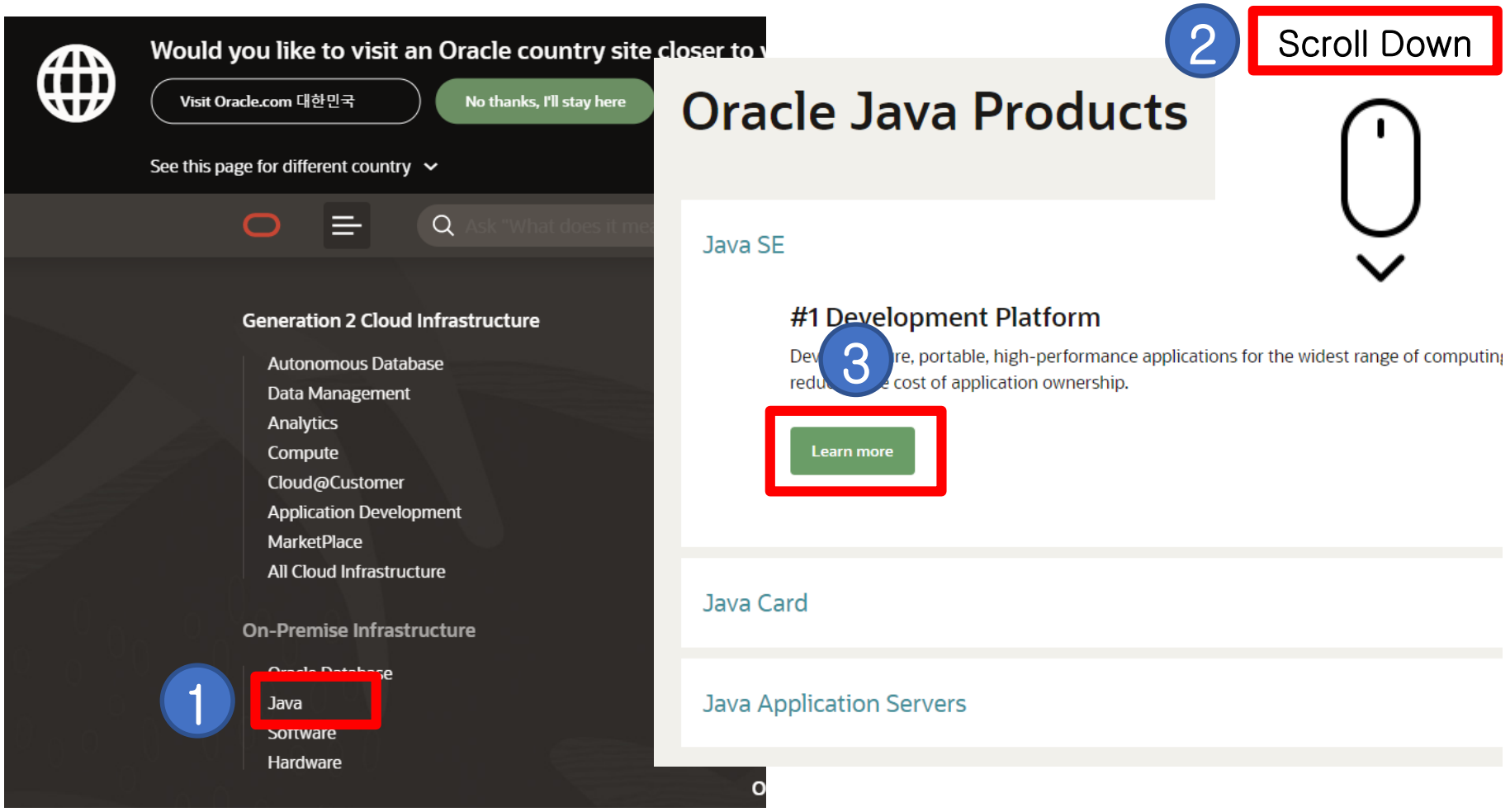
1강

JAVA

환경설정

JAVA 설치

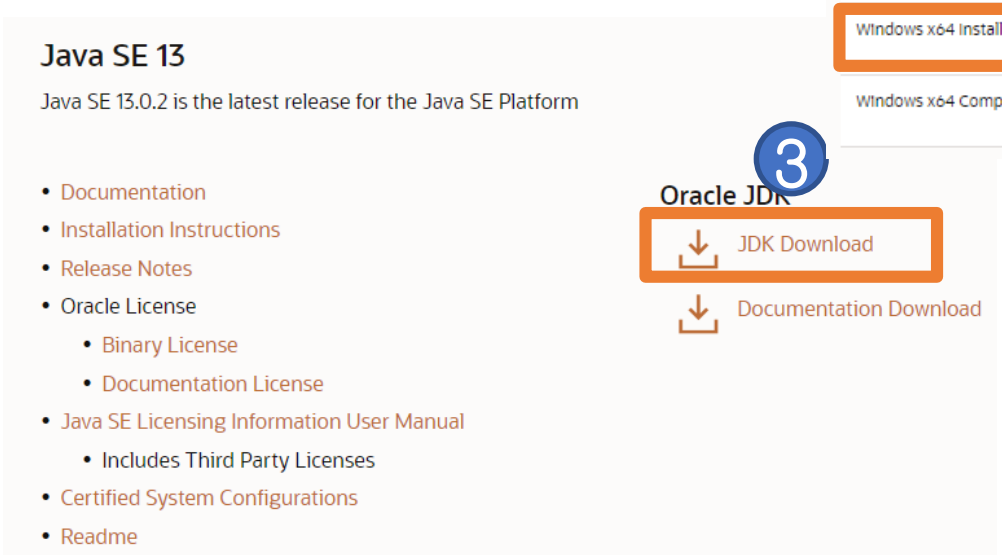
https://www.oracle.com

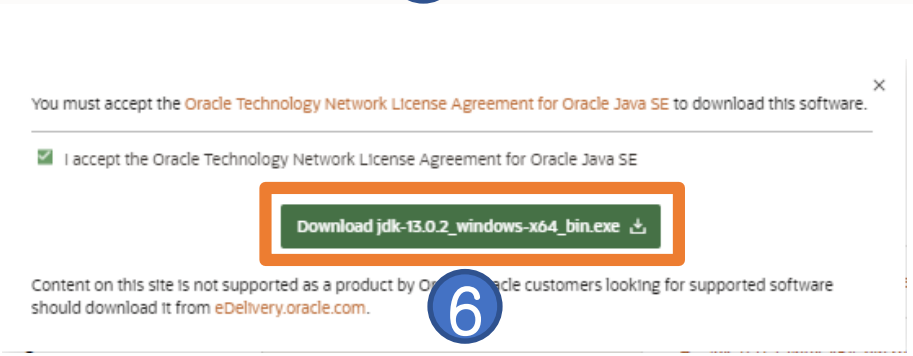


JAVA 설치


Overview
Downloads
Documentation
Community
Technologies
Training
1
2
DOWNLOAD
Java Platform, Standard Edition
Java SE 12.0.2

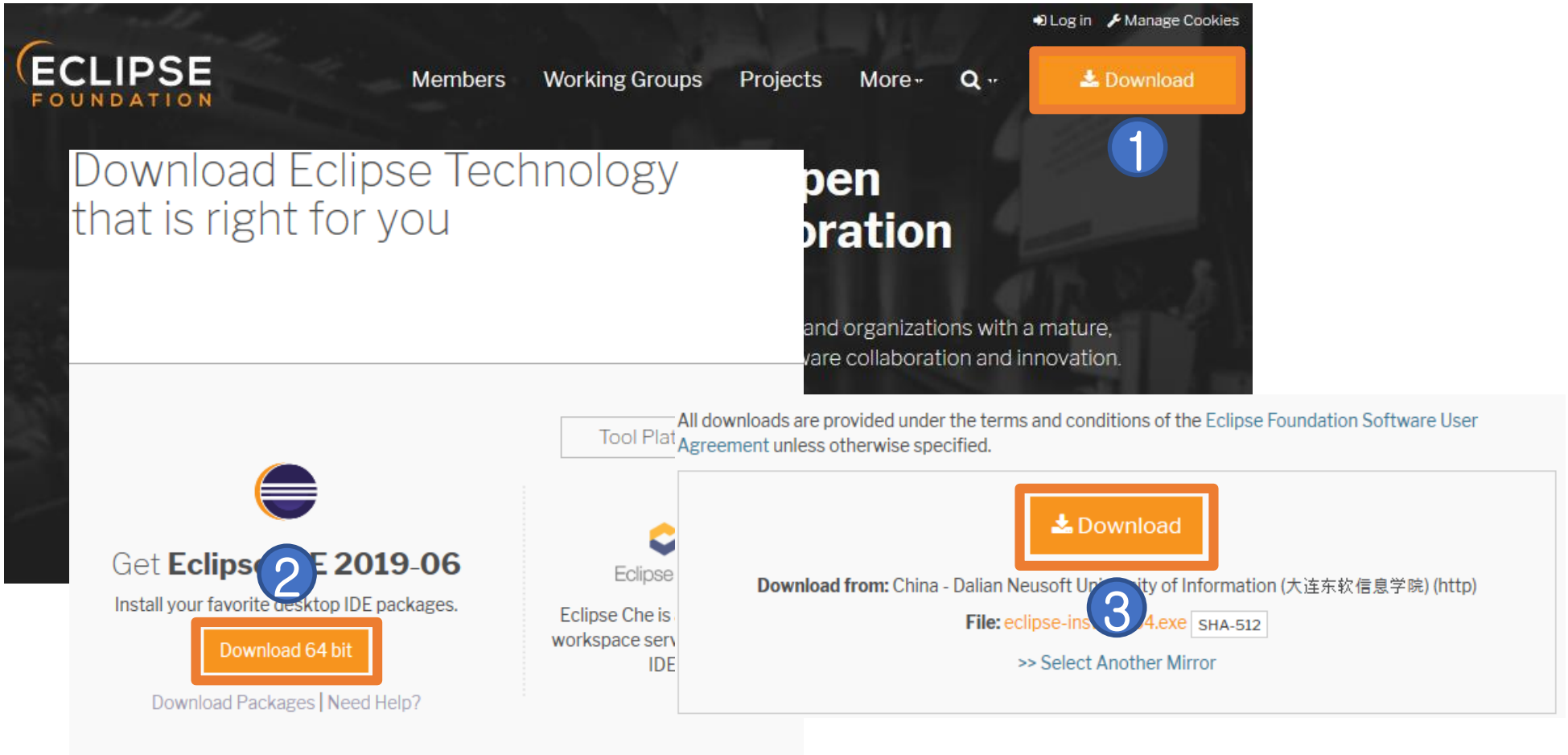

Java SE Development Kit 13.0.2
Product / File Description
Linux Debian Package 155.72 MB jdk-13.0.2_linux-x64_bin.deb
Linux RPM Package 162.66 MB jdk-13.0.2_linux-x64_bin.rpm
Linux Compressed Archive 179.41 MB jdk-13.0.2_linux-x64_bin.tar.gz
macOS Installer 173.3 MB jdk-13.0.2_osx-x64_bin.dmg
macOS Compressed Archive 173.7 MB jdk-13.0.2_osx-x64_bin.tar.gz
Windows x64 Installer 159.83 MB jdk-13.0.2_windows-x64_bin.exe
Windows x64 Compressed Archive 178.99 MB jdk-13.0.2_windows-x64_bin.zip
4
Scroll Down
5


Java SE 13
Java SE 13.0.2 is the latest release for the Java SE Platform
Documentation
Installation Instructions
Release Notes
Oracle License
Binary License
Documentation License
Java SE Licensing Information User Manual
Includes Third Party Licenses
Certified System Configurations
Readme
3
Oracle JDK
JDK Download
Documentation Download


You must accept the Oracle Technology Network License Agreement for Oracle Java SE to download this software.
I accept the Oracle Technology Network License Agreement for Oracle Java SE
Download jdk-13.0.2_windows-x64_bin.exe
Content on this site is not supported as a product by Or
should download it from eDelivery.oracle.com.
6

이클립스 설치

https://www.eclipse.org/



용어 정리

용어 정리

❖ **용도별 tool 구분**

- JAVA SE (Java Platform Standard Edition) : 데스크탑 및 서버에서 응용 프로그램 개발
- JAVA EE (Java Platform Enterprise Edition) : SE를 포함하고 있으며 웹 프로그램 개발
- JAVA ME (Java Platform Micro Edition) : 임베디드 개발

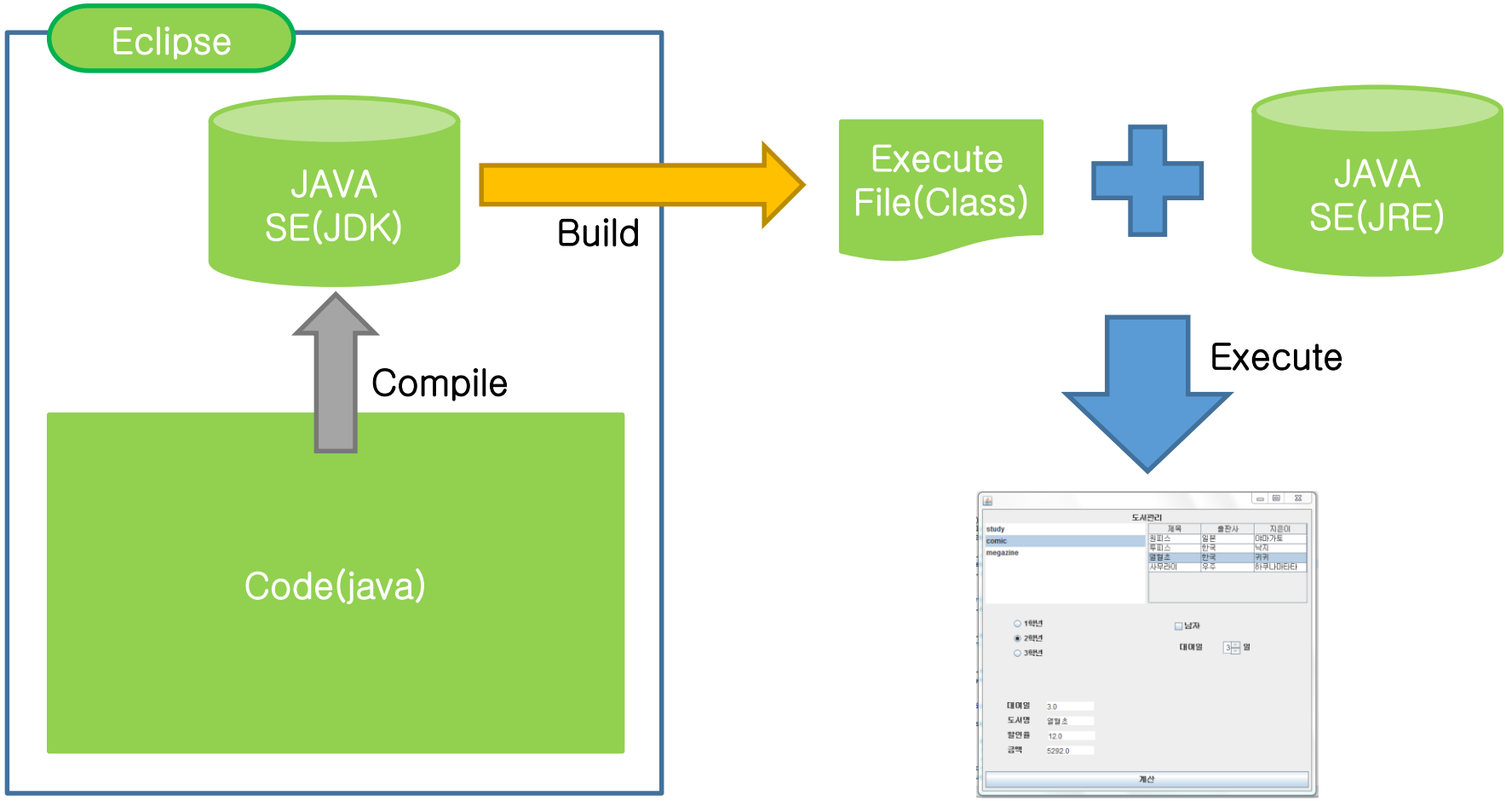
❖ **기능별 Tool 구분**

- JDK(Java Development Kit) : 개발 도구
- JRE(Java Runtime Environment) : 배포 도구

❖ **IDE(Integrated Development Environment)**

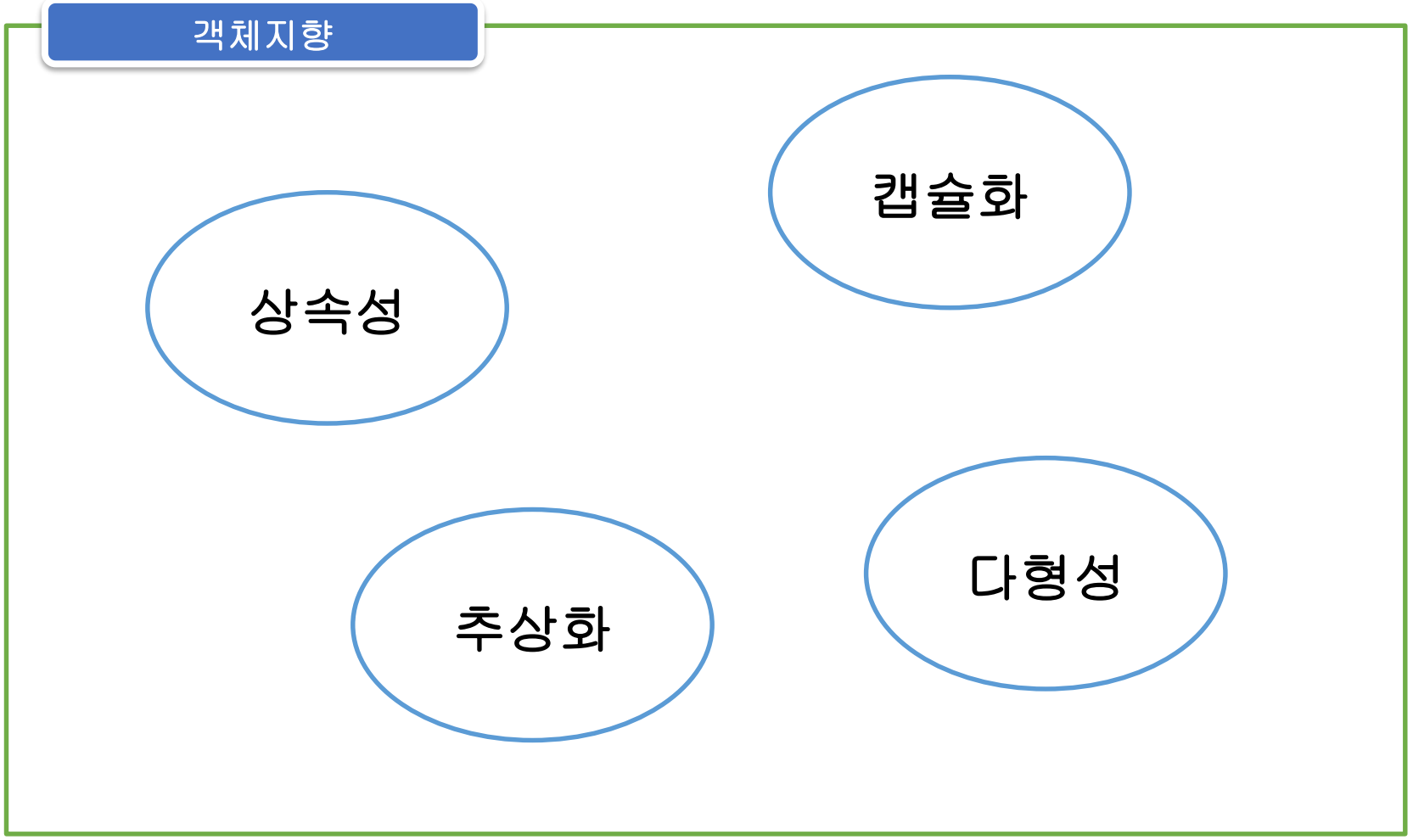
- Eclipse(무료)
- NetBeans(무료)
- vscode (무료)
- IntelliJ IDEA(유료)

동작 방식 이해


Eclipse
JAVA
SE(JDK)
Build
Compile
Code(java)
Execute
File(Class)
JAVA
SE(JRE)
Execute

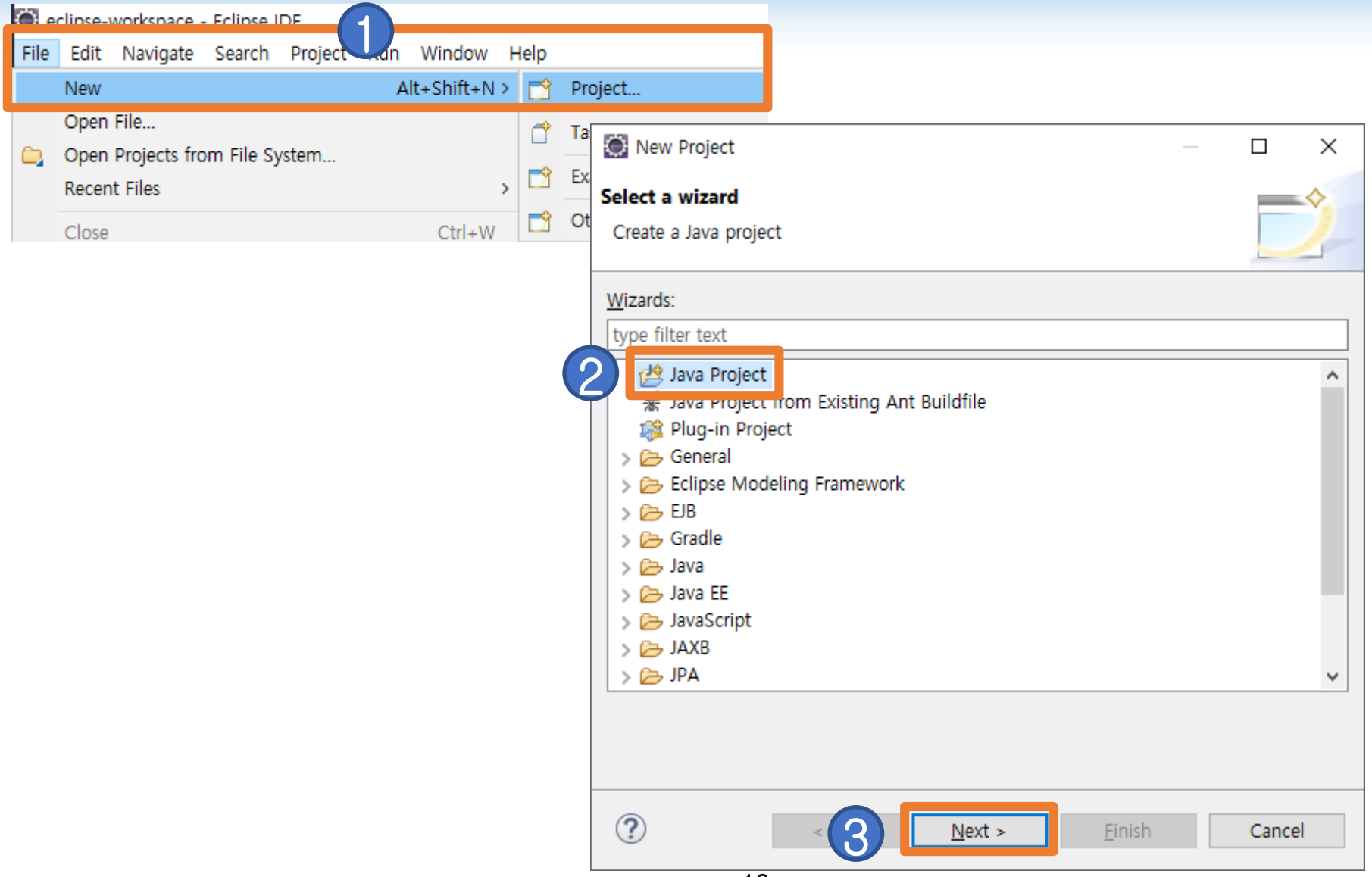
Java 기초 다지기

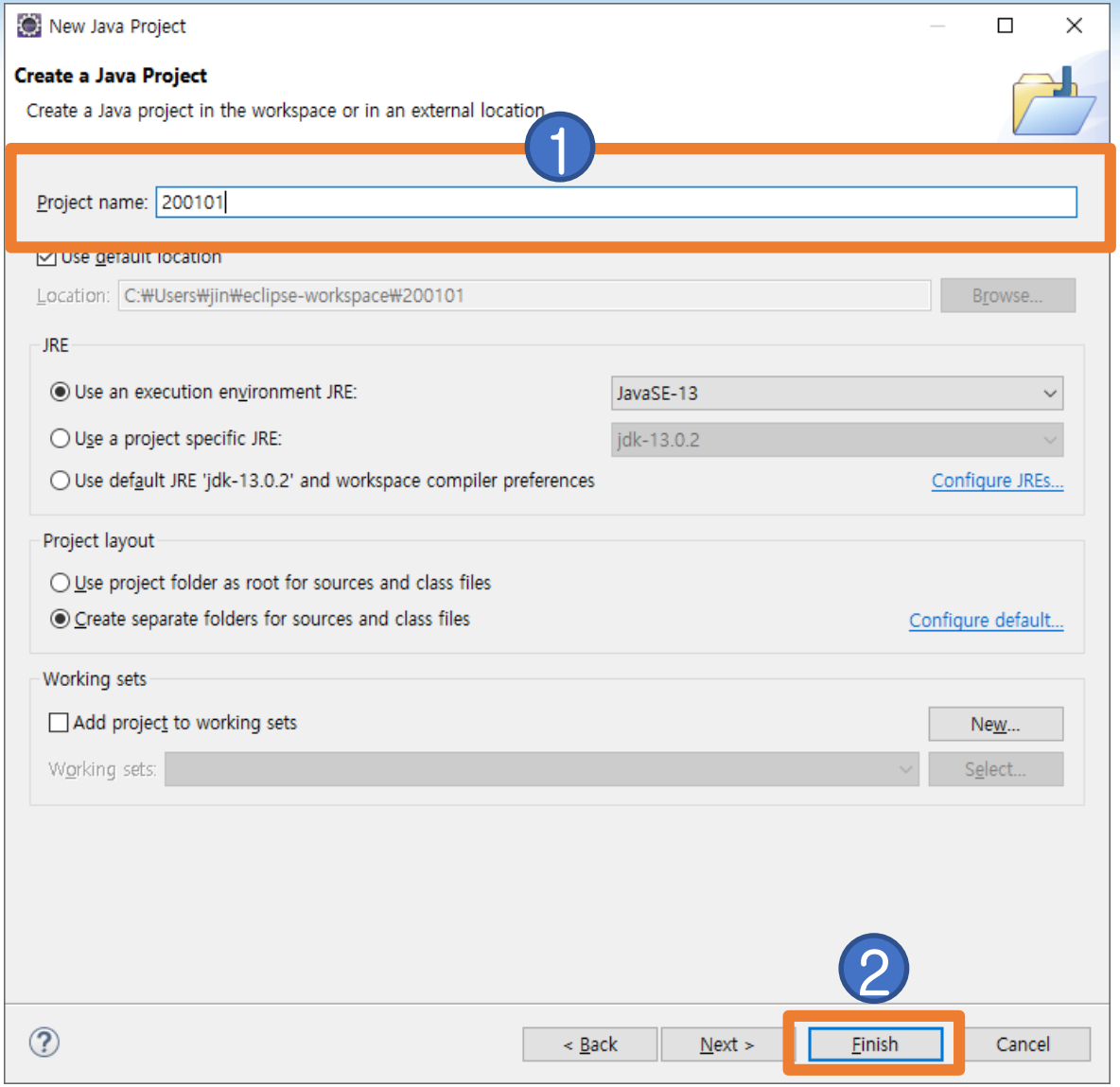
자바의 이해



이클립스 실행

프로젝트 만들기

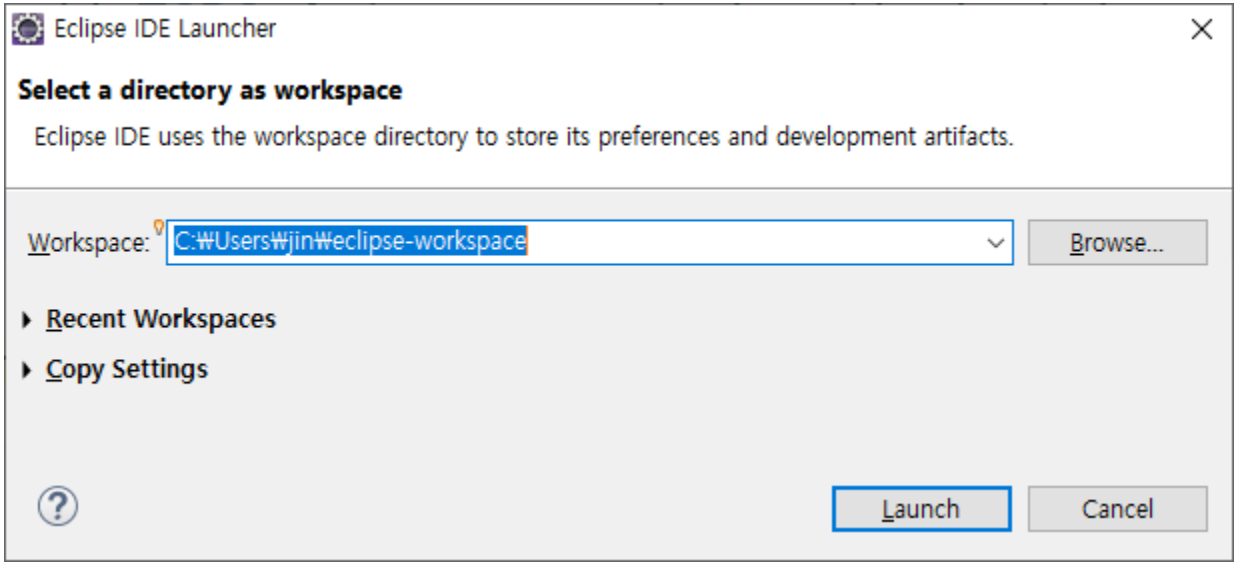

eclipse-workspace - Eclipse IDE
1
File Edit Navigate Search Project Run Window Help
New Alt+Shift+N >
Open File...
Open Projects from File System...
Recent Files
Close Ctrl+W
Project...
New Project
Select a wizard
Create a Java project
Wizards:
type filter text
2
Java Project
Java Project from Existing Ant Buildfile
Plug-in Project
General
Eclipse Modeling Framework
EJB
Gradle
Java
Java EE
JavaScript
JAXB
JPA
3
< Next > Finish Cancel


New Java Project
Create a Java Project
Create a Java project in the workspace or in an external location.
1
Project name: 200101
Use default location
Location: C:₩Users₩jin₩eclipse-workspace₩200101
Browse...
JRE
Use an execution environment JRE:
JavaSE-13
Use a project specific JRE:
jdk-13.0.2
Use default JRE 'jdk-13.0.2' and workspace compiler preferences
Configure JREs...
Project layout
Use project folder as root for sources and class files
Create separate folders for sources and class files
Configure default...
Working sets
Add project to working sets
New...
Working sets:
Select...
2
< Back
Next >
Finish
Cancel

workspace

❖ 이해하기

- 프로젝트 생성 시 저장되는 공간
- 최초 실행 시 기본 값으로 계정명₩eclipse-workspace 생성



기본 구조 익히기

❖ 실행 => ctrl + F11

```
package Java01.Basic.Ex01;

public class Ex01 {
    public static void main(String[] args) {
        System.out.print("Hello Java");
    }
}
```

주석

❖ 이해하기
- 코드에 대한 설명 붙이기
- 컴파일 또는 실행 시 무시됨

❖ 주석의 종류

주석 기호	설명
//	//부터 라인 끝가지 주석으로 처리한다. (행 주석)
/* ~ */	/*와 */ 사이에 있는 모든 범위를 주석으로 처리한다. (범위 주석)

출력

ESCAPE 문자

❖ 이해하기
- 특수문자
- ₩문자

❖ 종류

ESCAPE	기 능
₩n	새로운 줄, 개행
₩t	TAB
₩”	“
₩₩	₩

PRINT

```
public class Ex01 {
public static void main(String[] args) {
	System.out.println("안녕하세요!");
	System.out.println("자바 시간입니다.");

	System.out.print("안녕하세요!");
	System.out.print("자바 시간입니다.");
}
}
```

ESCAPE 문자

```
public class Ex02 {
public static void main(String[] args) {
	System.out.println("1. Hello Java");
	System.out.println("2. Hello\nJava");
}
}
```

ESCAPE 문자

```
public class Ex03 {
public static void main(String[] args) {
	System.out.println("Have\ta\tGood\tTime");
	System.out.println("1234567\t1\t12345678\t123");
	System.out.println("\"fun\"\\java");
}
}
```

출력

```
public class Ex04 {
public static void main(String[] args) {
	System.out.println(100 + 100);
	System.out.println(200.22 + 200);

	System.out.println("100" + 100);
	System.out.println("200.22" + 200);
}
}
```

출력

```
public class Ex05 {
public static void main(String[] args) {
	System.out.println(100 + "원");
	System.out.println("결과 : " + 200.22);
}
}
```

변수 이해하기

변수

❖ 이해하기
- 메모리에 일부분

❖ 명명 규칙
- 1. 영문 대/소문자 , 숫자 , _ (underScore) , $ 의 조합으로 구성
- 2. 변수명 시작은 숫자 안됨.
- 3. 예약어, 공백 안됨

기본 자료형

값의 종류	기본 타입	메모리 사용 크기		저장되는 값의 범위
정수	byte	1 byte	8 bit	2^7~2^7-1 (-128~127)
	char	2 byte	16 bit	0~2^{16}-1 (유니코드: ₩u0000~₩uFFFF, 0~65535)
	short	2 byte	16 bit	-2^{15}~2^{15}-1 (-32,768~32,767)
	int	4 byte	32 bit	-2^{31}~2^{31}-1 (-2,147,483,648~2,147,483,647)
	long	8 byte	64 bit	-2^{63}~2^{63}-1
실수	float	4 byte	32 bit	(+/-)1.4E-45 ~ (+/-)3.4E38
	double	8 byte	64 bit	(+/-)4.9E-324 ~ (+/-)1.7E308
논리	boolean	1 byte	8 bit	true, false

리터럴(literal)

❖ 이해하기
- 직접 입력된 변수의 초기값

❖ 사용예
- int n=10;
- long data=100L;
- float f1 = 3.14F;
- char ch='R';

변수의 범위

❖ 이해하기
- 설정된 지역 내에서만 사용 가능
- 지역은 {}를 의미함.

❖ 사용예

```
public class Ex01 {
    public static void main(String[] args) {
        int n1=2;          //main 내에서 사용 가능
        if(n1==2) {
            int n2=10; //if에서만 사용 가능
        }
        System.out.println(n2);//error
    }
}
```

변수 입/출력

```
public class Ex01 {
public static void main(String[] args) {
    int age = 5;
    double weight = 23.5;
    double height = 123.5;
    System.out.println("나의 나이는 " + age + "세 입니다.");
    System.out.println("나의 몸무게는 " + weight + "(kg)입니다.");
    System.out.println("나의 신장은 " + height + "(cm)입니다.");
    }
}
```

변수 입/출력

```
public class Ex02 {
public static void main(String[] args) {
	int data = 123, result;
	char ch = 'D';

	data =  data + 10;
	ch = 'D' + 3;
	result = data + ch;
	System.out.println("data = " + data);
	System.out.println("ch = " + ch);
	System.out.println("result : " + result);
}
}
```

변수 입/출력

```
public class Ex03 {
public static void main(String[] args) {
	String data = "Java";
	System.out.println("data : " + data);
	data = "JAVA Programming";
	System.out.println("data : " + data);
}
}
```

변수 입/출력

```
public class Ex04 {
public static void main(String[] args) {
	boolean flag = false;
	System.out.println("flag : " + flag);
	flag = true;
	System.out.println("flag : " + flag);
}
}
```

자료형

- Promotion
  - 작은 범위의 자료 형 => 큰 범위의 자료 형 대입
  - long = int
- Casting
  - 큰 범위의 자료 형 => 작은 범위의 자료 형 대입
  - int = long
  - Cast 연산자
- boolean, String 안됨.

자료형

```
public class Ex01 {
public static void main(String[] args) {
	char ch = 'A';
	int num = 98;

	System.out.println("ch : " + ch);
	System.out.println("(int)ch : " + (int)ch);

	System.out.println("num : " + num);
	System.out.println("(char)num : " + (char)num);
}
}
```

자료형

```
public class Ex02 {
public static void main(String[] args) {
    byte b = 97;
    short s = 20;
    char c = 'A';
    float f = 1.23f;

    s = b;
    c = (char) b;
    s = (short) c;
    c = (char) s;
    f = (float) 5.11;
}
}
```

입력

입력

```
public class Ex01 {
public static void main(String[] args) throws IOException{
	char data;
	System.out.print("데이터 입력 : ");
	data = (char)System.in.read();
	System.out.println("출력 : " + data);
}
}
```

입력

```
public class Ex02 {
public static void main(String[] args) throws IOException{
    int data1, data2, data3;
    System.out.print("첫번째 데이터 입력 : ");
    data1 = System.in.read();
    System.out.println("첫번째 데이터 출력 : " + data1);

    System.out.print("두번째 데이터 입력 : ");
    data2 = System.in.read();
    System.out.println("두번째 데이터 출력 : " + data2);

    System.out.print("세번째 데이터 입력 : ");
    data3 = System.in.read();
    System.out.println("세번째 데이터 출력 : " + data3);
}
}
```

입력

```
public class Ex03 {
public static void main(String[] args) throws IOException{
    int data1, data2;
    System.out.print("첫번째 데이터 입력 : ");
    data1 = System.in.read();
    System.out.println("첫번째 데이터 출력 : " + data1);

    System.in.read();System.in.read();

    System.out.print("두번째 데이터 입력 : ");
    data2 = System.in.read();
    System.out.println("두번째 데이터 출력 : " + data2);
}
}
```

입력

```
public class Ex04 {
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    String name;
    int age;
    System.out.print("이름 입력 : ");
    name = input.next();
    System.out.print("나이 입력 : ");
    age = input.nextInt();
    System.out.println("이름 : " + name);
    System.out.println("나이 : " + age);
}
}
```