

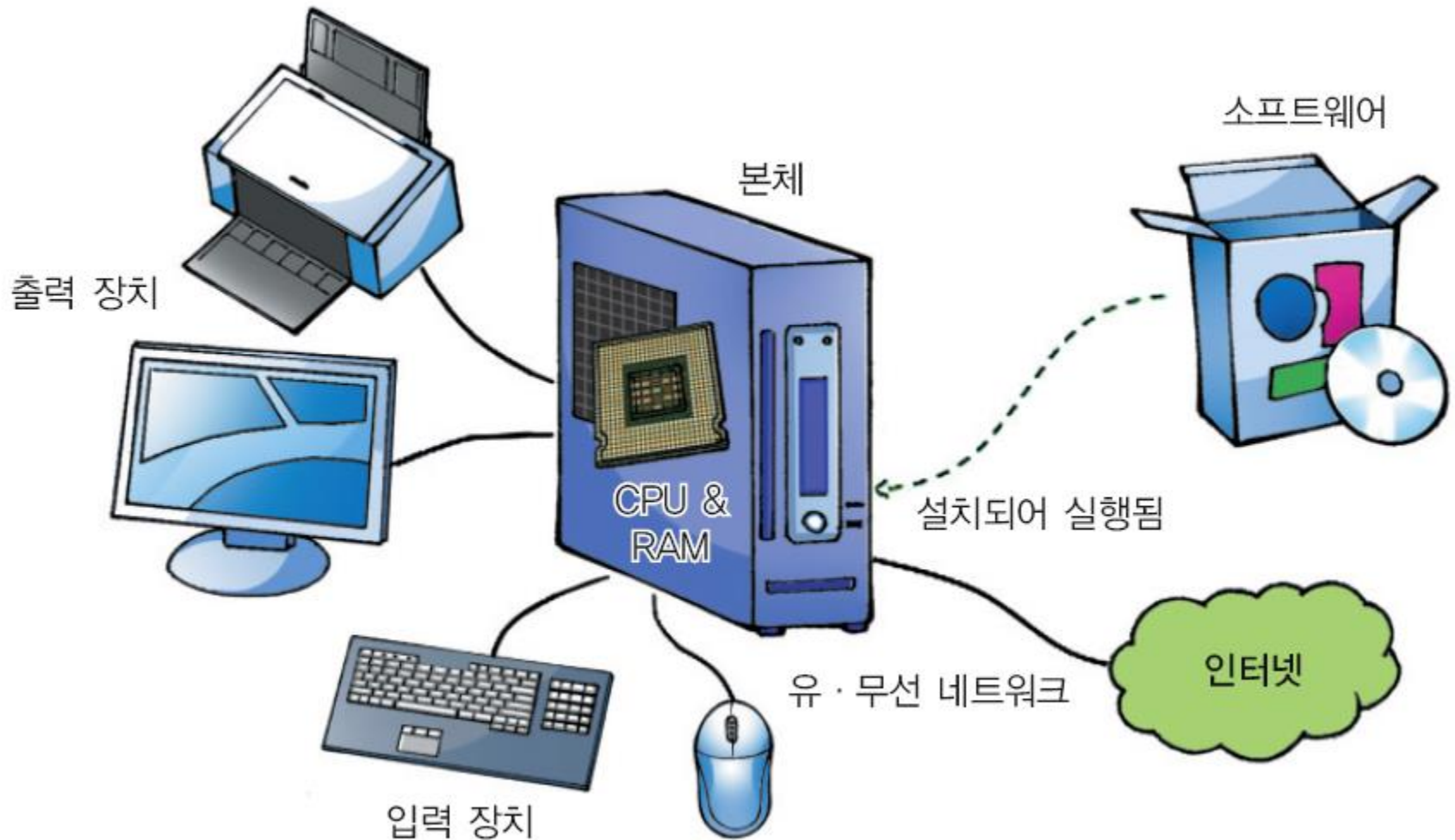
제1장 자바 시작

2018년1학기

컴퓨터공학과 김 명 교수

컴퓨터의 하드웨어와 소프트웨어

2



프로그래밍 언어 (1/2)

3

□ 기계어 (machine language)

- 0, 1 이진수로 구성
- 컴퓨터의 CPU는 기계어만 이해하고 처리함

□ 어셈블리어 (assembly language)

- 기계어의 명령을 ADD, SUB, MOVE 등과 같이 표현하기 쉬운 상징적인 단어인 니모닉 기호 (mnemonic symbol)로 1:1 대응시킨 언어

```
; Example of IBM PC assembly language
; Accepts a number in register AX;
; subtracts 32 if it is in the range 97-122;
; otherwise leaves it unchanged.

SUB32  PROC                ; procedure begins here
        CMP  AX,97          ; compare AX to 97
        JL   DONE           ; if less, jump to DONE
        CMP  AX,122         ; compare AX to 122
        JG   DONE           ; if greater, jump to DONE
        SUB  AX,32          ; subtract 32 from AX
DONE:   RET                 ; return to main program
SUB32   ENDP               ; procedure ends here
```

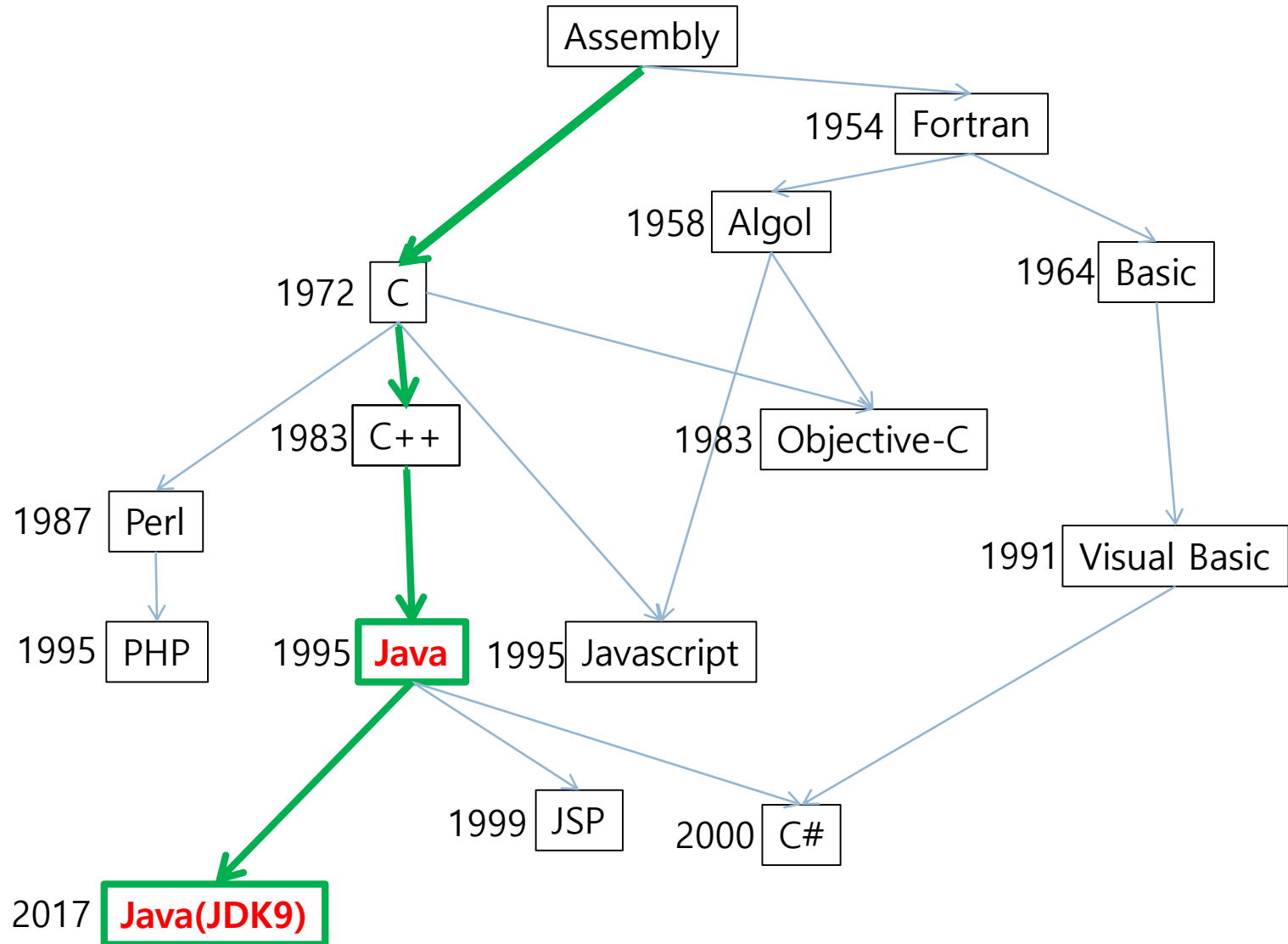
프로그래밍 언어 (2/2)

4

□ 고급언어 (high level language)

- 사람이 이해하기 쉽고,
복잡한 작업, 자료 구조, 알고리즘을 표현하기 위해 고안된 언어
- C/C++, Java, C#, 등
- 절차 지향 언어와 객체 지향 언어로 나눌 수 있음
 - Structured programming
 - Object oriented programming

5



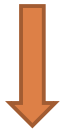
프로그램 편집, 컴파일 및 실행

6

- ▣ Java : .java → .class
- ▣ C : .c → .obj → .exe
- ▣ C++ : .cpp → .obj → .exe



소스 프로그램
편집 및 개발



```
if (i>0) {  
    i = i*10;  
}  
i = i - j;  
System.out.println(i);
```

소스 프로그램



Compiler

컴파일



```
01010000010001  
01010011110101  
10101010010111  
01010101010000  
10001110000000
```

기계어

프로그램 실행



자바의 태동

7

- Sun Microsystems 의 제임스 고슬링(James Gosling) 이 개발
 - ▣ 1991년에 가전 제품에 들어갈 소프트웨어를 위해 개발 시작
 - ▣ 1995년 자바 발표
- 목적
 - ▣ 플랫폼 호환성 문제 해결, 플랫폼 독립적인 언어 개발
 - ▣ 메모리 사용량이 적고 다양한 플랫폼을 가지는 가전 제품에 적용
- 초기 이름 : 오크(OAK)
 - ▣ 인터넷 환경에 적합하게 발전시켜 Java 발표
- 2009년에 Sun Microsystems 를 Oracle 사에서 인수

플랫폼 종속성(platform dependence)

8

인텔 CPU를 가진
리눅스 환경에서 개발



C/C++
프로그램

컴파일

기계어

플랫폼 = 하드웨어 플랫폼 + 운영체제 플랫폼

프로그램의 플랫폼 호환성 없는 이유

- 기계어가 CPU마다 상이함
- 운영체제마다 API가 서로 다름
- 운영체제마다 실행파일 형식이 서로 다름

실행



인텔 CPU + 리눅스

실행되지 않음



Apple사의 MAC PC

실행되지 않음



인텔 CPU + 윈도우 노트북



임베디드 컴퓨터
+ 리눅스

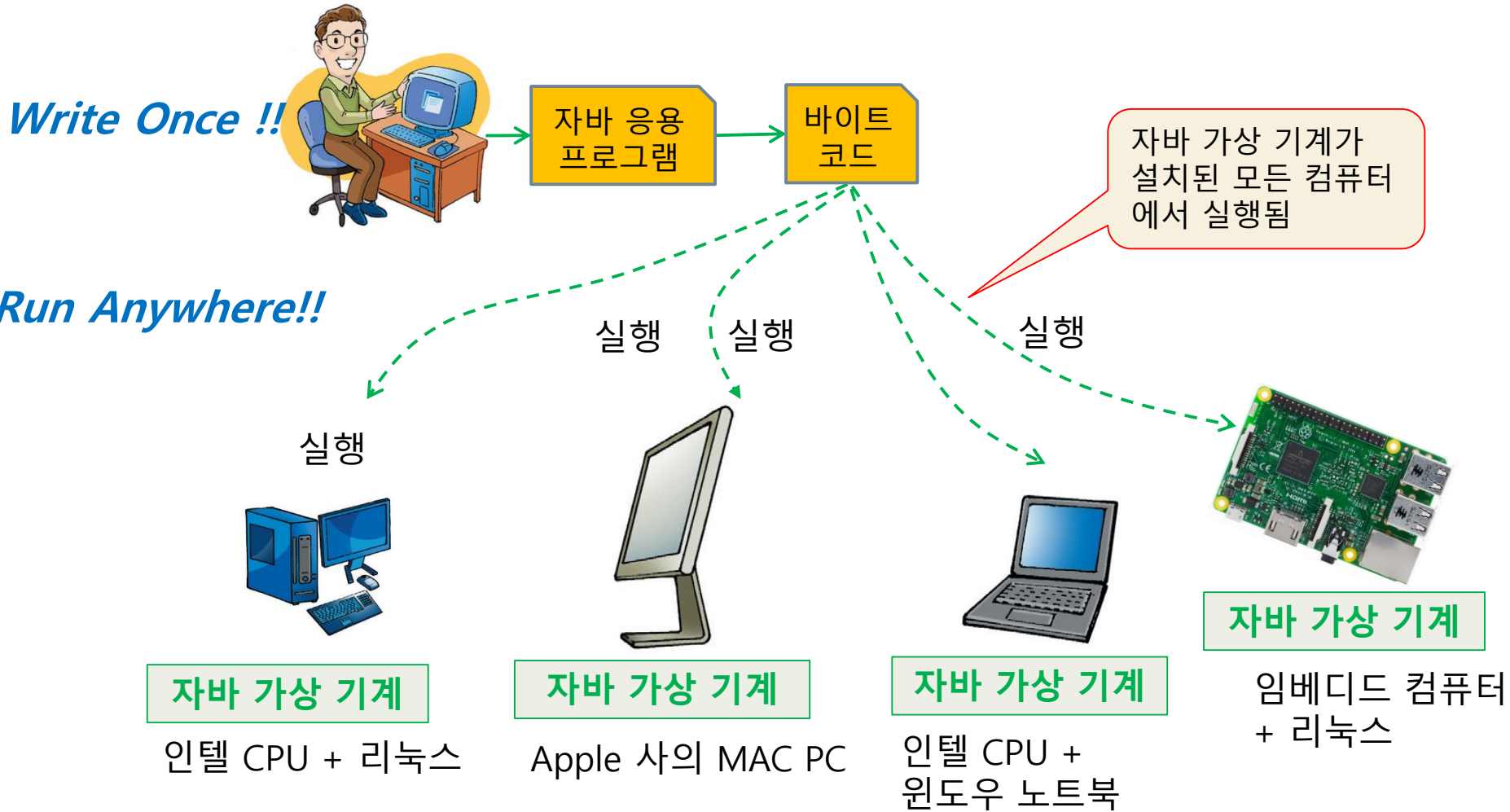
WORA(Write Once Run Anywhere)

9

- 한 번 만 작성하여 모든 플랫폼에서 실행가능
 - ▣ C/C++ 등과 같은 기존 언어의 플랫폼 종속성 극복
 - ▣ OS, HW에 상관없이 JVM이 있는 곳이면 어디서나 자바 프로그램에 동일한 실행환경을 제공하며 실행 결과 기대
 - ▣ 네트워크에 연결된 클라이언트들에서 실행 가능
 - 웹 브라우저, 분산 환경에서 지원됨
- WORA를 가능하게 하는 자바의 특징
 - ▣ 바이트 코드 (byte code)
 - ▣ 자바 가상 기계, JVM (Java Virtual Machine)

자바의 플랫폼 독립성, WORA

10



자바의 실행 환경

11

□ 바이트 코드 (Bytecode)

- 자바 가상 기계에서 실행 가능한 바이너리 코드
- 클래스 파일 (.class) 에 저장

□ 자바 가상 기계(JVM : Java Virtual Machine)

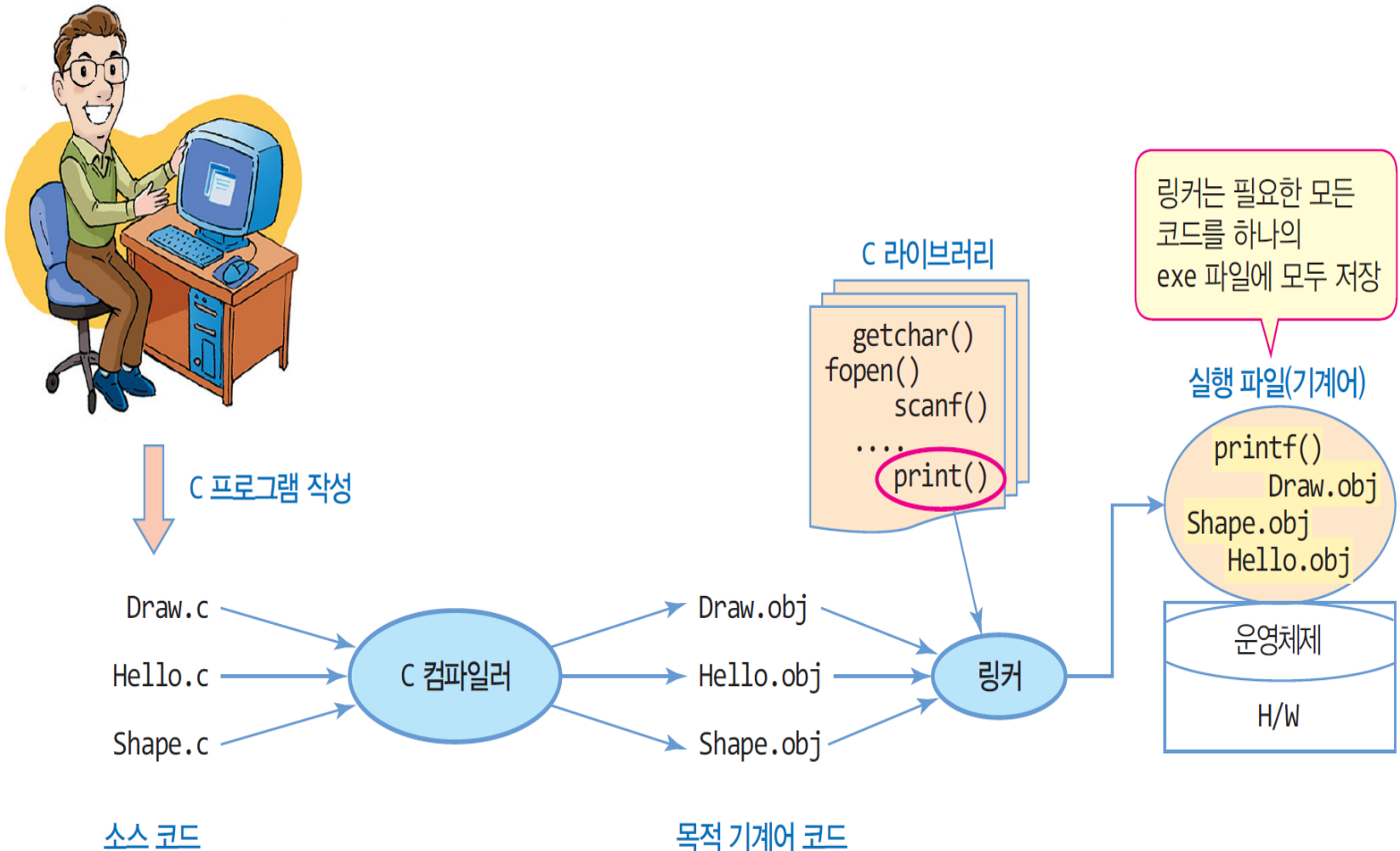
- 가상 기계로서 각기 다른 플랫폼에서 동일한 자바 실행환경 제공
- 자바 가상 기계 자체는 플랫폼에 종속적
- 인터프리터 (interpreter) 방식으로 바이트 코드 해석하고 실행
- 자바 가상 기계는 다양한 회사에서 개발되어 공급

□ 자바의 실행

- 자바 가상 기계가 클래스 파일 (.class) 의 바이트 코드 실행

C/C++ 프로그램의 개발 및 실행 환경

12



자바 가상 기계와 자바 응용프로그램의 실행

13



자바 프로그래밍

Draw.java
Hello.java
Shape.java

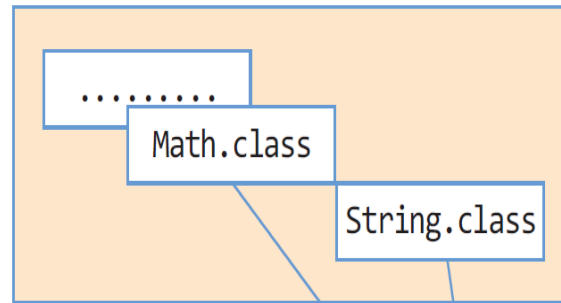
소스 코드

자바 컴파일러

Draw.class
Hello.class
Shape.class

바이트 코드

자바 클래스 라이브러리(JDK)



클래스 로딩

자바 가상 기계

운영체제

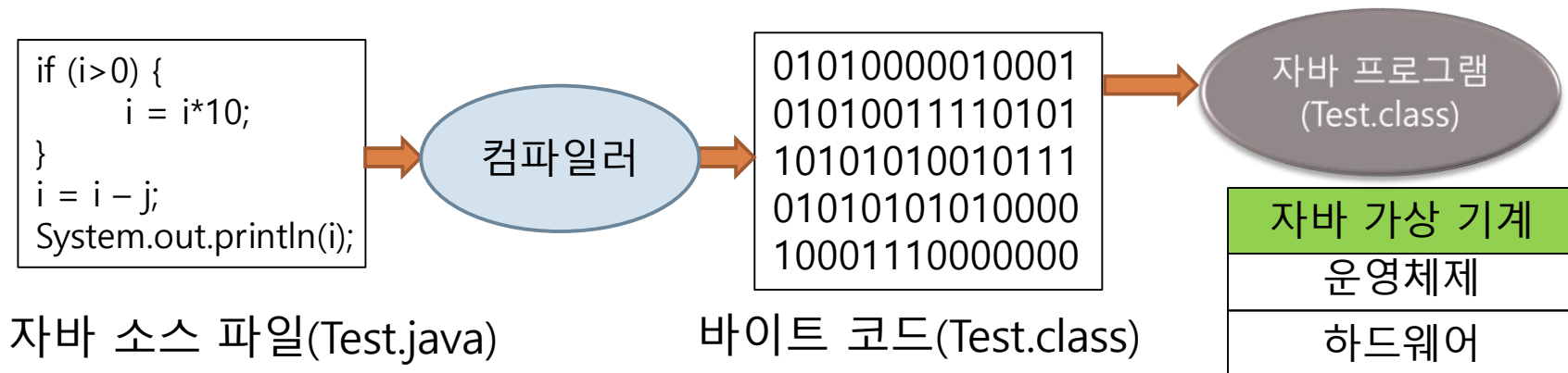
H/W

필요할 때 클래스 파일을 로딩하여 실행

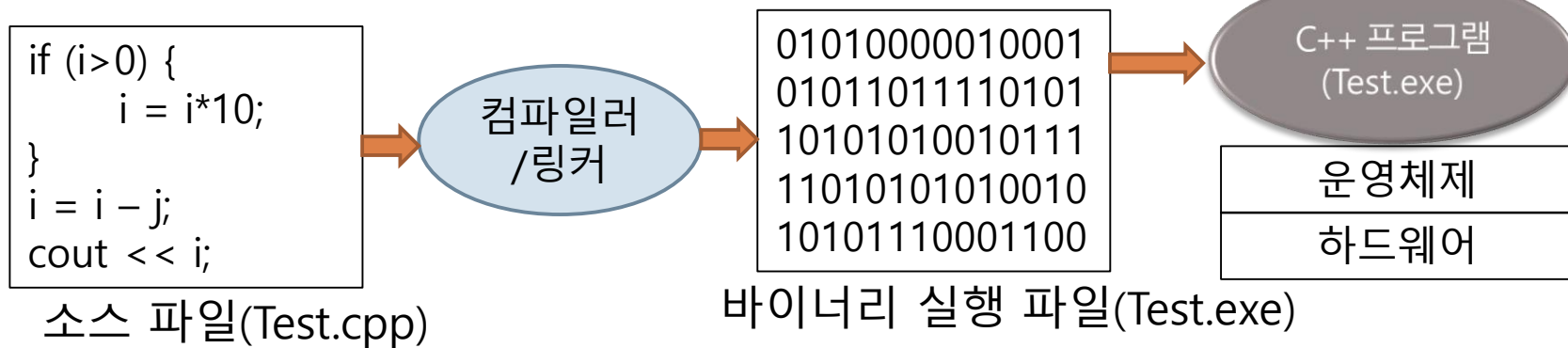
자바와 C/C++의 실행 환경 차이

14

□ Java



□ C/C++



자바와 C/C++ 실행 환경과 실행 과정

15

□ Java

- 링크 과정 없이 컴파일러가 바로 바이트 코드 생성한다.
- 바이트 코드는 JVM 에서만 실행 가능하다.
- 런타임에 필요한 클래스들은 JVM 에 의해 링크되며, 클래스 로더가 동적으로 필요한 클래스를 로딩한다.

□ C/C++

- 컴파일러가 오브젝트 코드를 생성한 후, 링커가 필요한 라이브러리를 링크하여 최종 실행 가능한 실행 파일 (exe 파일)을 생성한다.
- 정적 라이브러리들은 실행 파일에 포함되므로 실행 파일의 크기가 커진다.
- 동적 라이브러리의 경우 런타임에 링크가 일어난다.
- 오브젝트 코드와 실행파일은 플랫폼마다 다르다.

자바의 배포판 종류

16

□ Java SE

클릭



- 자바 표준 배포판
- 데스크탑과 서버 응용 개발 플랫폼

□ Java ME

- 자바 마이크로 배포판
- 휴대폰 등 제한된 리소스를 갖는 하드웨어에서 응용 개발을 위한 플랫폼
- Java SE의 subset + 임베디드 및 가전 제품을 위한 API 정의

□ Java EE

- 자바 기업용 배포판
- 자바를 이용한 다중 사용자, 기업용 응용 개발을 위한 플랫폼
- Java SE + 인터넷 기반의 서버측 컴퓨팅 관련 API 추가

자바 (Java SE) 설치

17

The screenshot shows the Oracle Java SE Downloads page. The browser address bar displays `www.oracle.com/technetwork/java/javase/downloads/index.html`. The Oracle logo is in the top left, and a 'Menu' button is next to it. A search bar is on the right. Below the header, a breadcrumb trail reads 'Oracle Technology Network / Java / Java SE / Downloads'. A sidebar on the left contains a list of links: 'Java SE', 'Java EE', 'Java ME', 'Java SE Advanced & Suite', 'Java Embedded', 'Java DB', 'Web Tier', 'Java Card', 'Java TV', 'New to Java', 'Community', and 'Java Magazine'. A red arrow points to the 'Java SE' link. The main content area has tabs for 'Overview', 'Downloads', 'Documentation', 'Community', 'Technologies', and 'Training'. A red arrow points to the 'Downloads' tab. Under the 'Downloads' tab, there are two download buttons: 'Java Platform (JDK) 9' and 'NetBeans with JDK 8'. Below these, a section titled 'Java Platform, Standard Edition' contains information about 'Java SE 9.0.4', stating it includes important bug fixes and is recommended for all Java SE 9 users. A red arrow points to the 'JDK DOWNLOAD' button in this section.

Java SE - Downloads | C x

www.oracle.com/technetwork/java/javase/downloads/index.html

ORACLE

Menu

Sign In Cour

Oracle Technology Network / Java / Java SE / Downloads

Java SE

Java EE

Java ME

Java SE Advanced & Suite

Java Embedded

Java DB

Web Tier

Java Card

Java TV

New to Java

Community

Java Magazine

Overview Downloads Documentation Community Technologies Training

Java SE Downloads

Java Platform (JDK) 9

NetBeans with JDK 8

Java Platform, Standard Edition

Java SE 9.0.4

Java SE 9.0.4 includes important bug fixes. Oracle strongly recommends that all Java SE 9 users upgrade to this release.

Learn more

Installation Instructions

Release Notes

JDK

DOWNLOAD

자바 (Java SE) 설치

18

Oracle Technology Network / Java / Java SE / Downloads

Java SE

Java EE

Java ME

Java SE Advanced & Suite

Java Embedded

Java DB

Web Tier

Java Card

Java TV

New to Java

Community

Java Magazine

Overview

Downloads

Documentation

Community

Technologies

Training

Java SE Development Kit 9 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- [Java Developer Newsletter](#): From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- [Java Developer Day hands-on workshops \(free\) and other events](#)
- [Java Magazine](#)

JDK 9.0.4 checksum

Java SE Development Kit 9.0.4

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

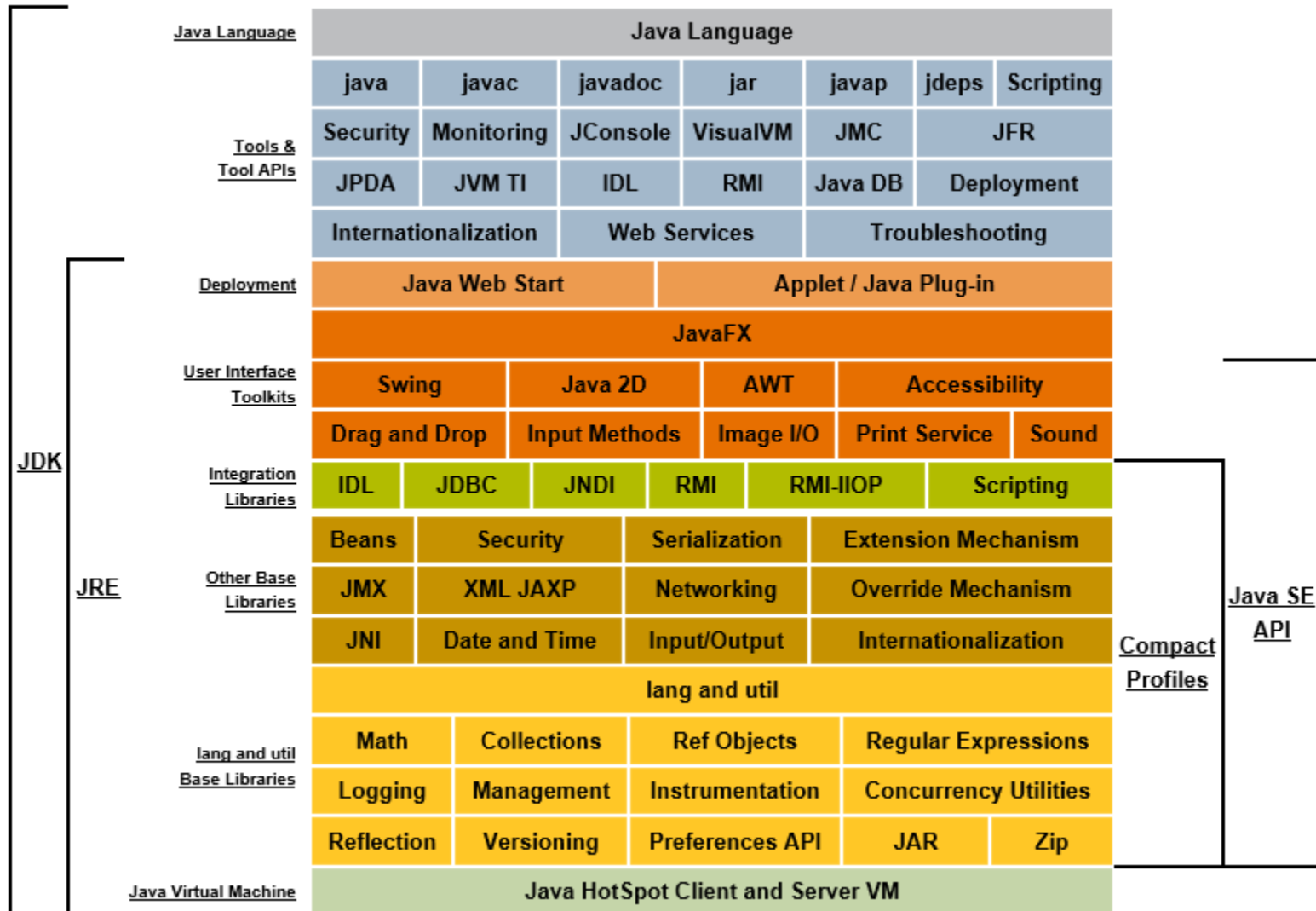
☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux	305.07 MB	jdk-9.0.4_linux-x64_bin.rpm
Linux	338.21 MB	jdk-9.0.4_linux-x64_bin.tar.gz
macOS	382.11 MB	jdk-9.0.4_osx-x64_bin.dmg
Windows	375.56 MB	jdk-9.0.4_windows-x64_bin.exe
Solaris SPARC	206.97 MB	jdk-9.0.4_solaris-sparcv9_bin.tar.gz

Java SE 구성

출처: <http://download.oracle.com/javase/8/docs/>

19



JDK와 JRE

20

□ JDK : Java Development Kit

- 자바 응용 개발 환경으로, 개발에 필요한 도구 포함
 - 컴파일러 및 도구, JRE (Java Runtime Environment), 클래스 라이브러리의 소스, 자바 응용 프로그램의 샘플 소스, 등

□ JRE : Java Runtime Environment

- 자바 실행 환경으로 JVM이 포함되어 있음
- 자바 실행 환경만 필요한 경우 JRE만 따로 다운 가능

□ JDK와 JRE의 개발 및 배포

- 오라클의 Technology Network의 자바 사이트에서 다운로드
- <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

자바 응용프로그램 실행기

21

- JDK의 bin 디렉토리에 포함된 주요 개발 도구
 - ▣ **javac** : 자바 소스를 바이트 코드로 변환하는 컴파일러
 - ▣ **java** : 자바 응용프로그램 실행기 (jre bin 에도 있음)
 - ▣ **javadoc** : 자바 소스프로그램으로부터 HTML 형식의 API 문서 생성
 - ▣ **jar** : 자바 클래스 파일을 압축한 자바 아카이브 파일 (JAR) 생성하고 관리하는 유틸리티
 - ▣ **jdb** : 자바 응용 프로그램의 실행 중 오류를 찾는 자바 디버거
 - ▣ **javap** : 클래스 파일에 담긴 바이트 코드를 자바 소스와 함께 보여주는 디어셈블러

자바 API (Application Program Interface)

22

□ 자바 API

- JDK에 포함된 클래스 라이브러리
- 주요한 기능들을 미리 구현한 클래스 라이브러리의 집합
- 개발자는 API를 이용하여 쉽고 빠르게 자바 프로그램 개발

□ 자바 패키지(package)

- 서로 관련된 클래스들을 분류하여 묶어 놓은 것
- 필요한 클래스가 속한 패키지만 import하여 사용
- 계층구조로 되어 있음

- 자바 API(클래스 라이브러리)는 JDK내에 패키지 형태로 제공
- 개발자가 자신의 패키지 생성 가능

자바 온라인 API 문서

23

<http://docs.oracle.com/javase/8/docs/api/>

The screenshot shows a web browser displaying the Java Platform Standard Ed. 8 API documentation for the `String` class. The browser's address bar shows the URL `https://docs.oracle.com/j`. The page has a navigation bar with tabs for OVERVIEW, PACKAGE, CLASS (selected), USE, TREE, DEPRECATED, INDEX, and HELP. Below the navigation bar, there are links for PREVIOUS CLASS, NEXT CLASS, FRAMES, and NO FRAMES. The main content area displays the class name `String` and its package `java.lang`. It lists the interfaces implemented by `String`: `Serializable`, `CharSequence`, and `Comparable<String>`. The class declaration is shown as `public final class String` extending `Object` and implementing `Serializable`, `Comparable<String>`, and `CharSequence`. A description states that the `String` class represents character strings and that all string literals in Java programs are implemented as instances of this class. It also notes that strings are constant and their values cannot be changed after creation. An example code snippet is provided: `String str = "abc";`. The left sidebar shows a list of packages and classes, with `String` highlighted.

Java™ Platform Standard Ed. 8

OVERVIEW PACKAGE **CLASS** USE TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

compact1, compact2, compact3
java.lang

Class String

java.lang.Object
java.lang.String

All Implemented Interfaces:
Serializable, CharSequence, Comparable<String>

public final class **String**
extends `Object`
implements `Serializable`, `Comparable<String>`, `CharSequence`

The `String` class represents character strings. All string literals in Java programs, such as "abc", are implemented as instances of this class.

Strings are constant; their values cannot be changed after they are created. String buffers support mutable strings. Because `String` objects are immutable they can be shared. For example:

```
String str = "abc";
```

자바 통합 개발 환경 (자바 IDE) - Eclipse

24

□ IDE

- Integrated Development Environment의 약자
- 통합 개발 환경
- 편집, 컴파일, 디버깅을 한번에 할 수 있는 통합된 개발 환경

□ Eclipse

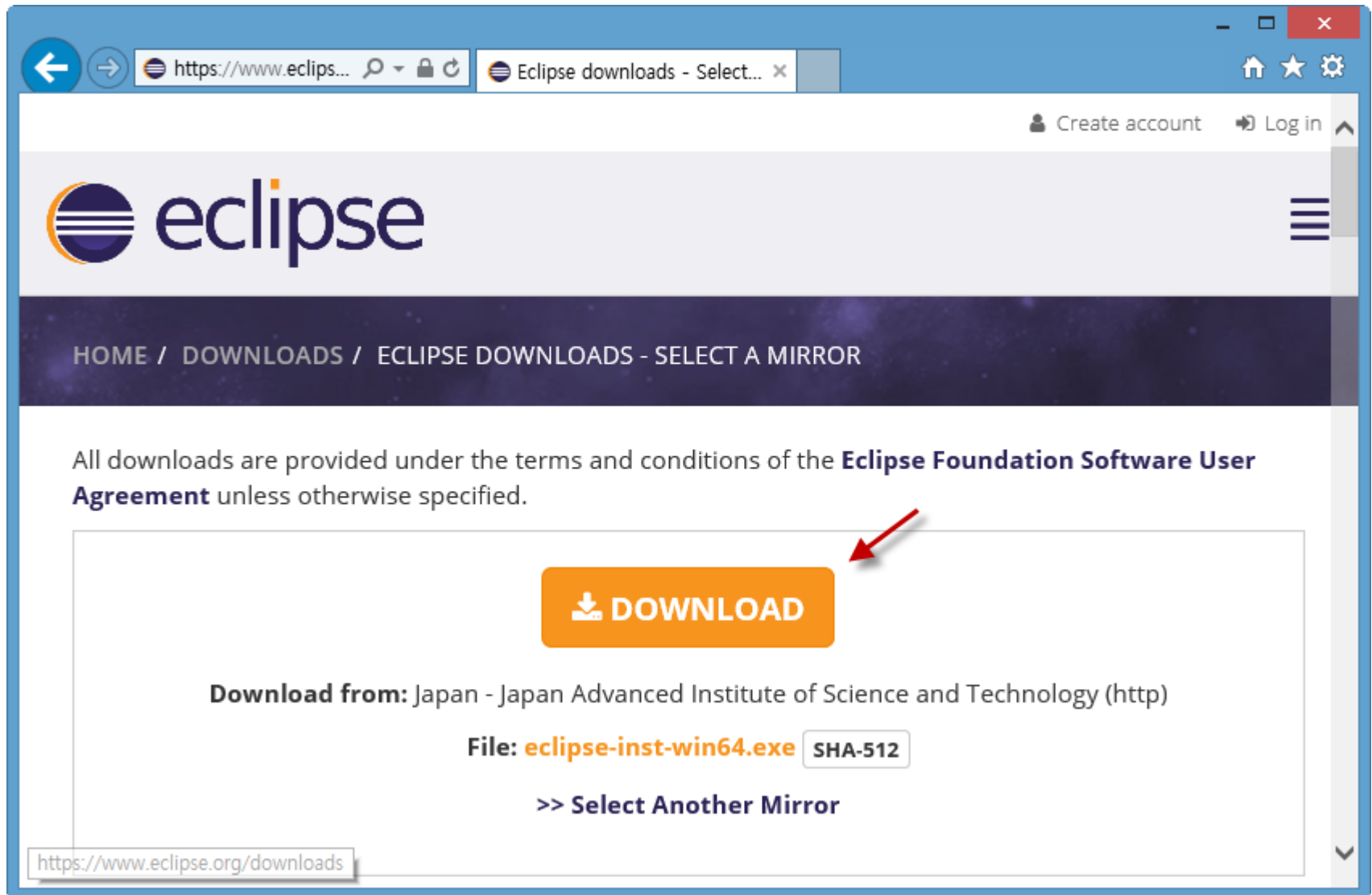
- 자바 응용 프로그램 개발을 위한 통합 개발 환경
- IBM에 의해 개발된 오픈 소스 프로젝트
- <http://www.eclipse.org/downloads/> 에서 다운로드

클릭

<http://www.eclipse.org/downloads/>

25






The screenshot shows a web browser window with the URL `https://www.eclips...` and a tab titled "Eclipse downloads - Select...". The page features the Eclipse logo and a navigation bar with links for "HOME", "DOWNLOADS", and "ECLIPSE DOWNLOADS - SELECT A MIRROR". A red arrow points to an orange "DOWNLOAD" button. Below the button, the text indicates the download source as "Japan - Japan Advanced Institute of Science and Technology (http)" and the file as "eclipse-inst-win64.exe" with a "SHA-512" checksum. A link to "Select Another Mirror" is also present.

26


← → `https://www.eclips...` Eclipse downloads - Select... ×

Create account Log in

 eclipse

HOME / DOWNLOADS / ECLIPSE DOWNLOADS - SELECT A MIRROR

All downloads are provided under the terms and conditions of the **Eclipse Foundation Software User Agreement** unless otherwise specified.

 **DOWNLOAD**

Download from: Japan - Japan Advanced Institute of Science and Technology (http)

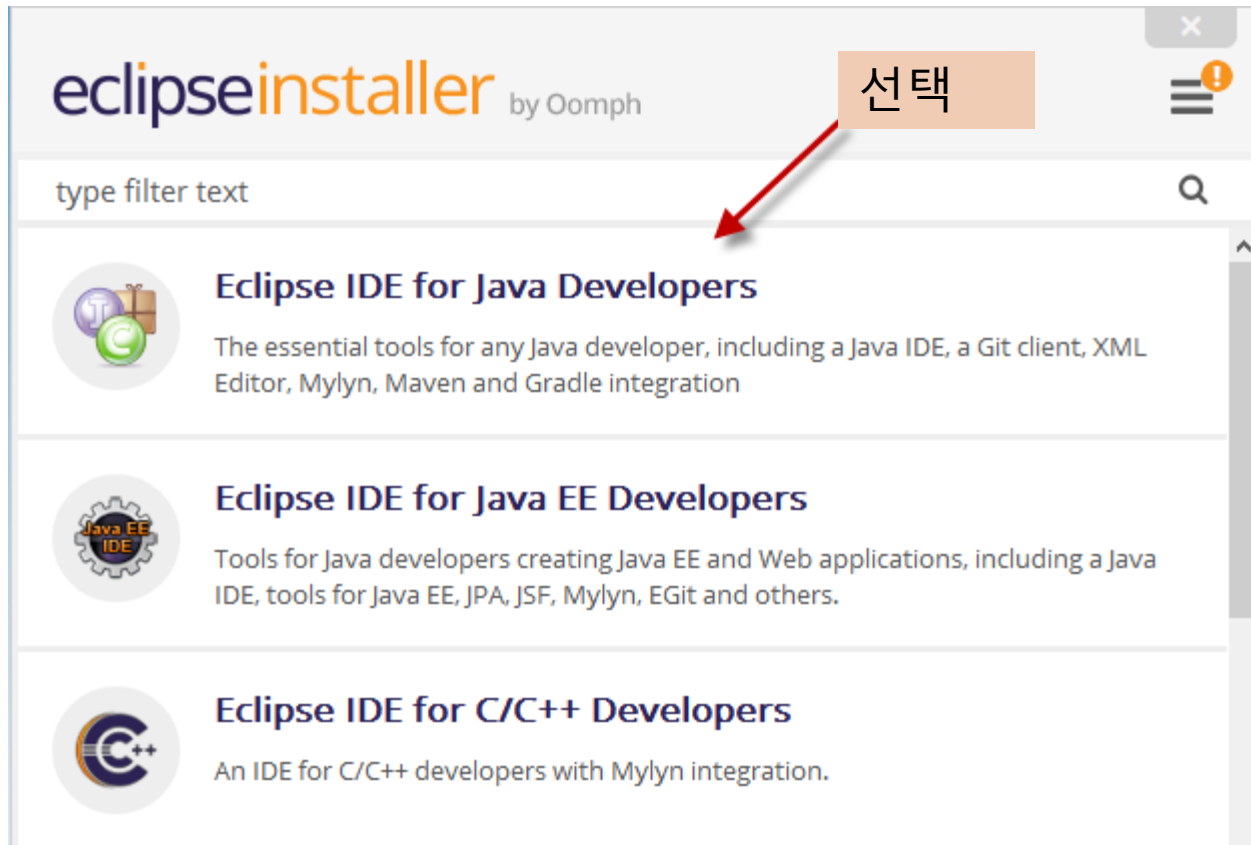
File: **eclipse-inst-win64.exe** **SHA-512**

>> Select Another Mirror

`https://www.eclipse.org/downloads`

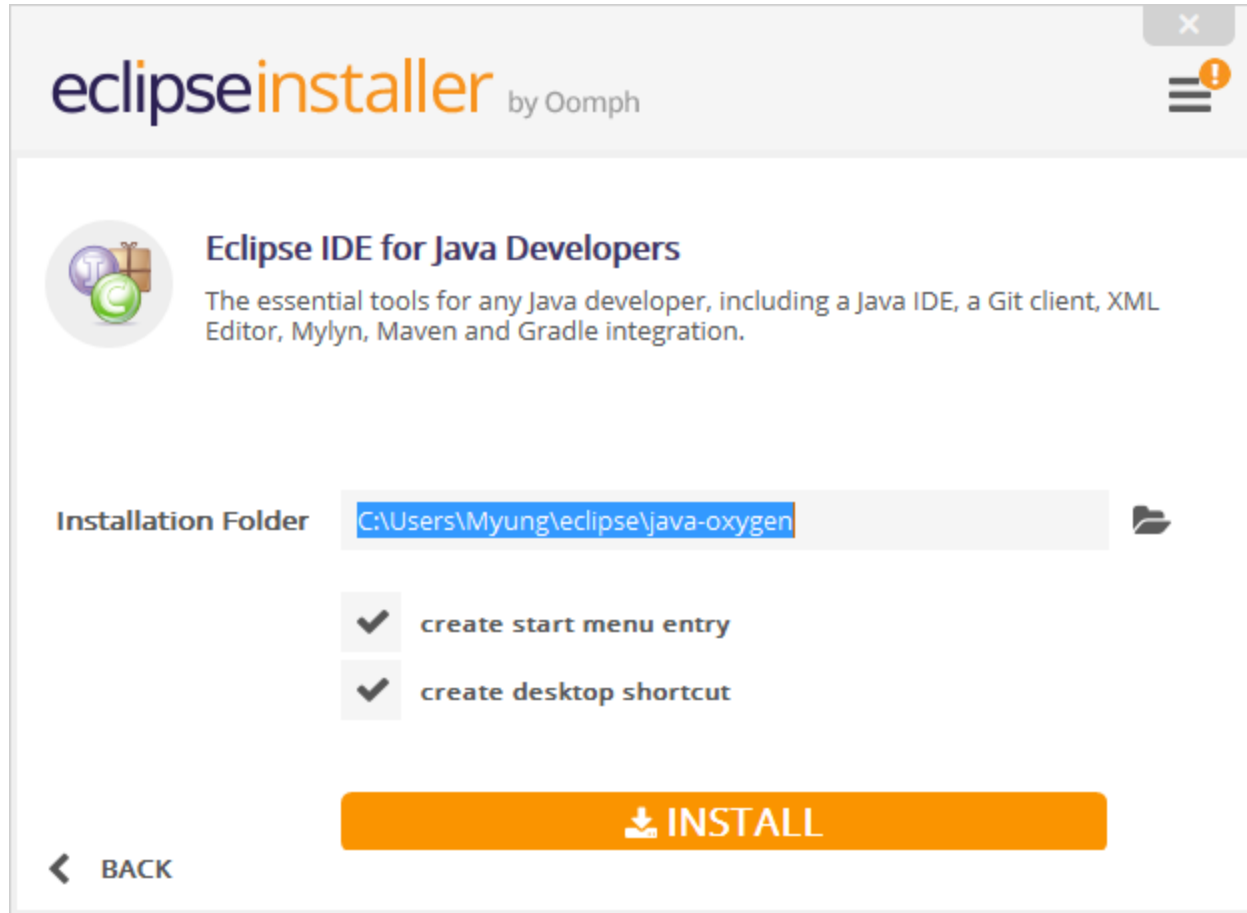
Eclipse 설치

27



Eclipse 설치의 진행...

28

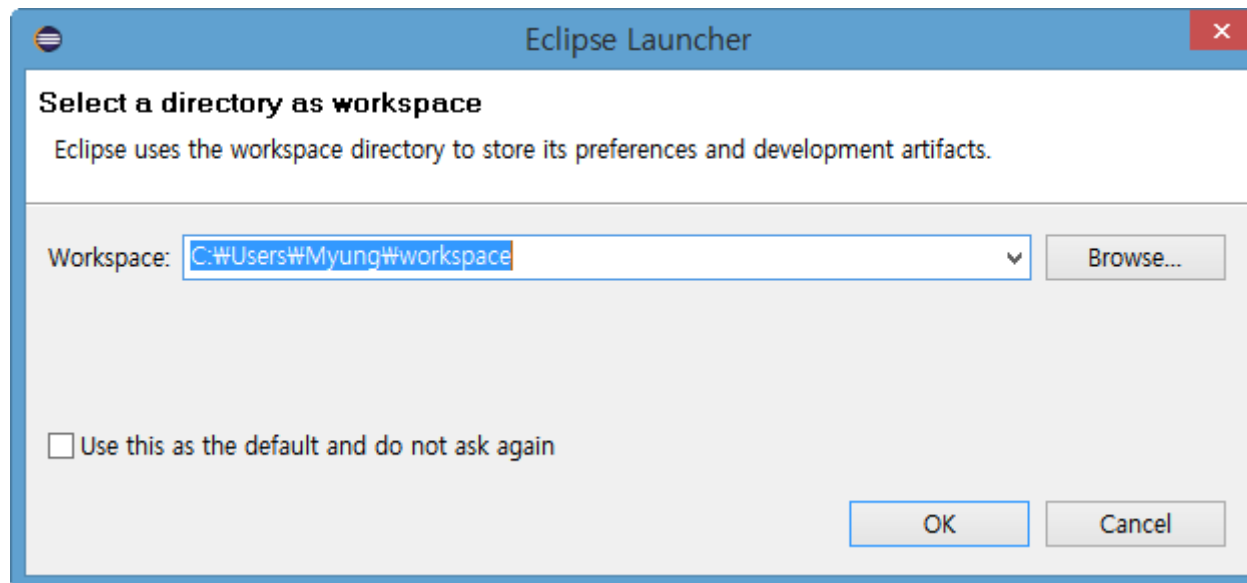


Eclipse 실행 (1/2)

29



프로젝트 저장할 폴더 지정



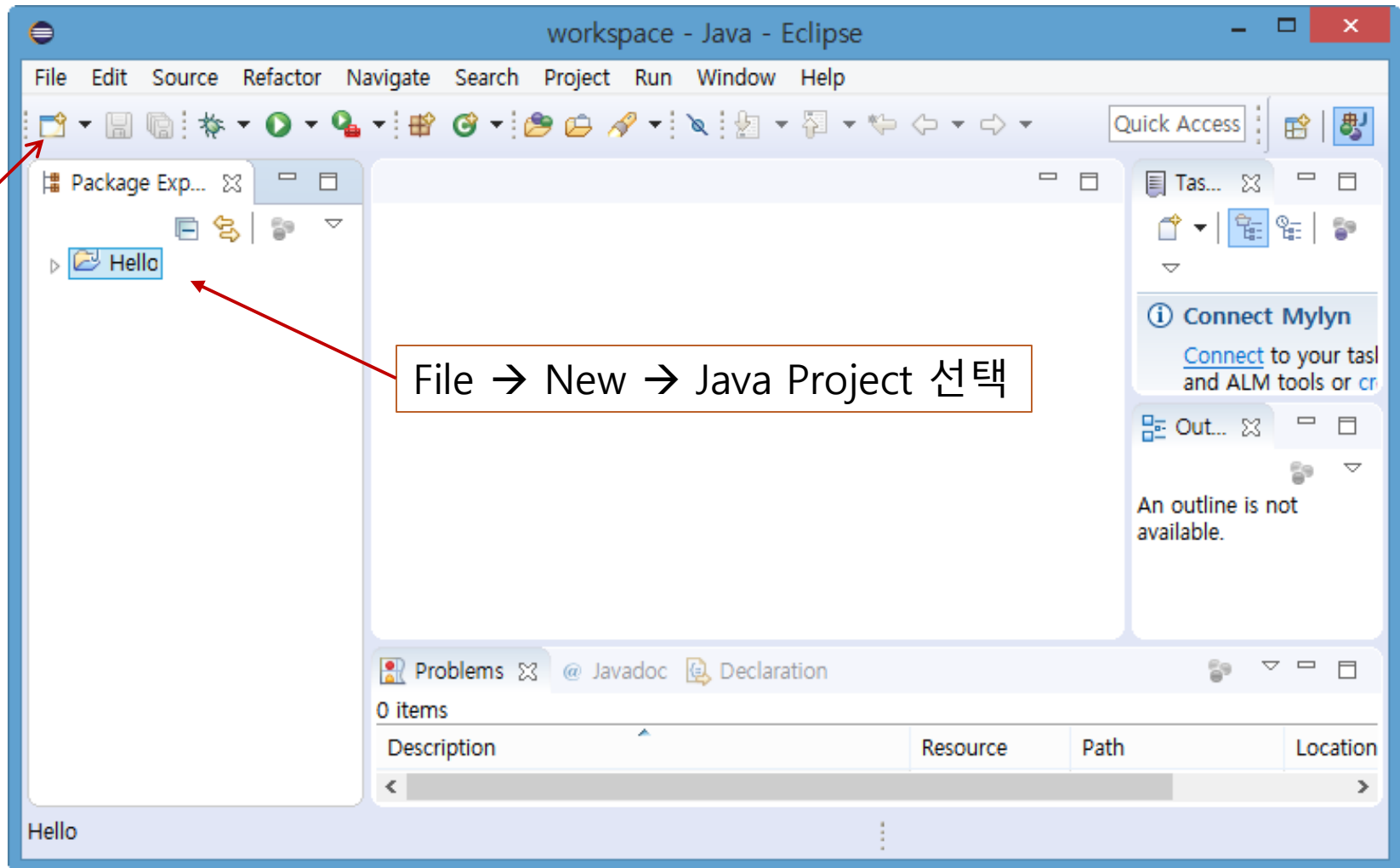
Eclipse 실행 (2/2)

30



자바 프로그램 개발 예제

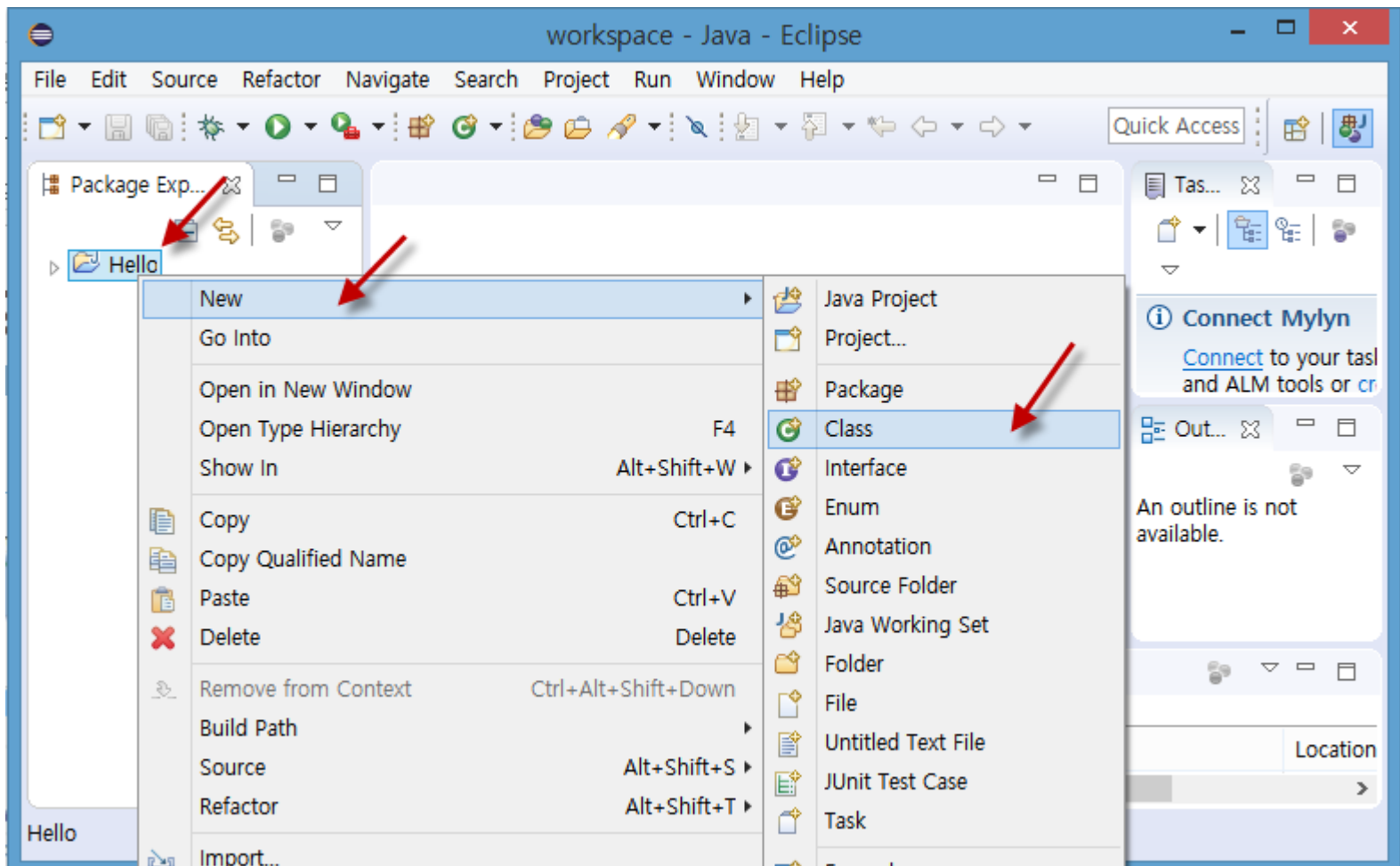
Eclipse : 프로젝트 생성



Eclipse : 클래스 생성

32

□ java class 생성



Eclipse : 클래스 생성 과정

33

New Java Class

Java Class
⚠ The use of the default package is discouraged.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

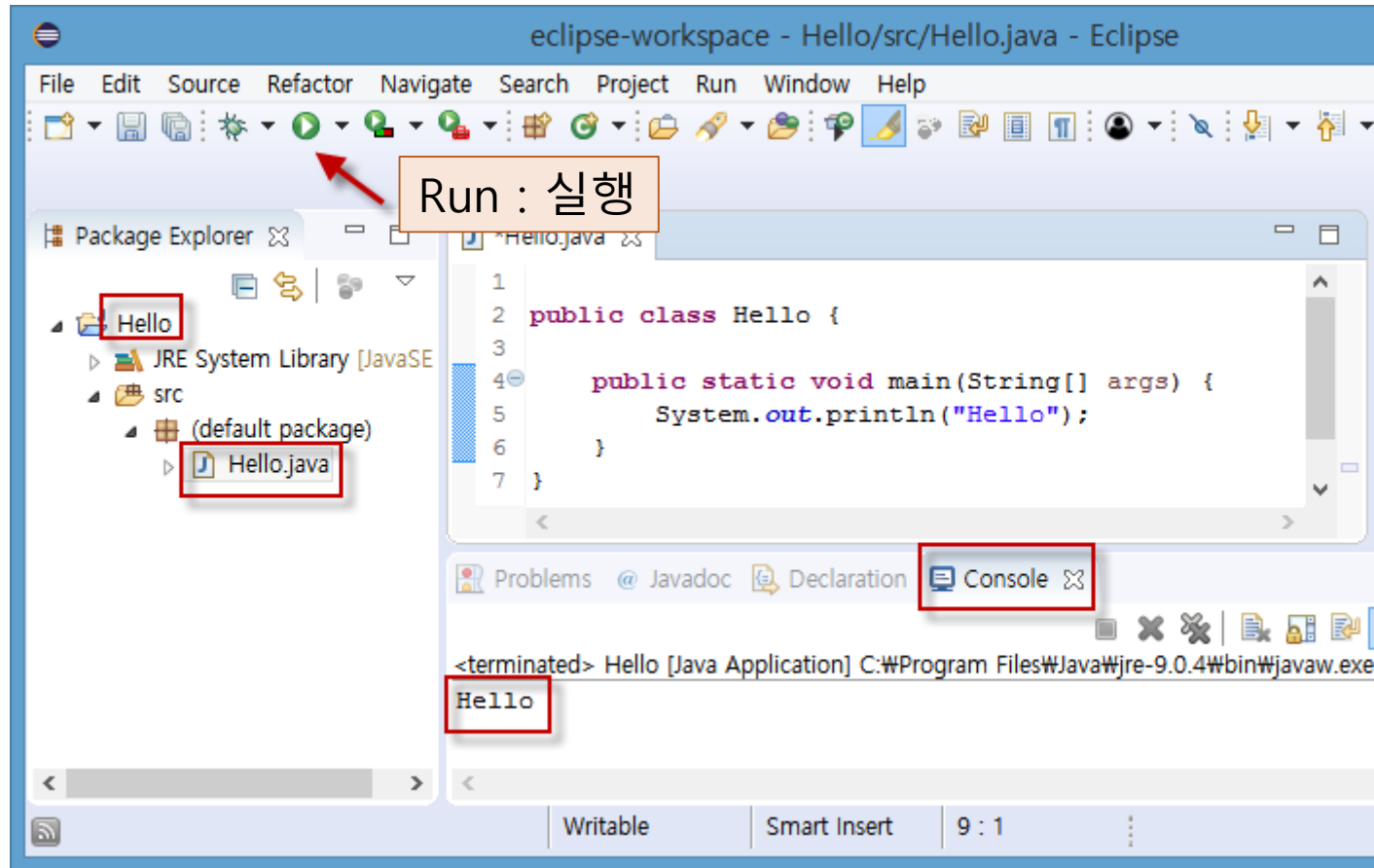
☒ `public static void main(String[] args)`
☐ Constructors from superclass
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

Eclipse : 클래스 코드 작성, 실행

34

- eclipse 에서 프로젝트 생성, 프로그램 작성과 실행



Hello 프로그램 설명

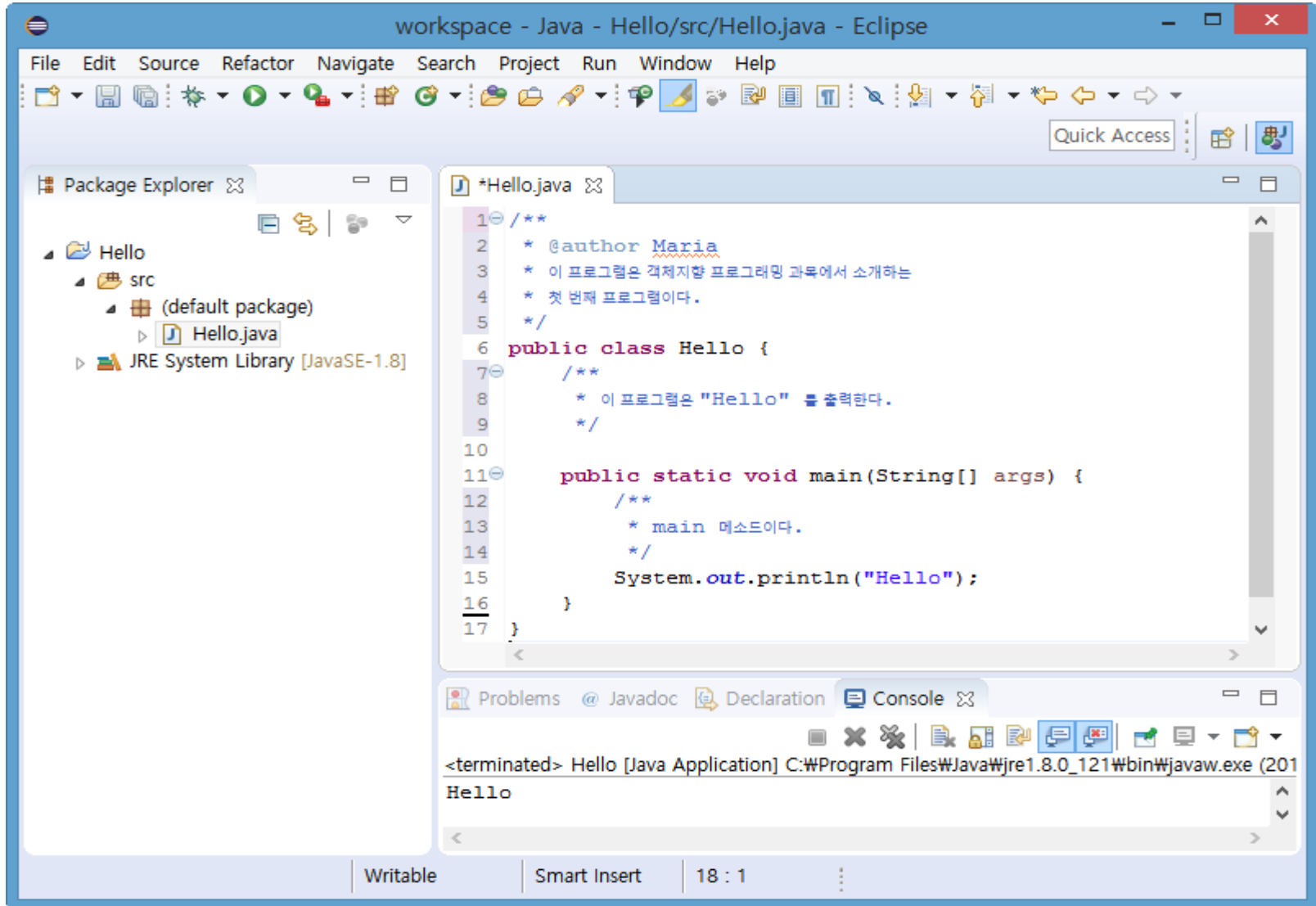
35

- public class Hello
 - ▣ 클래스의 선언
 - ▣ Hello 는 클래스 이름
 - ▣ 클래스는 { } 사이에 정의
 - ▣ 자바는 하나 이상의 클래스로 구성
- public static void main(String[] args)
 - ▣ 자바 프로그램은 main 메소드에서 실행 시작
 - ▣ 실행을 시작하는 클래스에 main() 메소드가 반드시 하나만 존재
- System.out.println("Hello");
 - ▣ 화면에 "Hello"을 출력하는 실행문
 - ▣ JDK에서 제공하는 java.lang.System.out 객체 이용

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

Javadoc : 자신만의 API 도큐먼트 생성 (1/5)

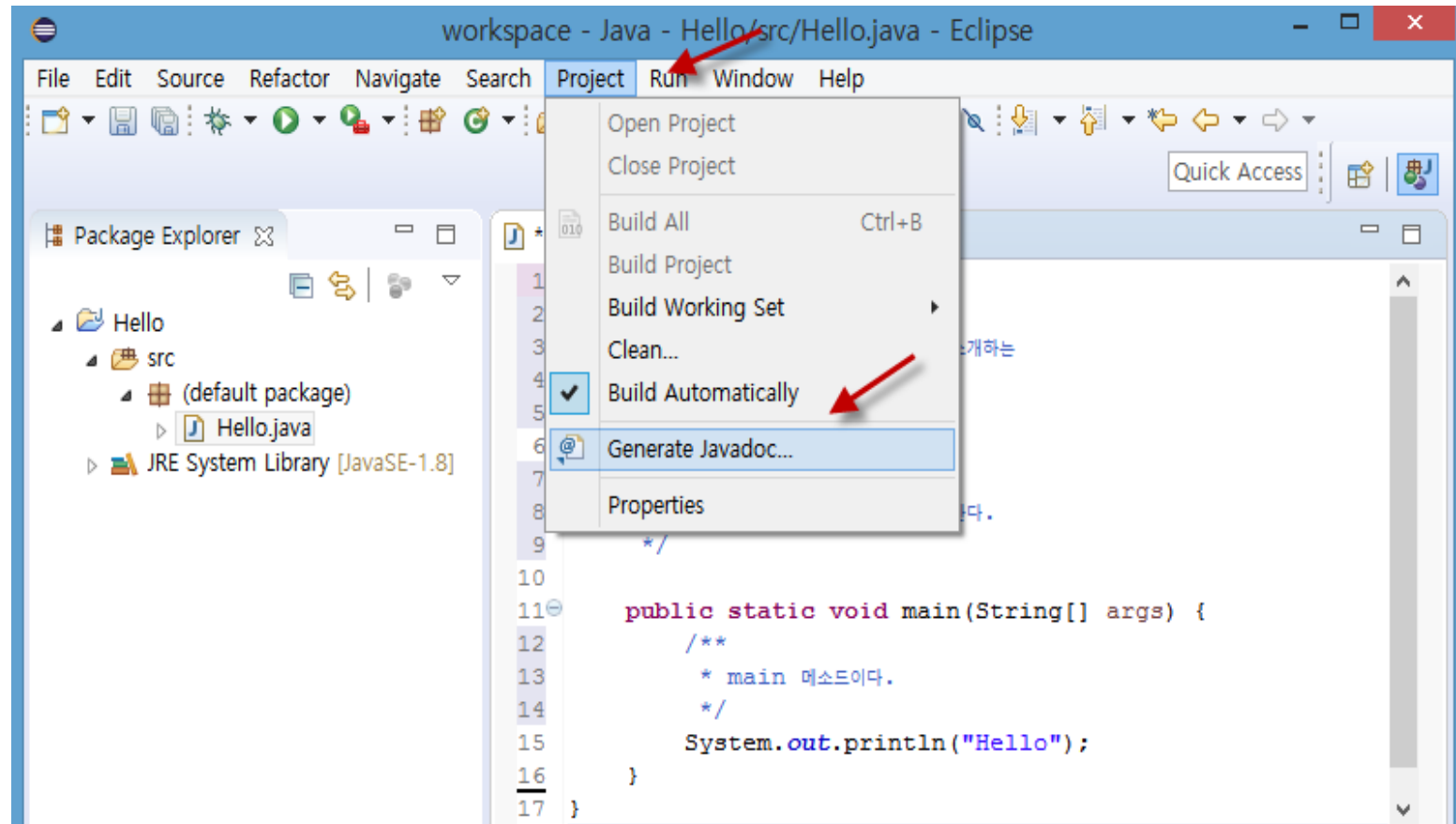
36



Javadoc : 자신만의 API 도큐먼트 생성 (2/5)

37

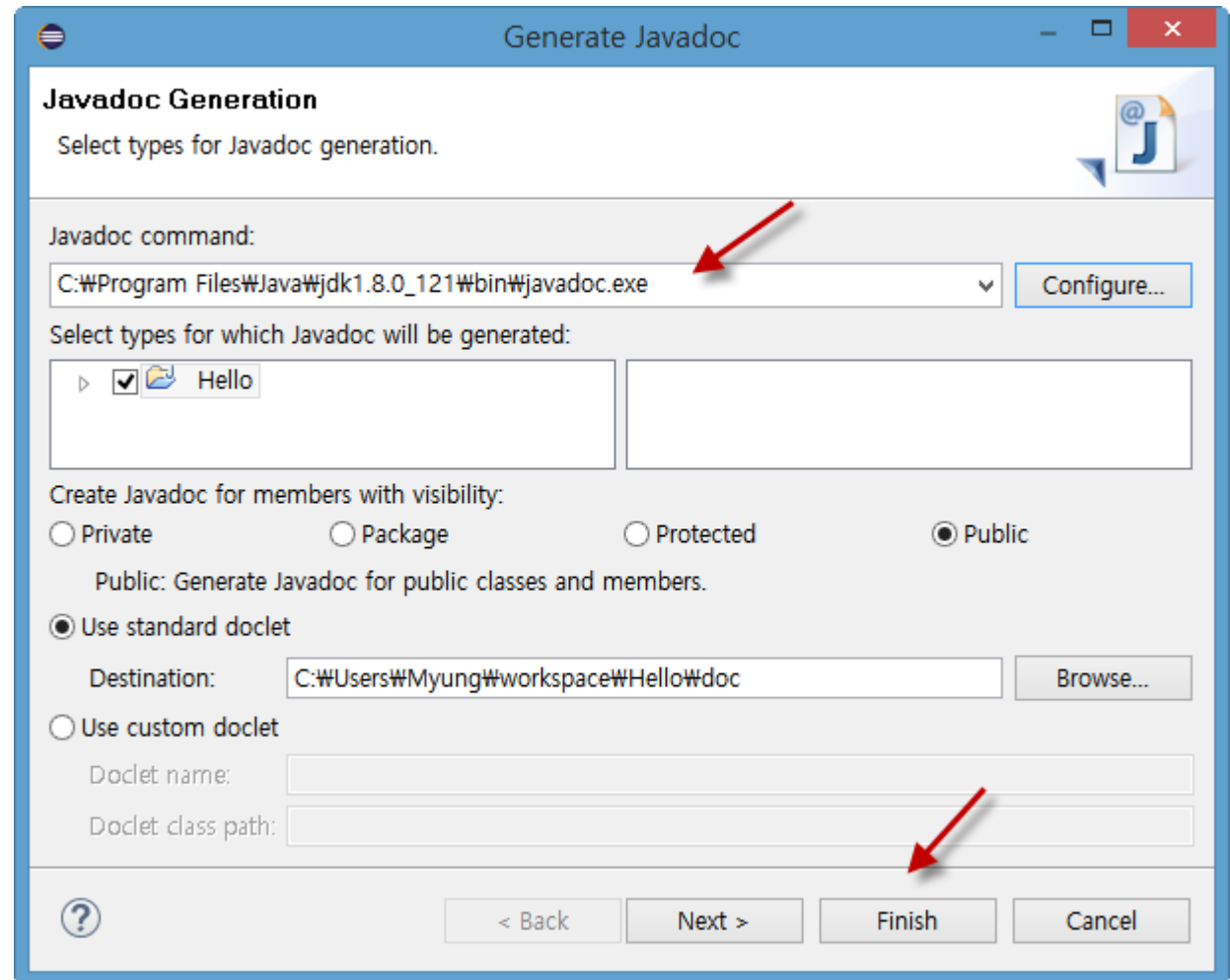
- Project → Generate Javadoc



Javadoc : 자신만의 API 도큐먼트 생성 (3/5)

38

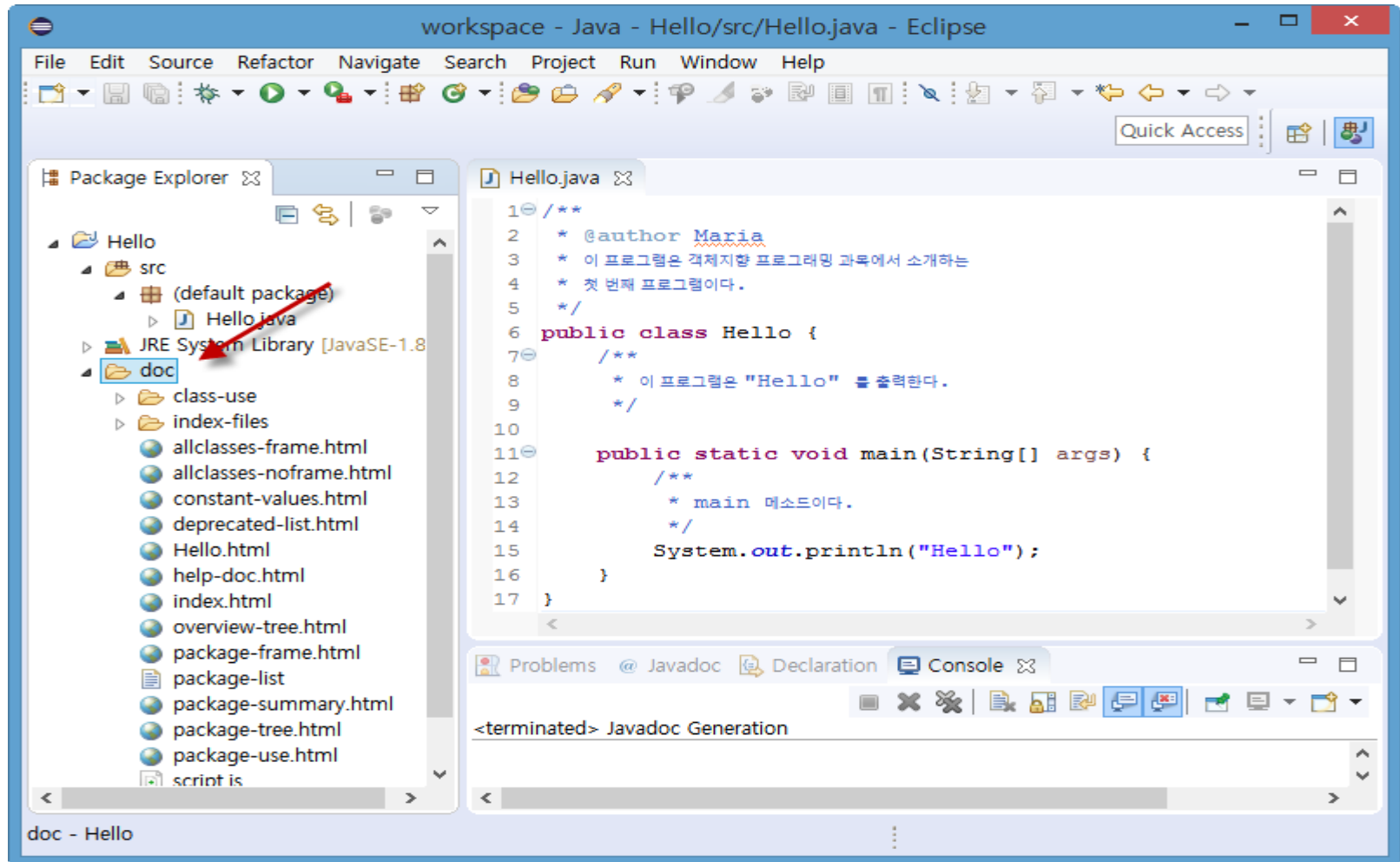
- Javadoc
command 패스
설정 후, finish



Javadoc : 자신만의 API 도큐먼트 생성 (4/5)

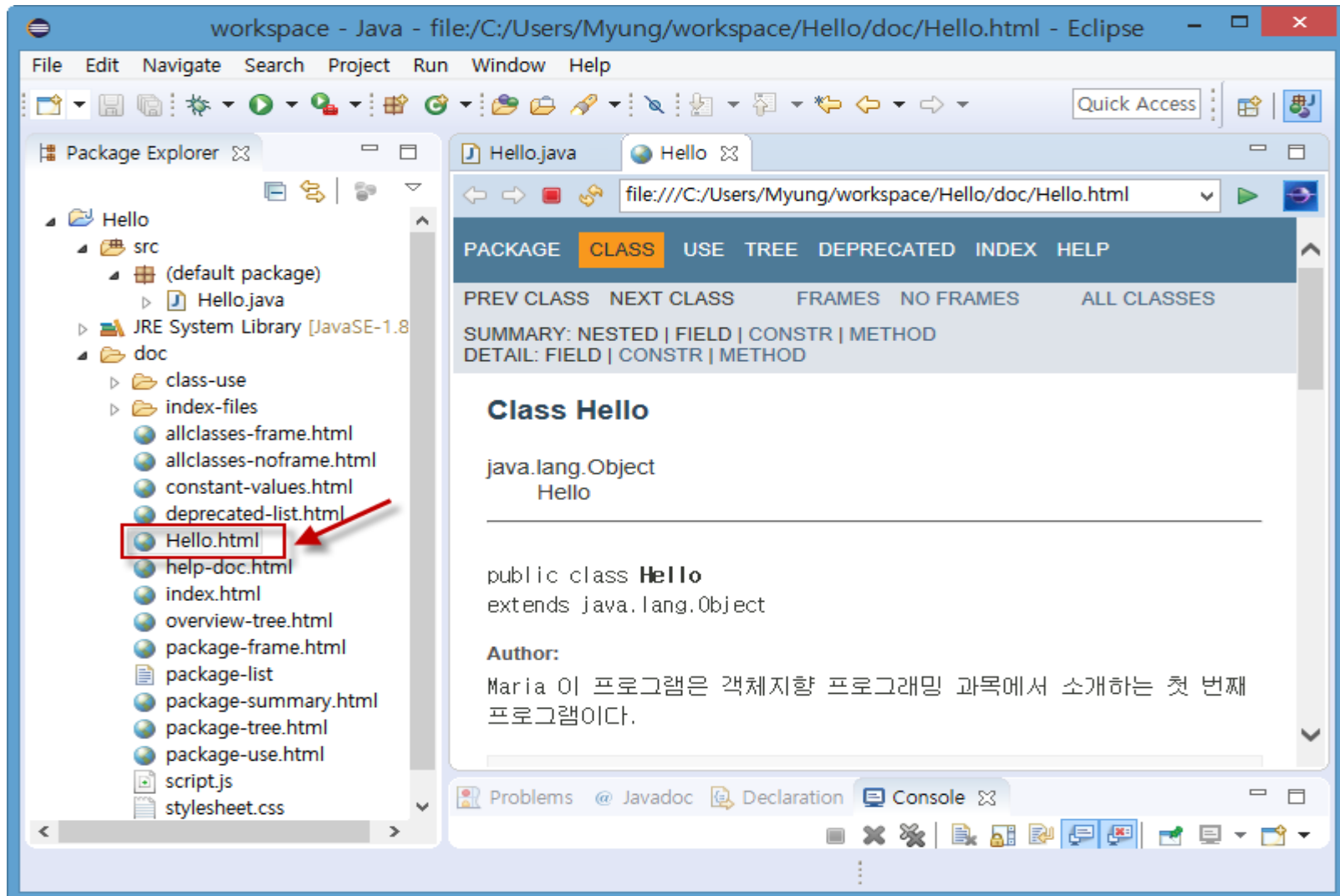
39

- doc 아래에 Hello.html 이 생성됨



Javadoc : 자신만의 API 도큐먼트 생성 (5/5)

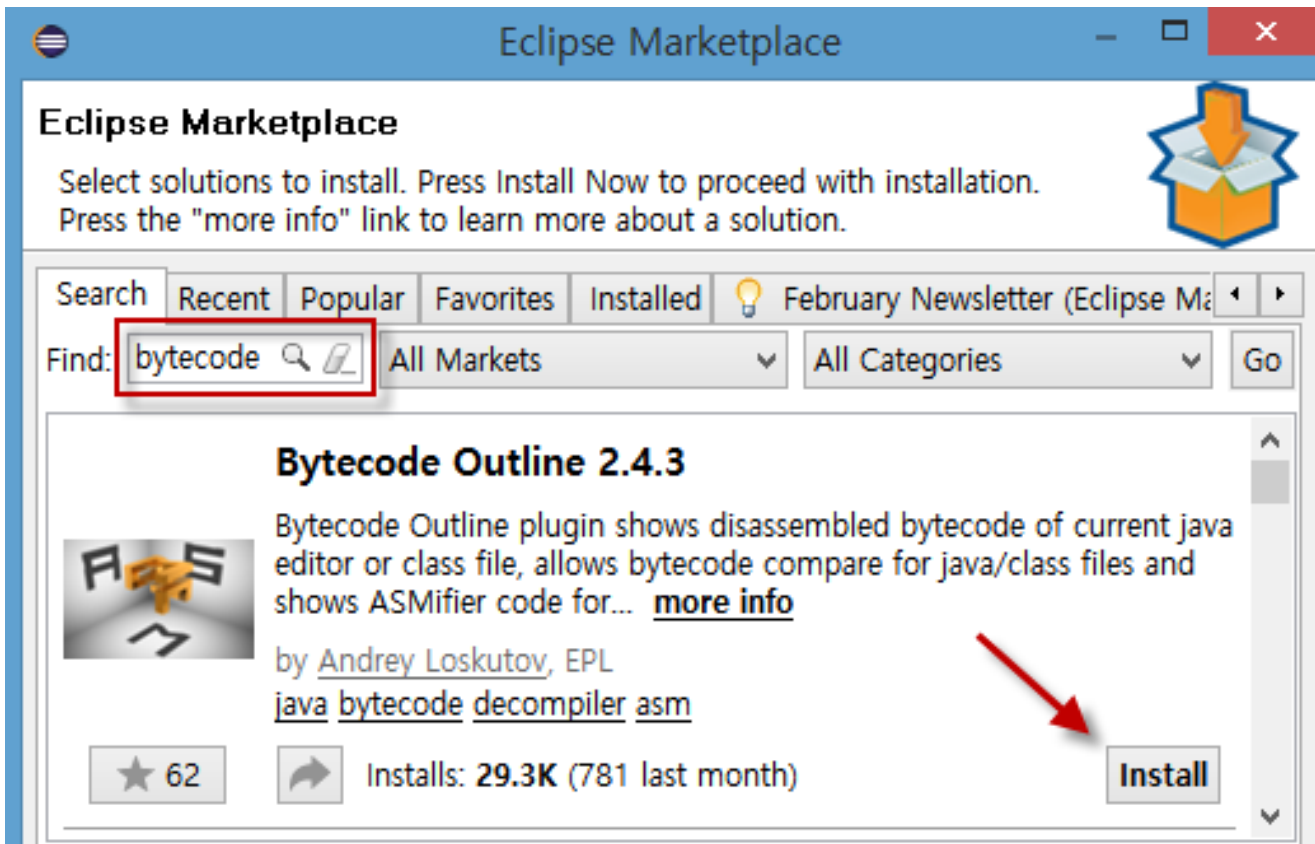
40



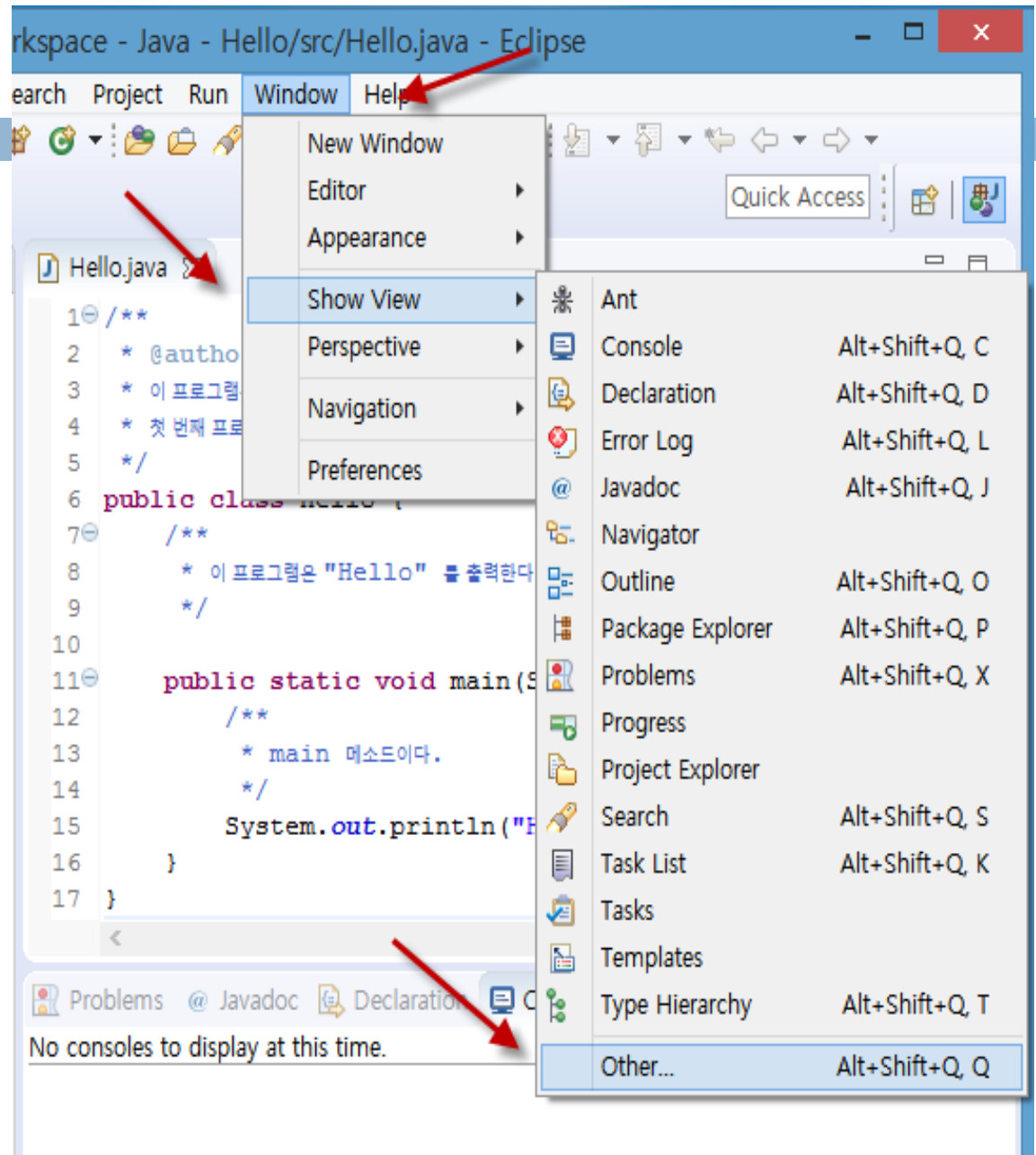
Eclipse 에서 bytecode 보기 기능 추가

41

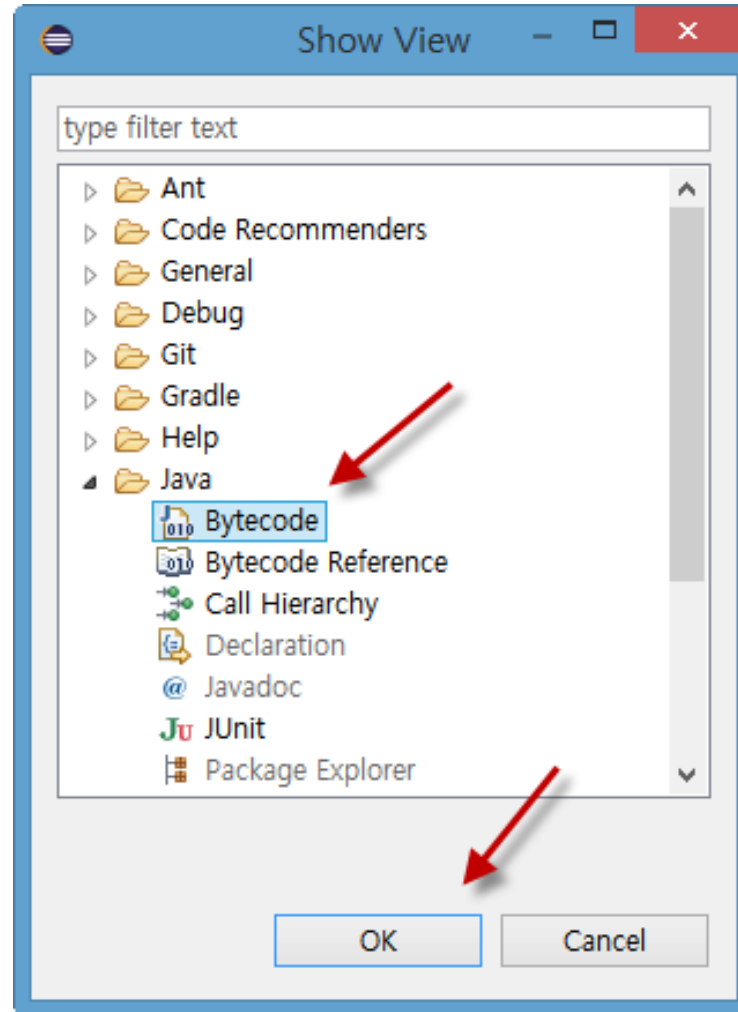
Help → Eclipse Marketplace → bytecode 검색 후 설치



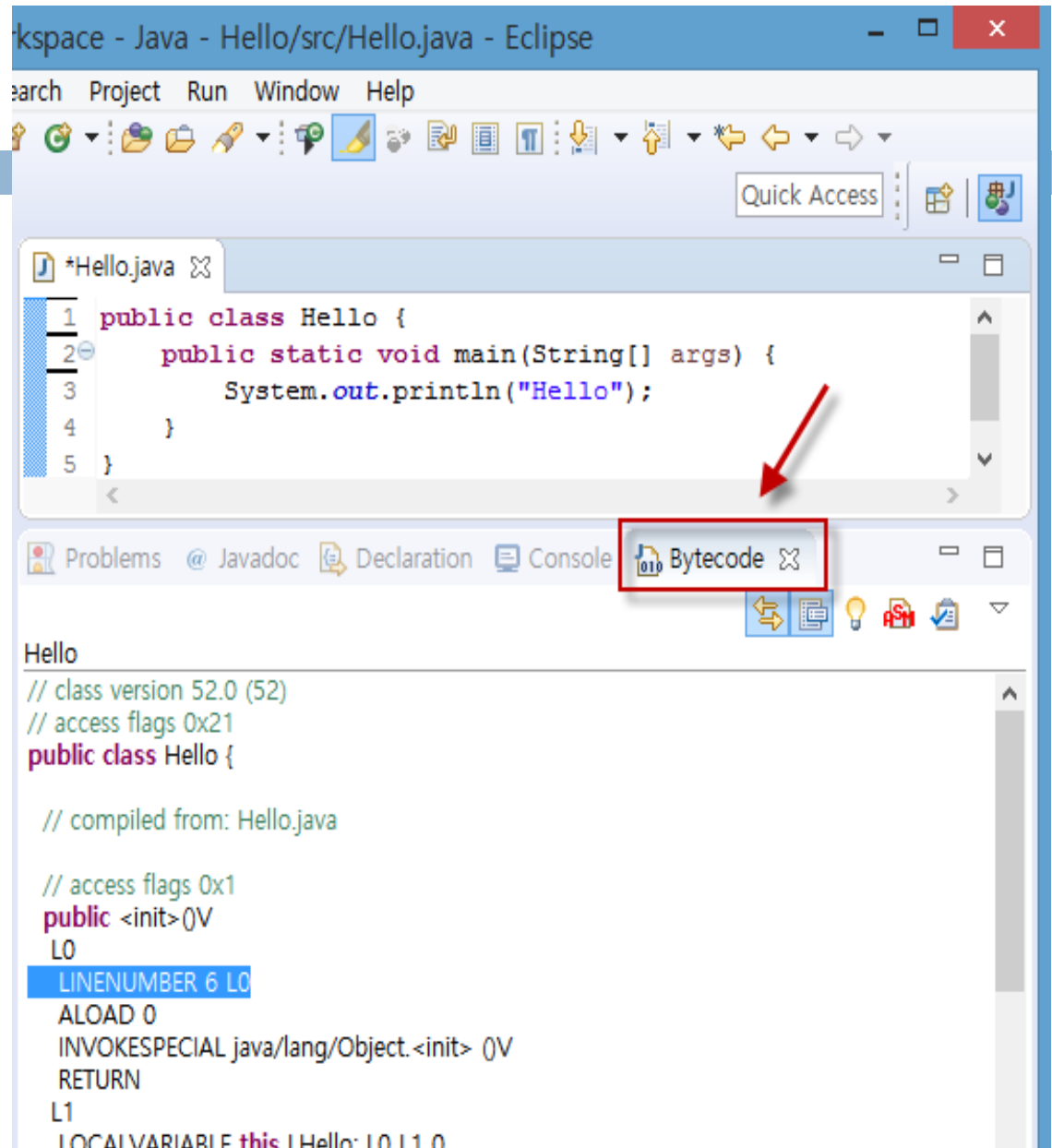
Eclipse 에서 bytecode 보기 (1/3)



Eclipse 에서 bytecode 보기 (2/3)



Eclipse 에서 bytecode 보기 (3/3)



요약 (1/2)

45

- 자바는 플랫폼 독립적, 객체지향 프로그래밍 언어이다.
- 자바 배포판: Java SE, Java ME, Java EE 이 있다.
- 자바 프로그램을 개발하고 실행하기 위한 환경은 JDK 에서 제공, IDE 환경을 제공하는 도구로는 이클립스(Eclipse) 가 있다.
- 자바 프로그램은 클래스 정의로 시작하며, 클래스 내에서 모든 변수나 메소드를 정의한다. 실행은 `main()` 메소드부터 시작한다.
- 자바 프로그램은 하드웨어 독립적인 바이트 코드로 컴파일되며, 이는 JVM 상에서 실행된다. (`javap` 사용 또는 eclipse 안에서 바이트코드 보기 기능을 추가하면 바이트 코드를 볼 수 있다.)

요약 (2/2)

46

- 자바 소스코드의 확장자는 .java 이고, 컴파일된 파일의 확장자는 .class 이다.
- 하나의 클래스 파일에는 반드시 하나의 자바 클래스가 포함되어 있어야 한다.
- 다수의 클래스 파일로 자바 응용프로그램이 구성된 경우 jar 파일 형태로 압축하여 배포 및 실행이 가능하다.
- Javadoc 을 사용하여 자신이 만든 프로그램의 document 를 손쉽게 html 형식의 파일로 만들 수 있다.