추상 클래스

추상 클래스란?

* 추상적이라는 것은 구체이지 않고 막연한 것을 뜻합니다.
* 어떤 클래스가 추상적이다라는 말은 구체적인 않은 클래스 라는 뜻 입니다.
* 추상 클래스를 영어로 표현하면 abstract class이고, 추상 클래스가 아닌 클래스는 concrete class라고 합니다. 지금까지 클래스는 모두 concrete class 였습니다.

추상클래스 문법

* 추상 클래스는 항상 추상 메서드를 포함 합니다.
* 추상 메서드는 {} 로 감싼 부분인 함수의 구현 코드가 없습니다.
* 함수의 구현코드가 없다는 것은 함수 몸체(body)가 없다는 뜻 입니다.

// {} 안의 내용이 함수의 몸체 (구체적인(구현된) 메서드) int add(int x, int y) { return x + y; }

* 이 부분이 없는 함수는 추상 함수(abstract function)이고 자바에서는 추상 메서드(abstract method)라고 합니다.
* 추상 메서드 예약어 : abstract
* {}(구현부)대신 ; 를 씁니다.

abstract int add(int x, int y);

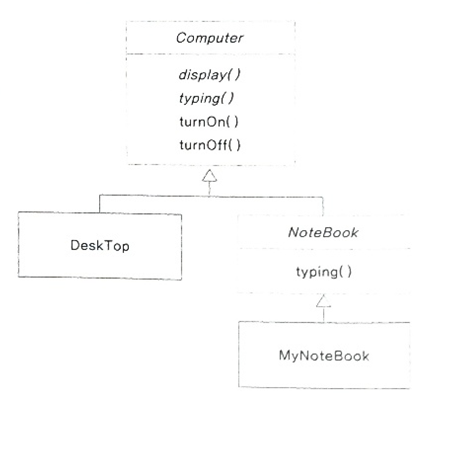
* \*\* 자바에서 추상메서드는 abstract 예약어를 사용하여 선언만 하는 메서드 입니다.\*\*

메서드 선언의 의미

int add(int num1, int num1);

* 상기 코드처럼 선언한 메서드를 보면 두 개의 정수를 입력받은 후 더해서 그 결과 값을 반환한다는 것을 유추할 수 있습니다.
* 즉, 이 메서드의 선언부(declatration)만 봐도 어떤 일을 하는 메서드인지 알 수 있습니다. 함수의 선언부 즉, 반환 값, 함수 이름, 매개변수를 정의한다는 것은 곧 함수의 역할이 무엇인지, 어떻게 구현해야 하는지를 정의한다는 뜻힙니다.
* 따라서 함수 몸체를 구현하는 것보다 중요한 것은 함수 선언부를 작성하는 것입니다.
* 자바에서 사용하는 메서드 역시 마찬가지로 메서드를 선언한다는 것은 메서드가 해야 할 일을 명시해 두는 것입니다.

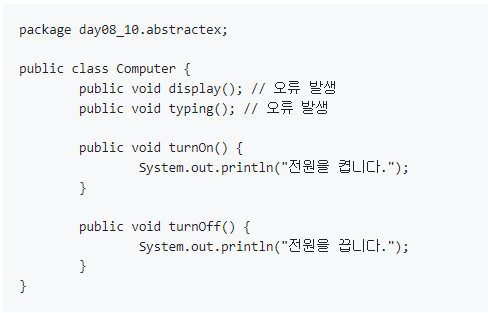
추상 클래스 구현하기

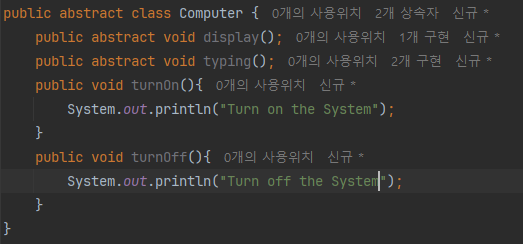
[[](https://raw.githubusercontent.com/yonggyo1125/curriculum300H/main/1.JAVA(84%EC%8B%9C%EA%B0%84)/8~10%EC%9D%BC%EC%B0%A8(9h%20-%20%EA%B0%9D%EC%B2%B4%EC%A7%80%ED%96%A5%20%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D2/images/%EC%B6%94%EC%83%81%ED%81%B4%EB%9E%98%EC%8A%A41.png)](https://raw.githubusercontent.com/yonggyo1125/curriculum300H/main/1.JAVA(84%EC%8B%9C%EA%B0%84)/8~10%EC%9D%BC%EC%B0%A8(9h%20-%20%EA%B0%9D%EC%B2%B4%EC%A7%80%ED%96%A5%20%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D2/images/%EC%B6%94%EC%83%81%ED%81%B4%EB%9E%98%EC%8A%A41.png)

>추상 클래스, 추상 메서드는 기울임꼴로 표시

* Computer 클래스는 추상클래스 입니다.
* 컴퓨터 종류에는 데스크톱과 노트북이 있습니다. 그리고 노트북의 종류에는 MyNoteBook이 있습니다.
* Computer 클래스는 추상 클래스이며 이를 상속받은 두 클래스 중 DeskTop 클래스는 일반 클래스이고 NoteBook 클래스는 추상 클래스입니다. 마지막으로 NoteBook 클래스를 상속받은 MyNoteBook 클래스도 일반 클래스 입니다.
* display()와 typing()은 추상 메서드이고 turnOn()과 turnOff()는 구현코드가 있는 메서드입니다.

day08\_10/abstractex/Computer.java



* Computer 클래스 내부에 추상 메서드 display()와 typing()을 선언하고, 구현 메서드 turnOn()과 turnOff()를 작성합니다.
* 그러면 완전하게 구현되지 않은 두 추상메서드에서 오류가 발생합니다.
* display()나 typing() 위에 마우스를 올리면 오류를 해결할 수 있는 방법으로 다음 두가지를 제시합니다.

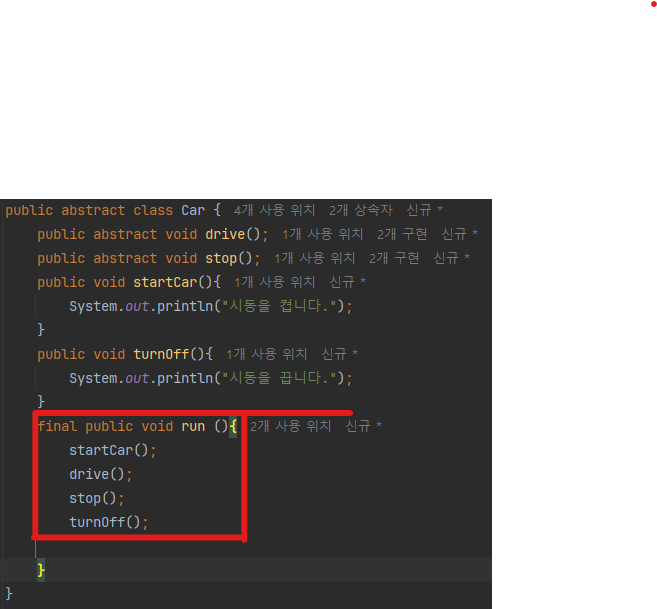
추상 클래스를 만드는 이유

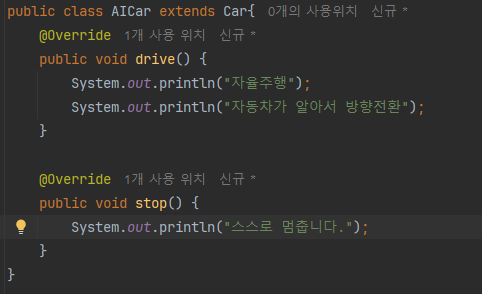
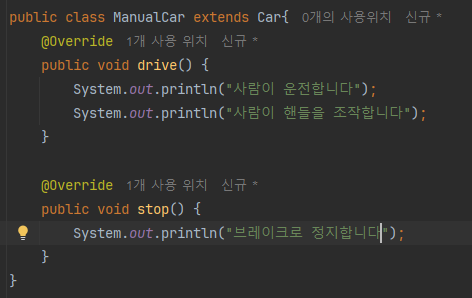
1. 추상 클래스는 인스턴스로 생성할 수 없다.
   * 추상 클래스는 모든 메서드가 구현되지 않았으므로 인스턴스로 생성할 수 없습니다. 하지만 추상 클래스에서도 형변환을 사용 할 수는 있다.
2. 추상 클래스에서 구현하는 메서드
   * 생성할 수 없는 추상 클래스는 상속을 하기 위해 만든 클래스입니다.
   * 추상클래스에는 추상메서드와 구현된 메서드가 함께 사용 될 수 있습니다.
   * 구현된 메서드는 하위 클래스에서도 사용할 즉, 하위 클래스에서도 구현 내용을 공유할 메서드를 구현합니다.
   * 실제 하위 클래스에서 내용을 각각 다르게 구현해야 한다면, 구현 내용을 추상 메서드로 남겨 두고 하위 클래스에 구현을 위임하는 것입니다.
   * 구현된 메서드 : 하위 클래스에서 공통으로 사용할 구현 코드, 하위 클래스에서 재정의할 수도 있음.
   * 추상 메서드 : 하위 클래스가 어떤 클래스냐에 따라 구현 코드가 달라짐

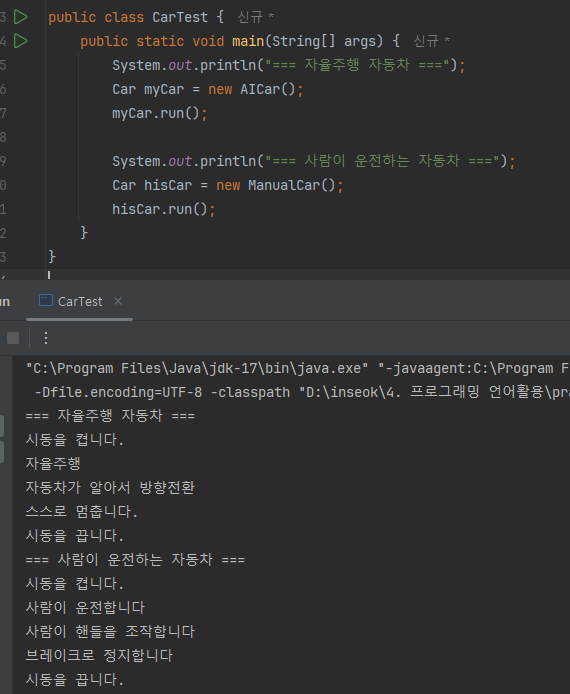
추상클래스와 다형성

* 상위 클래스인 추상 클래스는 하위에 구현된 여러 클래스를 하나의 자료형(상위 클래스 자료형)으로 선언하거나 대입할 수 있습니다.
* 추상 클래스에 선언된 메서드를 호출하면 가상 메서드에 의해 각 클래스에 구현된 기능이 호출됩니다.
* 즉, 하나의 코드가 다양한 자료형을 대상으로 동작하는 다형성을 활용할 수 있습니다.

템플릿 메서드 패턴

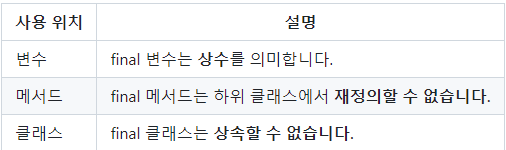




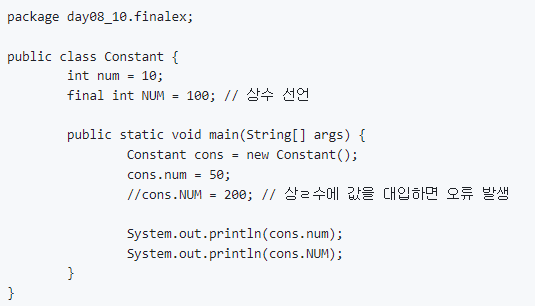


final 예약어

* final은 마지막이라는 의미입니다.
* 즉, 마지막으로 정한 것이니 더 이상 수정할 수 없다는 뜻입니다.
* 자바 프로그램에서는 final 예약어는 변수, 메서드, 클래스에 사용할 수 있습니다.

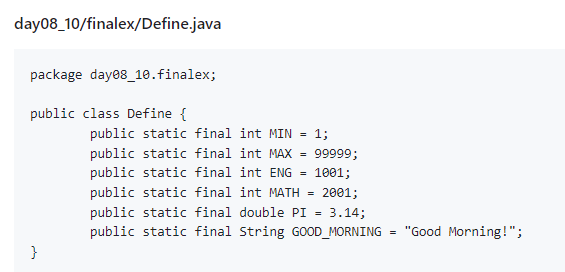


상수를 의미하는 final 변수



여러 자바 파일에서 공유하는 상수 값 정의하기

* 하나의 자바 파일에서만 사용하는 상수 값은 해당 파일 안에서 정의해서 사용할 수 있습니다. 그러나 프로젝트를 하다 보면 여러 파일에서 똑같이 공유해야 하는 상수 값도 있습니다.
* 자바로 프로젝트를 진행할 때 여러 파일에서 공유해 하는 상수 값은 한 파일에 모아 public static final로 선언하여 사용할 수 있습니다.



상속할 수 없는 final 클래스

클래스를 final로 선언하면 상속할 수 없습니다.

재정의 할 수 없은 final 메서드

메서드를 final로 선언하면 하위클래스에서 재정의 할 수 없습니다.