

## 14. 메서드 중첩(overloading)

하나의 클래스 내에서 같은 이름을 가지는 메서드를 여러 개 중첩으로 정의할 수 있다. 중첩된 메서드들이 구별되기 위해서는 매개변수의 개수나 타입이 **달라야** 한다. 메서드 중첩은 OOP 언어에서 **다형성**을 제공한다.

---

```
class Overload
{
    String foo(){ return "인수가 없음"; }
    int foo(int a){ return a * a; }
    int foo(int a, int b){ return a * b; }
    int foo(int a, int b, int c){ return a * b * c; }
}
class OverloadDemo
{
    public static void main(String args[])
    {
        Overload g = new Overload();
        System.out.println(g.foo());           //실행 결과
        System.out.println(g.foo(5));         //인수가 없음
        System.out.println(g.foo(5));         //25
        System.out.println(g.foo(4, 5));      //20
        System.out.println(g.foo(2, 3, 4));  //24
    }
}
```

---

- 메서드 중첩(overloading) : 인수형, 인수 개수가 서로 달라야 한다.
- 메서드 중첩은 인수 이름은 같아도 상관없다.
- 따라서, 다음 두 구조는 같은 것이다.

`foo(int a, int b, float c) ≡ foo(int x, int y, float z)`

### [tip] 중첩(overloading)과 재정의(overriding)

- 중첩과 재정의는 동적바인딩으로 처리되므로 프로그램 실행 속도가 늦어지는 단점이 있다.
- 동적바인딩(dynamic binding)은 프로그램이 수행되는 도중에 어떤 결정이 이루어진다.

기출문제 분석

1. 다음 Java 프로그램을 수행시킨 후의 결과는? [2006년 국가 9급]

```
-----  
class A  
{  
    public void a1() { System.out.println("a1"); }  
    public void a2() { System.out.println("a2"); }  
    public void a3() { System.out.println("a3"); }  
    public void a4() { System.out.println("a4"); }  
}  
class B extends A{  
    public void b1(char c){ super.a1(); }  
    public void b1(int i){ this.b2(); }  
    public void b2(int i){ this.b3(); }  
    public void b2( ){ super.a3(); }  
    public void b3( ){ super.a4(); }  
}  
class Test{  
    public static void main(String args[]){  
        B b;  
        b = new B();  
        b.b1(3);  
    }  
}  
-----
```

- ① a1                      ② a2  
③ a3                      ④ a4

♣ 메서드 오버로딩(overloading, 중첩)

- b.b1(3);  
    ↓ int형 인수가 있는 b1() 메서드 호출  
b1(int i){ this.b2() };  
    ↓ 자신 클래스의 메서드 중 인수가 없는 b2() 메서드 호출  
b2( ){ super.a3(); }  
    ↓ 부모 클래스의 a3() 메서드 호출  
public void a3() { System.out.println("a3"); } //출력 : a3

정답 : ③

2. 자바 프로그래밍 언어에 대한 설명으로 옳은 것은? [2021년 지방 9급]

- ① 클래스에서 상속을 금지하는 키워드는 this이다.
- ② 인터페이스(interface)는 추상메서드를 포함할 수 없다.
- ③ 메서드 오버라이딩(overriding)은 상위클래스에 정의된 메서드와 하위클래스에서 재정의되는 메서드의 매개변수 개수와 자료형 등이 서로 다른 것을 의미한다.
- ④ 메서드 오버로딩(overloading)은 한 클래스 내에 동일한 이름의 메서드가 여러 개 있고 그 메서드들의 매개변수 개수 또는 자료형 등이 서로 다른 것을 의미한다.

☞ 자바 프로그래밍 언어

- ① 클래스에서 상속을 금지하는 키워드는 **this**이다.(x)  
→ **this** : 자신의 객체를 참조한다.(각 클래스의 모든 객체는 묵시적으로 this를 가진다)
- ② 인터페이스(interface)는 **추상메서드**를 포함할 수 없다.(x)  
→ 인터페이스는 상수와 **추상메서드**를 선언한 집합이다.
- ③ 메서드 오버라이딩(overriding)은 상위클래스에 정의된 메서드와 하위클래스에서 재정의되는 메서드의 **매개변수 개수와 자료형 등이 서로 다른 것을 의미한다.**(x)  
→ 오버라이딩은 **메서드 이름, 매개변수 개수 및 자료형, 반환형**이 같아야 한다.

- Overloading : 동일한 클래스 내에서 같은 이름의 메서드를 정의하여 다형성 지원
- Overriding : 상속 관계의 클래스에서 같은 이름의 메서드를 정의하여 다형성 지원

```
class Overload { //메서드 오버로딩
    int foo(int a){ return a * a; }
    int foo(int a, int b){ return a * b; }
    int foo(int a, int b, int c){ return a * b * c; }
}
class OverloadDemo {
    public static void main(String args[]) {
        Overload g = new Overload();
        //실행 결과
        System.out.println(g.foo(5)); //25
        System.out.println(g.foo(4, 5)); //20
        System.out.println(g.foo(2, 3, 4)); //24
    }
}
```