

Homework #1

객체 지향 프로그래밍에서 추상화, 캡슐화, 은닉화, 상속, 다형성이라는 특징에 대해 알아보세요.

추상화 (abstraction)

공통적인 특징(데이터, 메소드 등)을 뽑아내는 것을 의미한다.
OOP에서 대표적인 추상화는 클래스를 만드는 것이다.
(추상적인 클래스를 만들고, 클래스로 객체(실체)를 만드는 것)

캡슐화 (encapsulation)

특징들(데이터, 메소드 등)을 하나로 묶는 것을 의미한다.
OOP에서 대표적인 캡슐화는 클래스, 객체를 보며 되는데
특정 클래스/객체가 추상/독립적으로 특징들을 묶어서 가진다.

은닉화 (hidden)

OOP는 객체 단위로 캡슐화하여 데이터와 메소드를 가지는데
캡슐화 할 때에, 외부에서 객체 내의 데이터와 메소드에 직접적으로 접근할 수 없게 하는 것을 은닉화라고 한다.
외부에서 접근 가능한 메소드를 통해, 접근할 수 없는 데이터와 메소드에 간접적으로 접근할 수 있게 설계할 수 있다.

상속 (inheritance)

새로운 클래스가 기존의 클래스를 상속하면, 기존 클래스의 데이터와 메소드를 이용할 수 있다.
여기서 상속 받은 클래스를 상위/부모 클래스라 부르고, 상속 한 클래스를 하위/자식 클래스라 부른다.
상속을 통해, 클래스 코드의 재사용성과 유연성을 가질 수 있다.

다형성(polymorphism)

동일한 이름을 다양한 기능(형태)으로 정의하거나 사용하는 것을 의미한다.
대표적으로 메소드의 오버로딩과 오버라이딩이 있다.

- 오버로딩(Overloading) : 같은 이름의 메소드가 매개변수의 자료형이나 개수에 따라서 다른 기능을 하는 것
- 오버라이딩(Overriding) : 상속관계에 있는 여러 클래스에서 같은 이름의 메소드가 재정의되어 다른 기능을 하는 것

구분	오버로딩	오버라이딩
메소드 이름	동일	동일
매개변수, 타입	다름	동일
리턴 타입	상관없음	동일