

## # this

객체(인스턴스) 자기자신을 의미

지역변수와 전역변수를 구별하는 기능도 있다.

하나의 클래스 필드에서 여러 객체를 다루어야 하는데, 만약 여러개의 객체가 접근한다면 어떤 객체가 필드에 접근하는지 알 수 없다.

그래서 this를 통해 필드에 접근한 객체가 누구인지 알려주고, 접근한 객체가 가지고 있는 필드의 주소값을 this에 자동으로 저장된다.

## # 변수의 종류

- 지역변수 : 메소드 영역에 선언된 변수
- 인스턴스 변수(전역변수) : 클래스 영역에 선언되는 변수
- 클래스 변수(정적변수) : 클래스 영역에 선언되는 static이 붙은 변수

클래스 영역: class 블록 { 시작부터 } 끝 / 메소드 영역: method 블록 { 시작부터 } 끝

	지역변수 (lv)	인스턴스 변수 (iv) (전역변수)	클래스 변수 (cv) (정적변수)
초기화	수동	자동	자동
생성시기	변수 선언문이 수행 됐을 때	객체가 생성됐을 때	클래스가 메모리에 올라갔을 때
생명주기	} (닫는 중괄호)	프로그램 종료 시	프로그램 종료 시

1. 지역변수는 메소드가 호출되어 작업하는 동안에만 사용된다.
2. 정적변수는 메모리에 자동으로 올라가기 때문에 객체를 생성할 필요가 없다.
3. 전역변수와 정적변수는 자동으로 초기화되는데, 자료형의 타입에 따라 초기화되는 값이 다르다.

## # 메소드 종류

### - 인스턴스 메소드

객체생성 후, 참조변수를 통해 메소드를 호출한다.

### - static 메소드

객체 생성이 필요없기 때문에 클래스 이름을 통해 호출 가능하다.

static메소드 내에서는 인스턴스 멤버들을 사용할 수 없다. (인스턴스 멤버들은 객체생성을 해야 사용가능)

클래스 변수와 지역변수만 사용가능

## #static

static은 인스턴스멤버에 붙여서 사용할 수 있다.

※ 언제 static을 붙이면 될까?

변수의 경우, 객체가 공통적으로 공유해야 하는 속성에 static을 붙여서 사용.

메소드의 경우, 인스턴스 멤버들을 사용하지 않을 때 static을 붙여서 사용.

지역변수를 통해 작업을 하기 때문에 인스턴스 멤버가 필요없다.

Q1. static메소드에서 인스턴스 변수 사용 가능? no

Q2. 인스턴스 메소드에서 static 변수 사용 가능? yes

Q3. static 메소드에서 인스턴스 메소드 사용 가능? no