

제12장

제어자와 다형성



11. 매개변수의 다형성

- ▣ 참조타입 매개변수는 메서드 호출 시, **자신과 같은 타입이거나 또는 자손타입의 주소로 즉, 인스턴스**를 넘겨주도록 한다.

```
class Product {  
    int price; // 제품가격  
    int bonusPoint; // 보너스점수  
}  
  
class Tv extends Product {}  
class Computer extends Product {}  
class Audio extends Product {}  
  
class Buyer { // 물건사는 사람  
    int money = 1000; // 소유금액  
    int bonusPoint = 0; // 보너스점수  
}
```

```
Buyer b = new Buyer();  
  
Tv tv = new Tv();  
Computer com = new Computer();  
  
b.buy(tv);  
b.buy(com);
```

```
Product p1 = new Tv();  
Product p2 = new Computer();  
Product p3 = new Audio();
```

```
void buy(Tv t) {  
    money -= t.price;  
    bonusPoint += t.bonusPoint;  
}
```

```
void buy(Product p) {  
    money -= p.price;  
    bonusPoint += p.bonusPoint;  
}
```

(중요) 여기서, 매개변수(Product p)의 개념을 잡는 것이 중요하다. 자신과 같은 타입이나 혹은 자손타입을 참조할 수 있다는 의미인 것을 분명히 이해하고 넘어가자.

12. 여러 종류의 객체를 배열로 다루기

■ 조상타입의 배열에는 조상 뿐만 아니라, 자손들의 객체도 담을 수 있다.(다형성)

```
Product p1 = new Tv();  
Product p2 = new Computer();  
Product p3 = new Audio();
```

```
Product p[] = new Product[3];  
p[0] = new Tv();  
p[1] = new Computer();  
p[2] = new Audio();
```

```
class Buyer { // 물건사는 사람  
    int money = 1000; // 소유금액  
    int bonusPoint = 0; // 보너스점수  
  
    Product[] cart = new Product[10]; // 구입한 물건을 담을 배열  
  
    int i=0;  
  
    void buy(Product p) {  
        if(money < p.price) {  
            System.out.println("잔액부족");  
            return;  
        }  
  
        money -= p.price;  
        bonusPoint += p.bonusPoint;  
        cart[i++] = p;  
    }  
}
```

13. Vector활용하여 다형성 적용하기

- **Vector클래스(모든 종류의 객체를 저장할 수 있음.)**는 컬렉션 프레임워크의 **List계열입(JDK 1.2) - 배열 기반**

메서드 / 생성자	설 명
<code>Vector()</code>	10개의 객체를 저장할 수 있는 Vector인스턴스를 생성한다. 10개 이상의 인스턴스가 저장되면, 자동적으로 크기가 증가된다.
<code>boolean add(Object o)</code>	Vector에 객체를 추가한다. 추가에 성공하면 결과값으로 true, 실패하면 false를 반환한다.
<code>boolean remove(Object o)</code>	Vector에 저장되어 있는 객체를 제거한다. 제거에 성공하면 true, 실패하면 false를 반환한다.
<code>boolean isEmpty()</code>	Vector가 비어있는지 검사한다. 비어있으면 true, 비어있지 않으면 false를 반환한다.
<code>Object get(int index)</code>	지정된 위치(index)의 객체를 반환한다. 반환타입이 Object타입이므로 적절한 타입으로의 형변환이 필요하다.
<code>int size()</code>	Vector에 저장된 객체의 개수를 반환한다.

(중요)위의 메서드 중 매개변수 타입이 Object가 있다. 이것이 무슨 의미인가?

즉, Object는 모든 클래스의 조상이므로 어떠한 클래스가 매개변수로 들어와도 된다는 것을 의미한다.

14. Vector 활용하여 다형성 활용하기

```
Product[] cart = new Product[10];  
//...  
void buy(Product p) {  
    //...  
    cart[i++] = p;  
}
```



```
Vector cart = new Vector();  
//...  
void buy(Product p) {  
    //...  
    cart.add(p);  
}
```

메서드 / 생성자
Vector()
boolean add(Object o)
boolean remove(Object o)
boolean isEmpty()
Object get(int index)
int size()

감사합니다.

