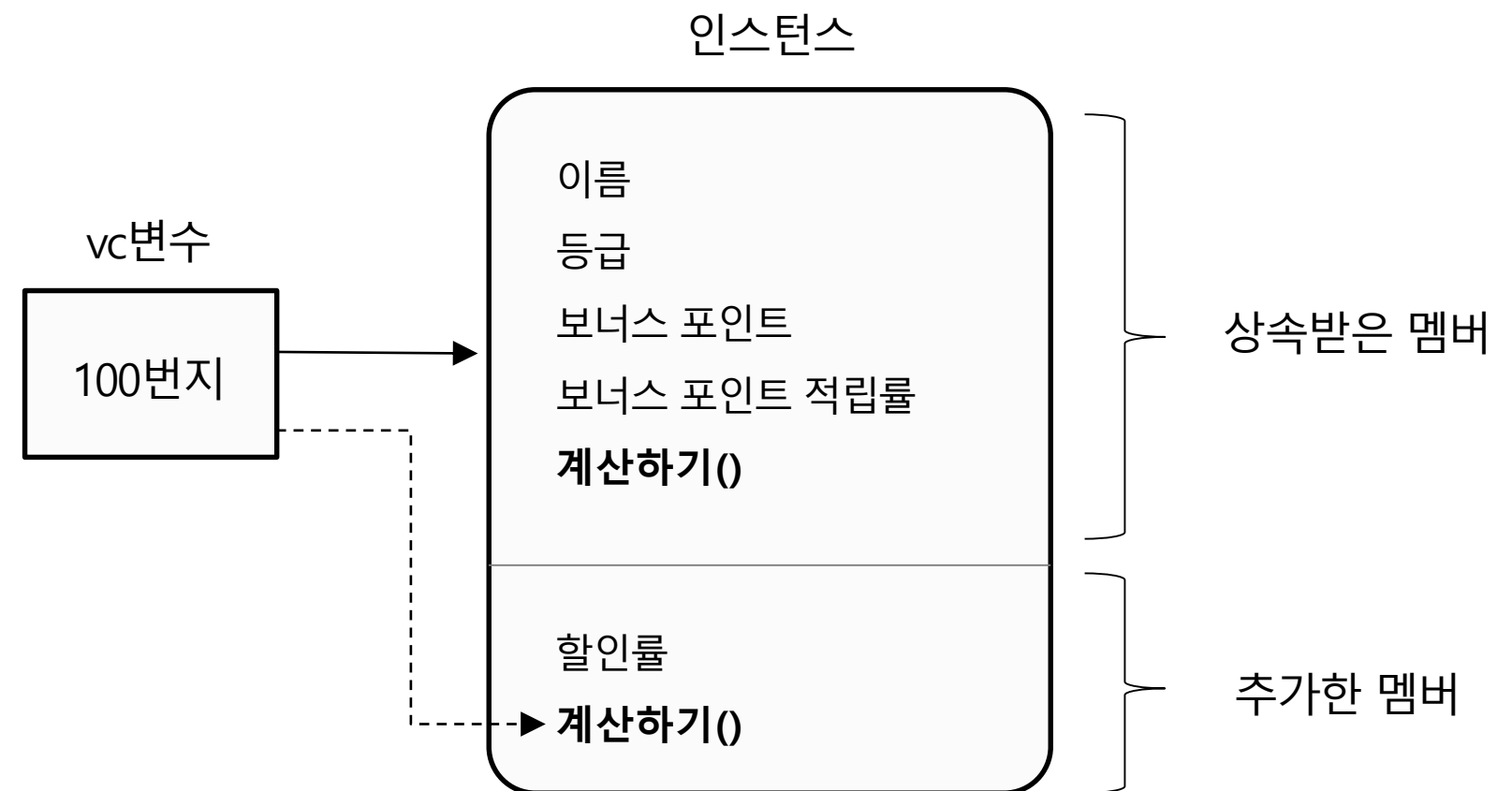
VIP회원은 9000원을 지불했습니다.

클래스를 형변환하고 메소드를 호출하면, 실제 객체의 타입에 따라 오버라이드된 메소드가 호출된다.

```
Customer vc = new VIPCustomer();  
vc.calcPrice(10000);
```

[실행결과]

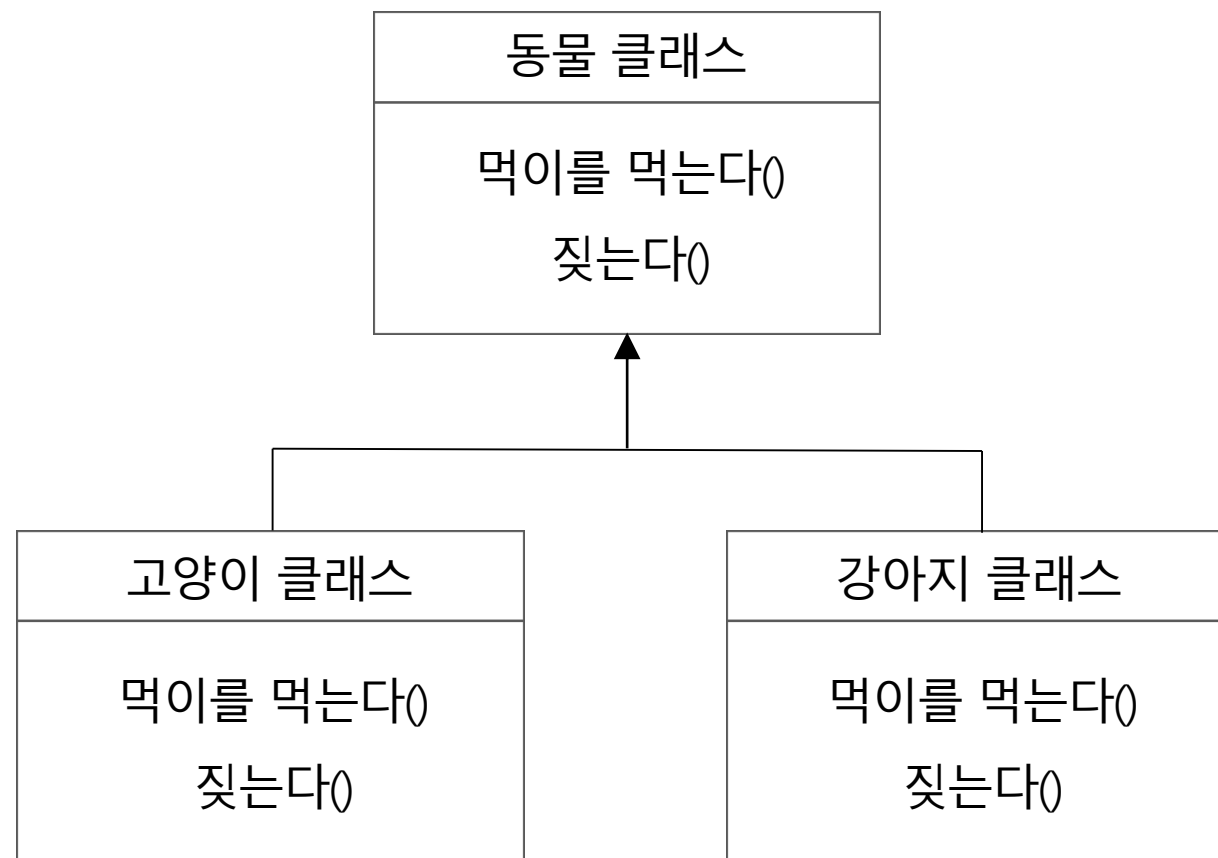
vc회원은 9000원을 지불했습니다.



아래 그림은 동물과 고양이, 강아지 클래스의 상속관계를 표현한 것입니다.

다음과 같이 클래스를 구현하고, 객체를 생성하세요.

그리고 다음과 같이 결과가 출력되도록 메소드를 재정의하세요.



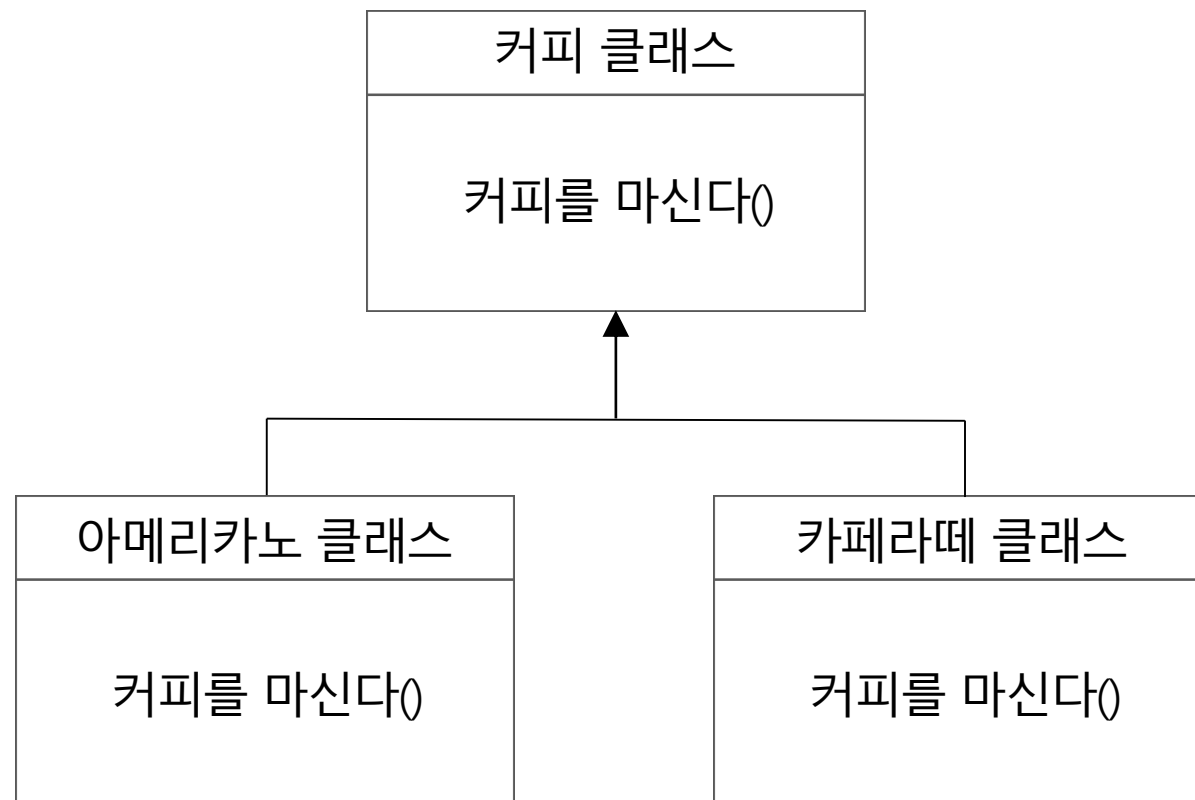
테스트 코드

```
Cat cat = new Cat();
Dog dog = new Dog();
cat.eat();
cat.sound();
dog.eat();
dog.sound();
```

테스트 결과

```
Problems @ Javadoc Declaration Console X
<terminated> Quiz1 (18) [Java Application] C:\Users\Wimjiyeo
고양이는 쥐를 먹는다
고양이는 야옹하고 짖는다
개는 고기를 먹는다
개는 멍멍하고 짖는다
```

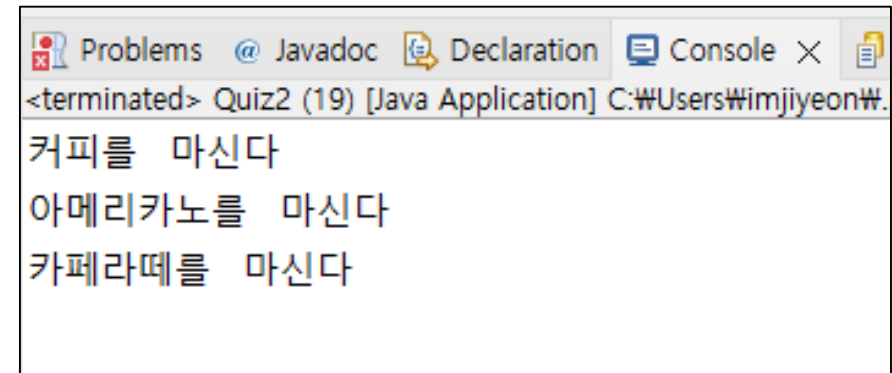
아래 그림은 커피와 아메리카노, 카페라떼 클래스의 상속관계를 표현한 것입니다.
다음과 같이 클래스를 구현하고, 객체를 생성하세요.
그리고 다음과 같이 결과가 출력되도록 메소드를 재정의하세요.



테스트 코드

```
Coffee coffee1 = new Coffee();
coffee1.taste();
Coffee coffee2 = new Americano();
coffee2.taste();
Coffee coffee3 = new CafeLatte();
coffee3.taste();
```

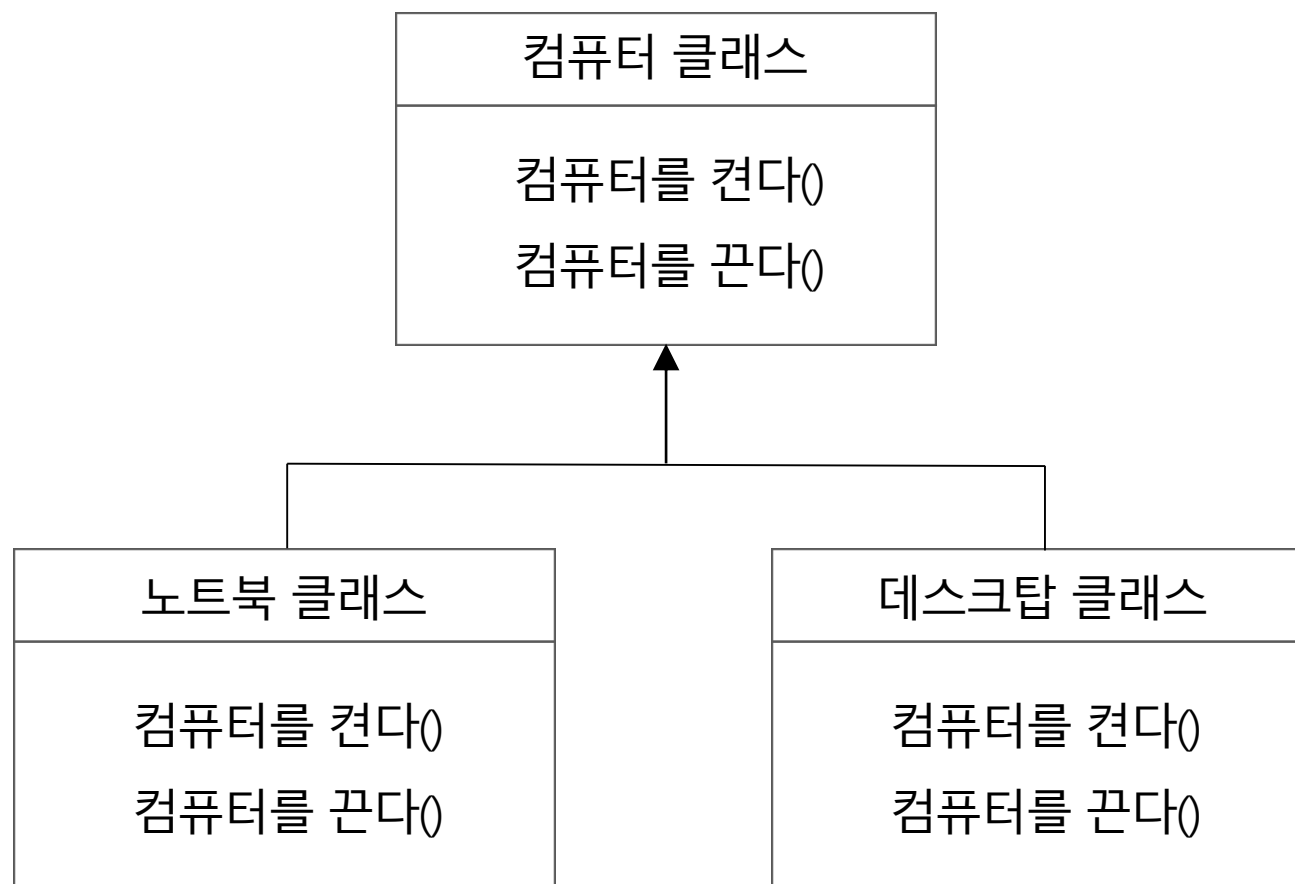
테스트 결과



아래 그림은 컴퓨터와 노트북, 데스크탑 클래스의 상속관계를 표현한 것입니다.

다음과 같이 클래스를 구현하고, 객체를 생성하세요.

그리고 다음과 같이 결과가 출력되도록 메소드를 재정의하세요.



테스트 코드

```

Computer computer1 = new Computer();
computer1.on();
computer1.off();
Computer computer2 = new Notebook();
computer2.on();
computer2.off();
Computer computer3 = new Desktop();
computer3.on();
computer3.off();
  
```

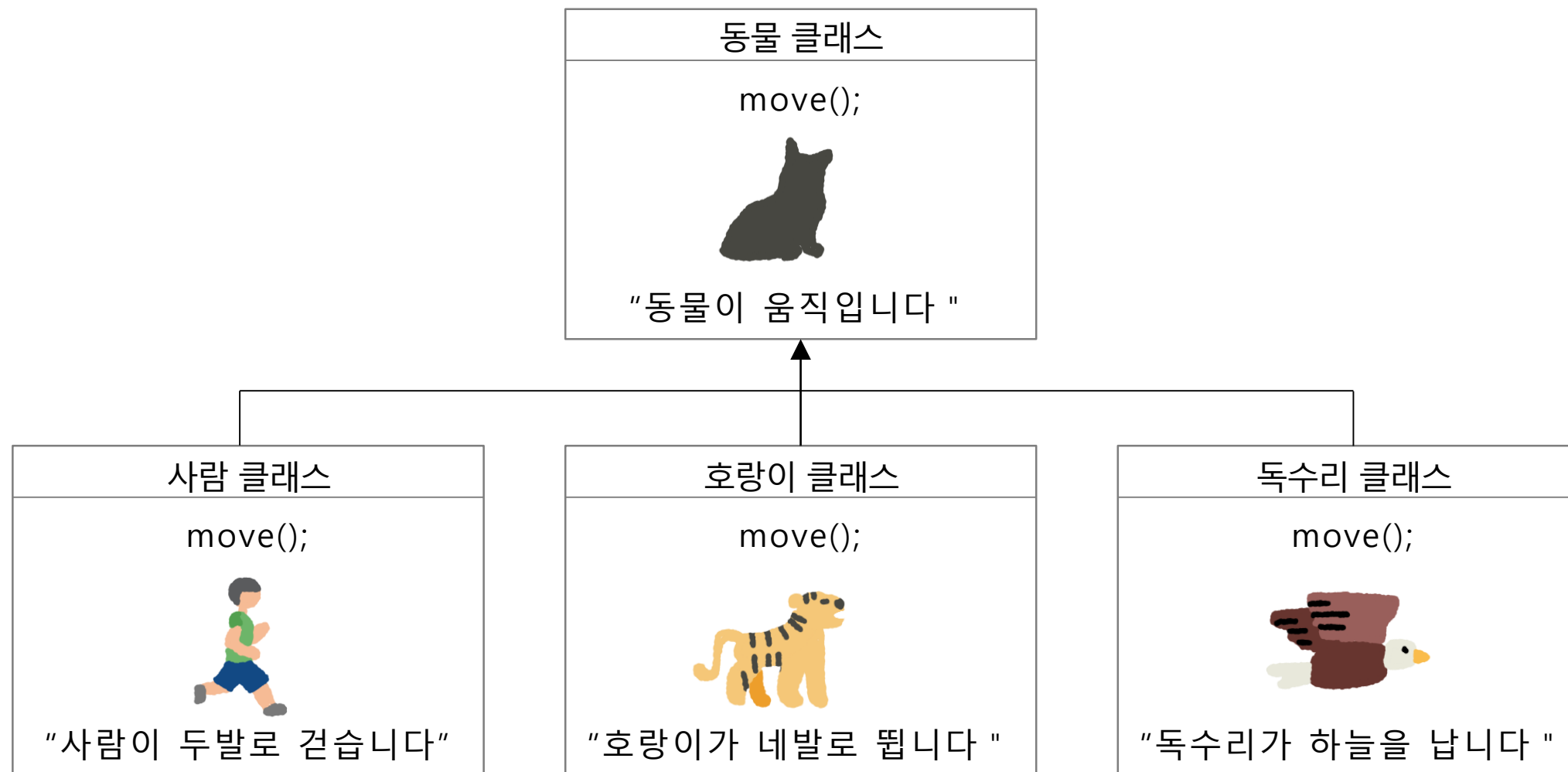
테스트 결과

```

Problems @ Javadoc Declaration Console x
<terminated> Quiz3 (11) [Java Application] C:\Users\jimjiyeon
컴퓨터를 켭니다
컴퓨터를 끕니다
노트북을 켭니다
노트북을 끕니다
데스크탑을 켭니다
데스크탑을 끕니다
  
```

다형성이란?

- 다형성은 객체지향 프로그래밍의 기본적인 개념이다.
- 다형성은 하나의 클래스를 사용하여, 다양한 객체를 처리할 수 있다는 의미이다.



다형성의 특징

- 부모클래스 타입으로 선언된 변수를 통해 실제 객체의 메소드를 호출할 수 있다.
- 동일한 코드를 실행하더라도 실제 인스턴스에 따라 결과가 달라진다.

```
Animal human = new Human();
```

```
Animal tiger = new Tiger();
```

```
Animal eagle = new Eagle();
```

```
human.move();
```

```
tiger.move();
```

```
eagle.move();
```

[실행 결과]

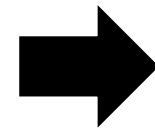
사람이 두발로 걷습니다

호랑이가 네발로 뛸니다

독수리가 하늘을 날니다

메소드의 매개변수로 부모타입을 사용하면 모든 자식 클래스를 받을 수 있다.

```
void moveAnimal(Human human) {  
    human.move();  
}  
void moveAnimal(Tiger tiger) {  
    tiger.move();  
}  
void moveAnimal(Eagle eagle) {  
    eagle.move();  
}
```

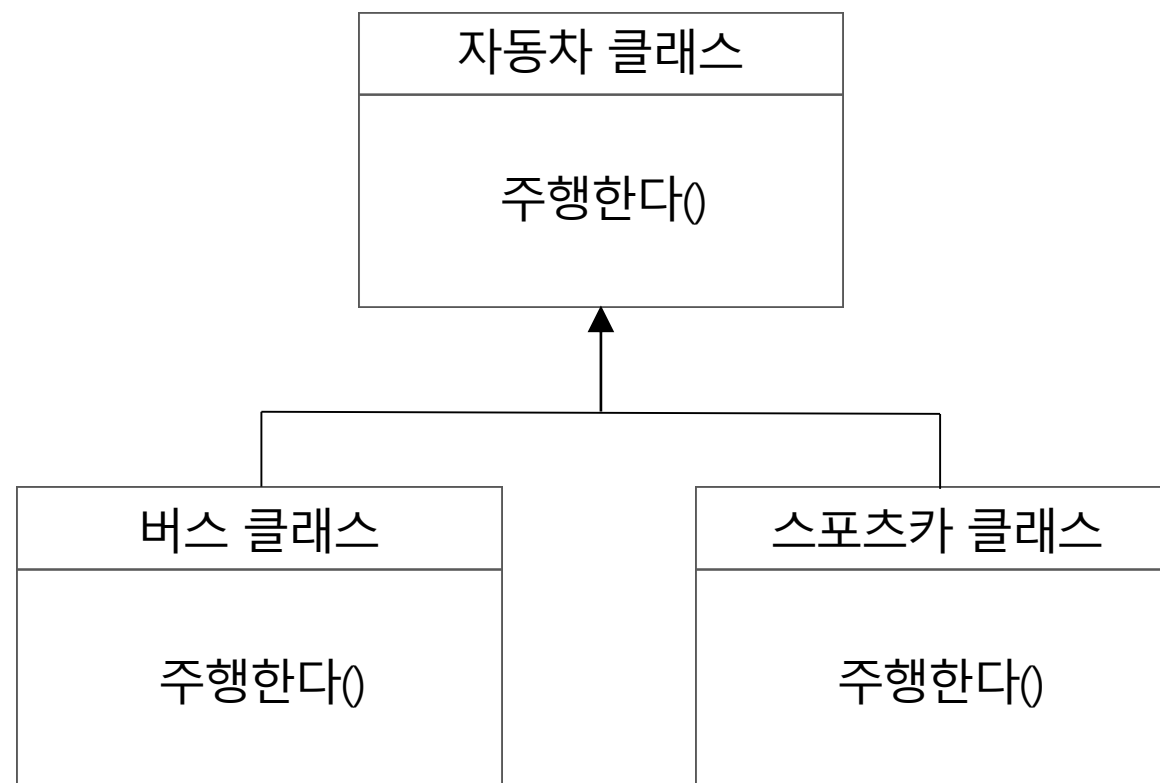


```
void moveAnimal(Animal animal) {  
    animal.move();  
}
```

사람, 호랑이, 독수리



아래 그림은 자동차와 버스, 스포츠카 클래스의 상속관계를 표현한 것입니다.
다음과 같이 클래스를 구현하세요.
그리고 테스트용 클래스에 runCar 메소드()를 만들어서 다음과 같이 결과를 출력하세요.



테스트 코드

```
runCar(new Bus());
runCar(new SportsCar());
```

테스트 결과

```
<terminated> Quiz1 (19) [Java Application] C:\Users\Wimjiyeon...
버스가 달린다
스포츠카가 달린다
```

리스트에 다형성을 활용하면, 프로그램을 유지보수하는데 매우 편리하다.
리스트 자료형에 부모클래스의 타입을 사용하면, 자식클래스의 객체도 담을 수 있다.

```
ArrayList<Customer> list = new ArrayList<>();
```

회원 리스트

```
list.add(new Customer("둘리"));
```

```
list.add(new Customer("또치 "));
```

```
list.add(new Customer("도우너 "));
```

```
list.add(new VIPCustomer("마이콜"));
```

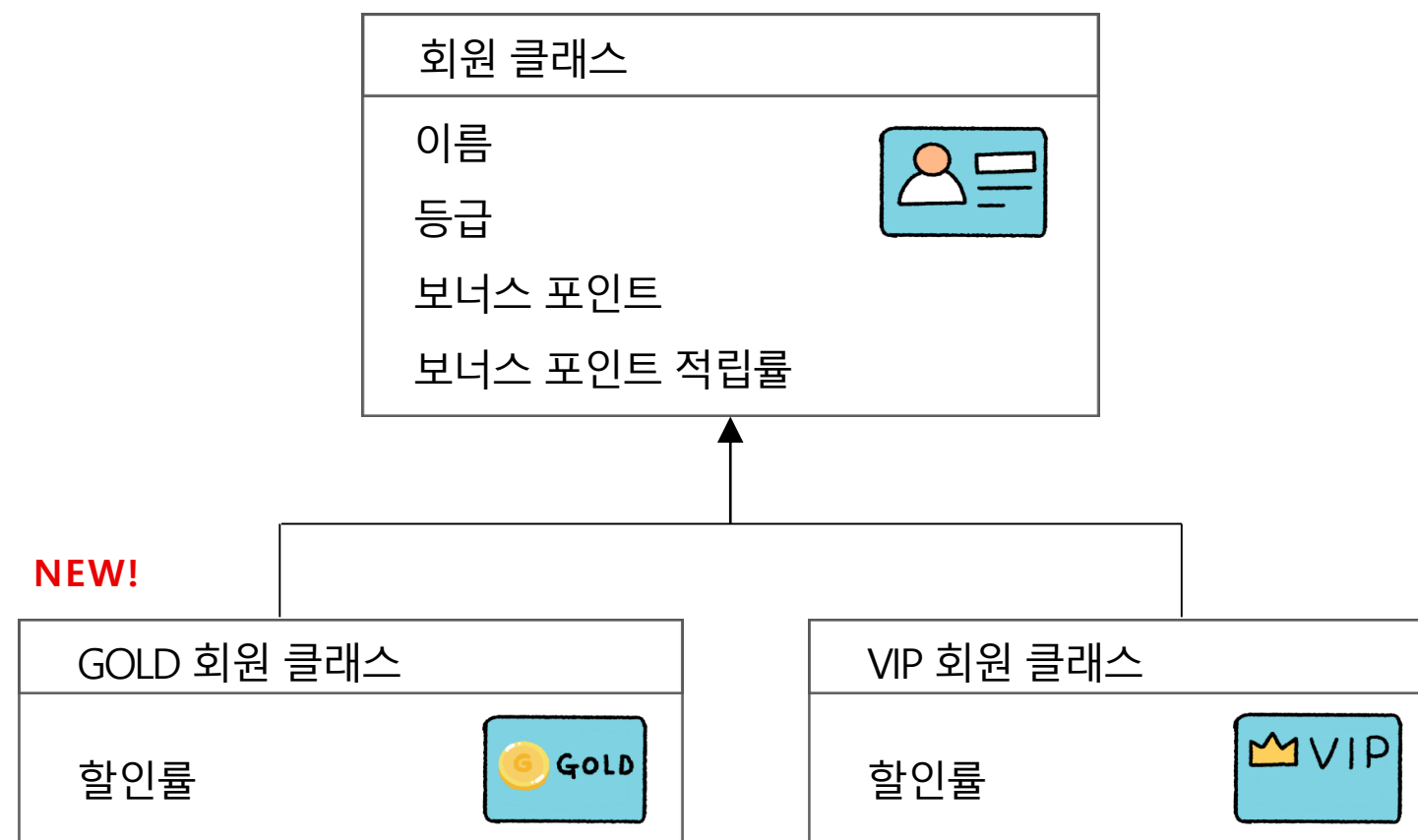
```
list.add(new VIPCustomer("고길동"));
```

쇼핑몰에 회원이 늘어 GOLD 회원 등급을 하나 추가합니다.

- 제품을 살 때 10%를 할인해준다.
- 보너스 포인트를 2% 적립해준다.

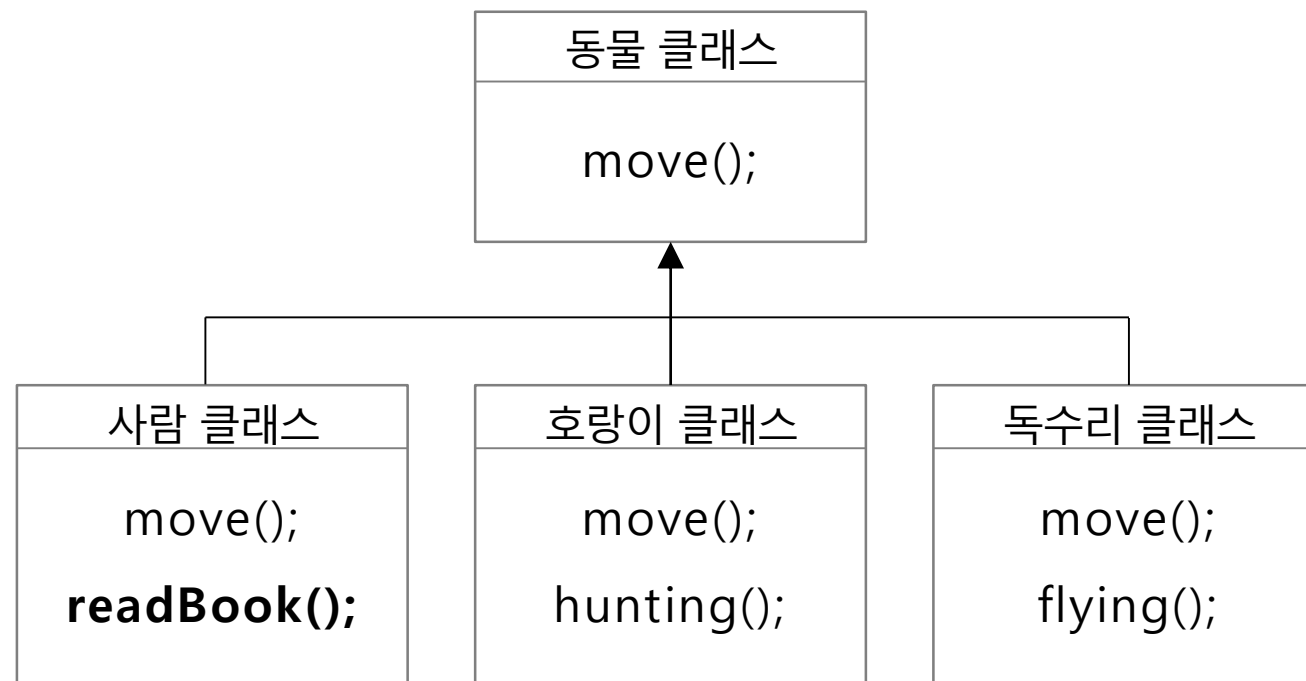
리스트를 생성하고 일반회원, 골드회원, VIP 회원을 1명씩 총 3명 저장하세요.

모든 회원은 10000원짜리 제품을 구매합니다.



다운캐스팅이란?

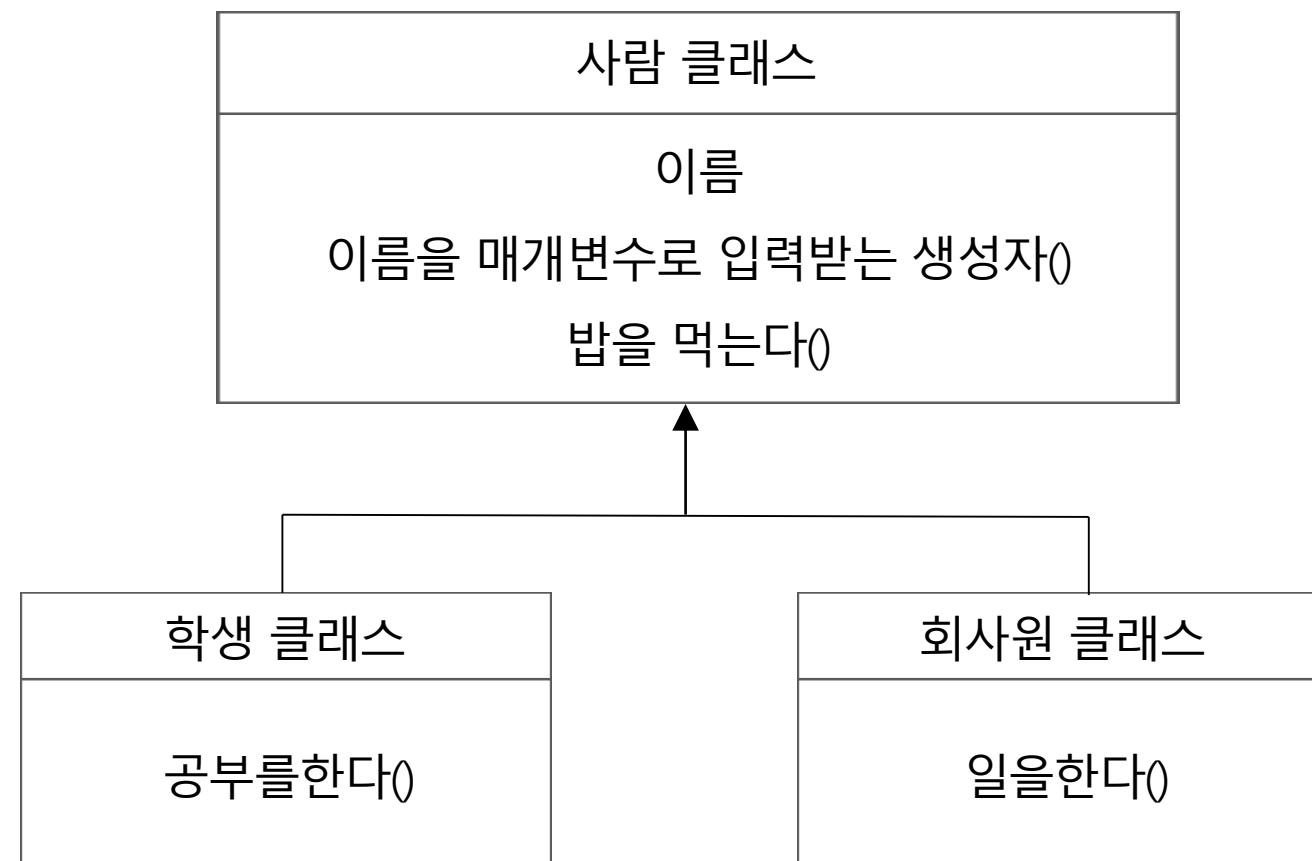
- 형변환된 변수를 다시 자식클래스로 변경하는 것을 '다운캐스팅'이라고 한다.
- 다운캐스팅 하기 전에 instanceof 연산자를 사용하여 형변환이 안전한지 확인해야 한다.



```
Animal ani = new Human() ;
ani.readBook();
```

```
if ( ani instanceof Human ) {
    Human human = (Human) ani;
    human.readBook();
}
```

아래 그림은 사람과 학생, 회사원 클래스의 상속관계를 표현한 것입니다.
 상속관계와 테스트 코드를 참고하여 다음과 같이 테스트 결과를 출력하세요.
 Person형으로 리스트를 생성하고 학생과 회사원을 1명씩 저장하세요.
 그리고 리스트에 저장된 학생과 회사원을 꺼내서 가지고 있는 모든 기능을 호출하세요.



테스트 코드

```

ArrayList<Person> list = new ArrayList<Person>();
list.add(new Student("둘리"));
list.add(new Employee("또치"));
... 생략
  
```

테스트 결과

```

<terminated> Quiz1 (22) [Java Application] C:\Users\Wimjiyeon\...
둘리가 밥을 먹는다
또치가 밥을 먹는다
둘리 학생이 공부를 한다
또치 회사원이 일을 한다
  
```