

---

---

---

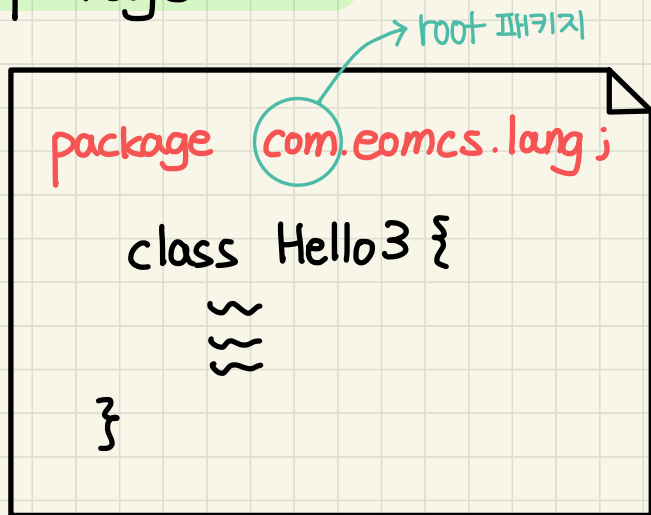
---

---



# \* package

\* 옵션+값. 한 묶음  
단, 소스파일 경로 뒤에 접착X



소스파일의 경로

src/main/java/com/eomcs/lang/Hello3.java

컴파일

\$javac -d bin/main src/ ~

클래스파일의 경로

bin/main/com/eomcs/lang/Hello3.class

\* 패키지 이름은 유일성을 보장하기 위해  
보통 도메인 명을 사용한다.

조직명.회사명.그룹명 → dev.bitcamp.com

제품명.국가명 → nls.go.kr

서비스명

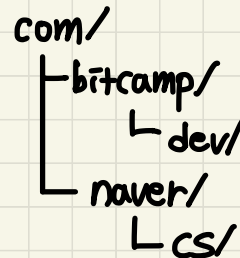
하위 서비스명.서비스명. "

\* 패키지 무소속은 바로 bin/main에 생성

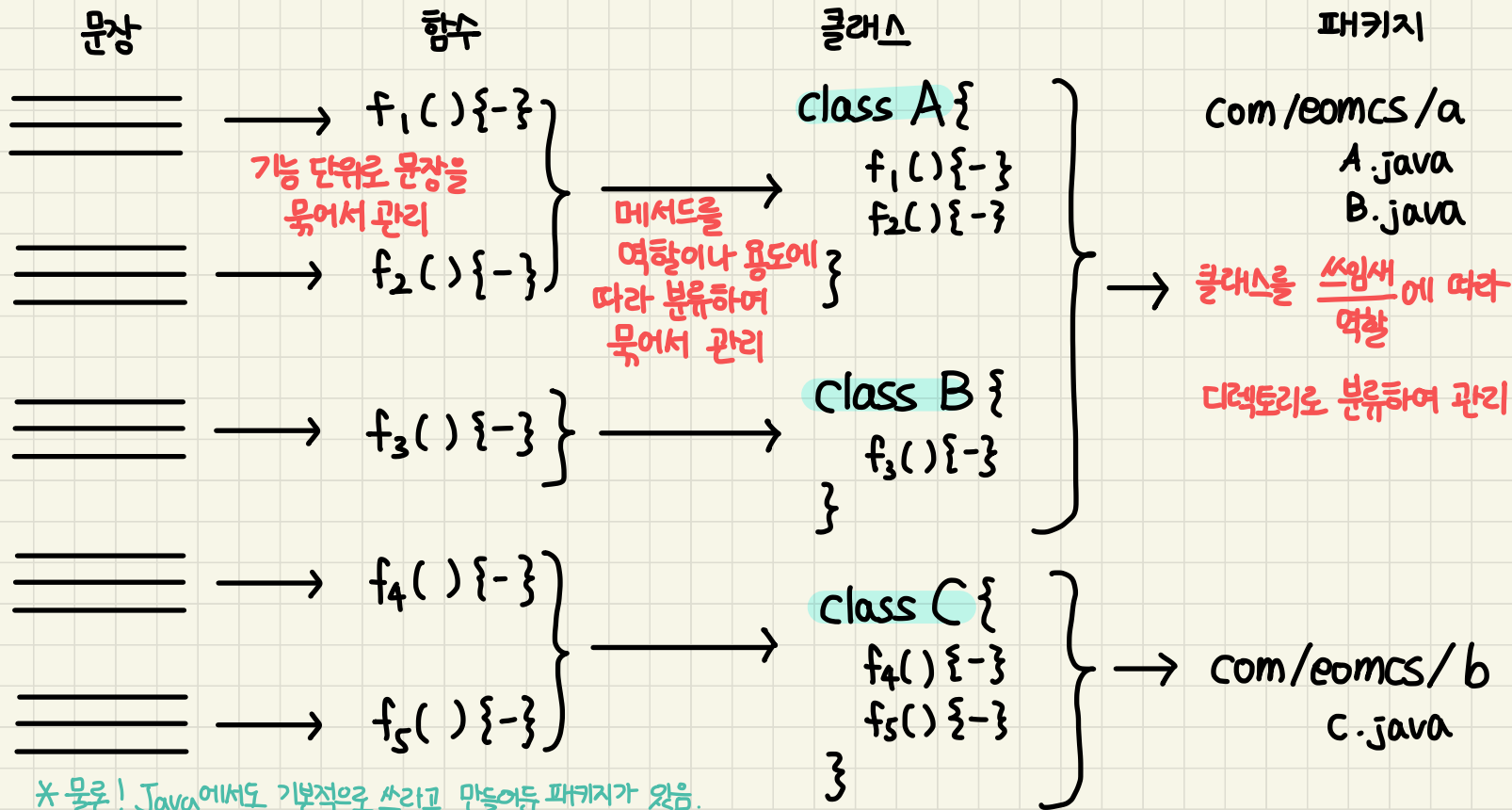
\* 패키지에 소속된 클래스를

실행할 때는 무조건 패키지명 ⇒ 루트 패키지가 있는  
폴더까지만 적기

디렉토리를  
구성하기 쉽게  
도메인 명을  
거꾸로 사용한다



# \* 함수, 클래스, 패키지



\* 물론! Java에서도 기본적으로 쓰라고 만들어진 패키지가 있음.

패키지를 관리하기 쉽게 묶은 것이 module \*

## \* 소스 파일과 클래스 블록

↳ 중괄호 { }

\* 자바의 컴파일 단위 : { } 블록 단위

```
Exam0120.java  
  
class A{ }  
class B{ }  
class C{ }
```

컴파일  
⇒

A.class

~~Exam0120.class~~

B.class

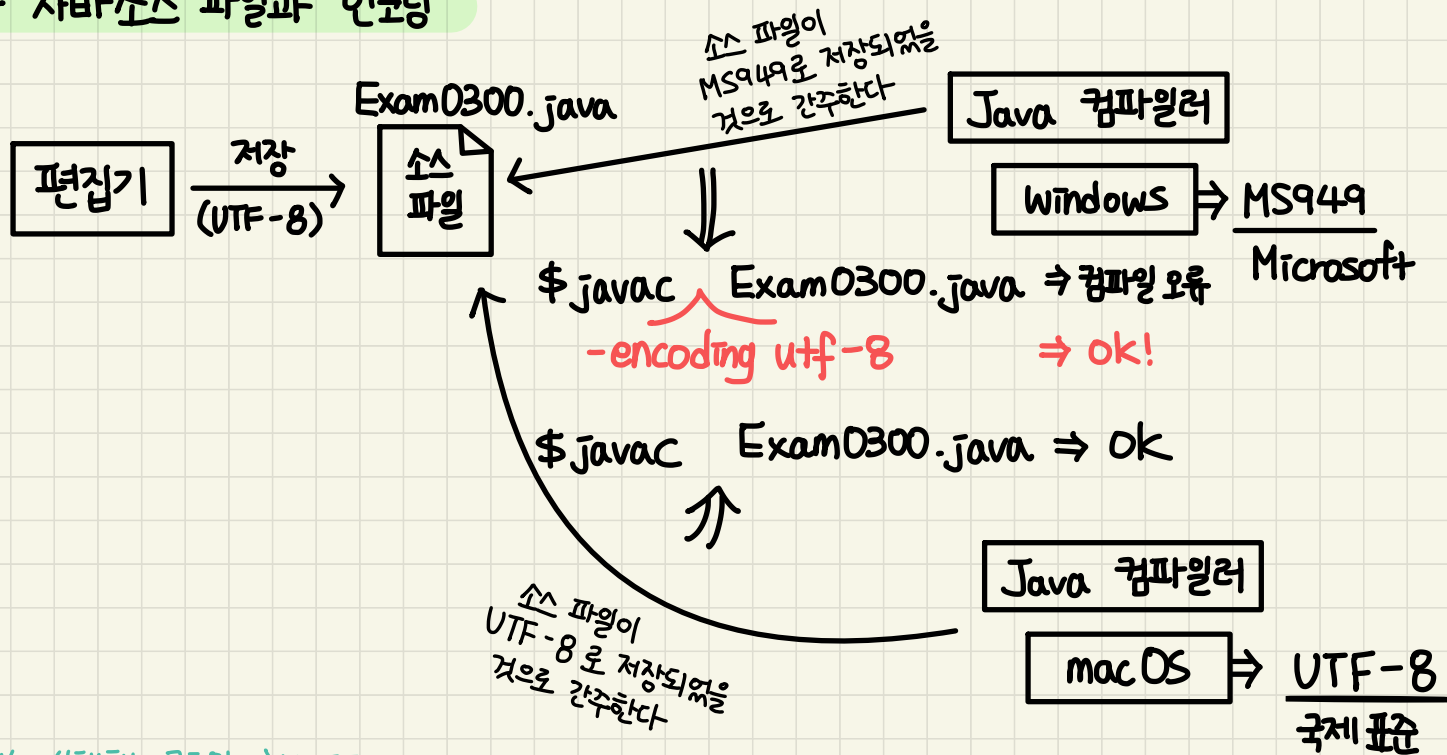
C.class

\* 유지 보수를 잘하려면 한 파일 안에  
class 하나만 두는게 관리하기 편함

\* public class로 하면

파일 이름과 클래스 명이 같아야 함

## \* 자바소스 파일과 인코딩



\* 실행하는 코드의 최소 코드

```
public class
```

```
public static void ~
```

\* `/** ~ */` → javadoc 주석

\* javadoc

\$ javadoc -encoding utf-8 -charset utf-8 —

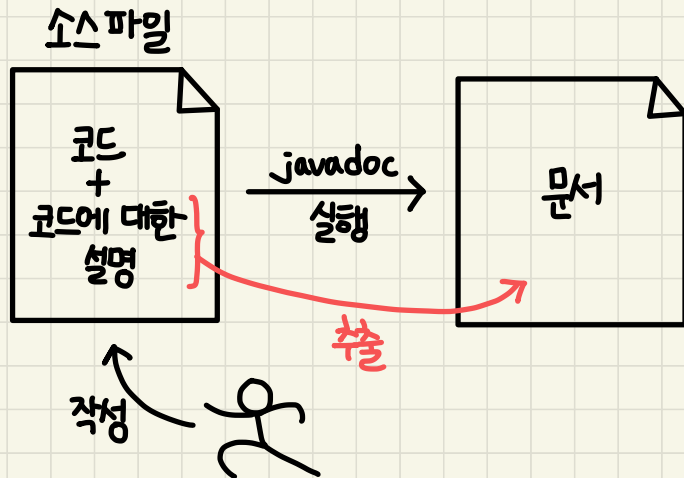
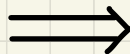
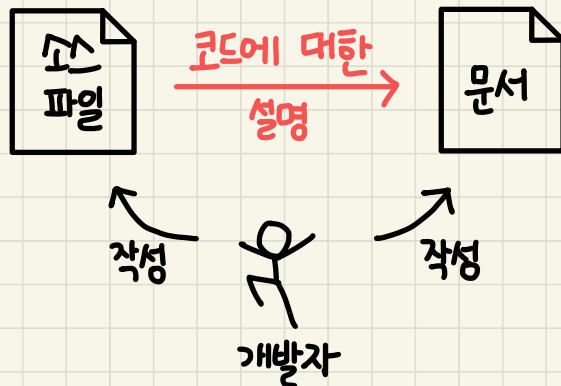
Exam0200.java

```
/**  
=  
*/  
class Exam0200 {  
    /**  
    = */  
    변수 선언  
    /**  
    =  
    */  
    메서드 정의  
}
```

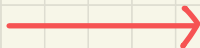
javadoc 실행

HTML  
문서

## \* javadoc 사용 전/후



\* 소스파일을 변경하면  
그에 대한 설명도 변경해야 하는데  
실제로 귀찮아서 문서를 변경하지 않는  
경우가 많다.

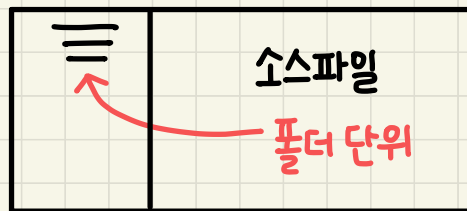


\* 소스파일을 작성할 때  
코드에 대한 설명을  
특별한 주석 (`/** ~ */`)으로  
적어 넣는다.  
그리고 javadoc을 실행하여  
문서를 자동 생성한다.

## \* 프로젝트 폴더와 편집기

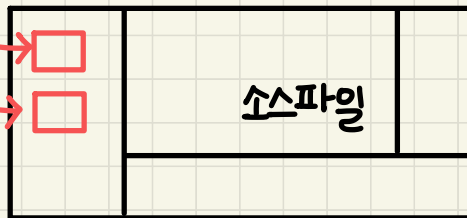
bitcamp-study/  
└ java-basic  
└ Study-project

VS Code 편집기 ⇒ 편집, 실행, 디버깅 등



더 막강한  
가능

Eclipse / IntelliJ ⇒ 편집, 컴파일, 실행,  
(무료) (유료) 디버깅, 배포 등



프로젝트  
단위로  
관리

## \* IDE

{ Integrated  
Development  
Environment

편집기, 컴파일, 디버깅을  
다양한 개발도구를  
수용하여 개발작업을 돕는다.



## \* Eclipse IDE

IBM → Eclipse 재단

↓ 개발

Eclipse IDE

+

자바 개발  
도구 →

JDT (Java Development Tool)

+

자바 웹 개발 도구 →

WTP

⇓

Java IDE for Enterprise and Web Developers

✓ 확장이 용이한 개발툴

↓

누구나 플러그인 작성 가능

↓

Eclipse 에 설치

## \* Eclipse Project의 조건

프로젝트폴더/

- project ← eclipse 에게 프로젝트 이름과 유형에 대한 정보를 제공한다  
Java/C/C++ 등  
→ 이 파일이 없으면 프로젝트 폴더로 간주하지 않는다
- classpath ← Java 프로젝트인 경우 이 파일이 있어야 한다  
→ 소스 폴더와 클래스 파일을 두는 폴더에 대한 정보  
→ 프로젝트에서 사용하는 외부라이브러리 (.jar)의 경로 정보
- settings/ ← 기타 설정 정보를 갖고 있는 보관

≡

## \* Gradle을 이용하여 Eclipse 프로젝트 관련 파일 자동 생성하기

```
[ build.gradle ]
```

```
    plugins {
```

```
        ≡
```

```
        id 'eclipse'
```

```
    }
```

'eclipse' 플러그인 추가

← 이클립스 IDE와 관련된 작업을 수행할 수 있다