

2강. 스프링 프로젝트 만들기

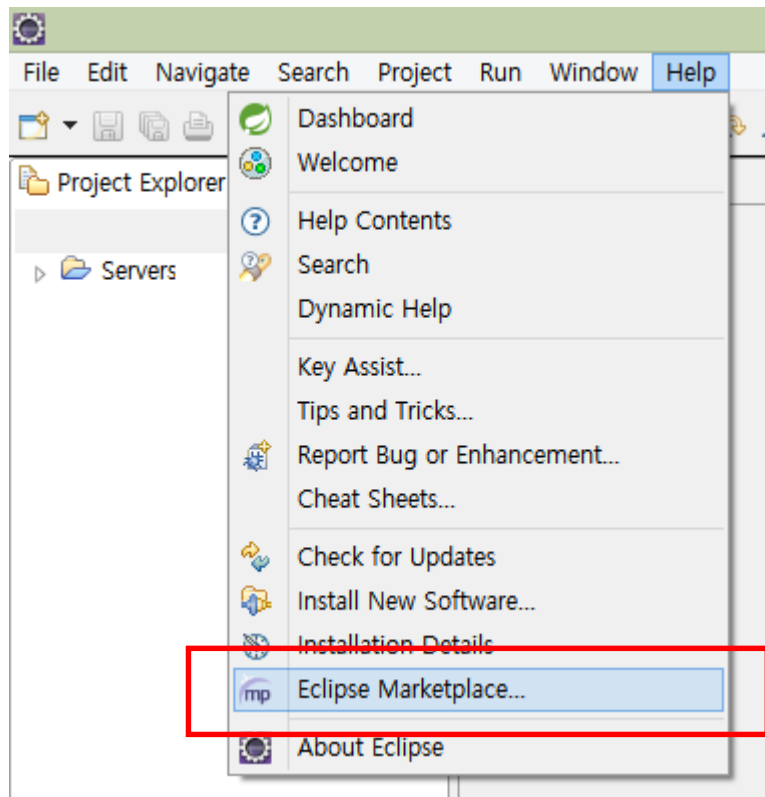
- 이클립스에 스프링 프로그램 설치(STS)
- 처음 만들어 보는 스프링 프로젝트
- DI(Dependency Injection)와 IOC컨테이너

Lecturer Kim Myoung-Ho
Nickname 불스
blogstudy@naver.com

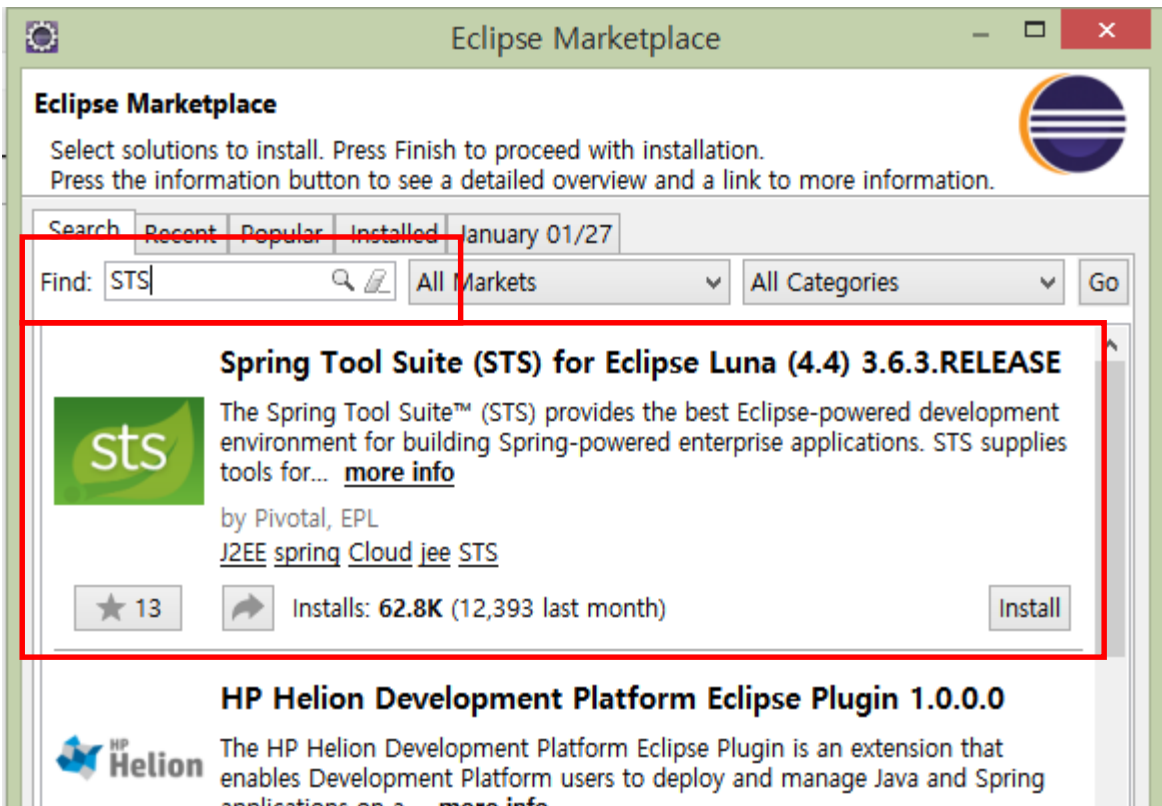
2-1. 이클립스에 스프링 플로그인 설치(Spring Tool Suite)

이클립스에서 스프링 프로젝트를 만들기 위해서는 스프링 플로그인이 필요 합니다.

이클립스 실행 및 Marketplace.... 진입



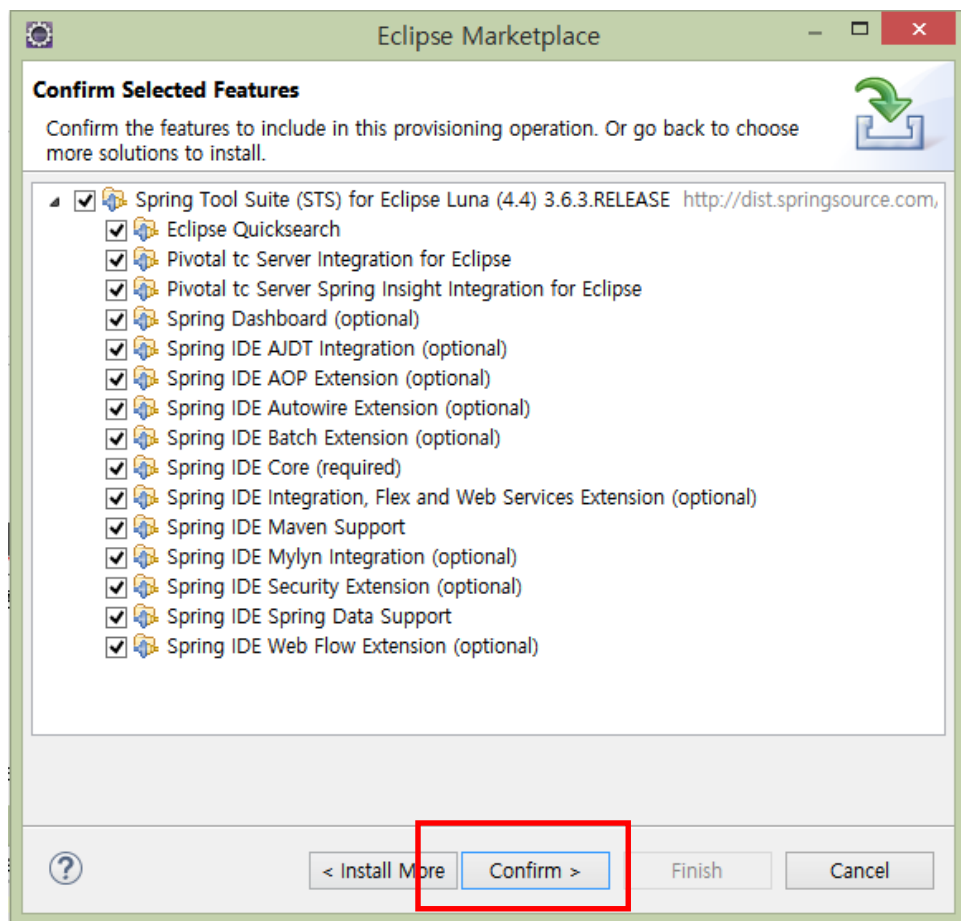
STS 플러그인 검색 하기



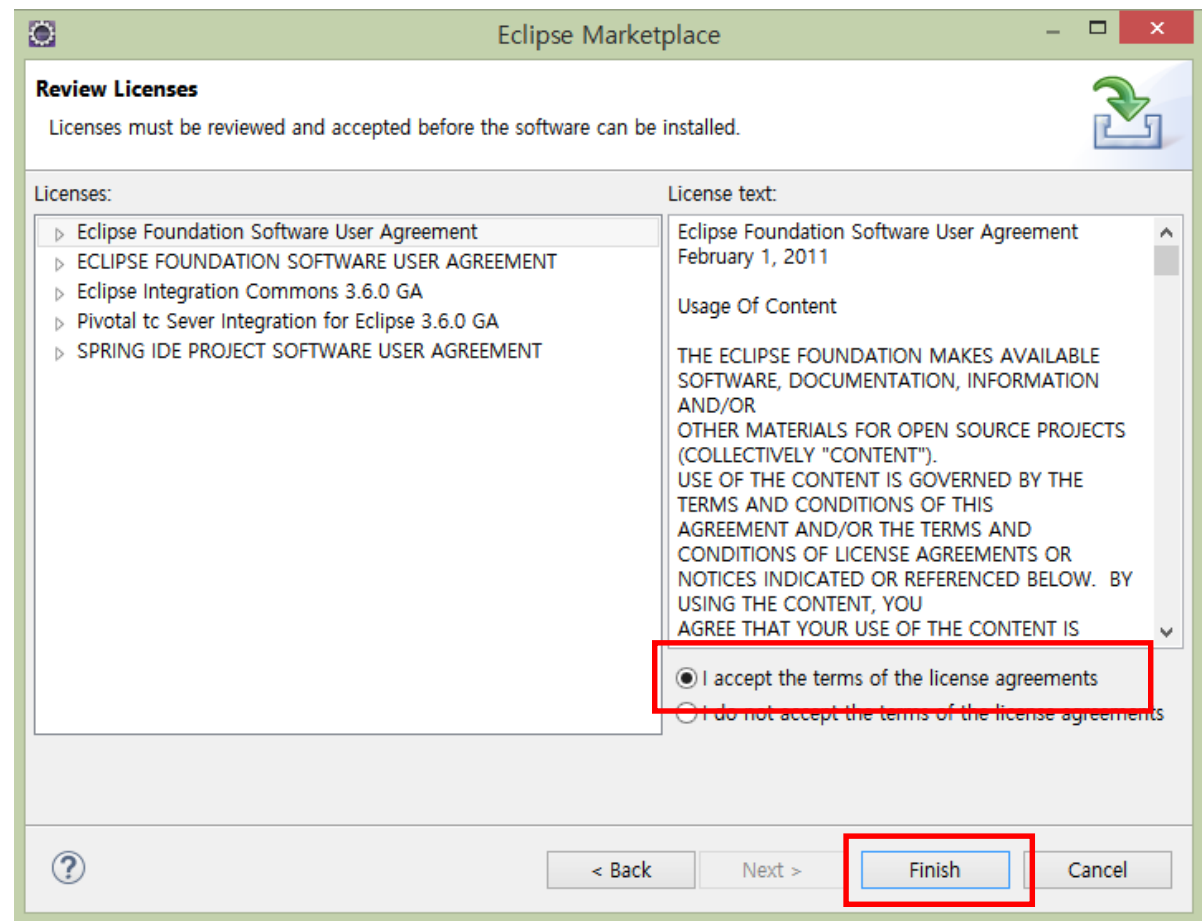
2-1. 이클립스에 스프링 플로그인 설치(Spring Tool Suite)

이클립스에서 스프링 프로젝트를 만들기 위해서는 스프링 플러그인이 필요 합니다.

플러그인 항목들 모두 선택 하고 Confirm 버튼 클릭



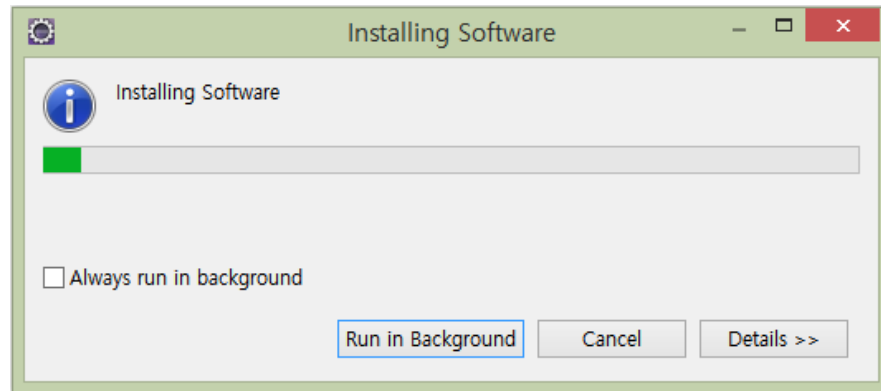
약관에 동의 후 진행



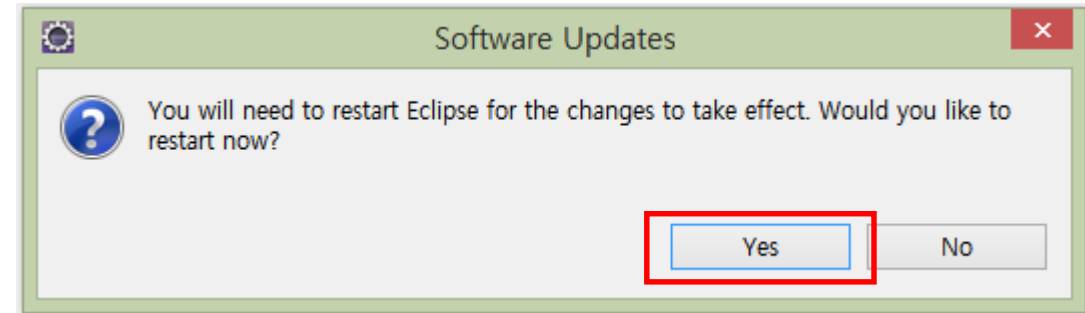
2-1. 이클립스에 스프링 플로그인 설치(Spring Tool Suite)

이클립스에서 스프링 프로젝트를 만들기 위해서는 스프링 플로그인이 필요 합니다.

설치 진행중



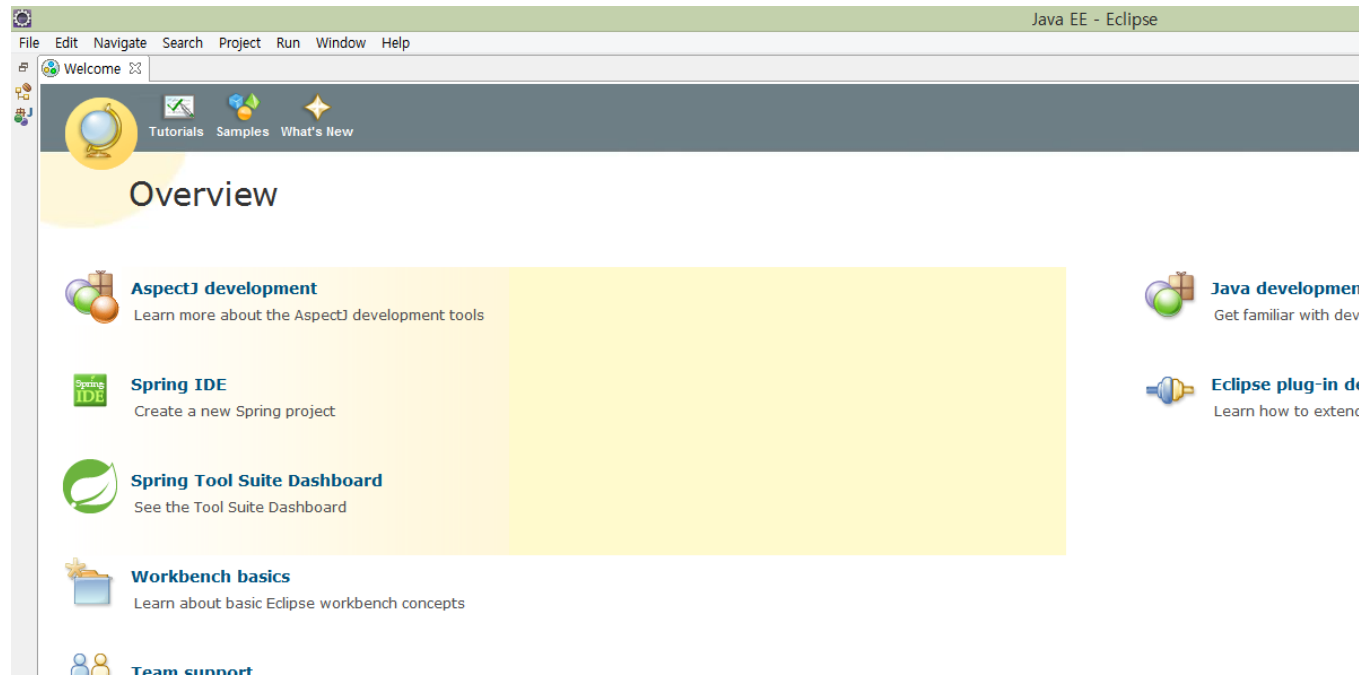
설치 완료 후 재부팅



2-1. 이클립스에 스프링 플로그인 설치(Spring Tool Suite)

이클립스에서 스프링 프로젝트를 만들기 위해서는 스프링 플로그인이 필요 합니다.

Overview 및 프로젝트 준비 완료



2-2. 처음 만들어 보는 스프링 프로젝트

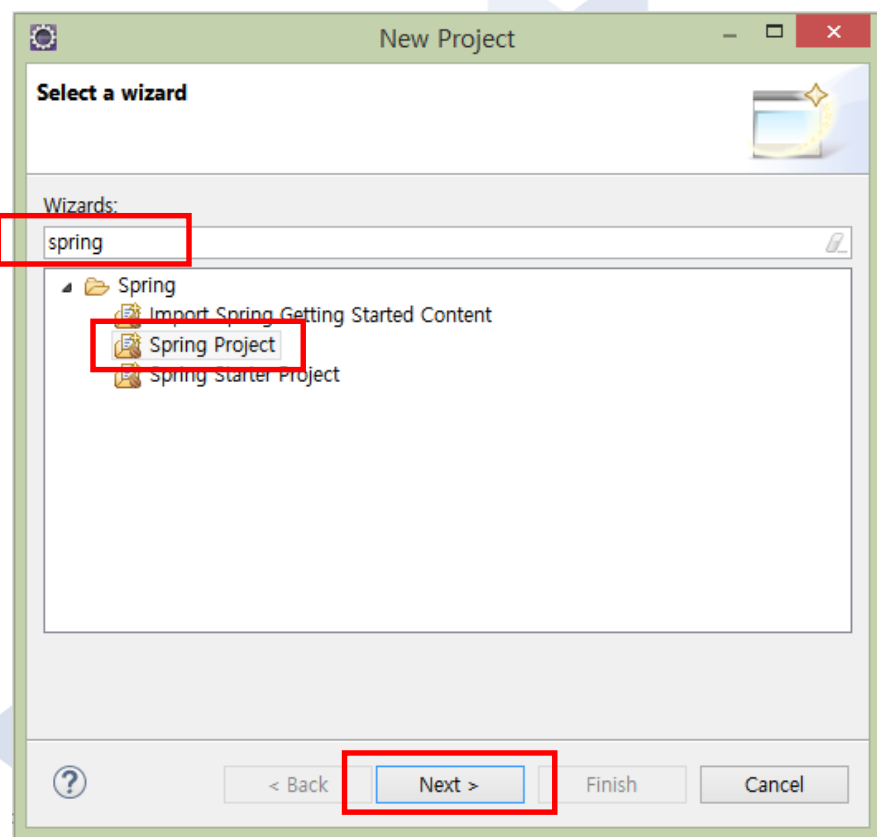
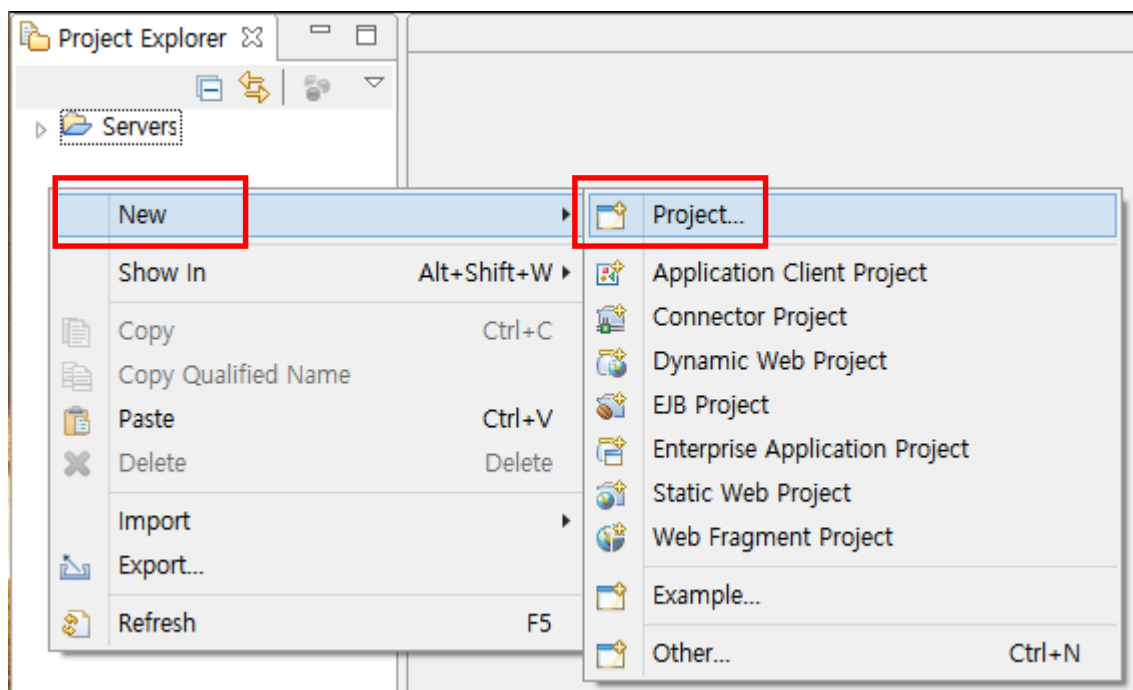
프로그래밍 언어를 처음 접할 때는 최대한 쉽게 접근 해야 합니다.

이유는 처음부터 많은 내용과 깊이 내용을 학습하여 하다 보면, 잘못된 학습으로 이어질 수 있으며, 조기에 지쳐서 학습에 흥미를 잃어 버릴 수 있습니다.

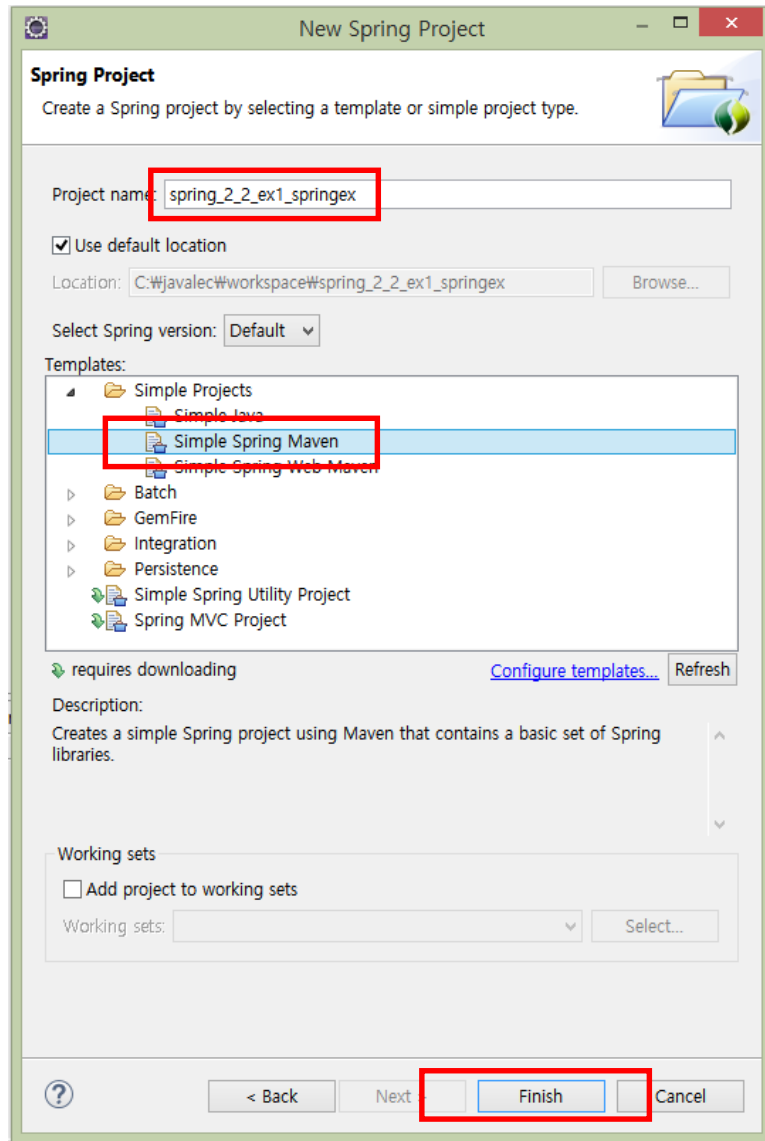
하여, 앞으로의 스프링 강의는 아주 쉽고 기초를 튼튼히 하기 위한 커리큘럼으로 진행 할 예정 입니다.

스프링을 최대한 쉽게 접근하려는 또 다른 의도는 기초부분에 대한 학습이 튼튼하게 되면, 추후에 스스로 스프링 레퍼런스 문서(웹문서 포함)들을 참고로 해서 스스로 학습 할 수 있기 때문 입니다.

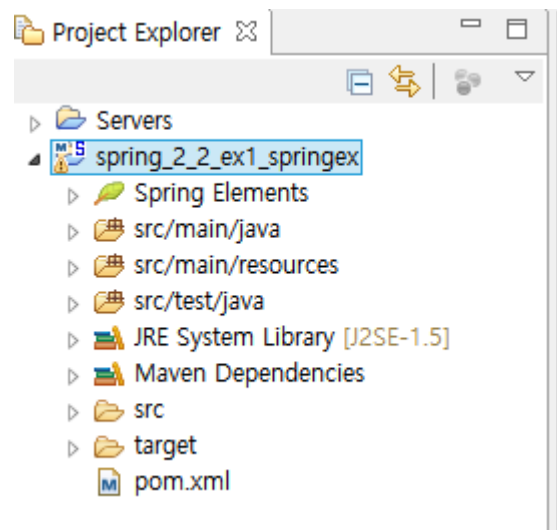
스프링 프로젝트 생성 하기



2-2. 처음 만들어 보는 스프링 프로젝트

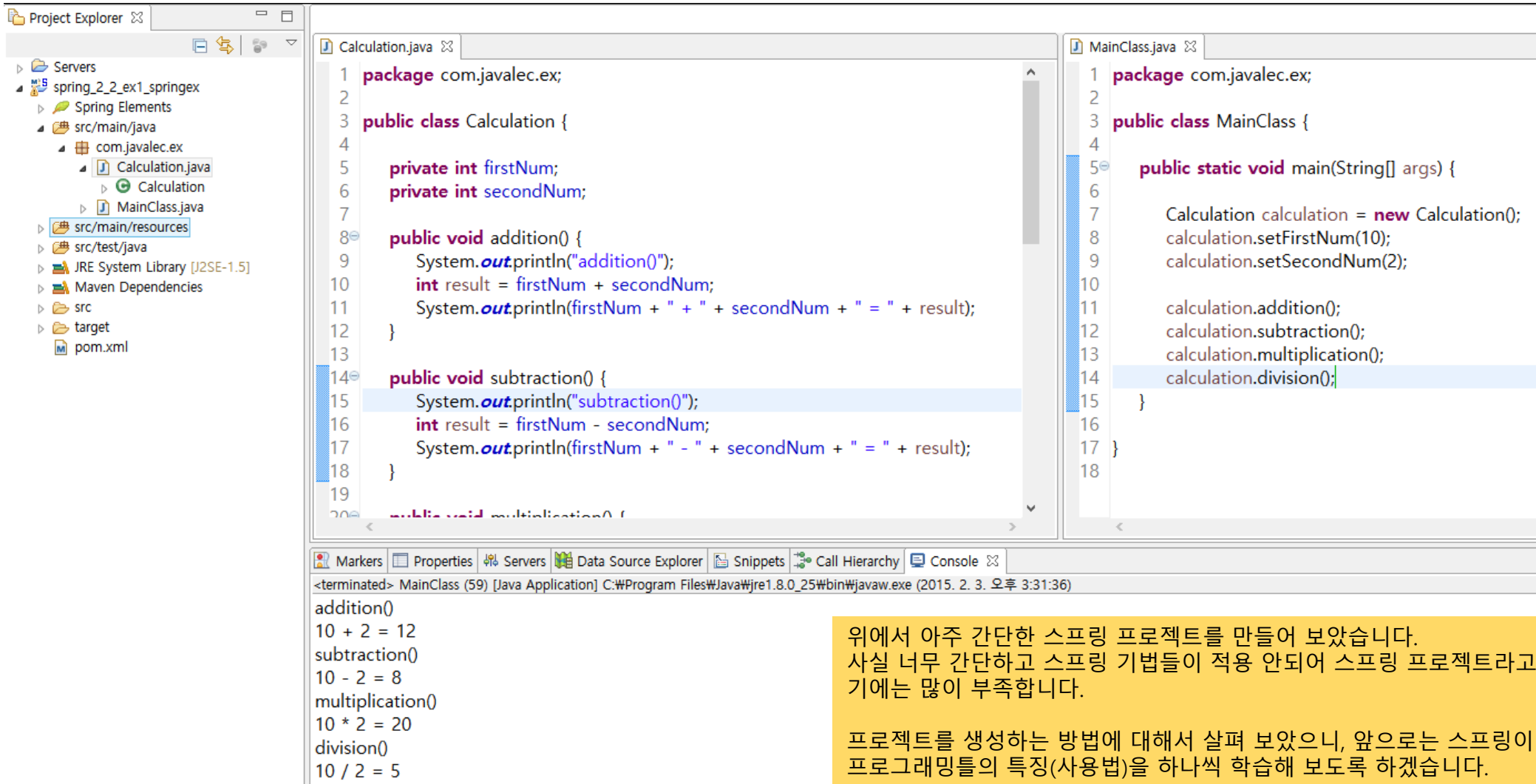


스프링 프로젝트 생성 완료



2-2. 처음 만들어 보는 스프링 프로젝트

스프링 프로젝트를 이용한 사칙연산 프로그래밍 (spring_2_2_ex1_springex)



The screenshot displays an IDE interface for a Spring project named 'spring_2_2_ex1_springex'. The Project Explorer on the left shows the project structure, including 'src/main/java' and 'src/main/resources'. The main editor shows two files: 'Calculation.java' and 'MainClass.java'.

Calculation.java

```
1 package com.javalec.ex;
2
3 public class Calculation {
4
5     private int firstNum;
6     private int secondNum;
7
8     public void addition() {
9         System.out.println("addition()");
10        int result = firstNum + secondNum;
11        System.out.println(firstNum + " + " + secondNum + " = " + result);
12    }
13
14    public void subtraction() {
15        System.out.println("subtraction()");
16        int result = firstNum - secondNum;
17        System.out.println(firstNum + " - " + secondNum + " = " + result);
18    }
19
20    public void multiplication() {
```

MainClass.java

```
1 package com.javalec.ex;
2
3 public class MainClass {
4
5     public static void main(String[] args) {
6
7         Calculation calculation = new Calculation();
8         calculation.setFirstNum(10);
9         calculation.setSecondNum(2);
10
11         calculation.addition();
12         calculation.subtraction();
13         calculation.multiplication();
14         calculation.division();
15     }
16 }
17
18 }
```

Console Output

```
<terminated> MainClass (59) [Java Application] C:\Program Files\Java\jre1.8.0_25\bin\javaw.exe (2015. 2. 3. 오후 3:31:36)
addition()
10 + 2 = 12
subtraction()
10 - 2 = 8
multiplication()
10 * 2 = 20
division()
10 / 2 = 5
```

위에서 아주 간단한 스프링 프로젝트를 만들어 보았습니다.
사실 너무 간단하고 스프링 기법들이 적용 안되어 스프링 프로젝트라고 하기에는 많이 부족합니다.

프로젝트를 생성하는 방법에 대해서 살펴 보았으니, 앞으로는 스프링이라는 프로그래밍틀의 특징(사용법)을 하나씩 학습해 보도록 하겠습니다.

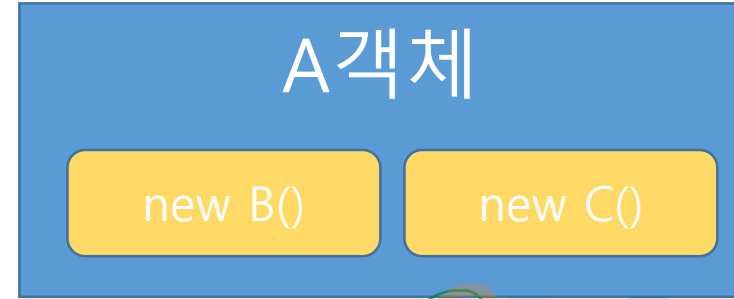
2-3. DI(Dependency Injection)와 IOC컨테이너

외부에서 의존하는 객체를 만들고 주입해주는 것.



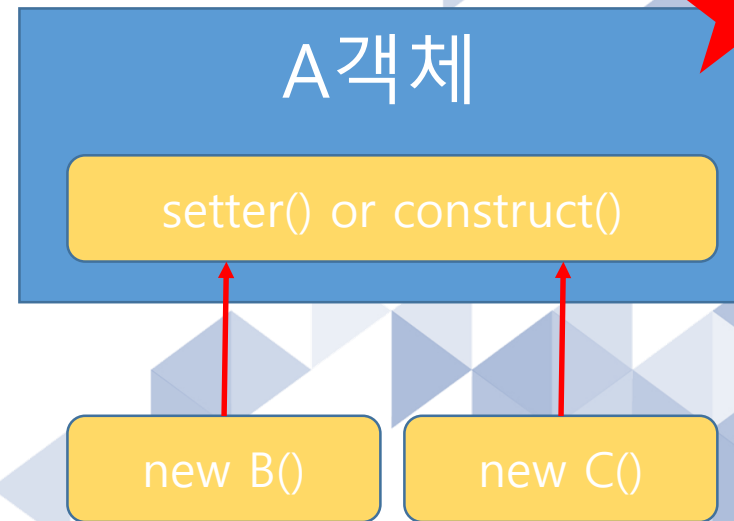
A객체는 B/C객체에 의존 한다.

방법1



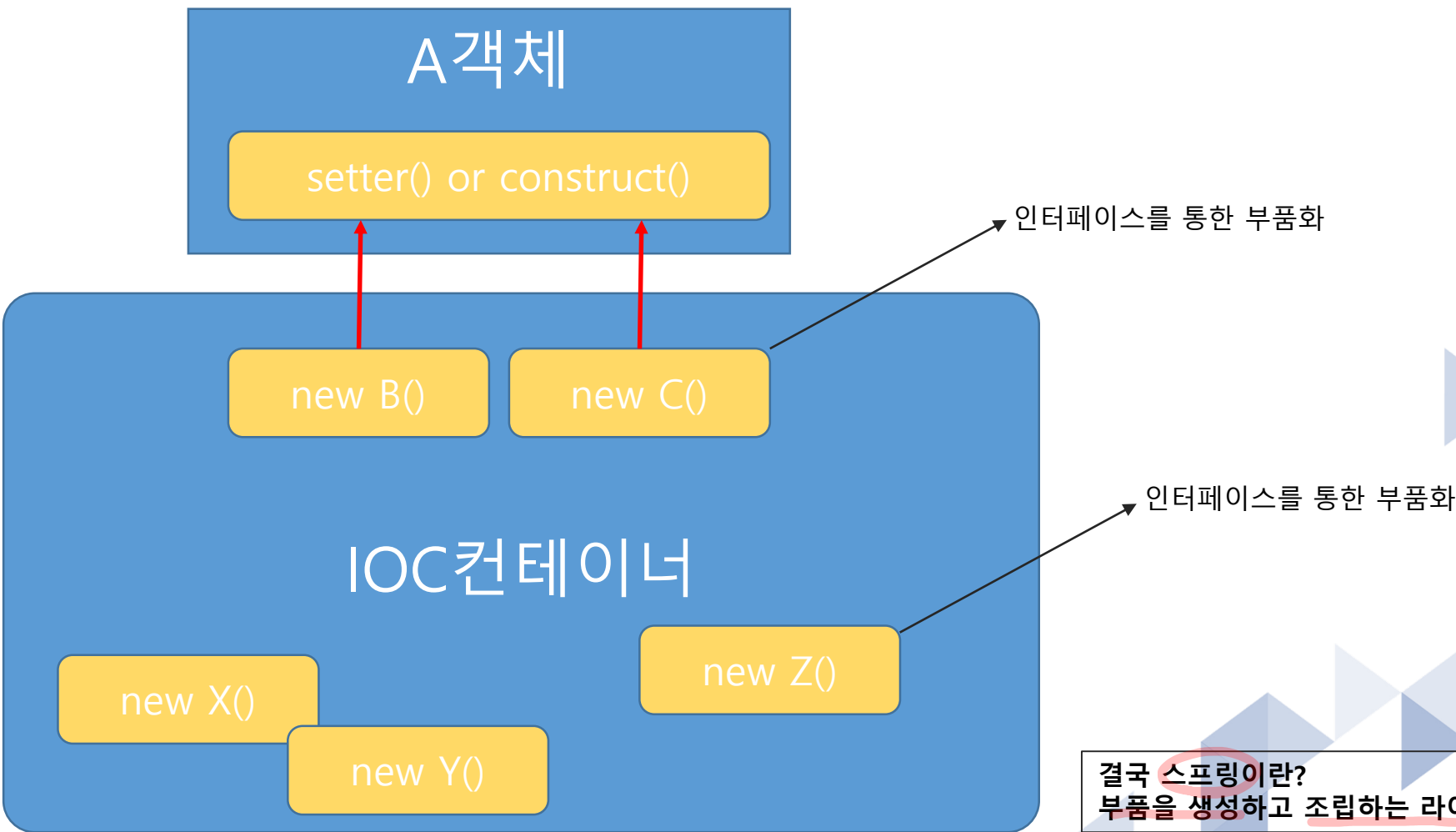
A객체가 B/C객체를 직접 생성 한다.

방법2



B/C객체 외부에 생성하여 A객체에 넣어 준다.

2-3. DI(Dependency Injection)와 IOC컨테이너



결국 스프링이란?
부품을 생성하고 조립하는 라이브러리 집합체 라고 할 수 있습니다.