https://imasoftwareengineer.tistory.com/73?category=769427

# Java

## 패키지(Package)란 무엇인가?

자바 패키지는 자바 클래스를 여러 namespace로 정리하고, 패키지에 소속된 각 class에게 고유한 namespace를 제공한다.

장점

1. 유지보수 : 관련 있는 class들을 패키지로 묶으면 특정 class의 위치를 찾을 때 더 빠르게 찾을 수 있다.
2. NameSpace 기능 : 동일한 이름의 파일에 대해서 namespace 기능을 제공한다.

# namespace란?

동일한 이름의 파일을 구분 할 수 있게 해주는 공간을 의미한다.  
( ex – A.B.C.java, B.D.C.java )

# 빌트 인 패키지(Built-in Package)

자주 쓰이는 패키지는 import를 하지 않아도 사용할 수 있도록 만들어져 있다.

(Ex – java.lang)

## 접근 제어자

객체 지향 디자인/개발 방법의 원칙 중에는 캡슐화라는 항목이 있다.

캡슐화는 어떤 변수나 메서드의 접근을 최대한 제한하는 것.

즉, 다시말해 공개하면 안되는 클래스/변수/메서드, 공개하지 않는 것이 좋은 클래스/변수/메서드 등은 공개하지 않는 것이 캡슐화이다.

접근제어가 없어서 코드를 짜고 코드를 패키지로 만들어서 배포되어 패키지가 제공하는 모든 변수와 메서드가 공개되어 있다면 보안 취약점 문제나, 비즈니스 로직의 상당 부분이 공개되는 문제가 발생할 수 있다.

Public : 같은 프로젝트

Default : 같은 패키지

Protect : 자기 자신과 자기를 상속하는 자식 클래스에게만 변수/메서드를 공개

Private : 같은 class 만