

프로그래밍 기초

▶ 프로그래밍

✓ 프로그램(Program)

컴퓨터가 인식할 수 있는 명령어의 나열(집합)

✓ 프로그래밍(Programming)

프로그램을 작성하는 과정 = 코딩

✓ 프로그래머(Programmer)

프로그램을 작성하는 사람

▶ 프로그래밍 언어

프로그램을 작성하기 위한 언어체계, 사람이 컴퓨터와 소통하게 하는 요소

Worldwide, Jun 2020 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	31.6 %	+4.3 %
2		Java	17.67 %	-2.4 %
3		Javascript	8.02 %	-0.2 %
4		C#	6.87 %	-0.4 %
5		PHP	6.02 %	-0.9 %
6		C/C++	5.69 %	-0.2 %
7		R	3.86 %	-0.1 %
8		Objective-C	2.5 %	-0.3 %
9		Swift	2.24 %	-0.1 %
10	↑	TypeScript	1.86 %	+0.2 %
11	↓	Matlab	1.72 %	-0.2 %
12	↑	Kotlin	1.54 %	+0.2 %
13	↑↑	Go	1.29 %	+0.2 %
14		VBA	1.24 %	-0.1 %
15	↓↓↓	Ruby	1.23 %	-0.1 %
16		Scala	1.0 %	-0.1 %
17		Visual Basic	0.88 %	-0.2 %
18	↑↑	Rust	0.78 %	+0.2 %
19	↑↑↑↑↑	Dart	0.5 %	+0.2 %
20	↓↓	Perl	0.46 %	-0.1 %

Jun 2020	Jun 2019	Change	Programming Language	Ratings	Change
1	2	↑	C	17.19%	+3.89%
2	1	↓	Java	16.10%	+1.10%
3	3		Python	8.36%	-0.16%
4	4		C++	5.95%	-1.43%
5	6	↑	C#	4.73%	+0.24%
6	5	↓	Visual Basic	4.69%	+0.07%
7	7		JavaScript	2.27%	-0.44%
8	8		PHP	2.26%	-0.30%
9	22	↑↑	R	2.19%	+1.27%
10	9	↓	SQL	1.73%	-0.50%
11	11		Swift	1.46%	+0.04%
12	15	↑	Go	1.02%	-0.24%
13	13		Ruby	0.98%	-0.41%
14	10	↓↓	Assembly language	0.97%	-0.51%
15	18	↑	MATLAB	0.90%	-0.18%
16	16		Perl	0.82%	-0.36%
17	20	↑	PL/SQL	0.74%	-0.19%
18	26	↑↑	Scratch	0.73%	+0.20%
19	19		Classic Visual Basic	0.65%	-0.42%
20	38	↑↑	Rust	0.64%	+0.38%

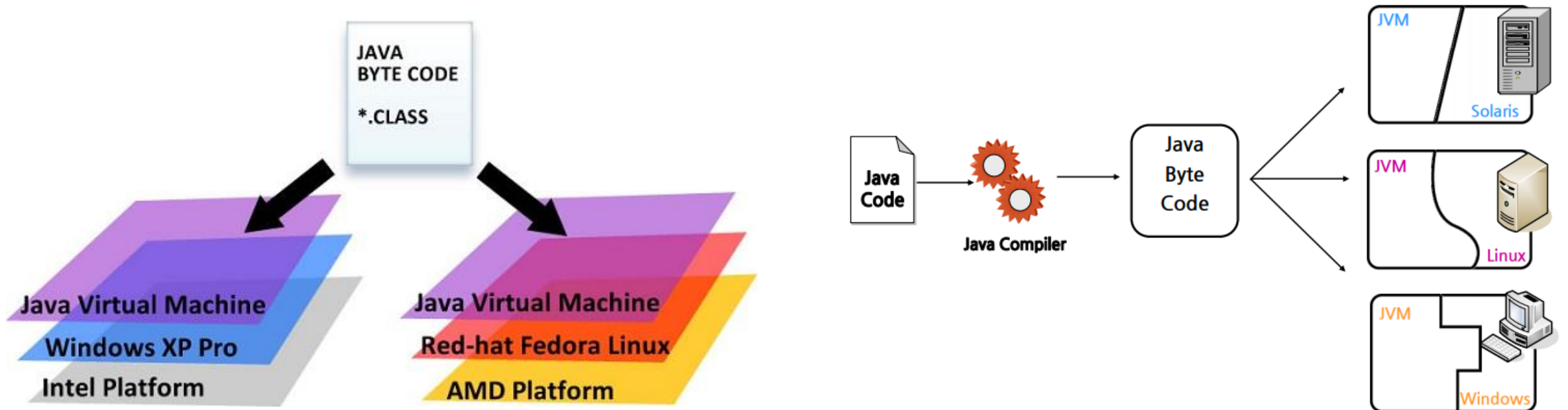
▶ 자바 언어 특징



1. 운영체제에 독립적(이식성이 높음)
2. 사용하기 쉬운 언어
 - 다른 언어의 단점 보완
 - 객체 지향 언어
 - 능률적이고 명확한 코드 작성 가능
3. 자동 메모리 관리(Garbage Collection)
4. 네트워크와 분산환경 지원
5. 멀티쓰래드 지원
6. 동적 로딩 지원

▶ JVM(Java Virtual Machine)

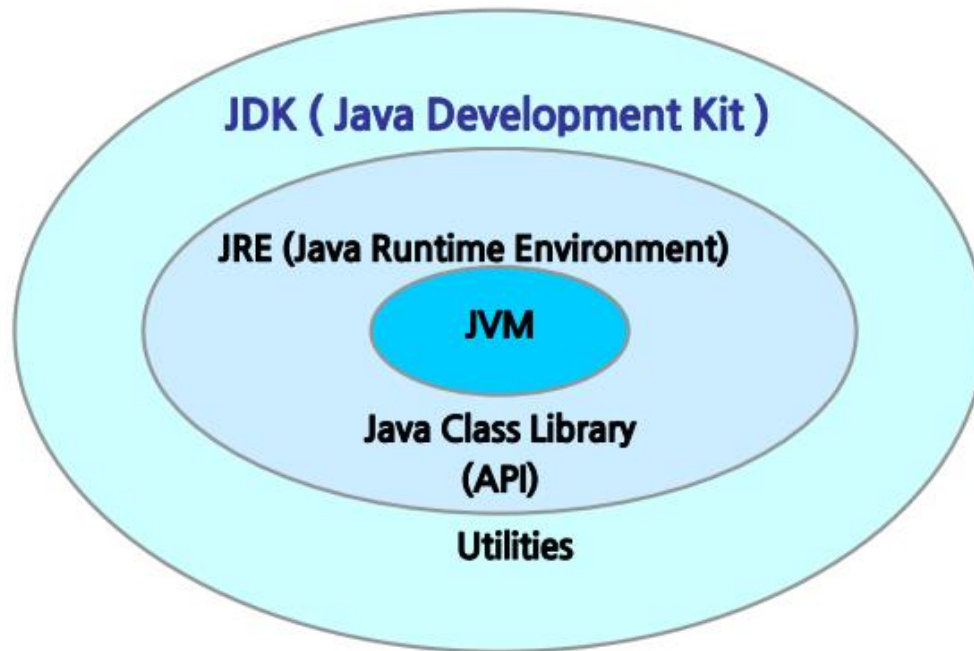
자바를 실행하기 위한 가상 기계로 플랫폼에 의존적
byte code(class파일)를 해석하고 실행하는 interpreter



▶ 자바 개발 환경

✓ 설치 범위

사용자 / 개발자 입장에 따라 설치하는 범위가 달라짐



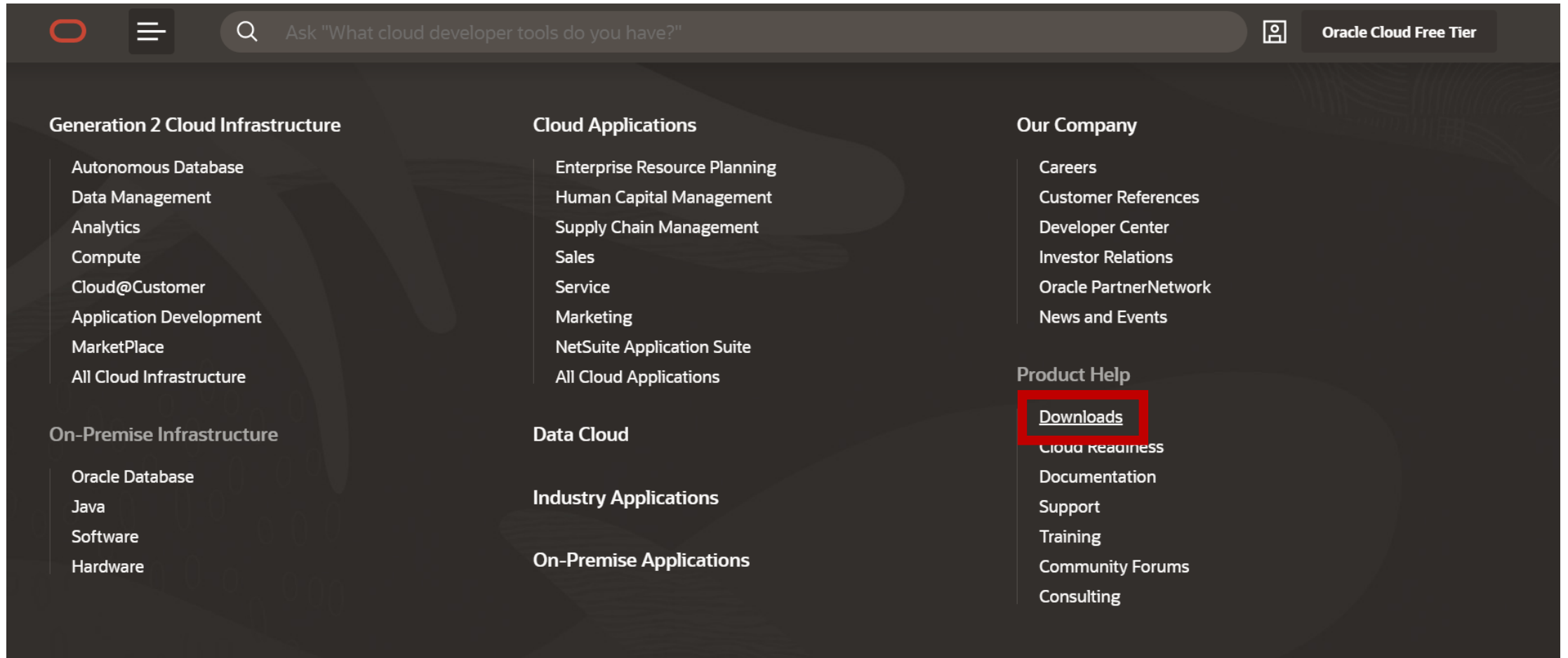
Java SE : Java Standard Edition

Java EE : Java Enterprise Edition

Java ME : Java Micro Edition

▶ 자바 설치

✓ Oracle 홈페이지 접속(www.oracle.com) - 메뉴 - 다운로드 클릭



▶ 자바 설치

✓ Java 탭 – Java(JDK) for Developers 클릭

Java

Java (JRE) for Consumers

Java (JDK) for Developers

Event Processing for Java Embedded

Java Card

Java EE & GlassFish Server

Java Embedded Suite

Java for Mobile

Java ME

Java ME Embedded

Java ME Embedded Client

Java ME SDK

Java Runtime Environment (JRE)

Java SE

Java SE Embedded

Java TV

▶ 자바 설치

✓ Java SE 8u231 / Java SE 8u231 - JDK DOWNLOAD 클릭

Java SE 8u231

Java SE 8u231 includes important bug fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.

[Learn more](#) ▶

- [Installation Instructions](#)
- [Release Notes](#)
- [Oracle License](#)
- [Java SE Licensing Information User Manual](#)
 - [Includes Third Party Licenses](#)
- [Certified System Configurations](#)
- [Readme Files](#)
 - [JDK ReadMe](#)
 - [JRE ReadMe](#)

JDK

DOWNLOAD ⬇

Server JRE

DOWNLOAD ⬇

JRE

DOWNLOAD ⬇

Which Java package do I need?

▶ 자바 설치

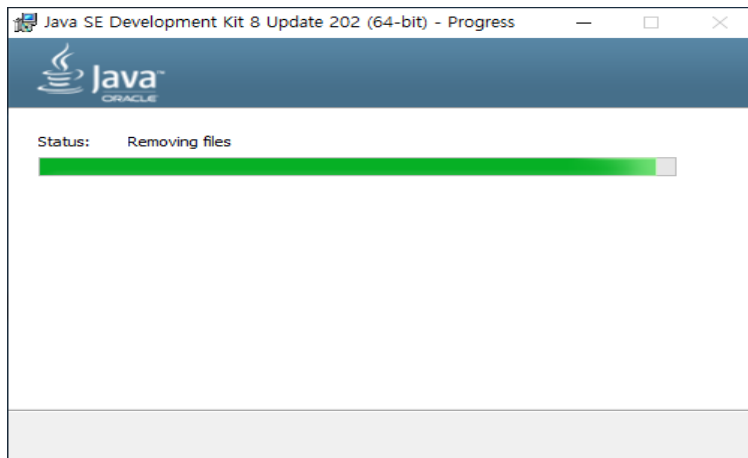
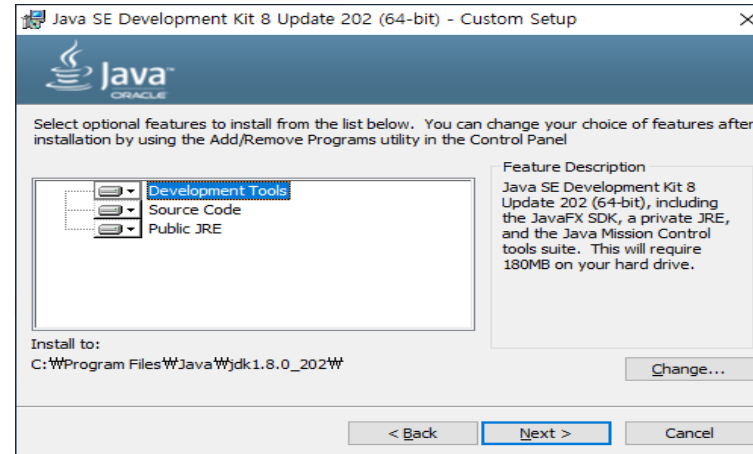
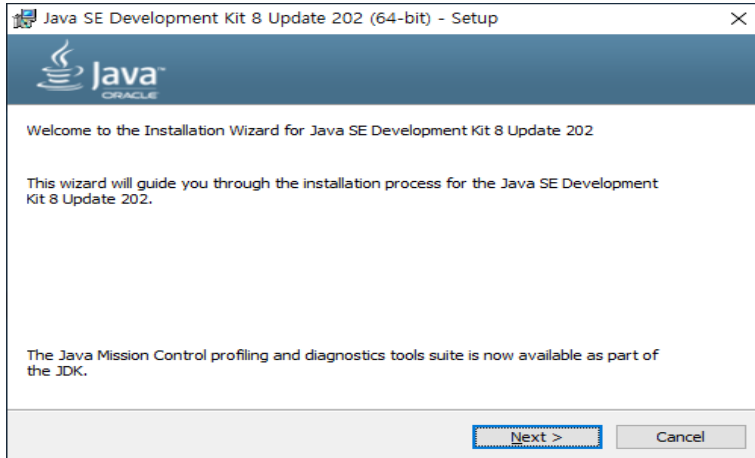
✓ Java SE Development Kit 8u231 - Accept License Agreement 체크

- 운영체제에 맞는 exe 파일 다운로드

Java SE Development Kit 8u231		
You must accept the Oracle Technology Network License Agreement for Oracle Java SE to download this software.		
<input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.9 MB	jdk-8u231-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	69.8 MB	jdk-8u231-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.93 MB	jdk-8u231-linux-i586.rpm
Linux x86	185.75 MB	jdk-8u231-linux-i586.tar.gz
Linux x64	170.32 MB	jdk-8u231-linux-x64.rpm
Linux x64	185.16 MB	jdk-8u231-linux-x64.tar.gz
Mac OS X x64	253.4 MB	jdk-8u231-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	132.98 MB	jdk-8u231-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	94.16 MB	jdk-8u231-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	133.73 MB	jdk-8u231-solaris-x64.tar.Z
Solaris x64	91.96 MB	jdk-8u231-solaris-x64.tar.gz
Windows x86	200.22 MB	jdk-8u231-windows-i586.exe
Windows x64	210.18 MB	jdk-8u231-windows-x64.exe

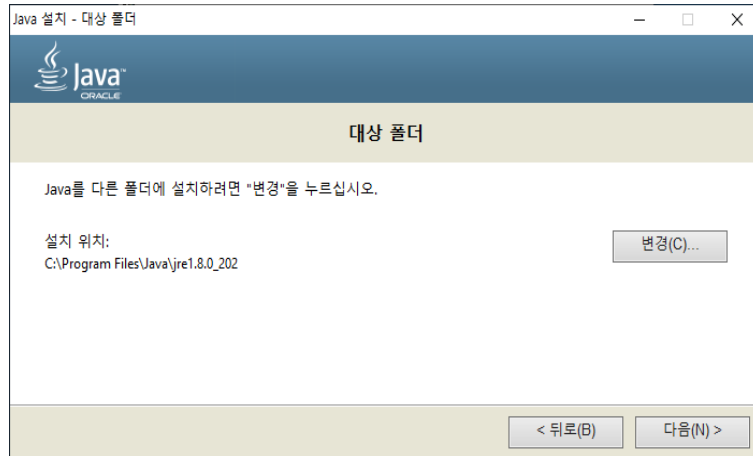
▶ 자바 설치

✓ 모두 Default로 놓고 next, 다음 버튼 클릭

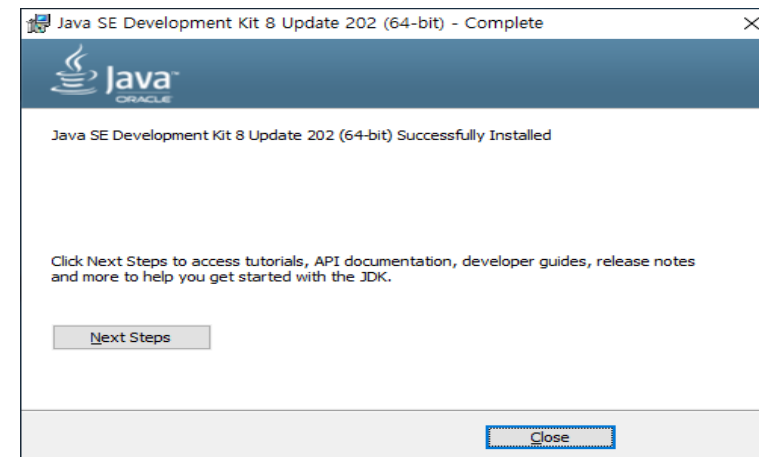


▶ 자바 설치

✓ 모두 Default로 놓고 next, 다음 버튼 클릭

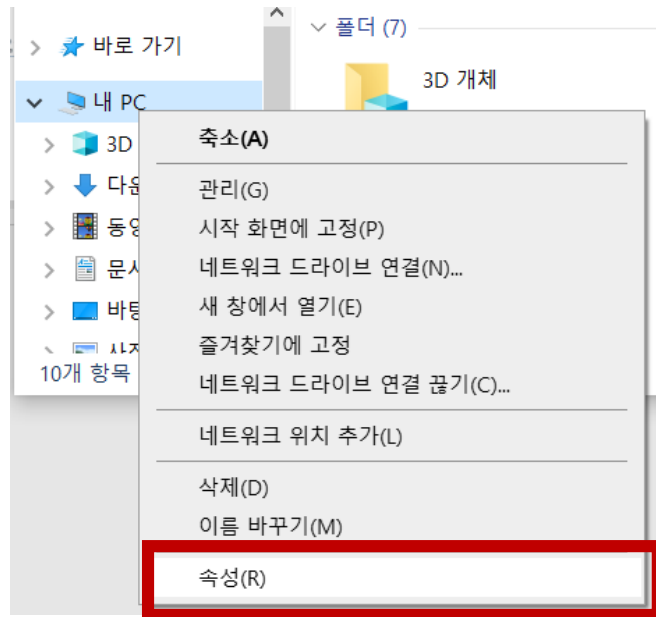


>> 폴더 변경 시 폴더 위치 기억할 것

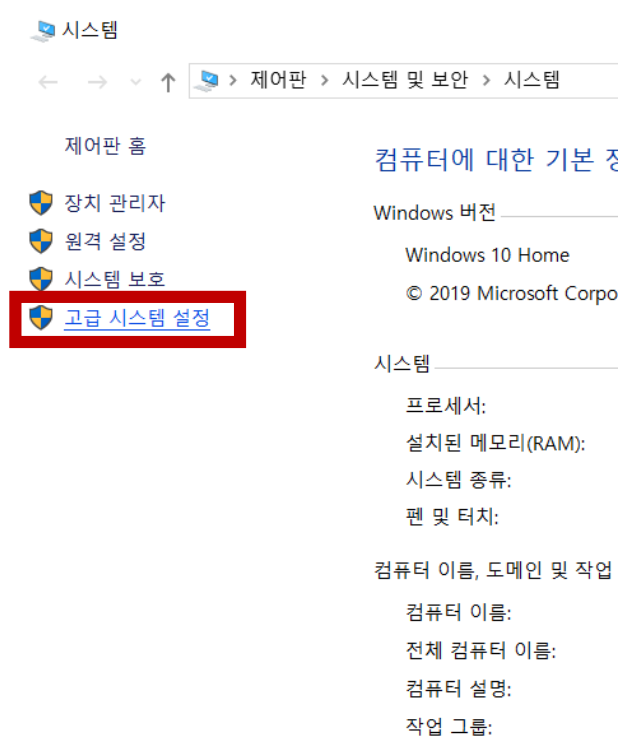


▶ 자바 설정

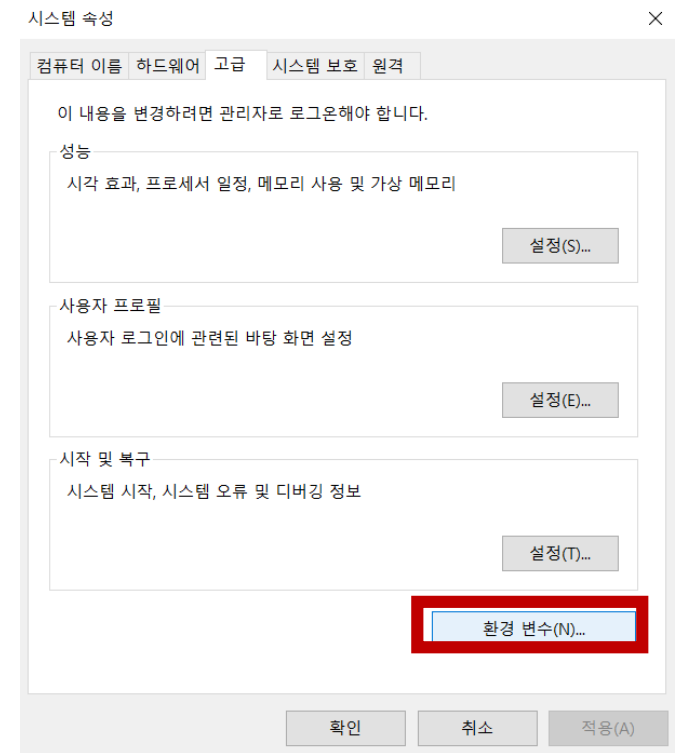
✓ '내 PC' 오른쪽 클릭 - 속성



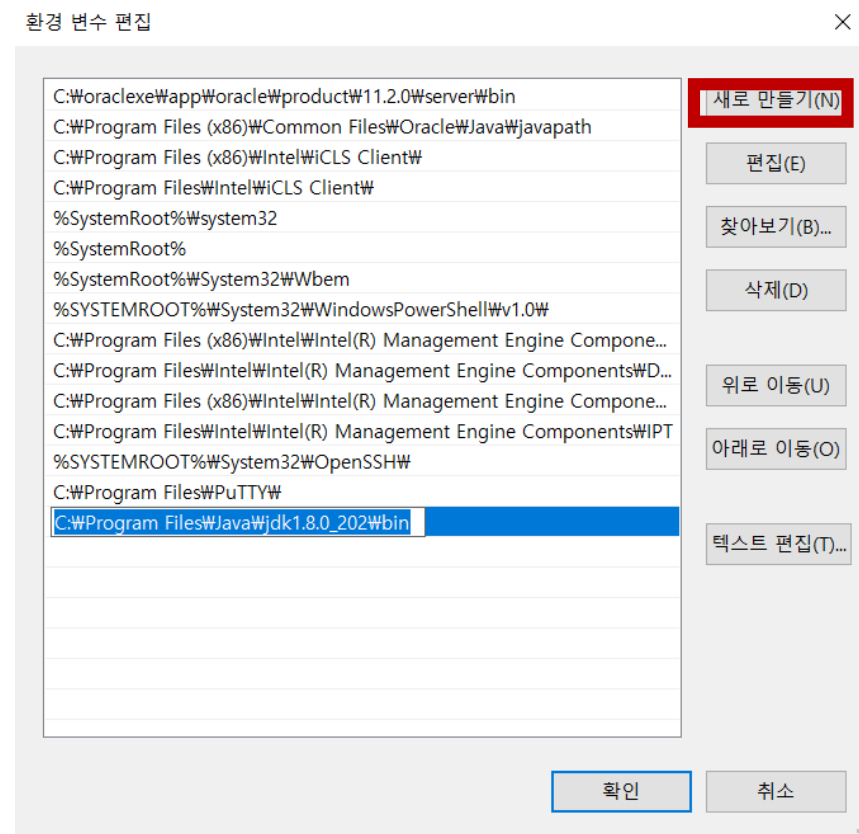
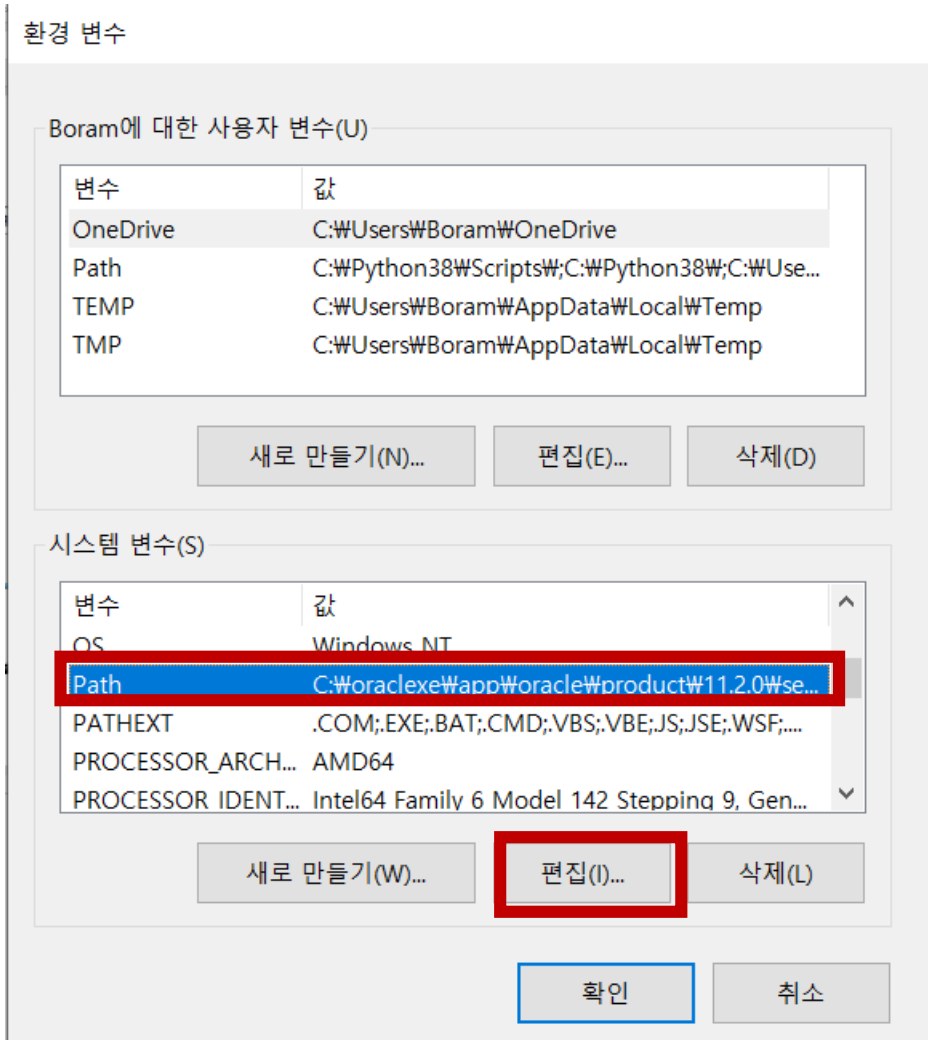
✓ 고급시스템 설정



✓ 고급 - 환경변수



▶ 자바 설정



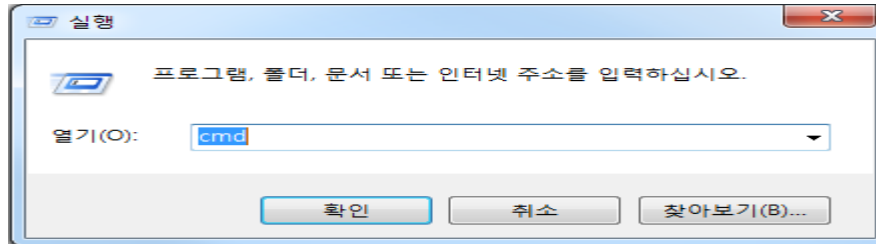
1) 새로만들기

2) JDK가 설치된 폴더 경로 작성

(C:\Program Files\Java\jdk1.8.0_202\bin)

▶ 자바 설정

✓ 환경변수 테스트



윈도우키 + R 버튼 누르고 실행창에서 cmd엔터

```
Microsoft Windows [Version 10.0.17134.648]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\>java -version
java version "1.8.0_202"
Java(TM) SE Runtime Environment (build 1.8.0_202-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.202-b08, mixed mode)

C:\>javac -version
javac 1.8.0_202

C:\>
```

java -version 엔터 / javac -version 엔터
위 화면같이 설치된 버전 정보 나오면 설정완료

▶ 자바 개발 환경

✓ Compile Test

1. Workspace (작업환경폴더) 만들기

C드라이브 폴더에 workspace 폴더명으로 새폴더 생성

2. HelloWorld.java 파일 생성

메모장을 활용하여 아래의 코드 작성 후 파일 저장 시 workspace 폴더 안에 HelloWorld.java 파일명으로 저장

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


▶ 자바 개발 환경

✓ Compile Test

3. 실행을 위해 cmd 창에서 workspace 폴더 경로 찾기

```
C:\> cd C:\workspace
```

4. HelloWorld.java 파일 compile 하기 (실행하기 위해 byte code로 변환시키는거임)

```
C:\workspace> javac HelloWorld.java
```

5. compile 된 클래스 실행 하기

```
C:\workspace> java HelloWorld
```


6. 실행 잘 되는지 확인하기

▶ 이클립스 설치

✓ 이클립스 다운로드 홈페이지(<https://www.eclipse.org/downloads/>)



Download Eclipse Technology
that is right for you




Get **Eclipse IDE 2019-09**
Install your favorite desktop IDE packages.

[Download 64 bit](#)

[Download Packages](#) | [Need Help?](#)

Tool Platf



Eclipse (

Eclipse Che is a developer
workspace server and cloud IDE.

MORE DOWNLOADS

- Other builds
- Eclipse 2019-06 (4.12)
- **Eclipse 2019-03 (4.11)**
- Eclipse 2018-12 (4.10)
- Eclipse 2018-09 (4.9)
- Eclipse Photon (4.8)
- Eclipse Oxygen (4.7)
- Eclipse Neon (4.6)
- Eclipse Mars (4.5)
- Eclipse Luna (4.4)
- Older Versions

Eclipse Packages | The Eclipse F x +


← → ↻ ⌂ 🔒 <https://www.eclipse.org/downloads/packages/>

Get **Eclipse IDE 2019-03**
Install your favorite desktop IDE packages.

[Download 64 bit](#)

[Download Packages](#) | [Need Help?](#)


Eclipse IDE 2019-03 R Packages



Eclipse IDE for Enterprise Java Developers
340 MB | 89,178 DOWNLOADS

Tools for Java developers creating Enterprise Java and Web applications, including a Java IDE, tools for Enterprise Java, JPA, JSF, Mylyn, Maven, Git and more. Click here to file a bug against Eclipse Web Tools Platform. Click here to file a bug against Eclipse Platform. Click here to file a bug against Maven integration for web projects.


[Windows 64-bit](#)
[Mac Cocoa 64-bit](#)
[Linux 64-bit](#)



Eclipse IDE for Java Developers
189 MB | 61,378 DOWNLOADS

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven and Gradle integration


[Windows 64-bit](#)
[Mac Cocoa 64-bit](#)
[Linux 64-bit](#)



Eclipse IDE for C/C++ Developers
229 MB | 17,628 DOWNLOADS

An IDE for C/C++ developers with Mylyn integration.

[Windows 64-bit](#)
[Mac Cocoa 64-bit](#)
[Linux 64-bit](#)



Eclipse IDE for Eclipse Committers
316 MB | 7,614 DOWNLOADS

Package suited for development of Eclipse itself at Eclipse.org: based on the Eclipse Platform adding PDE, Git, Marketplace Client, source code and developer documentation.

[Windows 64-bit](#)
[Mac Cocoa 64-bit](#)
[Linux 64-bit](#)

RELATED LINKS

- Compare & Combine Packages
- New and Noteworthy
- Install Guide
- Documentation
- Updating Eclipse
- Forums
- Simultaneous Release

MORE DOWNLOADS

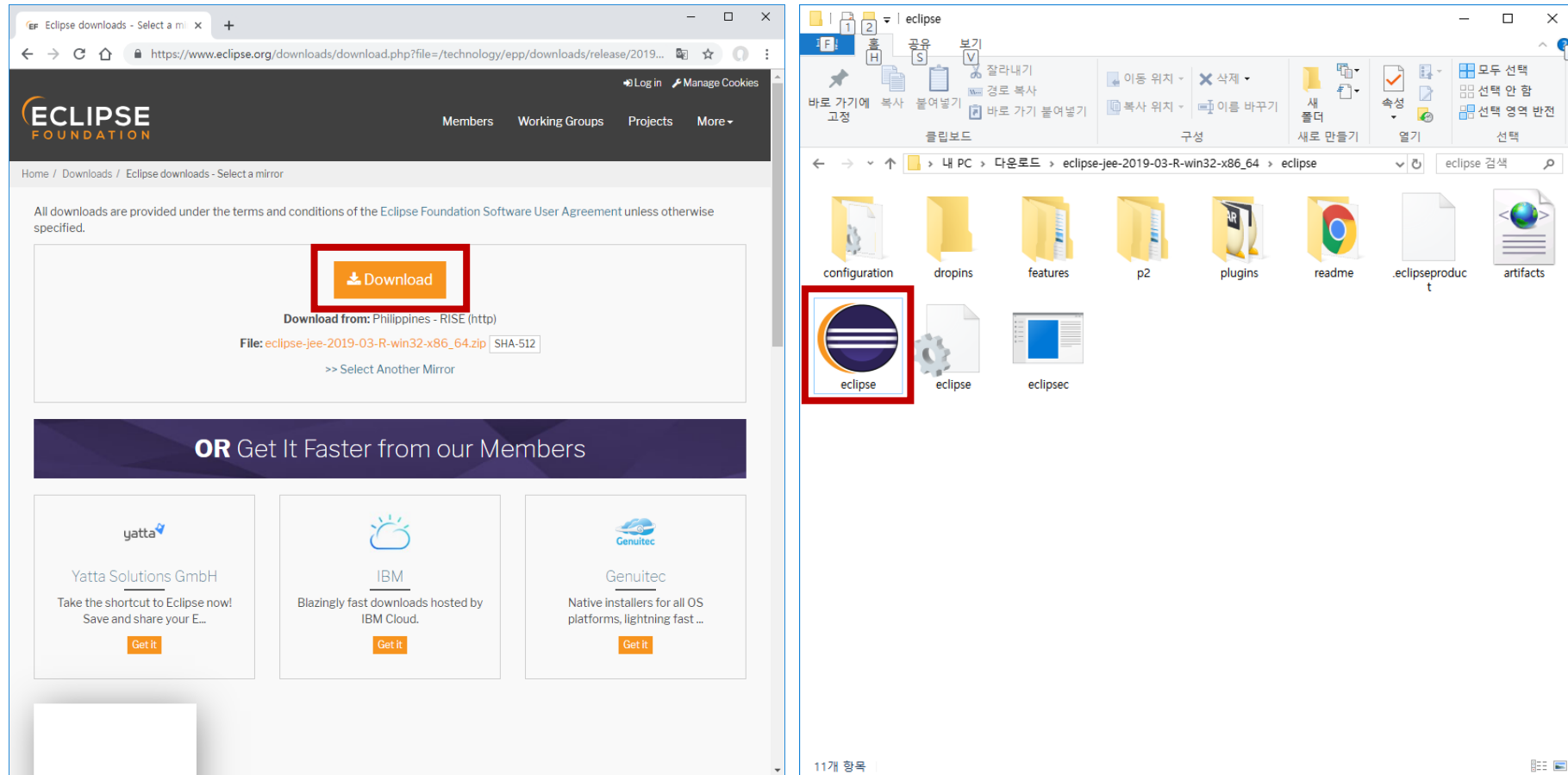
- Other builds
- Eclipse 2018-12 (4.10)
- Eclipse 2018-09 (4.9)
- Eclipse Photon (4.8)
- Eclipse Oxygen (4.7)
- Eclipse Neon (4.6)
- Eclipse Mars (4.5)
- Eclipse Luna (4.4)
- Eclipse Kepler (4.3)
- Older Versions

HINT

You will need a Java runtime environment (JRE) to use Eclipse (Java SE 8 or greater is recommended). All downloads are provided under the

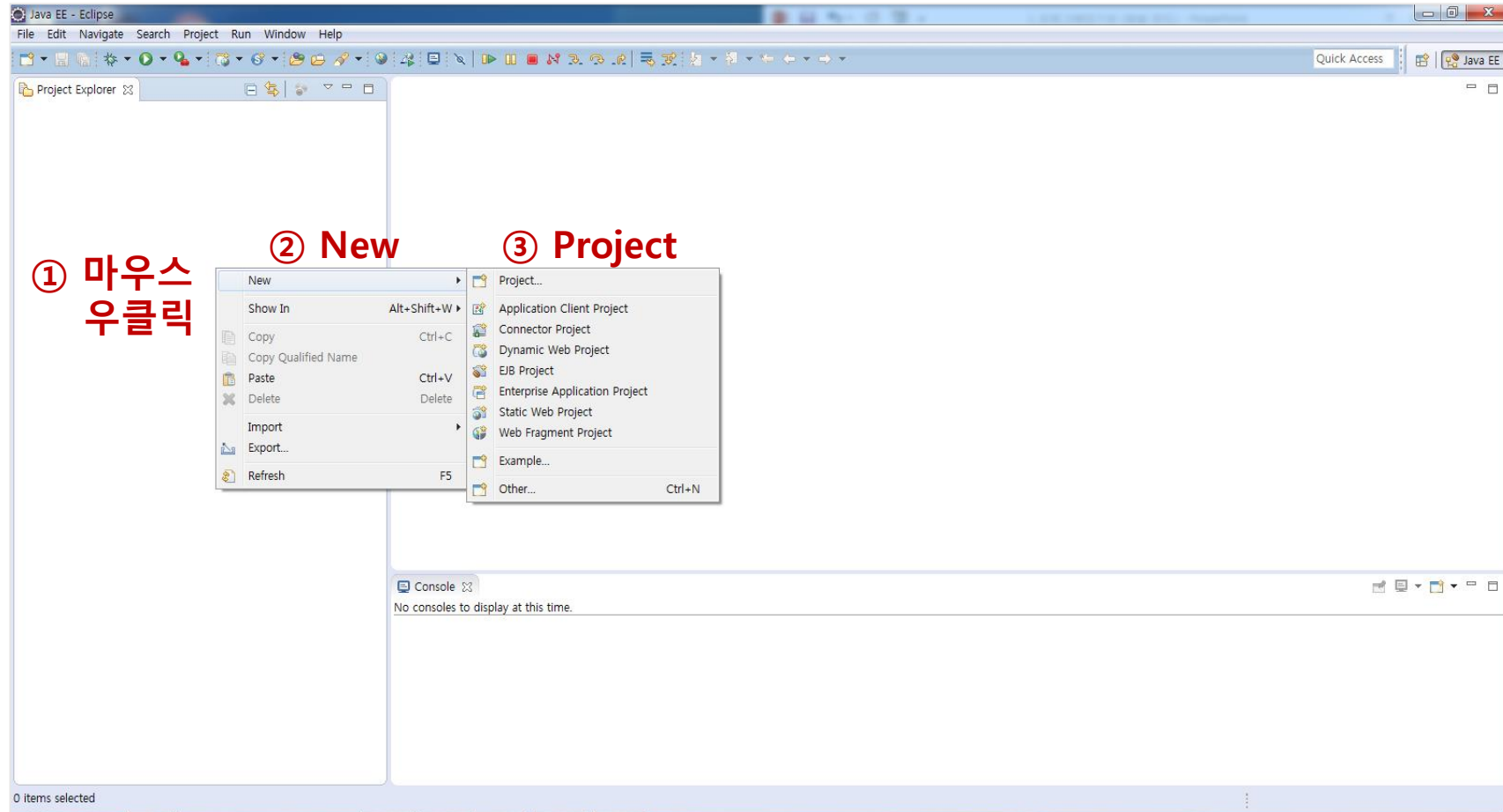
▶ 이클립스 설치

✓ zip 파일 다운로드 후 다운로드 파일 압축 해제 / eclipse.exe 실행



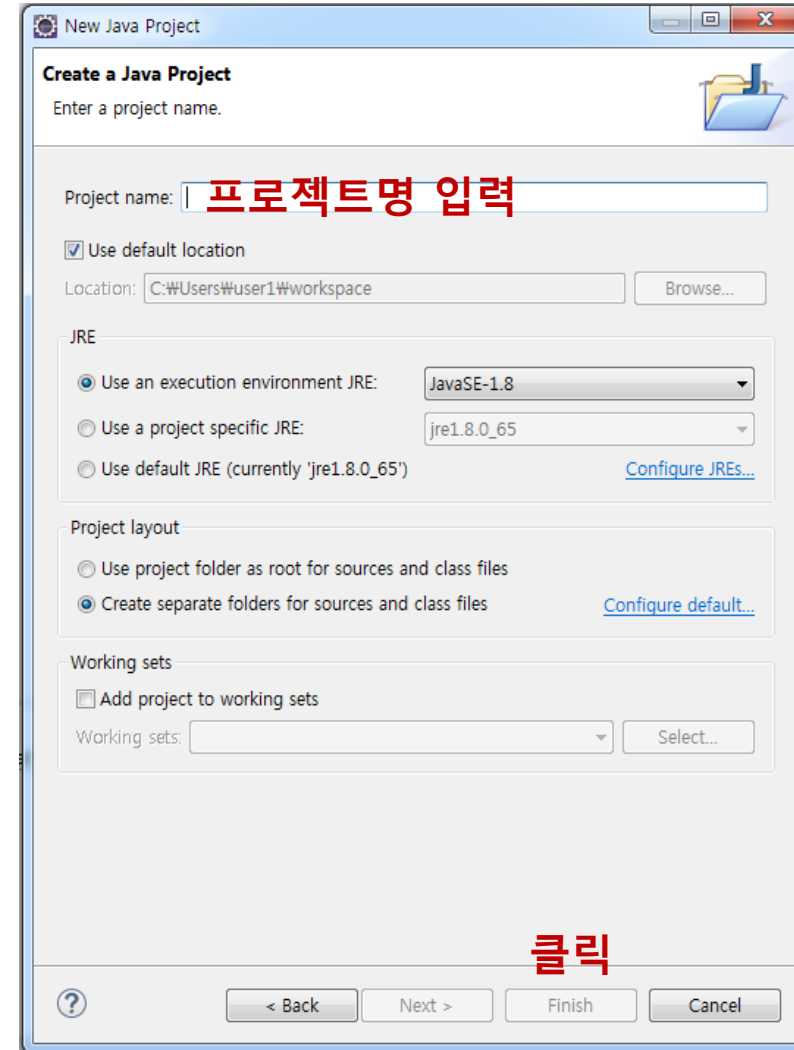
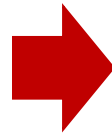
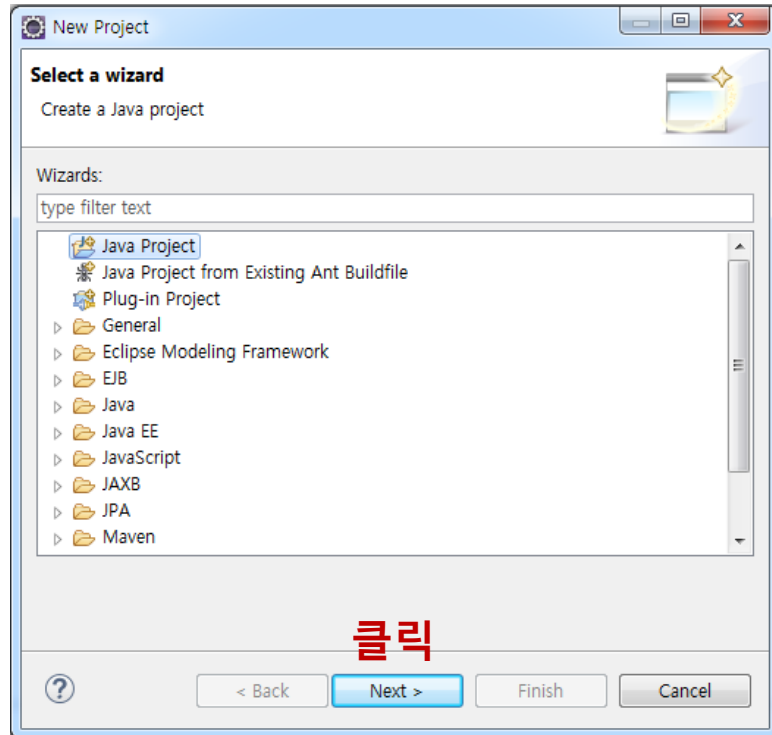
▶ Eclipse 환경에서의 자바 프로그래밍 순서

✓ Project 만들기



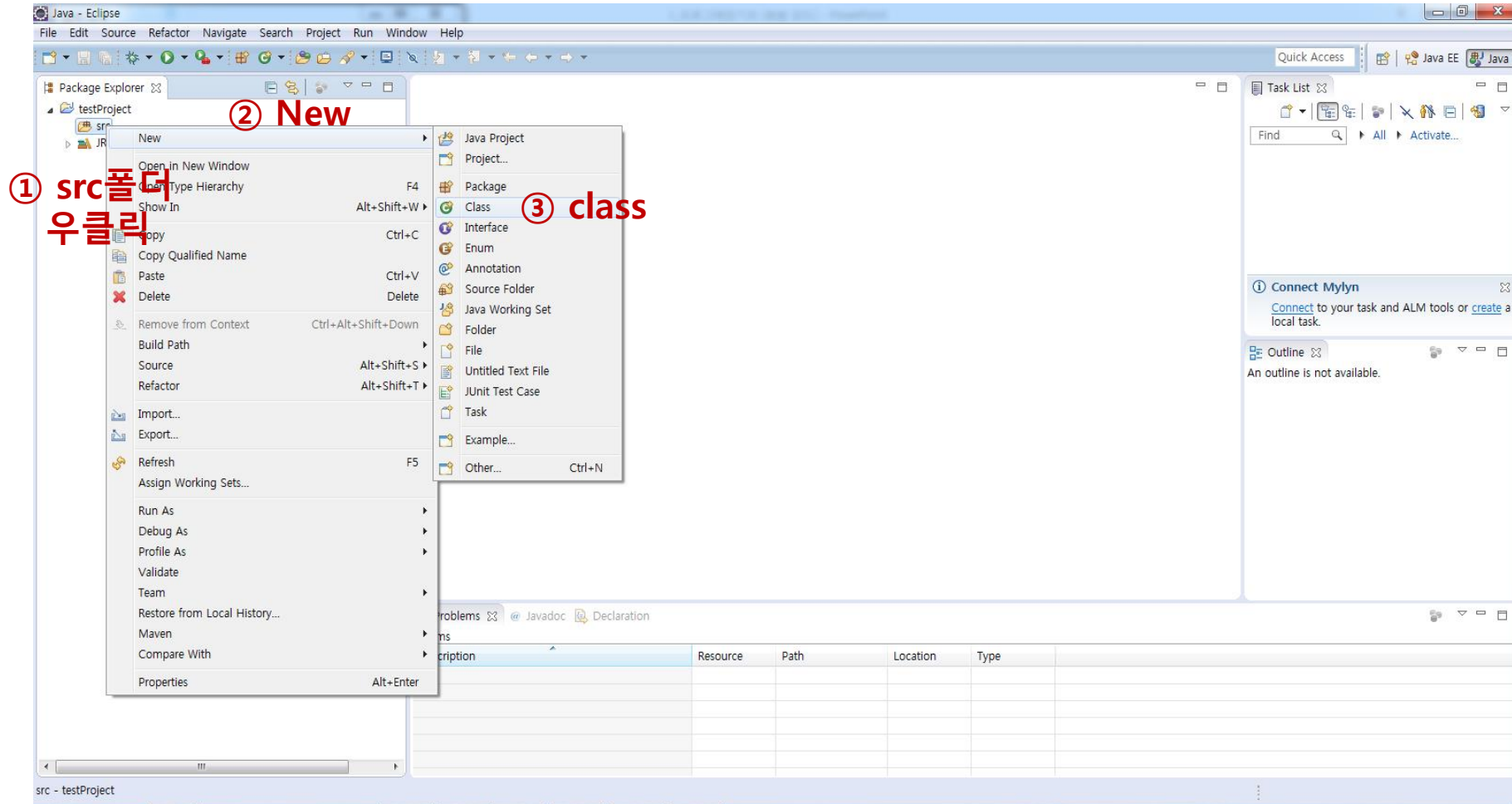
▶ Eclipse 환경에서의 자바 프로그래밍 순서

✓ Project 만들기



▶ Eclipse 환경에서의 자바 프로그래밍 순서

✓ Class 만들기



▶ Eclipse 환경에서의 자바 프로그래밍 순서

✓ Class 만들기

The image shows the 'New Java Class' dialog box in Eclipse. It has a title bar 'New Java Class' and a subtitle 'Java Class' with the instruction 'Create a new Java class.' and a green 'C' icon. The dialog contains several input fields and checkboxes. Annotations in red Korean text are present: '패키지명 입력' (Package name input) points to the 'Package' field; 'Class명 입력' (Class name input) points to the 'Name' field; '메인 메소드 필요시 체크' (Check if main method is needed) points to the 'public static void main(String[] args)' checkbox; and '클릭' (Click) points to the 'Finish' button at the bottom right.

Source folder: testProject/src Browse...

Package: testProject 패키지명 입력 Browse...

☐ Enclosing type: Browse...

Name: Class명 입력

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

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

Which method stubs would you like to create?

☐ public static void main(String[] args) ← 메인 메소드 필요시 체크
☐ Constructors from superclass
☒ Inherited abstract methods

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

Finish Cancel

▶ Eclipse 환경에서의 자바 프로그래밍 순서

✓ Class 작성 순서

① 패키지(package) 선언부 `package member.model.vo;`

② импорт(import) 선언부 `import java.util.Date;`

③ 클래스(class) 작성부 `public class Member {`

```
private String name;  
private int age;  
private Date enrollDate;
```

멤버 변수

생성자

```
public Member() {}
```

```
public Member(String name, int age, Date enrollDate) {  
    super();  
    this.name = name;  
    this.age = age;  
    this.enrollDate = enrollDate;  
}
```

```
public String getName() {  
    return name;  
}
```

```
public void setName(String name) {  
    this.name = name;  
}
```

멤버 함수

... 이하 생략...

```
}
```


▶ 자바 프로그래밍 기본

✓ 클래스

자바에서 모든 코드는 반드시 클래스 안에 존재해야 하며
서로 관련된 코드들을 그룹으로 나누어 별도의 클래스 구성
클래스들이 모여 하나의 Java 애플리케이션 구성

```
public class 클래스명 {  
  
    /*  
    * 주석을 제외한 모든 코드는 블록 { } 내에 작성  
    */  
  
}
```

▶ 자바 프로그래밍 기본

✓ 메소드

클래스의 기능(동작)에 해당하는 구현 부분으로
해당 메소드 호출 시 메소드의 블록({ })에 해당하는 부분 실행

```
public class 클래스명 {  
  
    public void 메소드명() {  
        // 실행될 문장들을 적는다  
    }  
}
```

▶ 자바 프로그래밍 기본

✓ 실행 메소드 (main 메소드)

'**public static void main(String[] args)**'는 main 메소드의 고정된 선언부

프로그램 실행 시 java.exe에 의해 호출됨

모든 클래스가 main메서드를 가지고 있어야 하는 것은 아니지만

하나의 Java애플리케이션에는 main메서드를 포함한 클래스가 반드시 하나는 있어야 함

```
public class 클래스명 {  
    //메인 메서드의 선언부  
    public static void main(String[] args) {  
        // 실행될 문장들을 적는다  
    }  
}
```

▶ 자바 프로그래밍 기본

✓ 주석(comment)

코드에 대한 설명이나 그 외 다른 정보를 넣을 때 사용하는 것으로
컴파일 시 컴파일러가 주석 부분은 건너 뛴다

`/* */` : 범위 주석, `/*`와 `*/` 사이 내용은 주석으로 간주

`//` : 한 줄 주석, `//` 뒤의 내용은 주석으로 간주

▶ 출력메소드

✓ `System.out.print()`

() 안의 변수, 문자, 숫자, 논리 값을 모니터에 출력해주는 메소드

✓ `System.out.println()`

print문과 동일하게 출력은 해주지만 출력 후 자동으로 출력창에 줄바꿈을 해주는 메소드

예) `System.out.print("안녕하세요");`
`System.out.print(123);`
`System.out.print(변수명);`

`System.out.println("안녕하세요");`
`System.out.println(123);`
`System.out.println(변수명);`