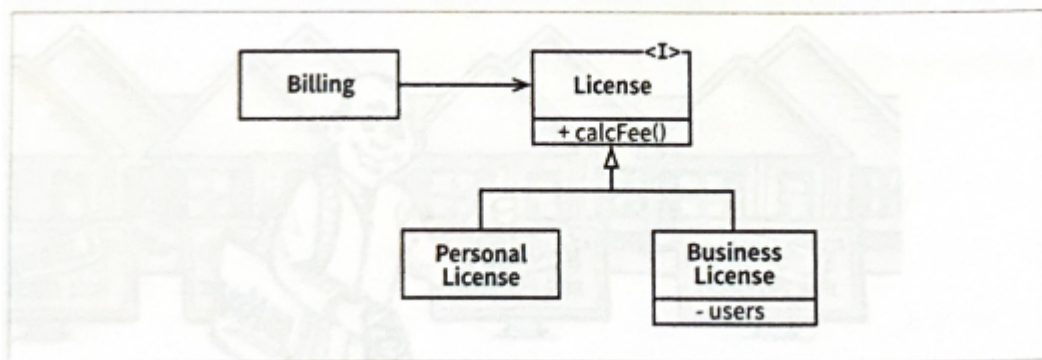


09장. LSP: 리스코프 치환 원칙

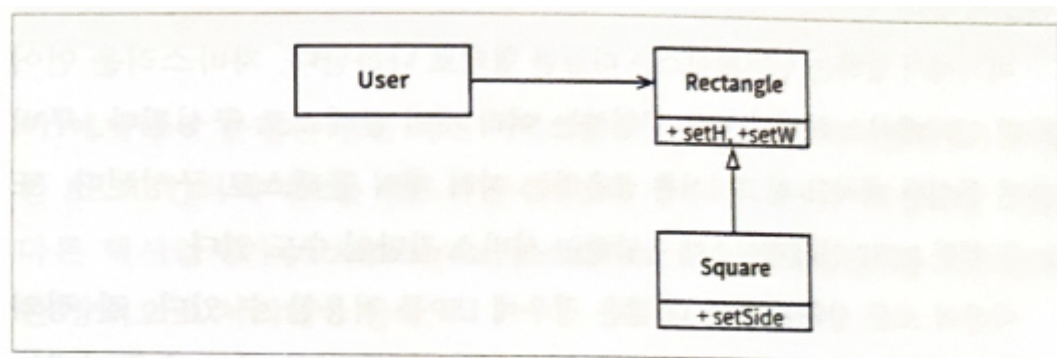
여기에서 필요한 것은 다음과 같은 치환원칙이다. S 타입의 객체 o1 각각 대응하는 T 타입 객체 o2가 있고, T 타입을 이용해서 정의한 모든 프로그램 P에서 o2의 자리에 o1을 치환하더라도 P의 행위가 변하지 않는다면, S는 T의 하위 타입이다.

상속을 사용하도록 가이드하기



- 위 사진은 LSP를 준수한다. Billing의 행위가 License 하위 타입중 어떤 것을 사용하는 지에 대해 의존하지 않기 때문이다.

정사각형/직사각형 문제



- 이 예제에서 Square는 Rectangle의 하위타입으로 적합하지 않다.
 - Rectangle은 직사각형, Square는 정사각형이기 때문
- LSP를 위반한다.

LSP와 아키텍처

- 상속을 넘어서 인터페이스와 구현체에도 적용되는 설계원칙으로 변했다.
- 아키텍처 관점에서 LSP를 이해하는 최선의 방법은 이 원칙을 어겼을 때 시스템 아키텍처에서 무슨일이 일어나는지 관찰하는 것이다.

LSP 위배 사례

REST API를 구현할 때도 같다. 인터페이스가 호환되지 않는다면 수많은 if문을 추가해야한다.

결론

LSP는 아키텍처 수준까지 확장할 수 있고, 반드시 확장해야만 한다.