**Day 04 ╰(\*°▽°\*)╯**

* 객체지향 프로그래밍 방법론

객체(DATA + CODE) 간의 통신, 속성(Attribute)과 메소드(Code)

객체 object = 단일책임, 단일 역할 = 그와 관련된 다수의 기능과 다수의 데이터

= 독립적 실행 단위 = Object 단위로 메모리에 저장됨(메소드 하나만 사용해도 객체단위로 메모리에 올라감)

1. 캡슐화 – 외부에 데이터를 감추는 것
2. 추상화 – 특성에 집중하도록, 전문성을 띄도록 상세 내역 없애는 과정, 보편, 일반적인 것을 제외하고 가지치기라는 것, 디테일을 삭제하는 것
3. 다형성 – 식별자(함수 이름,인수 타입, 개수)를 비교하여 불러온다. 하나의 객체가 여러 타입을 가질 수 있다는 의미

* 프로젝트 방법론

데드라인을 지키는 것이 비용을 절감하는 것에 큰 도움이 된다.

따라서 프로젝트를 효율적으로 진행하기위해 프로젝트 진행 방법론이 탄생했다.

1. 폭포수 모델 (초기 모델) : 소규모에서 적용, 단계가 끝날 때마다 **산출물** 작성!

**요구사항 파악**(what, 기획)[**요구사항 정의서: Use-case Diagram**]

-> **분석**(how : 아키텍처, ERD, UI/UX 등등)[**각자 관점에 따라 쓰는 도구가 다름**]

-> **설계**(Design)[**Class Diagram**] -> **구현**(test)

1. 애자일 방법론

* 화면 좌표계 (데카르트 좌표계)

모니터의 기준 좌표 (0,0)는 왼쪽 맨 위가 기준이다. 하지만 이렇게 하면 y축을 -로 해야 하지만 그렇게 하면 계산을 두 번해야 하기 때문에 그냥 양수를 쓴다. 하지만 현재 위치보다 위로 올라가야 할 때는 빼기를 쓴다.