SOLID란? 객체지향에서 지켜야 할 디자인 원칙인 SRP, OCP, LSP, ISP, DIP의 앞 글자를 따서 만든 용어이다.

SRP란? 객체는 단 하나의 책임을 가져야 한다는 의미다.

책임이란? 보통 ‘해야하는 것’ 으로 간주할 수 있다. 객체에 책임을 할당 할 때에는 어떤 객체보다도 작업을 잘 할 수 있는 객체에 책임을 할당해야 한다. 또한 객체는 책임에 해당하는 일을 자신만이 수행 할 수 있어야 한다.

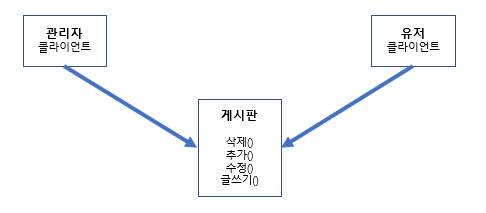
OCP란? 개방-폐쇄 원칙으로서 기존의 코드를 변경하지 않으면서 기능을 추가할 수 있도록 설계가 되어야 한다는 뜻이다.

LSP란? 리스코프 치환 원칙으로서 자식 클래스는 최소한 자신의 부모 클래스에서 가능한 행위는 수행할 수 있어야 한다는 뜻이다. LSP를 만족하면 프로그램에서 부모 클래스의 인스턴스 대신에 자식 클래스의 인스턴스로 대체해도 프로그램의 의미는 변화되지 않는다. 이를 위해 부모 클래스와 자식 클래스 사이는 행위가 일관 되어야한다.

DIP란? 의존 역전 원칙으로서 의존 관계를 맺을 때 변화하기 쉬운 것 또는 자주 변화하는 것보다는 변화하기 어려운 것, 거의 변화가 없는 것에 의존하라는 원칙이다.

ISP란? 인터페이스 분리 원칙으로서 객체지향 설계 원칙에는 클라이언트 자신이 이용하지 않는 기능에는 영향을 받지 않아야 한다는 의미이다. 즉, 인터페이스를 클라이언트에 특화되도록 분리시키는 설계 원칙이다.

ISP미적용 예시



ISP적용 예시

