



Java의 개요

컴퓨팅 환경의 변화

◆ 컴퓨팅 환경의 변화

- 70년대 메인 프레임 시대
- 80년대 중형 컴퓨터/PC, 클라이언트/서버
- 90년대 웹/PC (빌 게이츠 : 향후 컴퓨팅 환경은 웹과 PC의 결합)

JAVA 탄생

◆ 탄생

- 가전 제품에 들어갈 소프트웨어를 만들기 위해 탄생

◆ 개발자

- 선 마이크로시스템즈 사의 (James Gosling)
 - 가전 제품의 긴 수명으로 인한 완벽한 호환을 가진 소프트웨어의 개발 요구
 - 가전 제품에 사용될 소프트웨어의 높은 신뢰성 필요
 - C 언어의 문제점
 - ◆ 특정한 컴퓨터 칩에 대해 컴파일하여야 함
- 현재는 Oracle

Java의 탄생

- ◆ 1991년 Sun Microsystems의 James Gosling → Oak
- ◆ Oak을 WWW에 적용
- ◆ 1994년 Oak을 Java로 개칭
- ◆ 1995년 Java Beta 2
- ◆ 1996년 Java 1.0
- ◆ Hotjava
- ◆ Java 1.1, 1.2, 1.3,...1.4.2 J2SE 5.0, 1.7



What is Java?



- ◆ Programming Language
- ◆ Platform
 - Java Virtual Machine
 - Java API(Application Programming Interface)



What is Java?



- ◆ Programming language
 - Simple
 - Object–Oriented

Java의 특성

- ◆ 간단한 언어 : C++에서 혼동되는 개념을 제거
- ◆ 객체 지향 언어
- ◆ 플랫폼 독립적 : 하드웨어와 OS에 관계 없이 동작

Java Platform

- ◆ **자바 가상 기계(Java Virtual Machine ; JVM)**
 - : 자바 바이트 코드를 기계로 사용하는 컴퓨터
 - JDK에 포함되어 있는 인터프리터
 - Appletviewer
 - WWW 검색기(넷스케이프, 익스플로러, 핫자바 등)
- ◆ **자바 API(Application Programming Interface)**
 - : 자바 개발 환경에서 제공되는 거대한 패키지

자바 플랫폼

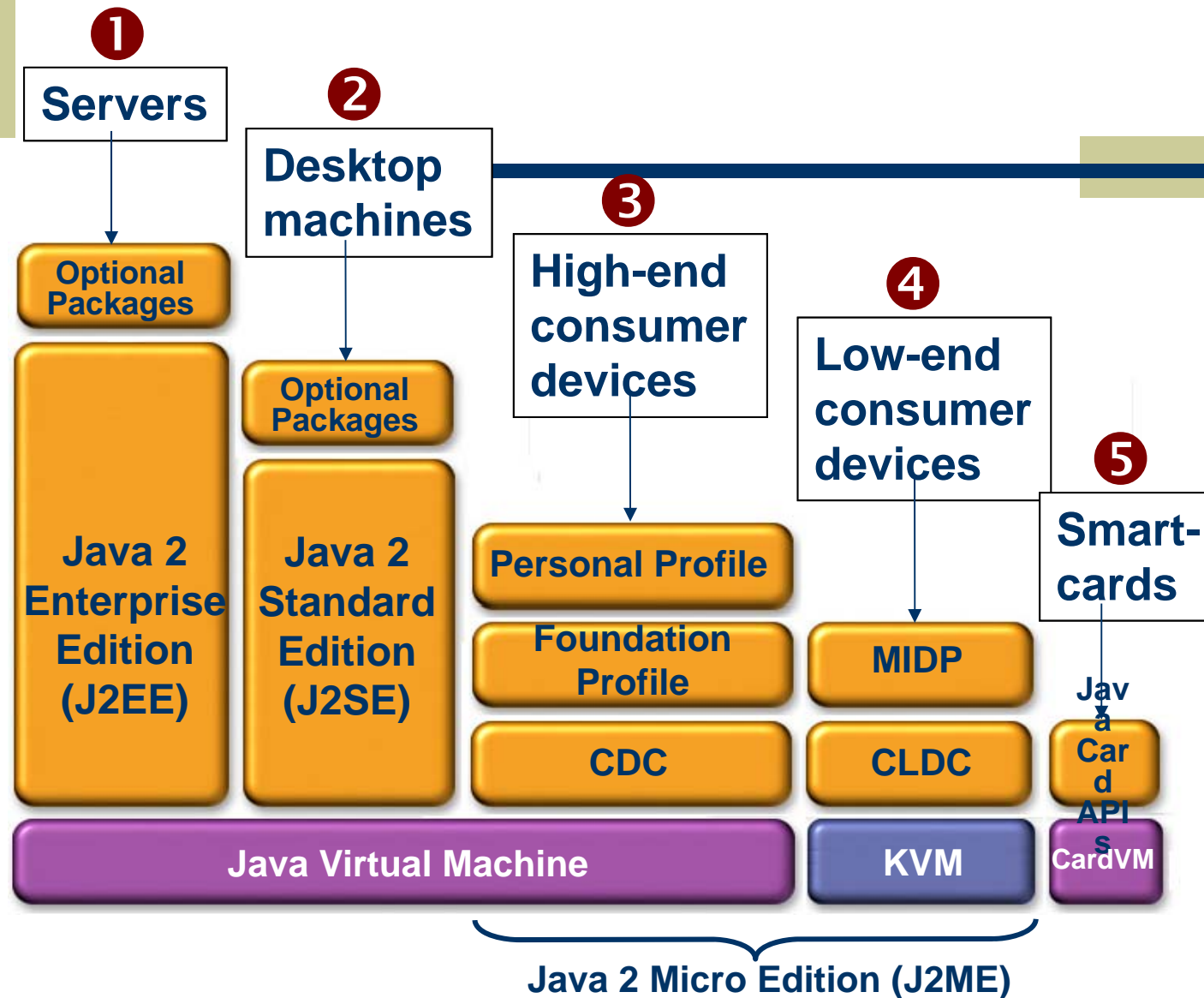


자바 프로그램

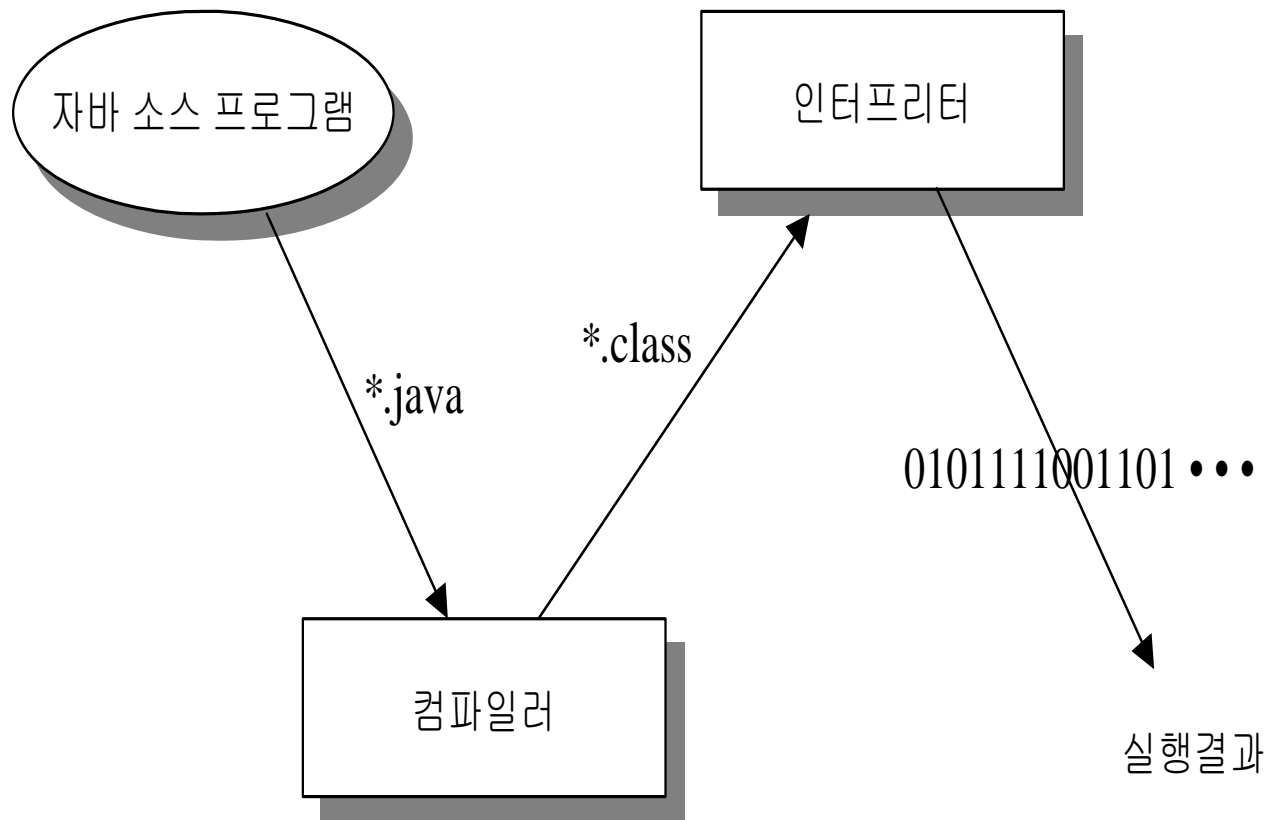
자바 API

자바 가상 기계

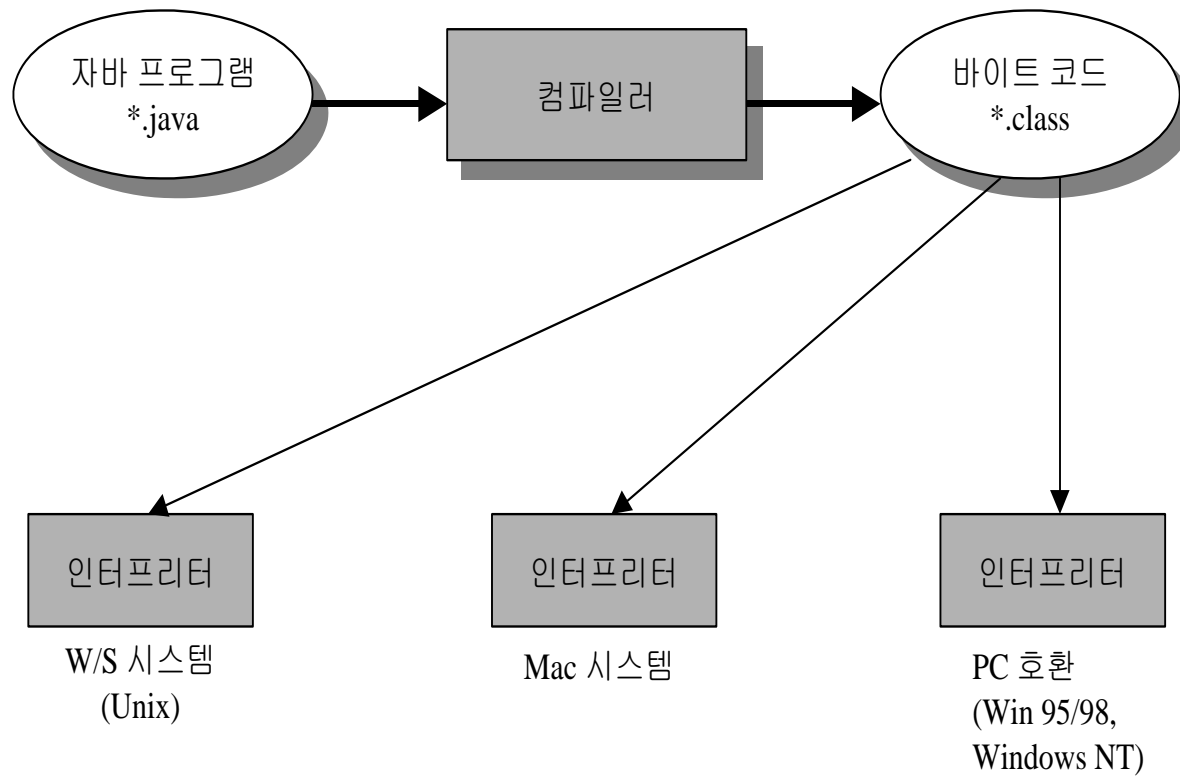
하드웨어 플랫폼



Compiled/Interpreted Lang.



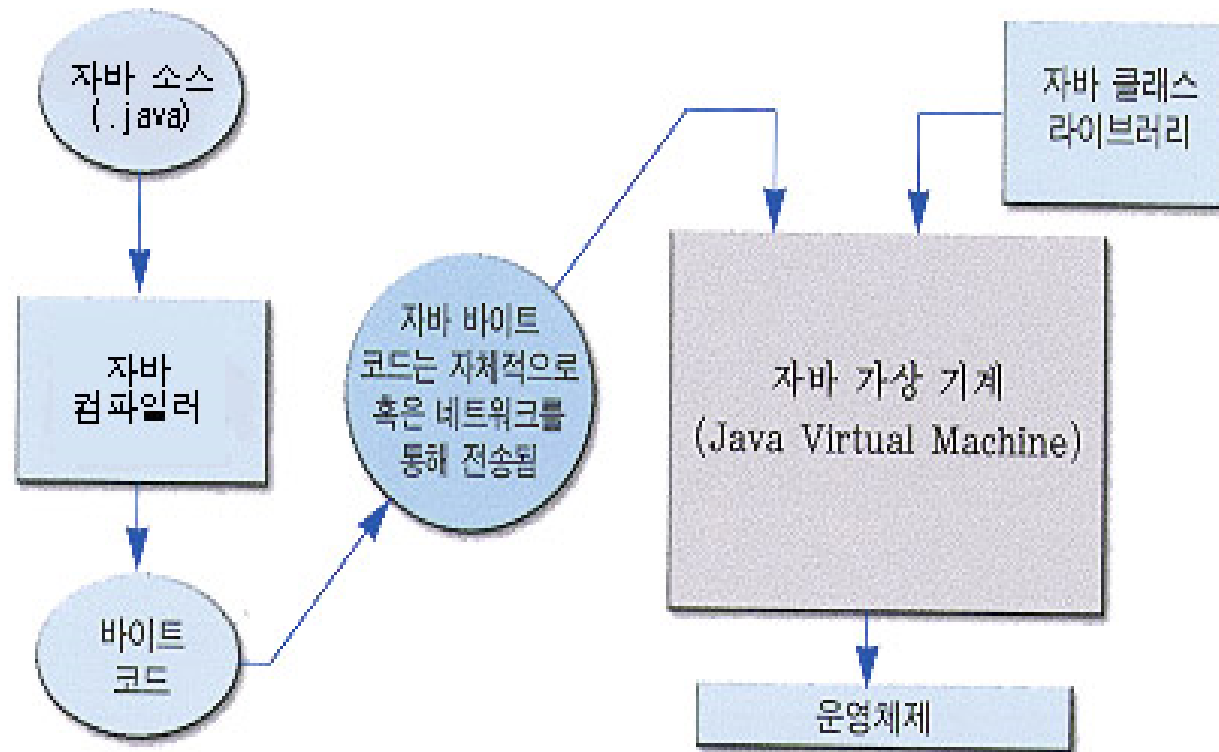
Independent Platform



Java의 실행 환경

컴파일 환경

실행 환경(자바 플랫폼)



Java의 장점

- ◆ 빨리 배울 수 있다
- ◆ 적은 코드로 프로그램을 작성할 수 있다.
- ◆ 좋은 품질의 프로그램을 작성할 수 있다.
- ◆ 프로그램 개발을 빨리 할 수 있다.
- ◆ 플랫폼에 독립적인 프로그램을 작성할 수 있다.
- ◆ 분산 소프트웨어를 쉽게 작성할 수 있다.



Java의 활용 분야



- ◆ 웹, 네트워크, 멀티미디어 분야
- ◆ 인트라넷 분야
- ◆ 주식, 증권 분야
- ◆ 인터넷 제어
- ◆ 서버



Java



- ◆ **Java SE(Standard Edition)**
 - **JDK(Java Development Kit)**
 - **JRE(Java Runtime Environment)**
- ◆ **JAVA EE(Enterprise Edition)**
- ◆ **Java ME(Micro Edition)**
- ◆ **JavaFX**

Java SE
Java EE
Java ME
Java SE Support
Java SE Advanced & Suite
Java Embedded
JavaFX
Java DB
Web Tier
Java Card
Java TV
New to Java
Community
Java Magazine

Overview	Downloads	Documentation	Community	Technologies	Training
----------	-----------	---------------	-----------	--------------	----------

Java SE Downloads

Next Releases (Early Access)

Embedded Use

Previous Releases



Java Platform (JDK) 7u25



JDK 7 + NetBeans

Java Platform, Standard Edition

Java SE 7u25

This release includes important security fixes. Oracle strongly recommends that all Java SE 7 users upgrade to this release.

[Learn more](#)

Which Java package do I need?

- **JDK:** (Java Development Kit). For Java Developers. Includes a complete JRE plus tools for developing, debugging, and monitoring Java applications.
- **Server JRE:** (Server Java Runtime Environment) For deploying Java applications on servers. Includes tools for JVM monitoring and tools commonly required for server applications, but does not include browser integration (the Java plug-in), auto-update, nor an installer. [Learn more](#)
- **JRE:** (Java Runtime Environment). Covers most end-users needs. Contains everything required to run Java applications on your system.

JDK

DOWNLOAD

JDK 7 Docs

- [Installation Instructions](#)
- [ReadMe](#)
- [Release Notes](#)
- [Oracle License](#)
- [Java SE Products](#)
- [Third Party Licenses](#)
- [Certified System Configurations](#)

Server JRE

DOWNLOAD

Server JRE 7 Docs

- [Installation Instructions](#)
- [ReadMe](#)
- [Release Notes](#)
- [Oracle License](#)
- [Java SE Products](#)
- [Third Party Licenses](#)
- [Certified System Configurations](#)

JRE

DOWNLOAD

JRE 7 Docs

- [Installation Instructions](#)
- [ReadMe](#)
- [Release Notes](#)
- [Oracle License](#)
- [Java SE Products](#)
- [Third Party Licenses](#)
- [Certified System Configurations](#)

JDK(Java Development Kit)

◆ Java 개발 tool

■ javac (자바 번역기)

- 자바 소스 코드를 바이트 코드로 번역하는 도구
- **javac** [*options*] *filename.java*

■ java (자바 해석기)

- 자바 바이트 코드를 해석하여 실행하는 도구
- 인터프리터의 파라미터(파일명)는 main() 메소드가 속해 있는 클래스명으로 해야 한다.
- **java** [*options*] *class_filename* [*arguments*]

JDK(Java Development Kit)

- appletviewer (애플릿 실행기)
 - 애플릿 프로그램을 실행하는 도구
 - (애플릿을 지정하는 HTML 파일을 생성하여 실행)
 - appletviewer [options] URL
- jdb (자바 디버거)
 - 프로그램의 버그(에러)를 찾는 데 사용되는 도구
 - jdb classfile_name
- javah
 - Java와 C로 작성된 소스 코드를 연결하는 도구
 - javah [options] filename.java [arguments]

JDK(Java Development Kit)

■ javadoc

- 자바 소스 파일에 대한 HTML 파일을 생성하는 도구
- (소스 프로그램에 있는 클래스, 인터페이스, 생성자 메소드, 클래스 계층구조 등에 관한 HTML 파일을 생성)
- **javadoc** [*options*] *filename.java*

■ jar (자바 압축도구)

- 여러 개의 클래스 파일을 하나의 *.jar 파일로 압축하거나 압축을 푸는 도구
- **jar** [*options*] [*jarfilename*] [*files*]

Java program

◆ Application

- C/C++ 언어로 만든 프로그램과 같은 응용 프로그램



◆ Applet

- Java 애플릿은 네트워크를 통하여 WWW 검색기나 애플릿 뷰어에서 실행되는 프로그램
- 애플릿은 HTML 파일속의 <applet> 태그에 실행할 위치와 옵션을 기술

Hello Java World!

- ◆ 예 : 응용 프로그램의 소스 프로그램 (Example1.java)

```
class Example1
{
    public static void main(String args[])
    {
        System.out.println("안녕하세요!! 첫 번째 자바프로그램입니다");
    }
}
```



```
MS-DOS 한글 MS-DOS
7 x 14
Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-1998.
C:\Windows>cd ..
C:\>cd jdk1.2\bin
C:\jdk1.2\bin>javac Example1.java
C:\jdk1.2\bin>java Example1
안녕하세요!! 첫 번째 자바 프로그램입니다
C:\jdk1.2\bin>
```

HelloWorld 애플리케이션

HelloWorld.java

```
class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!");
    }
}
```

컴파일 : javac HelloWorld.java (HelloWorld.class가 생성됨)

실행 : java HelloWorld