

스프링 3주차 공부

모각소 4주차 (02.16)

박수인

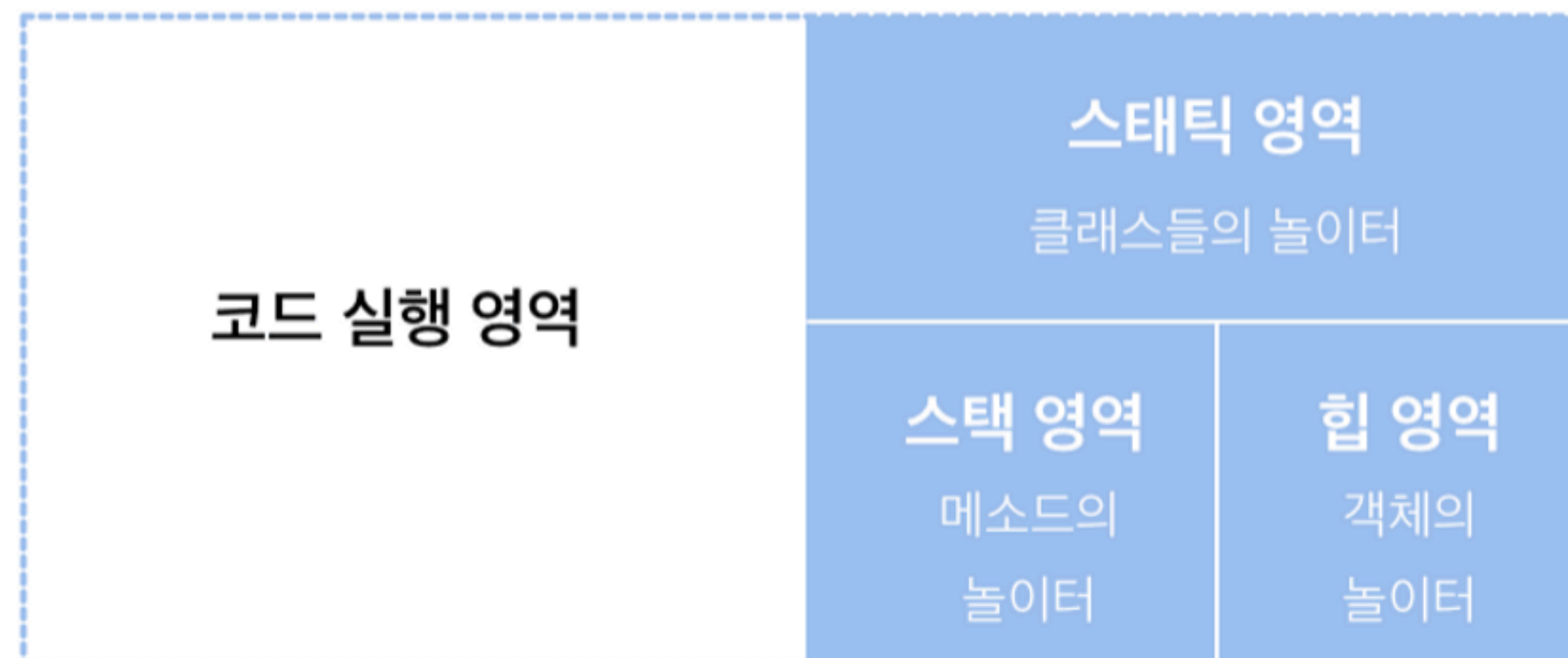
자바와 객체지향

- 클래스 : 객체 = 분류에 대한 개념 : 실체 = 펭귄 : 뽀로로

- 변수 유형

static 변수	클래스 [멤버] 속성, 정적 변수, 정적 속성 ...	스태틱 영역 (T메모리)
인스턴스 변수	객체 [멤버] 속성, 객체 변수 ...	힙 영역 (T메모리)
local 변수	지역 변수	스택 영역 (스택 프레임 내부)

아래를 T 메모리라고 임시로 부르자!



자바와 객체지향

- 객체 지향의 4대 특성

1. 캡슐화 Encapsulation: 정보은닉

- 접근 제어자
 - Private, default, protected, public

↑ 상속받은 클래스나,
같은 패키지 내 클래스에
서만 접근가능

2. 상속 Extends: 재사용 + 확장

- 하위 클래스 is a kind of 상위 클래스
Like 고래는 포유류이다

3. 추상화 Abstraction: 모델링

- 관심 영역 = Application Boundary
- [클래스] [객체 참조 변수] = new [클래스] ();

4. 다형성 Polymorphism: 사용편의성

- 오버라이딩 overriding: 재정의
- 오버로딩 overloading: 중복 정의

객체지향 설계 5원칙(SOLID)

' High Cohesion, Loose Coupling '

1. SRP - 단일 책임 원칙

- '어떤 클래스를 변경해야 하는 이유는 오직 하나뿐이어야 한다'

2. OCP - 개방 폐쇄 원칙 ★

- '자신의 확장에는 열려 있고, 주변의 변화에 대해서는 닫혀 있어야 한다'
- 해당 원칙을 위배하면 → 유연성, 재사용성, 유지보수성 등의 장점을 얻을 수 없음

3. LSP - 리스코프 치환 원칙

- '서브 타입은 언제나 자신의 기반 타입(base type)으로 교체할 수 있어야 한다'
- '하위 클래스의 인스턴스는 상위형 객체 참조 변수에 대입해 상위 클래스의 인스턴스 역할을 하는데 문제 없어야 한다'
- 하위 클래스 is a kind of 상위 클래스
- 구현 클래스 is able to 인터페이스
- 계층도/조직도 구조 - 해당 원칙 **위배**, 분류도 - 만족

객체지향 설계 5원칙(SOLID)

4. **I**SP - 인터페이스 분리 원칙

- '클라이언트는 자신이 사용하지 않는 메서드에 의존 관계를 맺으면 안된다'

5. **D**IP - 의존 역전 원칙

- '추상화된 것은 구체적인 것에 의존하면 안된다. 구체적인 것이 추상화된 것에 의존해야 한다'

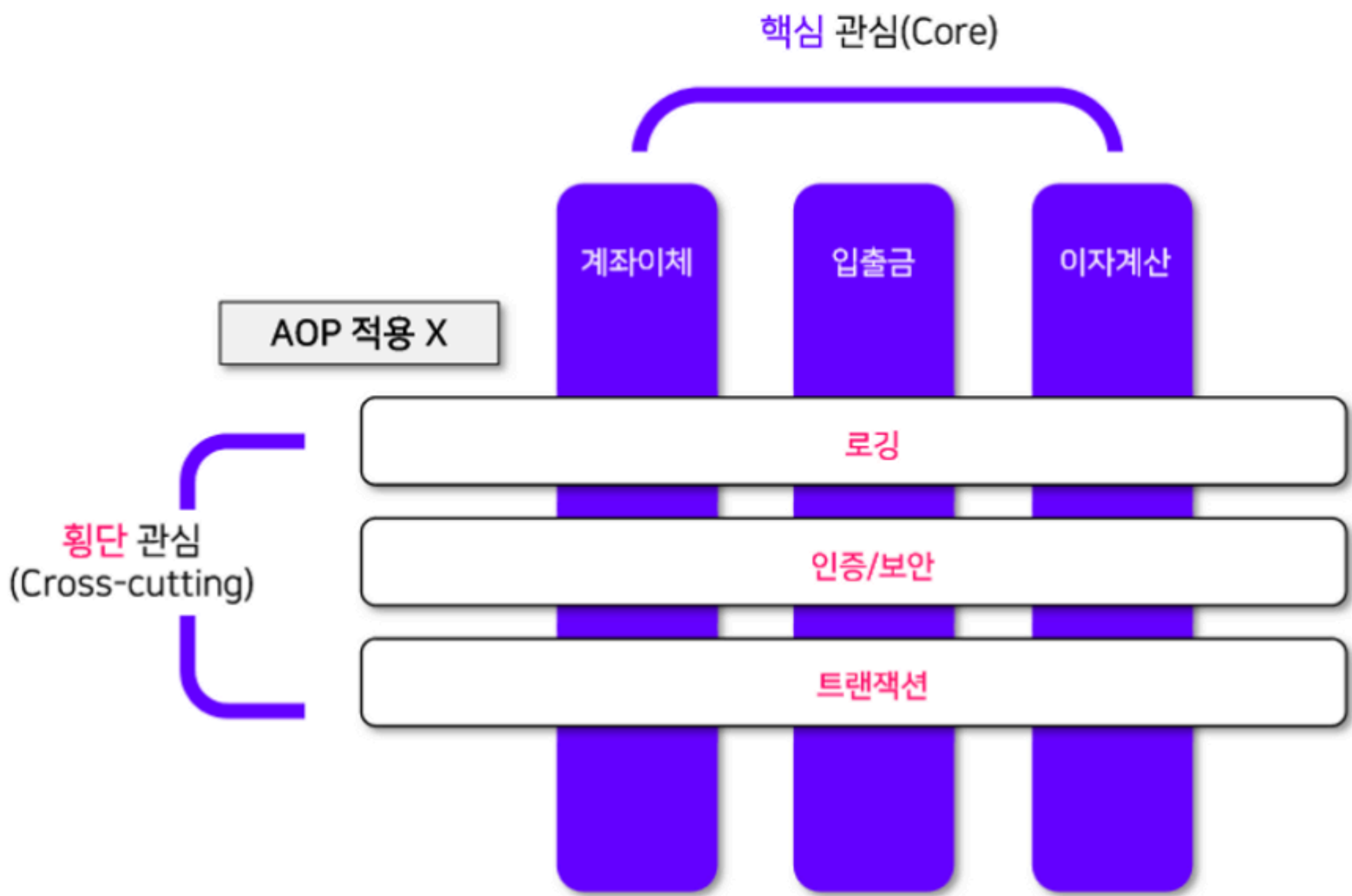
디자인 패턴

- 1. Adapter Pattern
- 2. Proxy Pattern
- 3. Decorator Pattern
- 4. Singleton Pattern
- 5. Template method Pattern
- 6. Factory Pattern
- 7. Strategy Pattern
- ...

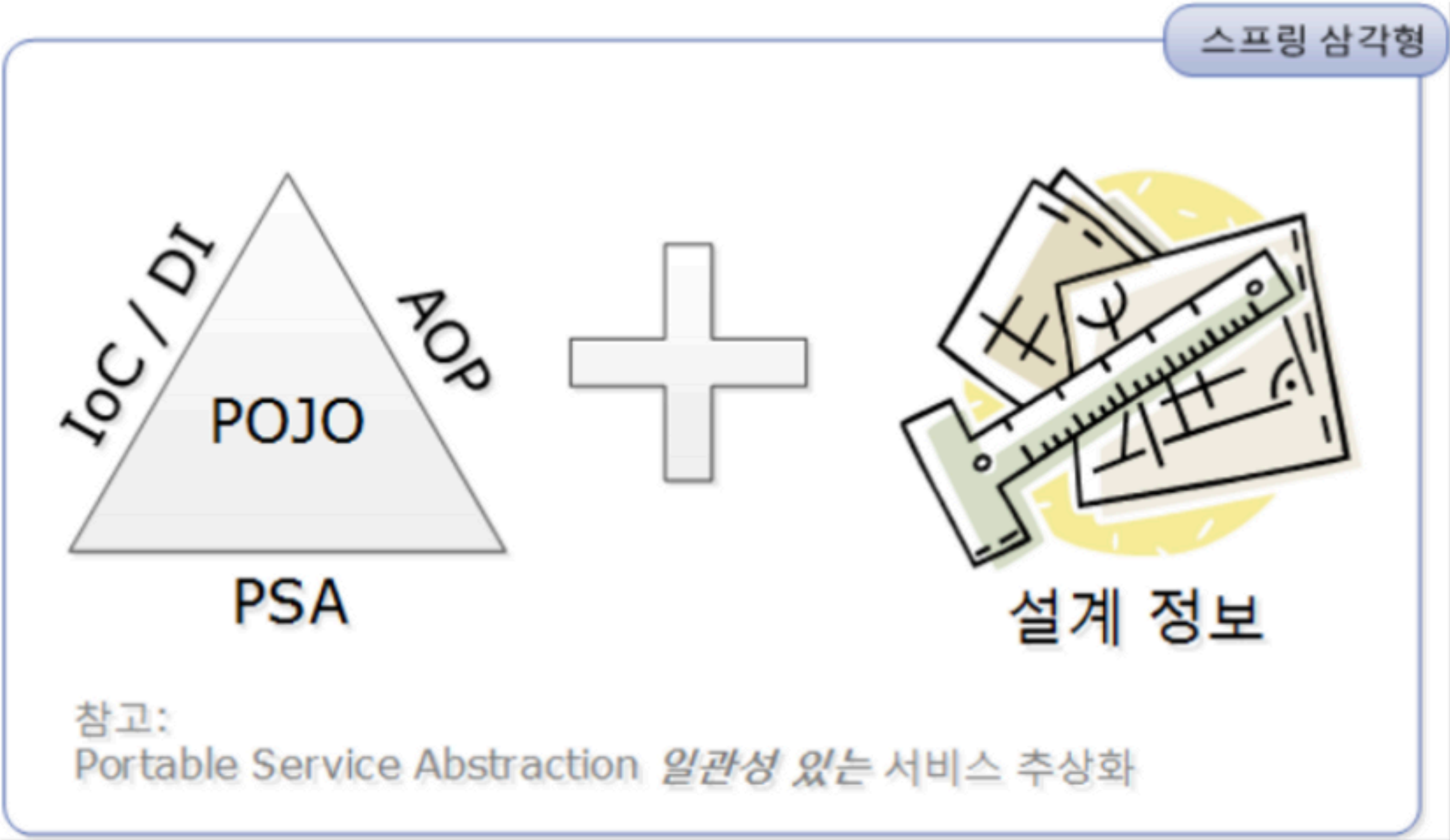
요리	OOP
요리도구	4대 원칙
요리도구 사용법	설계 원칙(SOLID)
레시피	디자인 패턴

스프링 삼각형과 설정정보

- 1. IoC/DI : 제어의 역전 / 의존성 주입
- 2. AOP : 관점 지향 프로그래밍



횡단 관심사 : 다수의 모듈에 공통적으로 나타나는 부분



- 3. PSA : 일관성 있는 서비스 추상화