

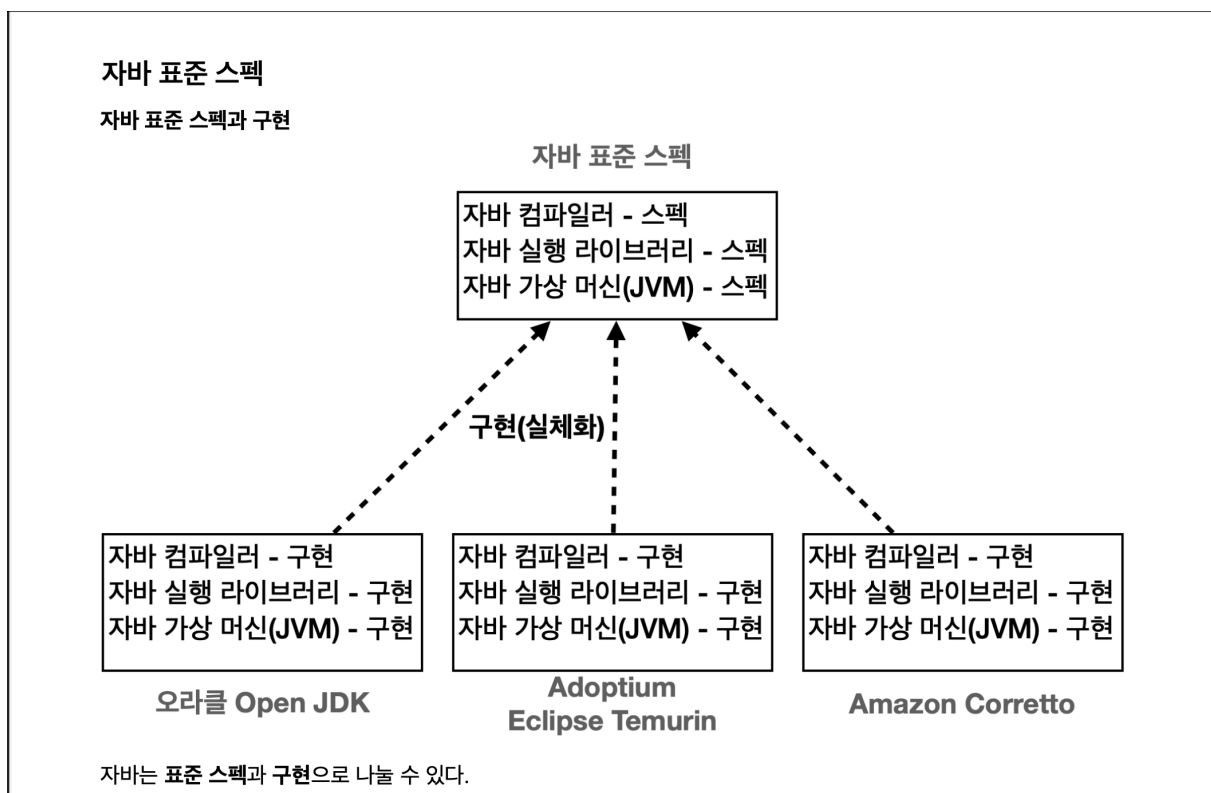
01-08 [Java]

👤 소유자	종수 김
☰ 태그	

Basic Java

자바 ?

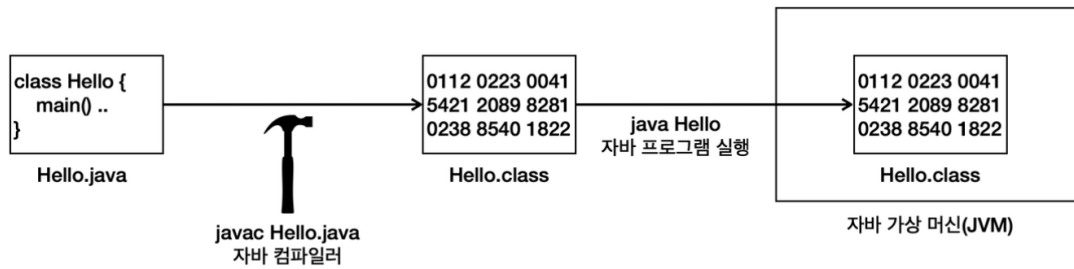
자바 표준 스펙



- 문서화 된 자바 표준 스펙을 기반으로 자바를 구현
- 구현된 자바는 모두 표준 스펙을 기반으로 구현되었으므로, 후에 다른 Java로 변경하더라도 문제 없이 잘 동작

컴파일과 실행

컴파일과 실행



자바 프로그램은 컴파일과 실행 단계를 거친다.

- Hello.java와 같은 자바 소스 코드를 개발자가 작성한다.
- 자바 컴파일러를 사용해서 소스 코드를 컴파일 한다.
 - 자바가 제공하는 javac 라는 프로그램을 사용한다.
 - .java → .class 파일이 생성된다.
 - 자바 소스 코드를 바이트코드로 변환하며 자바 가상 머신에서 더 빠르게 실행될 수 있게 최적화하고 문법 오류도 검출한다.
- 자바 프로그램을 실행한다.
 - 자바가 제공하는 java 라는 프로그램을 사용한다.
 - 자바 가상 머신(JVM)이 실행되면서 프로그램이 작동한다.

1. 코드작성

2. Java Compiler(Javac)를 통해 바이트 코드로 이루어진 .class파일로 변환

3. JVM은 바이트 코드로 이루어진 .class파일을 Class Loader를 이용해서 로드


4. JVM에 의해서 바이트 코드를 OS가 이해할 수 있는 기계어로 해석

5. Java Application 실행

▼ 정리

자바 컴파일과 실행순서

자바 컴파일과 실행과정을 설명

 <https://velog.io/@minbo2002/자바-컴파일과-실행과정>



- 자바 소스 코드를 바이트코드로 변환하며 자바 가상 머신에서 더 빠르게 실행될 수 있게 최적화.
- JVM에 의해 Java는 운영체제로부터 독립적
JVM은 운영체제에 종속적

public static void main(String [] args)

- 자바 애플리케이션이 시작하면 자바는 Main 메서드를 찾는다.
해당 메서드가 프로그램의 시작인 Entry Point

주석

- 소스코드가 복잡할 때, 소스 코드에 대한 설명을 추가
- 이는 컴파일러에 의해 해석되지 않는다.
- `//` : 한줄 주석
- `/* */` : 여러줄 주석

변수

변할 수 있는 수 / 값을 담아두는 공간
사용자의 입력 값, 변할 수 있는 값 등을 저장

`int a = 10;`

=는 equals의 의미가 아닌 대입을 의미.

이처럼 선언한 변수에 값을 대입하는 것을 '변수 초기화' 라고 부름.

변수를 선언하고 초기화 하지 않는다면? primitive 타입이더라도 초기화 하지 않으면 컴파일 에러,,,,

- 사용하지 않는 변수는 컴파일러가 최적화를 통해 삭제 (메모리만 먹을 뿐이니까,)
- 자바는 실행 시점에 변수의 값을 읽어서 사용
- 변수를 선언하면 컴퓨터 메모리 공간에 확보해서 그곳에 데이터를 저장
변수의 이름을 통해 메모리 공간에 접근
- 메모리는 여러 시스템이 함께 사용하는 공간이므로 어떠한 값들이 계속 저장.
변수를 선언하면 메모리 상의 어떤 공간을 차지하고 사용하는데, 그 공간에 기존에 어떤 값이 있었는지 모르므로, 초기화 하지 않으면 이상한 값이 출력될 수 있음. 때문에 자바는 초기화를 컴파일 오류로 강제
- 지역 변수는 개발자가 초기화 해주는 것이 필수 (일부 변수는 Java가 초기화해줌)

변수 타입

`int` : 정수

- `byte` : -128 ~ 127
- `short` : -32768 ~ 32767
- `int` : - 2,147,483,648 ~ 2,147,483,647

- long : -9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
L을 통해 명시적으로 Long임을 명시
- 자바는 기본적으로 4byte를 효율적으로 계산하도록 설계되어있음.
- 리터럴은 기본이 int이기 때문에 Long은 L

double : 실수

- float : f를 통해 명시적으로 float임을 명시
- double : float 보다 큼 / 정밀도가 높음 / 실수는 기본 double 사용

boolean : 참 or 거짓

char : 문자 하나를 다룰 때 사용

String : 문자열

- String은 기본형처럼 사용할 수 있게 만든 문자열을 다루는 클래스타입.

리터럴 : 코드에서 개발자가 직접 적어넣은 고정된 문자 / 글자 Ex) int a = 100;

- 변수에 크기에 따라 적절한 변수 타입을 통해 메모리 절약 가능
 - 변수 타입마다 차지하는 메모리 크기가 다르기 때문

변수 명명 규칙

규칙 : 필수 (컴파일 오류 발생)

- 숫자로 시작할 수 없음
- 공백이 들어갈 수 없음
- 예약어를 변수 이름으로 사용할 수 없음
- 영문자, 숫자, 달러 기호, 밑줄 만 사용 가능

관례 : 필수는 아니지만 사실상 규칙

- 소문자로 시작하는 낙타표기법
 - camelCase
- Class만 첫 문자를 대문자.
- 상수는 대문자 + 언더바
 - USER_LIMIT
- 패키지는 모두 소문자

- 변수 이름은 의미있고, 용도를 명확하게 설명.