


22. 08. 31 - 프로그래밍 언어 발전 패러다임

용어 정리

JAVA 메모리 구조

자바 교실

 https://cafe.naver.com/jjdev?iframe_url=/ArticleList.nhn%3Fsearch.clubid=25331457%26search.menuid=296%26search.boardtype=L



궁금한게 있을 땐
카페 스마트봇

- 모든 언어는 CPU에 의해 실행된다.
- CPU는 0과 1만 인식할 수 있다.
- 고급 언어는 번역기가 필요하다.

「 프로그래밍 언어 발전 패러다임 」

사람의 편리성의 관점 (저급 언어 → 고급 언어)

저급언어	→	고급언어	
	(자연어에 가까워짐)		
기계어, 어셈블리어		기계어, 어셈블리어를 제외한 언어	
번역기 필요 없음		번역기 필요함	1. Compiler
			2. Interpreter

재사용의 관점 (비구조적 언어 → 구조적 언어 → 객체지향적 언어)

비구조적 언어	→	구조적 언어	→	객체지향적 언어
---------	---	--------	---	----------

스파게티소스 코드, 흐름 제어 x, 재사용률 x	(흐름 제어문, 함수의 개념이 추가됨)		(객체간의 독립성의 개념이 추가됨)	
저급언어		1. 함수 단위 2. 범위 외에 함수를 불러올 수 없다 → 재사용률이 낮다		객체 단위
		재사용 가능		구조적 언어를 기반으로 두고 있음

※ JAVA 는 Only 객체 지향적

언어로 탄생

객체지향언어는 구조적언어를 품고 있다.

함수의 한계



하나의 함수를 사용할 때 다른 함수나 변수와 종속적인 경우가 있다.

함수의 한계를 극복하기 위한 OOP 개발!!

구조적언어의 프로그램 작성단위 = 함수 단위로 프로그램 작성

OOP 언어의 프로그램 작성 단위 = 객체 단위로 프로그램 작성

자바는 처음부터 객체지향적 언어를 모티브로 삼아서 설계 되어있기 때문에 OOP 특징이 그대로 있다.

JAVA

인터프리터 언어이다 (Interpreted)



자바는 정확하게 말하면 컴파일 언어인 동시에 인터프리터 언어이다. 자바는 먼저 텍스트 소스를 컴파일하여 2진 파일(클래스 파일)로 만든 다음 자바 런타임이 클래스 파일을 인터프리트하면서 실행한다. 먼저 시스템에 무관한 2진 파일을 만듦으로써 자바는 컴파일 언어에 가까운 속도와 시스템 독립성을 동시에 얻을 수 있었다.

플랫폼 독립적(OS) 프로그램 기법 제공



JVM (Java Virtual Machine) 을 이용하여 OS에 관계없이 개발이 가능하다.

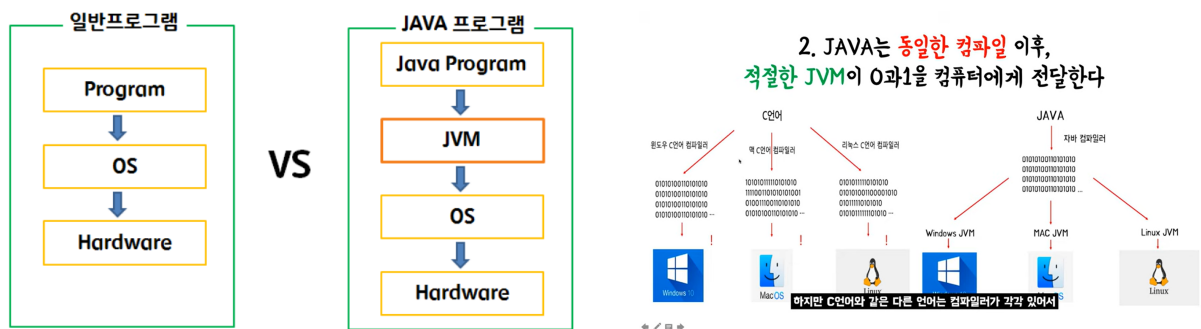
⇒ 자연어 → 바이트 코드(중간 코드)로 변환 → 바이트 코드를 기계어로 번역 해 줌

⇒ 자바와 운영체제 사이에서 중개자 역할을 수행하며, 자바가 운영체제에 구애 받지 않고 프로그램을 실행할 수 있도록 도와준다.

⇒ 스택 기반의 가상 머신

- OS마다 커널이 달라 OS별로 코드를 작성해야 함
→ **운영체제에 독립적**
- Oracle 사에서는 JVM(Java Virtual Machine)을 OS별로 제공해줌
→ **플랫폼 독립적**, 이식성이 좋다.

※ 가상머신이란 프로그램을 실행하기 위해 물리적 머신과 유사한 머신을 소프트웨어로 구현한 것

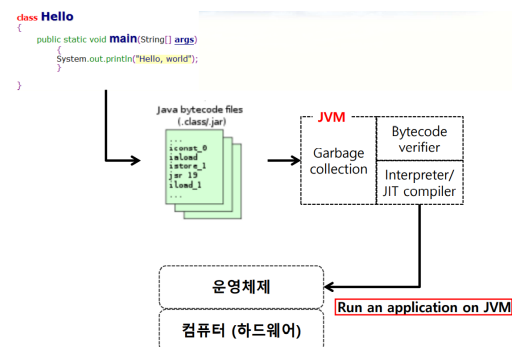


JVM의 단점



실행 속도가 느리다.

- 컴파일한 기계어 파일을 이용하는 것이 아니다.
- JVM을 통해 그때 그때 ByteCode 를 실행 시킨다.





보안에 취약하다.

- ByteCode에는 특정 패턴이 있다.
- ByteCode의 패턴을 분석해 소스 코드를 복원할 수 있다.

java프로그램은 **JVM(Java Virtual Machine : 자바가상머신)**이라는 프로그램만 있으면 실행이 가능한데, **JVM이 OS 로 부터 메모리 사용 권한을 할당 받고 JVM이 자바 프로그램을 호출하여 실행하게 된다.** OS에서는 독립되었지만 JVM이라는 프로그램에 종속적이게 된다. (JVM을 실행시키고 다시 JVM이 프로그램을 실행시키는 방식이다 보니 OS에 직접 제어 받는 방식보다는 속도면에서는 느리다는 단점을 가진다)

자바 단점 :

1. 실행할때마다 기계어로 한줄한줄 읽어가며 번역해서 느리다.
2. byte 코드의 특성을 사람이 알 수있기 때문에 보안에 취약하다.

자동 메모리 관리(**Garbage Collection**) 이터 , 가치가 없는[쓸모없는] 것

Garbage = 불필요한 데이터



프로그래머가 관리하던 메모리를 자동으로 관리한다. → 더 효율적인 관리 가능
어플리케이션이 생성한 객체의 생존 여부를 판단하여, 더이상 참조되지 않거나 null인 객체의 메모리를 해체 시켜 메모리 반납을 한다.

컴파일

코드를 jvm이 이해 할 수 있는 byte code 로 전환한다

객체를 기반으로 만들어서 객체지향적 언어라고 한다. oop

구조적 언어의 프로그램 작성 단위 = 함수 단위로 프로그램 작성

OOP 언어의 프로그램 작성 단위 = 객체 단위로 프로그램 작성

자바는 처음부터 객체지향적 언어를 모티브로 삼아서 설계 되어있기 때문에 OOP 특징이 그대로 있다.

JDK (Java Development Kit)



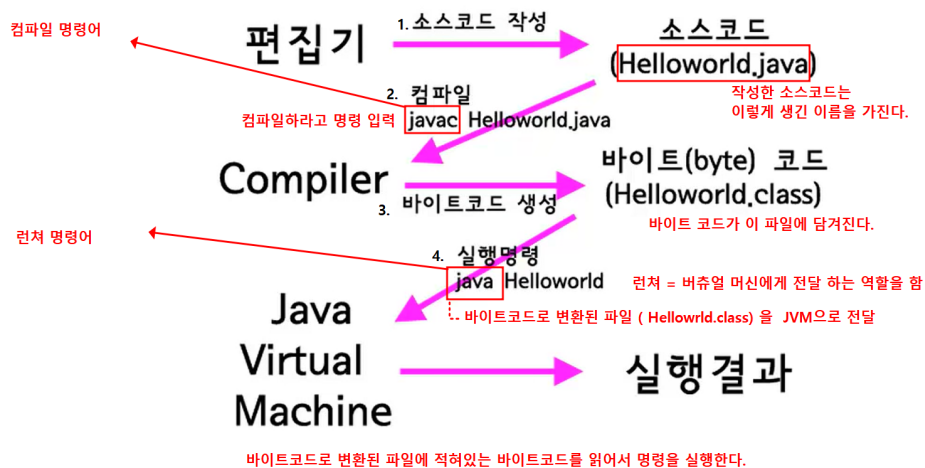
JRE + Compiler + .. (개발에 필요한 소프트웨어)

JRE(Java Runtime Environment) : JVM을 다운로드 함 → 클라이언트용

Java SE, JDK, JRE 차이점

JVMが動作するフロー

JAVA 실행 과정



하나의 패키지 형태로 묶어서 나타내는게 JRE +A 로 컴파일 등의 기능을 담고 있다.