

[자바 컨벤션]

1. 공통으로 사용되는 파일 이름

- 1) GNUmakefile : makefile 이름
* SW 빌드 시, gnumake 명령어를 사용
- 2) README : 디렉토리 내용 요약하는 파일 명

2. 파일 구조

=> 라인 수가 2,000줄이 넘으면 가독성이 떨어지므로 피하자.

3. 자바 소스 파일

- 1) 자바 소스 파일은 public class 또는 interface를 가진다.
- 2) public class는 파일에서 첫 번째 class 또는 interface이어야 한다.
- 3) 소스 파일 순서

1. 시작 주석	/* * 클래스 이름 * * 버전 정보 * * 날짜 * * 저작권 주의 */	
2. Package 문과 import 문	package 패키지 명; import java.파일 경로.파일 명;	package java.awt; import java.awt.peer.CanvasPeer;
3. Class와 Interface 선언		

4. 들여쓰기

=> 4개의 빈 칸(space)을 들여쓰기 단위로 사용

5. 한 줄의 길이

=> 70줄 미만을 사용

* 터미널(Terminal) 툴에서 사용하기 편하게 하기 위해서

6. 주석

=> 자바 프로그램은 구현 주석, 문서화 주석 두 종류의 주석을 가진다.

- 1) 구현 주석 : C/C++에서의 주석과 동일
- 2) 문서화 주석 : /** ... */에 의해서 경계가 결정된다.
 - 자바 SW에 포함되어 있는 javadoc 툴을 사용하면 문서화 툴을 포함하는 HTML 파일을 자동으로 만들 수 있음
 - 문서화 주석은 소스 코드가 없는 개발자들도 읽고 이해할 수 있도록, 실제 구현 코드와 상관이 없는 코드의 명세 사항(specification)을 포함한다.

* 주석은 코드에 대한 개요 또는 코드 자체만으로 알 수 없는 추가적 정보들을 제공하기 위해 사용되어야 한다.

ex) '패키지를 어떻게 만들 것인가'와 같은 정보는 주석에 포함되면 안된다.

7. 배치

- 1) 선언은 블록의 시작에 위치해야 한다.
- 2) 블록 안의 블록에서 동일한 변수 이름을 사용해서 선언하지 말기!

8. 명명(Naming) 규칙

1) 패키지(Packages)

- 패키지 이름의 최상위 레벨은 항상 ASCII 문자에 포함되어 있는 소문자로 쓰고, 가장 높은 레벨의 도메인의 이름 중 하나이어야 한다.
- 패키지 이름의 나머지 부분은 조직 내부의 명명 규칙을 따르기
 - * 부서 명, 팀 명, 프로젝트 명, 컴퓨터 이름, ... ex) com.sum.eng, com.apple.quicktime.v2

2) 클래스(Classess) / 인터페이스(Interfaces)

- 클래스 이름은 명사이어야 하며, 복합 단어일 경우 각 단어의 첫 글자는 대문자.
- 간단하고 명시적으로 작성해야 한다.
- 완전한 단어를 사용하고 두 문자어와 약어는 피하도록 한다.
 - * 단, 약어가 더 많이, 더 넓게 사용되고 있다면 약어를 사용하는 것도 괜찮다.

3) 메서드(Methods)

- 메서드의 이름은 동사이어야 하며, 복합 단어일 경우 첫 단어는 소문자로 시작하고 그 이후에 나오는 단어의 첫 문자는 대문자로 사용한다.
 - ex) run(), runFast(), ...

4) 변수(Variables)

- 변수 이름의 첫 번째 문자는 소문자로 시작하고, 각각 내부 단어의 첫 번째 문자는 대문자로 시작한다.
- 이름은 짧지만 의미 있어야 한다.
- 이름의 선택은 그 변수의 사용 의도를 알아낼 수 있도록 의미적이어야 한다.
- 한 문자로만 이루어진 변수 이름은 암시적으로만 사용하고 버릴 변수일 경우를 제외하고는 피해야 한다.
 - * 보통의 임시 변수들의 이름은 정수는 i, j, k, m, n을 사용하고, 문자는 c, d, e를 사용한다.

5) 상수(Constants)

- 상수 이름은 모두 대문자로 쓰고 각각의 단어는 언더바(_)로 분리 해야 한다.
- 디버깅을 쉽게 하기 위해서 ANSI 상수들의 사용은 자제하도록 한다.
 - ex) static final int MIN_WIDTH = 4;

9. 좋은 코드 예제

```
CodeConvention.java
1  /*
2   * @(#)CodeConvention.java      0.02 2016/11/17
3   *
4   * [저작권 및 라이선스 관련 정보를 여기에 작성한다.]
5   * Copyright (c) 2016 Myeongun Ryu.
6   * ComputerScience, ProgrammingLanguage, Java, Seoul, KOREA
7   * All rights reserved.
8   *
9   * This software is the confidential and proprietary information of Myeongun
10  * Ryu ("Confidential Information"). You shall not
11  * disclose such Confidential Information and shall use it only in
12  * accordance with the terms of the license agreement you entered into
13  * with Myeongun Ryu.
14  */
15
16 package myeonguni.codeconvention;
17
18 import myeonguni.*;
19
20 /**
21  * 클래스에 대한 설명을 여기에 작성한다.
22  *
23  * @version      1.00 2016년 11월 17일
24  * @author      류영운
25  */
26 public class CodeConvention extends Convention {
27     /* 클래스의 구현 주석을 여기에 작성한다. */
28
29     /** 클래스 변수 classVar1에 대한 설명을 여기에 작성한다. (문서 주석) */
30     public static int classVar1;
31
32     /**
33      * 클래스 변수 classVar2에 대한 설명이 (문서 주석이다)
34      * 한 줄 이상일 경우 이렇게 작성한다. (접근 제어자가 private일 경우 나오지는 않음.)
35      */
36     private static Object classVar2;
37
38     /** 인스턴스 변수 instanceVar1에 대한 설명을 여기에 작성한다. (문서 주석) */
39     public Object instanceVar1;
40
41     /** 인스턴스 변수 instanceVar2에 대한 설명을 여기에 작성한다. (문서 주석) */
42     protected int instanceVar2;
43
44     /** 인스턴스 변수 instanceVar3에 대한 설명을 여기에 작성한다. (문서 주석) 접근 제어자가 private일 경우 나오지는 않음. */
45     private Object[] instanceVar3;
46
47     /**
48      * ... 생성자 CodeConvention()에 대한 설명을 여기에 작성한다. (문서 주석) ...
49      */
50     public CodeConvention() {
51         // ... 여기에 실제 코드를 작성한다. ...
52     }
53
54     /**
55      * ... 메서드 doSomething()에 대한 설명을 여기에 작성한다. (문서 주석) ...
56      */
57     public void doSomething() {
58         // ... 여기에 실제 코드를 작성한다. ...
59     }
60
61     /**
62      * ... 메서드 doSomethingElse()에 대한 설명을 여기에 작성한다. (문서 주석) ...
63      * @param someParam 파라미터에 대한 설명
64      * @return String 반환값에 대한 설명
65      * @exception exception 예외사항에 대한 설명
66      */
67     public String doSomethingElse(Object someParam) {
68         // ... 여기에 실제 코드를 작성한다. ...
69     }
70 }
```

출처 : <https://myeonguni.tistory.com/1596>