

1. 객체지향 프로그래밍에서 추상화, 캡슐화, 은닉화, 상속, 다형성 이라는 특징에 대해 알아보세요.

객체지향 프로그래밍

: 프로그램을 어떻게 설계해야 하는지에 대한 방법론

: 프로그램을 수 많은 '객체'라는 기본 단위로 나누고 이 객체들의 상호작용으로 서술하는 방식

특징

1) 추상화(abstraction)

관련 있는 것을 묶어서 이름을 부여하는 것

객체들의 공통적인 특징을 뽑아내는 것이다. 즉, 우리가 구현하는 객체들이 가진 공통적인 데이터와 기능을 도출해 내는 것을 의미한다. 즉, "공통의 속성이나 기능을 묶어 이름을 붙이는 것"

-> 클래스를 정의하는 과정이 추상화 과정이라고 할 수 있다.

2) 캡슐화(encapsulation)

'데이터'와 '데이터 다루는 방법'을 묶는 것

데이터 구조와 데이터를 다루는 방법을 결합시켜 묶는 것을 말한다. 특정 객체가 독립적으로 역할을 제대로 수행하기 위해 필요한 데이터와 기능을 하나로 묶어 관리한다. 객체가 맡은 역할을 수행하기 위한 하나의 목적을 위해 데이터와 기능을 묶는 것이다.

다른 의미로도 쓰이는데 즉, 데이터는 은닉하고 그 데이터를 접근하는 기능을 밖으로 노출한다는 의미를 나타낼 때 캡슐화라는 용어를 쓴다.

->데이터를 기능이라는 캡슐로 보호한다는 것이다.

3) 상속성(inheritance)

하나의 클래스가 가지고 있는 특징(데이터+조작)을 그대로 다른 클래스가 물려 받는 것

객체지향의 꽃이다. 상속이란 상위개념의 특징을 하위 개념이 물려받는 특징을 말한다. 하나의 클래스가 가지고 있는 특징들을 그대로 다른 클래스가 돌려주고자 할 때 상속성의 특징을 사용한다.

4) 다형성(polymorphism) : ‘여러 형태’를 의미하는 그리스어

다른 방법으로 일을 하는 함수의 동일한 이름으로 호출해 주는 것을 말한다.
같은 하나의 명령이 다른 결과로 나타나는 것을 말한다.

5) 은닉화

내부 데이터, 내부 연산을 외부에서 접근하지 못하도록 은닉(hiding) or 격리(isolation)시키는 것을 의미한다.
변수를 private로 지정한다.