오버로딩과 오버라이딩의 차이

1. **오버 로딩 (Overloading)**

* 메서드 오버 로딩과 생성자 오버로딩이 있는데 둘다 같은 개념이다.
* 같은 이름 함수를 여러 개 정의 하고, 매개변수의 유형과 개수를 다르게 하여 다양한 유형의 호출에 응답하게 한다.
* 같은 클래스 내의 같은 이름의 메서드를 사용하는 것

1. **오버 라이딩(Overriding)**

* 상속 관계에 있는 클래스 간에 같은 이름의 메서드를 정의하는 기술
* 부모클래스가 정의한 것을 자식클래스가 재정의 하는 것

1. **오버 로딩(Overloading)과 오버 라이딩(Overriding) 성립조건**

|  |  |  |
| --- | --- | --- |
| 구분 | 오버로딩 | 오버 라이딩 |
| 메서드 이름 | 동일 | 동일 |
| 매개변수, 타입 | 다름 | 동일 |
| 리턴 타입 | 상관 없음 | 동일 |

인터페이스

인터페이스는 추상 메서드와 상수로만 구성된 특수 형태의 클래스로 다중 상속이 가능하지만 객체 생성이 불가능하다..

구현 객체의 같은 동작을 보장하기 위한 목적

일반 클래스의 단일 상속 단점을 보완하고 프로그램 모듈 연결하는 이음매 역할

Class 대신 **interface**를 써서 정의하고 상속을 받을 시 **implements**를 사용한다.

상속인 **extends**도 사용가능, 인터페이스 멤버는 반드시 public이다.

하나의 인터페이스가 여러 개의 인터페이스 구현 가능

Ex2) class C implements A,B{

public int a(){

return 1;

}

}

Ex1) public interface A{

Public int a(){}

}

public interface B{

Public int b(){}

}

모든 멤버 필드는 public static final 처리로 생략 가능하고, 모든 메서드는 추상 메서드로 public abstract는 생략 가능하다.