

Java 언어 소개

Java 언어의 탄생

- 1991년 제임스 고슬링 주축인 그린 프로젝트로 시작
- TV 셋톱박스 등의 다양한 종류의 기기에서 동작하는 언어를 지향
- 1995년 Sun Microsystems에서 Java 1.0 공식 발표
- 2021 현재 Java 18까지 출시 됨



Java 언어의 특징 1

- **WORA – Write Once Run Anywhere**
- **Java 이전 C, C++은 하드웨어에 의존적**
 - 다른 종류의 하드웨어에서 실행하기 위해서는 재 컴파일 필요
- **Java는 한번 작성 및 컴파일 된 결과물이 어디서든 동작 가능**
 - JVM(Java Virtual Machine)이 핵심!

Java 언어의 특징 2

- 객체 지향 언어
- Java 코드는 class를 기본 단위로 구성 됨
- 상속, 다형성, 캡슐화 등의 객체지향적 요소를 언어적으로 지원

Java 언어의 특징 3

- 자동화 된 메모리 관리
- 프로그램은 컴퓨터 메모리를 사용
 - 메모리 자원은 한정적이며, 효율적인 관리가 필수
- JVM의 Garbage Collector가 사용되지 않는 메모리 자원을 회수
 - 개발자가 직접 메모리 관리를 하지 않아도 됨

Java 언어 적용 분야

- Java 언어는 다양한 분야에 적용 될 수 있음

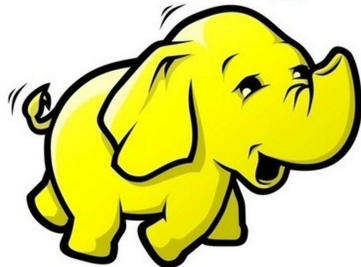


android



spring

hadoop



elasticsearch

Java 언어 적용 분야

- 모바일 애플리케이션 개발
- 웹 애플리케이션 개발
- 빅데이터 솔루션 개발
- 데이터베이스 개발
- 데스크탑 GUI 애플리케이션 개발
- 빅데이터 분석, 머신러닝
- 게임 개발
- 임베디드 SW 개발

Point!

- Write Once, Run Anywhere
- Java는 객체 지향 언어
- JVM의 Garbage Collector가 메모리를 자동 관리

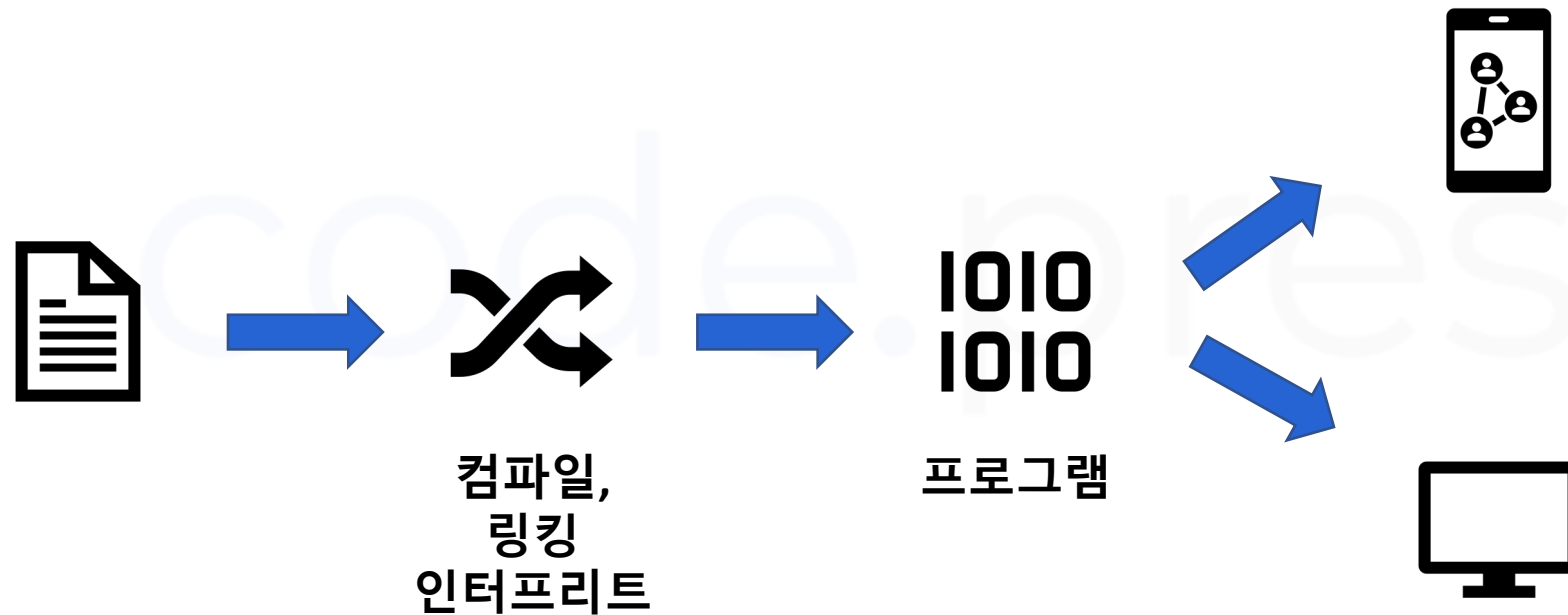
Java 주요 용어

Java 주요 용어

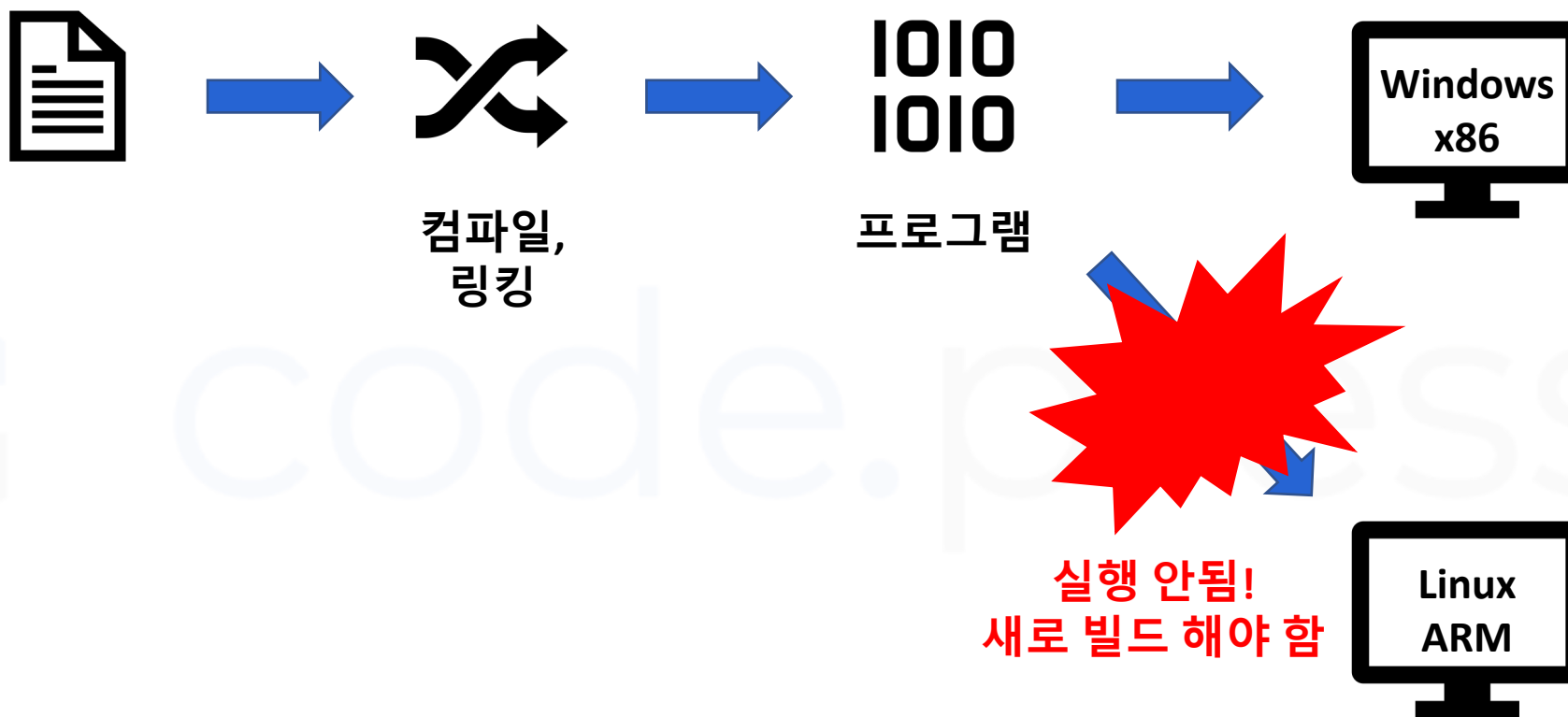
- Bytecode
- JVM
- JRE
- JDK

소스코드의 실행 과정

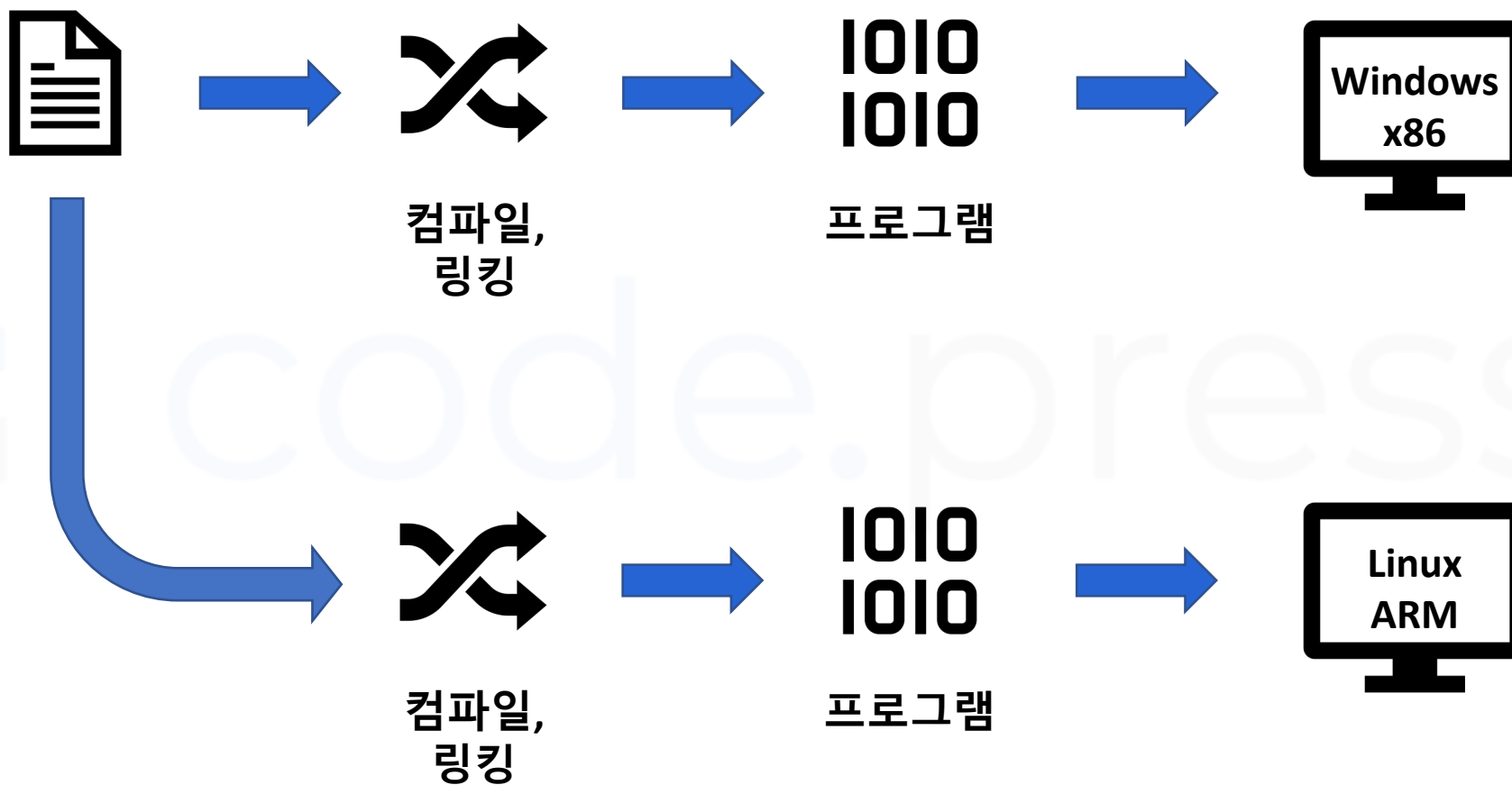
- 소스코드를 컴퓨터가 이해할 수 있는 실행 파일로 만들어 주는 과정



플랫폼 의존성 - C언어의 예



플랫폼 의존성 - C언어의 예



Bytecode와 JVM

■ Bytecode

- Java 소스코드를 빌드 했을 경우 생성 되는 중간 코드
- 어셈블리어와 유사한 형태
- 기계가 바로 실행할 수 없음

```
// Bytecode stream: 03 3b 84 00 01 1a 05
68 3b a7 ff f9
// Disassembly:

iconst_0      // 03
istore_0      // 3b
iinc 0, 1     // 84 00 01
iload_0       // 1a
iconst_2      // 05
imul          // 68
istore_0      // 3b
goto -7       // a7 ff f9
```

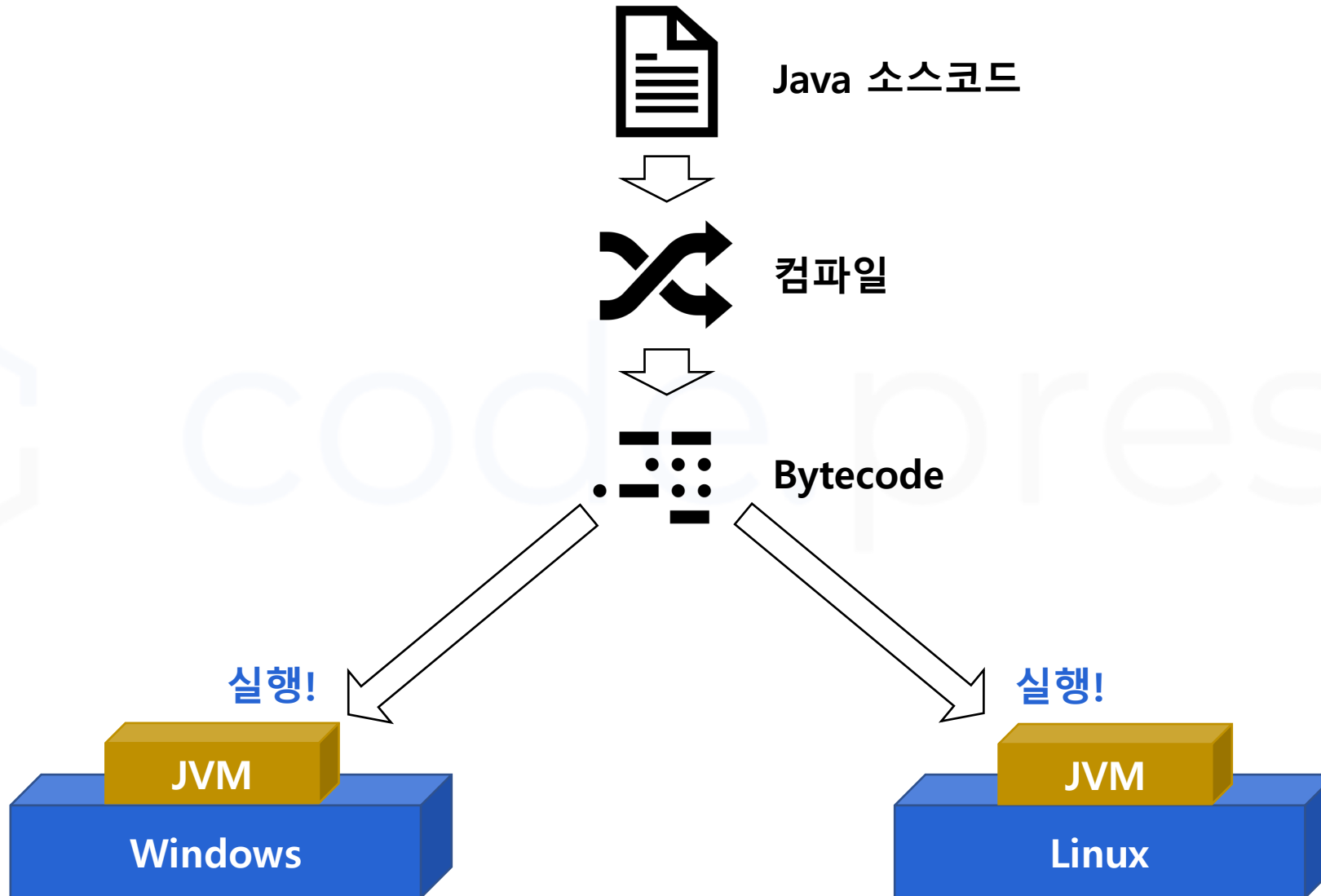
<https://www.jrebel.com/blog/java-bytecode-tutorial>

Bytecode와 JVM

■ JVM

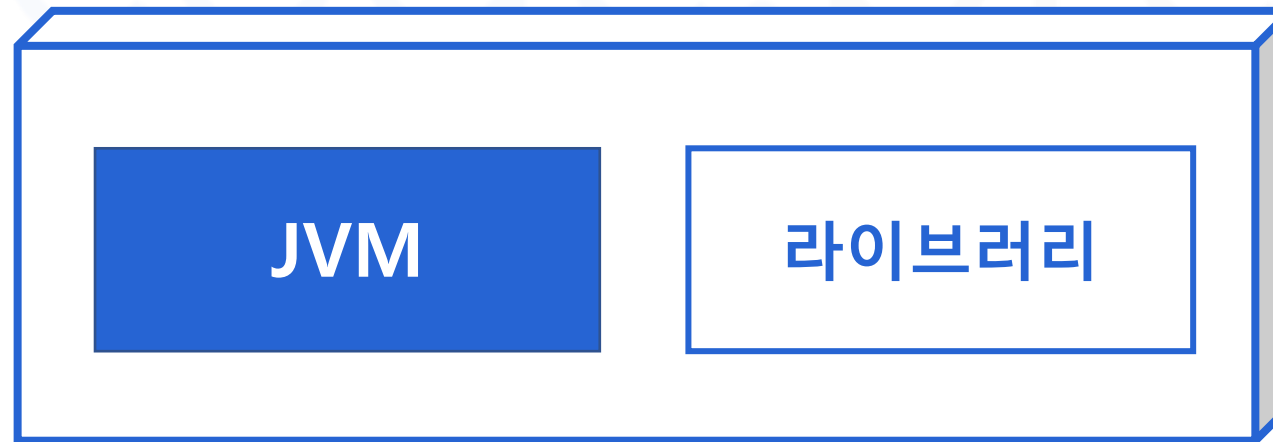
- Java Virtual Machine
- Bytecode를 실행하기 위한 가상 머신
- Write Once, Run Anywhere를 구현하는 핵심
- OS, HW 독립적으로 Bytecode를 실행 가능
 - Java Bytecode와 OS/HW 사이의 번역기

Bytecode와 JVM



■ JRE

- Java Runtime Environment
- JVM + Java 애플리케이션 실행에 필요한 추가적인 라이브러리 등을 포함
- Java 애플리케이션을 실행하기 위한 기본 단위

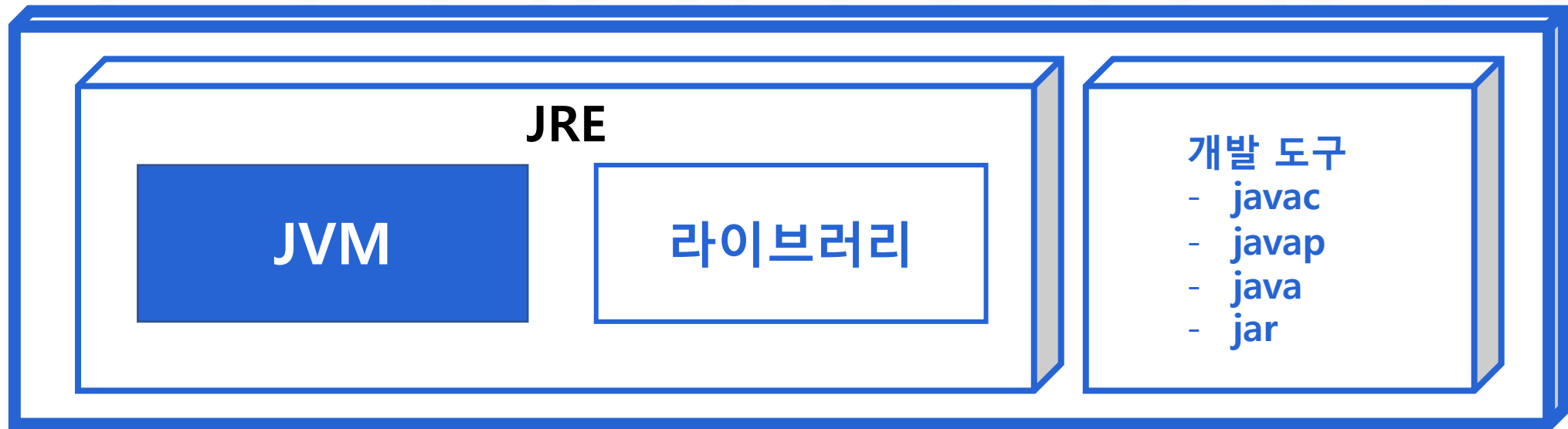


Java Runtime Environment

JRE와 JDK

■ JDK

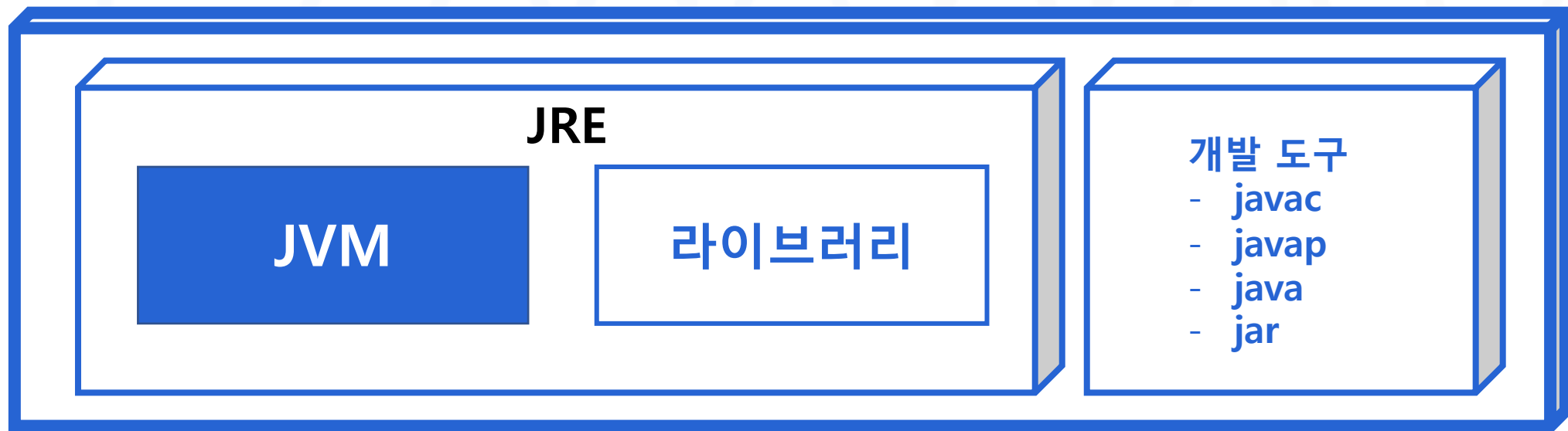
- Java Development Kit
- JRE + Java 개발 도구를 포함
- Java 개발을 위해서는 JDK 설치가 필수



Java Development Kit

Point!

- Java 소스코드를 컴파일하면 Bytecode라는 중간 코드가 생성 됨
- JVM으로 플랫폼 독립적인 WORA를 구현
- JRE는 실행 환경, JDK는 개발 도구를 제공



Java Development Kit