

객체지향개발(Object Oriented Programming)의 특성은 크게 추상화, 캡슐화, 상속성, 다형성이 있다.

### 1) 추상화(Abstraction)

-공통의 속성이나 기능을 묶어 이름을 붙이는 것.

객체 지향적 관점에서 클래스를 정의하는 것을 바로 추상화라고 정의 내릴 수 있겠다.

좀 더 살펴보면 물고기, 사자, 토끼, 뱀이 있을 때 우리는 이것들을 각각의 객체라 하며, 이 객체들을 하나로 묶으려 할 때, 만약 동물 또는 생물이라는 어떤 추상적인 객체로 크게 정의한다고 하자. 이때 동물 또는 생물이라고 묶는 것을 추상화라고 한다.

### 2) 캡슐화(Encapsulation)

-데이터 구조와 데이터를 다루는 방법들을 결합 시켜 묶는 것.

다시 한번 말하자면 변수와 함수를 하나로 묶는것을 말한다.

- ex)

```
public String test() {  
    string aa = "aaa";  
}
```

하지만 무작정 한데 묶으면 되는 것이 아니라객체가 맡은 역할을 수행하기 위한 하나의 목적을 한데 묶는다고 생각해야한다. 이것이 많이 들어본 의미로는 은닉화라고 한다.또한 데이터를 절대로 외부에서 직접 접근을 하면 안되고 오로지 함수를 통해서만 접근해야하는데 이를 가능하게 해주는 것이 바로 캡슐화이다.

따라서 캡슐화에 성공하면 당연히 은닉화도 자연스럽게 효력이 나타난다.

### 3) 상속성, 재사용(Inheritance)

- 상위 개념의 특징을 하위 개념이 물려받는 것

-

- 객체지향의 하이라이트 부분이라고 생각한다. 상속이란 개념이 없으면 객체지향이나 절차지향이나 도진개진  
- 간단히 예를 들자면, 자동차라는 부모클래스가 있다. 기름을 먹거나 달리는 기능을 하는 자동차인데,  
만약 지붕뚜껑이 열리는 특수한 기능을 추가하고 싶다면 기존의 자동차에서 스포츠카를 생성한다.  
그러면 스포츠카는 기름도 먹고 달리면서 지붕뚜껑이 열리는 기능도 갖춘 자동차가 되는 것

### 4) 다형성(Polymorphism)

- 부모클래스에서 물려받은 가상 함수를 자식 클래스 내에서 오버라이딩 되어 사용되는 것

간단히 예를 들자면 군대에서 나는 K2 소총을 잡았고 동기는 K1 소총을 잡았다. 사격 훈련이 있을 때 중대장이 '준비된 사수부터 발사!'라고 외치면 나와 내친구는 명령을 받고 앞의 사로를 향해 총을 쏜다. 이때 중대장은 추상적 객체를 상속받은 모든 객체들에게 명령을 내린것이고 그 병사가 총이 뭐든간에 그냥 발사를 하라는 명령을 한것이다. 즉, 다형성이 없다면 K1 소총을 든 병사 발사, K2 소총을 든 병사 발사 라며 명령을 하나하나 내려야 할 것이다

( \*\*단 objective-C 에서는 오버로딩은 지원하지 않는다. )