



Background

- ✓ Spring Overview
- ✓ Dependency Injection

Goal

- ✓ Spring Framework 의 주요 특징과 기능에 대해서 이해한다.
- ✓ Dependency Injection 의 다양한 방법을 이해하고 사용한다.

Problem

ITSKB 은행은 고객 관리 시스템의 Back-End 에 Spring Framework 도입을 검토하고 있다. 이를 위해 Spring Framework 의 다양한 기능을 이해하고 적용하려고 한다. 먼저 주요 특징 중 하나인 DI 에 대해 테스트를 진행하려고 한다. 직접 객체를 생성하는 대신 Spring Framework 의 DI 로 객체를 얻고 사용해 보자. 다음의 가이드대로 진행한다.

1. Spring Tool Suite 을 설치, 사용한다.
2. Spring MVC 프로젝트를 생성한다. (프로젝트명 : SP_WS_01)
기본 package 는 com.itskb.myapp 를 사용한다.
3. 기본 package 아래 calculator package 를 생성한다. 그 곳에 다음과 같은 파일을 작성한다.
 - 3-1. 저축 이자 및 대출 이자를 계산해 주는 InterestCalculator 인터페이스를 작성한다. 금액을 입력 받아 이율을 적용해 이자를 리턴하는 calculate() 를 가진다.
 - 3-2. 위 인터페이스를 구현하는 LoanInterestCalculator 와 SavingInterestCalculator 를 작성한다. 이율은 적절하게 부여한다.



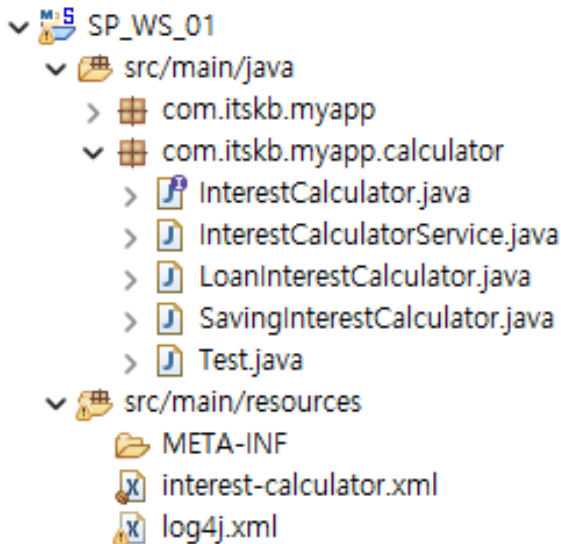
3-3 InterestCalculatorService 는 Spring Framework의 Component 이면서 Spring 의 DI 를 이용해서 InterestCalculator 를 구현한 객체를 얻고 사용한다.

3-4. Test 는 main() 를 이용해서 Spring Framework를 사용한다. interest-calculator.xml 을 이용해서 ClassPathXmlApplicationContext 를 생성한다. InterestCalculatorService 를 이용해서 1억원에 대한 대출 이자와 저축 이자를 계산하고 그 결과를 출력한다.

3-5. interest-calculator.xml 은 annotation 기반으로 Spring Component 를 인식하고 처리한다.

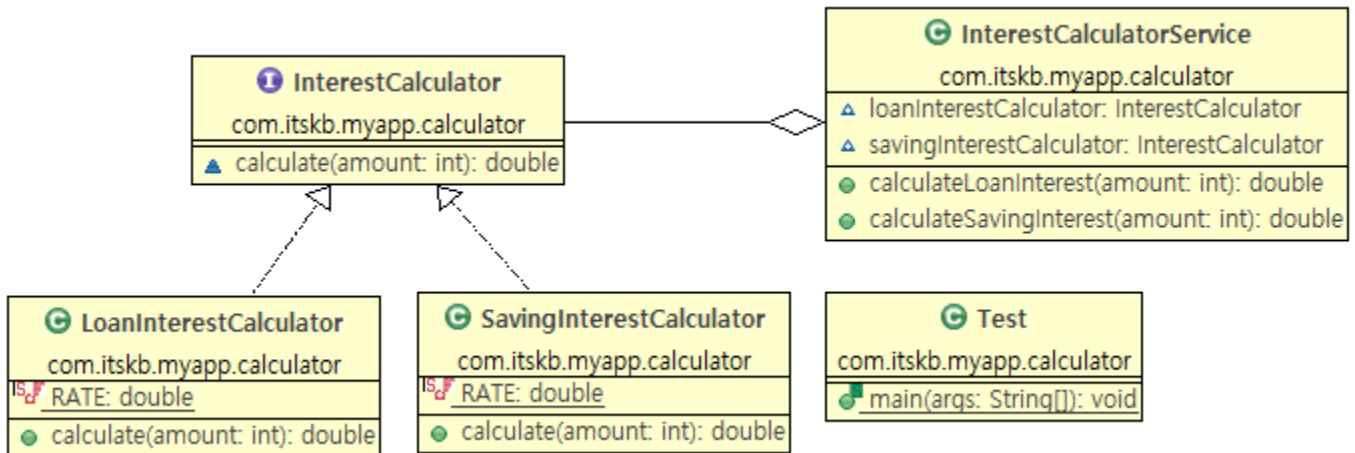
3-6. 대출 이자와 은행 이자를 구분하기 위한 annotation 을 이용해 본다.

4. 프로젝트의 구조 및 Test 파일 실행 내용은 아래와 같다.



```
LoanInterestCalculator - calculate()
Amount : 100000000 loanInterest is 5250000.00
SavingInterestCalculator - calculate()
Amount : 100000000 savingInterest is 3340000.00
```

5. Class Diagram.



❖ 제출방법

- 프로젝트명 : "SP_WS_01" 으로 작성
- 완성 후 프로젝트를 SP_WS_01_반_성명.zip으로 압축하여 제출