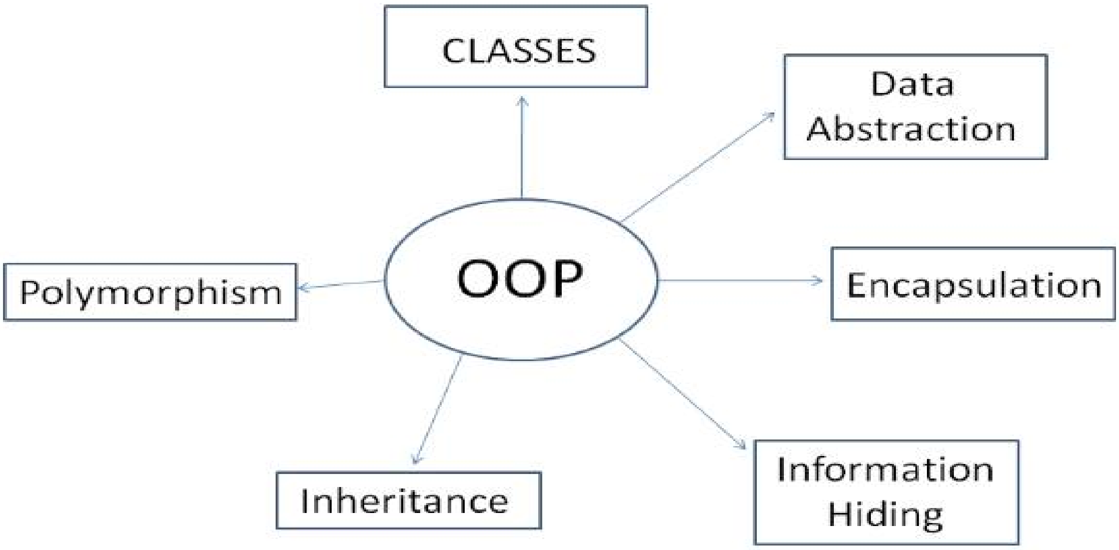
OOP 정리

Effective software design :

- KIS(Keep It Simple) No Overengineering, no Spaghetti code.
- DRY(Don't Repeat Yourself) Code duplication equals bug reuse.



Encapsulation :

- Information Hiding.

Classes :

- "Snowy is a dog." = "The Snowy object is an instance of the dog class."

Inheritance :

- "a dog is a mammal"
- What works for the parent class should also work for any subclass.

Polymorphism : 다형성

- Different subclasses can be treated like the parent class, but execute their specialized behavior.

Information Hiding : API만 알면 문제없이 사용 가능하게 한다.

- Be the principle of segregation of the design decisions in a computer program that are most likely to change.

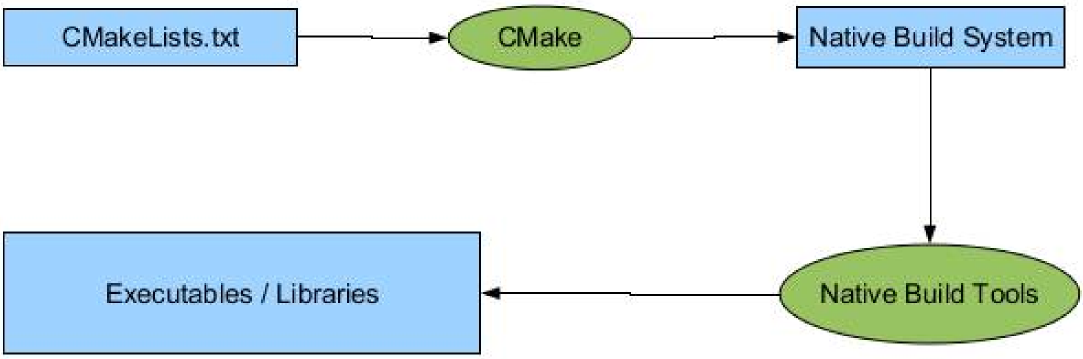
Can we make OOP applicable?

- It always helps to sketch with pen and paper.

CMake

Make vs CMake

- The make utility and Makefiles provide a build system that can be used to manage the compilation and recompilation of programs that are written in any programming language.
- CMake is a cross-platform MakeFile.



- Cmake는 makefile을 보다 쉽게 기술해 주는 일종의 meta-makefile이라고 할 수 있다.
- Cmake는 소스파일 내부까지 들여다보고 분석해서 의존성 정보를 스스로 파악한다.

Object-Oriented Programming in Python

1) 캡슐화
2) 상속
3) 오버로딩

Class = Function(Methods) + Data(Variables)