객체지향 프로그래밍

* 클래스 선언
  + 파일마다 1개 클래스를 선언하지만, 2개 이상의 클래스를 1개 파일로 선언 가능하다.
  + 하나의 파일에 클래스가 둘 이상 있다면 하나만 public으로 선언할 수 있고, 해당 클래스 이름은 소스 파일 이름과 동일해야 한다.
  + 클래스 이름은 한글이든 영어든 상관없지만, 한글로 클래스 이름을 만드는 경우는 거의 없다.
  + 통상적으로 클래스의 첫 글자는 대문자이다.

클래스를 선언한 다음, 컴파일을 했다면 객체를 생성할 설계도가 만들어진 샘입니다. 클래스로부터 객체를 생성하려면 다음과 같이 new 연산자를 사용하면 됩니다.

new 클래스();

new는 클래스로부터 객체를 생성시키는 연산자입니다. New 연산자 뒤에는 생성자가 오는데, 생성자는 클래스() 형태를 가지고 있습니다. new연산자로 생성된 객체는 메모리 힙 영역에 생성됩니다.

클래스의 구성 멤버

클래스에는 객체가 가져야 할 구성 멤버가 선언됩니다. 구성 멤버에는 필드, 생성자, 메소드가 있습니다. 이 구성 멤버들은 생략되거나 복수의 개수로 작성될 수 있습니다.

필드: 객체의 데이터가 저장되는 곳

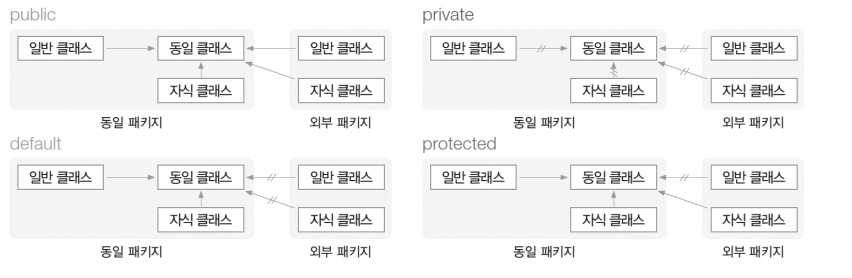
생성자: 객체 생성 시 초기화 역할 담당

메소드: 객체의 동작에 해당하는 실행 블록

접근 제한자

* 접근 제한자의 종류
  + Public: 모든 클래스의 접근을 허용
  + Protected: 동일 패키지 또는 상속 관계가 있는 클래스의 접근을 허용
  + Default: 동일 패키지에 존재하는 클래스의 접근 허용
  + Private: 모든 클래스의 접근을 거부

접근 제한자의 허용 범위

접근자와 설정자

필요성

: 클래스 내부에 캡슐화된 멤버를 외부에서 사용할 필요

(즉, private으로 지정된 필드는 클래스 내부에서만 사용할 수 있음)->이를 위해 접근자와 설정자 사용

값을 반환하는 접근자

:private필드를 외부로 반환하는 public 메소드

Get + 필드명의 형태로 작성, 반환 타입은 대상 필드왁 같음

Public 선언 -> 클래스 밖의 모든 코드에서 호출 가능

값을 변경하는 설정자

: private 필드를 외부에서 변경하게 하는 public메소드

Set + 필드명의 형태로 작성, 반환 타입은 void

Public 선언 -> 클래스 밖의 모든 코드에서 호출 가능

생성자

생성자의 역할: 객체를 생성하는 시점에서 필드를 다양하게 초기화

디폴트 생성자

: 모든 클래스는 최소한 하나의 생성자가 있음

만약 생성자를 선언하지 않으면 컴파일러가 자동으로 디폴트 생성자를 추가

인스턴스 멤버와 정적 멤버

Java는 static 키워드로 클래스의 필드를 공유할 수 있도록 지원

인스턴스 멤버와 정적 멤버

인스턴스 변수는 객체별로 관리

: 객체를 생성할 때 인스턴스 변수도 생성, 객체가 소멸될 때는 자동으로 소멸

정적 멤버의 활용

상수는 변경되지 않는 변수이기 떄문에 final 키워드로 지정

상수는 값이 변경되지 않으므로 객체마다 따로 기억 공간을 할당할 필요가 없음