

# 10. 다음으로

#1.인강/2.핵심원리 - 기본편/기본편#

## 다음으로

### 지금까지 학습한 내용 목차

1. 객체 지향 설계와 스프링
2. 스프링 핵심 원리 이해1 - 예제 만들기
3. 스프링 핵심 원리 이해2 - 객체 지향 원리 적용
4. 스프링 컨테이너와 스프링 빈
5. 싱글톤 컨테이너
6. 컴포넌트 스캔
7. 의존관계 자동 주입
8. 빈 생명주기 콜백
9. 빈 스코프

### 학습 내용 정리

#### 스프링 핵심 원리와 핵심 기능

- 스프링 핵심 원리
  - 객체 지향 설계
    - ◆ 다형성 - 역할과 구현의 분리
    - ◆ SOLID [OCP, DIP]
    - ◆ DI 컨테이너
- 스프링 핵심 기능
  - 스프링 컨테이너, 빈
  - 싱글톤
  - 컴포넌트 스캔
  - 의존관계 자동 주입
  - 빈 생명주기 콜백
  - 빈 스코프

지금까지 우리는 스프링의 핵심 원리와 핵심 기능에 대해서 깊이있게 학습했다. 스프링이 왜 만들어졌고, 왜 필요한지,

그리고 객체 지향 설계와 스프링이 왜 떨어 수 없는 관계인지 이해했다.

스프링의 핵심 원리와 핵심 컨셉을 제대로 학습했기 때문에 스프링 웹 MVC, 스프링 데이터 접근 기술, 스프링 부트를 포함해서 스프링의 핵심 기술을 활용하는 수 많은 스프링 기술들을 배우고 사용할 때도, 단순한 기능 사용을 넘어서 깊이있는 이해가 가능할 것이다.

## 스프링 완전 정복 시리즈(진행중)

스프링을 완전히 마스터 할 수 있는 다음 시리즈를 준비중이다.

실제 실무에서 사용하는 핵심 스프링 기능 위주로 설명하고 실무에서 사용하지 않거나 오래된 기능은 과감하게 삭제했다. 그리고 실무 노하우를 전수한다.

### 강의 목록

- [입문] 스프링 입문 - 코드로 배우는 스프링 부트, 웹 MVC, DB 접근 기술(오픈)
- [초급] 스프링 핵심 원리 - 기본편(오픈)
- [초급] 실전! 스프링 웹 MVC (진행중)
- [초급] 실전! 스프링 DB 접근 기술 (예정)
- [초급~중급] 실전! 스프링 부트 (예정)
- [중급] 스프링 핵심 원리 - 중급편(예정)

### 실전! 스프링 웹 MVC - 다음 출시 예정

**목표:** 이 강의로 웹 애플리케이션을 개발할 때 필요한 모든 웹 기술 완성

- 웹 기초부터 실무까지: 실무 개발을 위한 내용이 많지만, 웹 애플리케이션 개발 경험이 없는 개발자도 적응할 수 있게 구성
- 자바 웹 기술 이야기: 서블릿, 필터 같은 자바 기본 웹 기술부터 시작해서 왜 스프링 MVC가 나오게 되었는지 설명
- 실무 예제 중심: 실무에서 스프링으로 웹 애플리케이션을 개발할 때 필요한 모든 내용을 예제로 자연스럽게 학습, 실무 베스트 프랙티스 전수
- 간결함: 실무에서 사용하지 않는 오래된 기능은 과감하게 제외, 최신 실무에 주로 사용하는 내용 위주로 설명

## 스프링 부트와 JPA 실무 완전 정복 로드맵

최신 실무 기술로 웹 애플리케이션을 만들어보면서 학습하고 싶으면 **스프링 부트와 JPA 실무 완전 정복 로드맵**을 추천한다. URL: <https://www.infllearn.com/roadmaps/149>

### 강의 목록

- [초급] 자바 ORM 표준 JPA 프로그래밍
- [초급] 실전! 스프링 부트와 JPA 활용1 - 웹 애플리케이션 개발

- [중급] 실전! 스프링 부트와 JPA 활용2 - API 개발과 성능 최적화
- [중급] 실전! 스프링 데이터 JPA
- [중급] 실전! Querydsl

## 추천 학습 코스

### 목표: 최신 기술의 자바 백엔드 개발자

- 취업 준비생
- 실무 주니어 백엔드 개발자
- 최신 개발 트렌드를 깊이 이해하고 싶은 실무 백엔드 개발자

**핵심:** 실무에서 프로젝트를 진행할 때 어떻게 설계하고, 개발하는지, 그리고 최근에는 기술들을 사용하는지 경험해보고, 큰 그림을 이해한 상태에서, 그 다음에 세부적인 각각의 기술을 학습해야 한다. → 야생형 코스

- 학자형 코스 vs 야생형 코스
  - 학자형 코스: 이론을 먼저 차근차근 쌓아올린 다음에 실무 활용으로 넘어가는 방식
  - 야생형 코스: 일단 실무에서 어떤 기술을 어떤식으로 활용하는지, 깊이가 부족해도 일단 경험해본 다음에 필요에 의해서 이론 기술들을 공부하는 방식

분야마다 다를 수는 있지만 개발 공부는 야생형 스타일로 학습하는 개발자가 빨리 성장한다. 왜냐하면 내가 해결해야 할 문제가 뭔지, 알고난 다음에 문제 해결에 필요한 기술을 학습하기 때문이다. 내가 해결해야 할 문제가 뭔지도 모른 상태로 단순히 이론만 계속 학습하면 이 기술을 왜 배워야 하는지 이해하기도 쉽지 않고, 무엇보다 중간에 포기하기 쉽다.

### 추천 야생형 코스

1. 스프링 입문
2. 스프링 핵심 원리 - 기본편
3. 스프링 부트와 JPA 실무 완전 정복 로드맵
  1. 타입 B - 야생형(추천)

### 타입 B - 야생형(추천)

1. **활용편 1** - 먼저 활용편 1을 꼭 따라 치며 전체 기술과 백엔드 서비스 개발의 감을 익힙니다. (이해가 좀 부족해도 우선 진행하시길 권장합니다.)
2. **JPA 기본편** - JPA 기본 내용을 다집니다.

**3. 활용편 1 복습** - 이미 기본편을 학습한 상태이기 때문에 처음부터 다시 쭉 따라하다 보면 JPA의 깊이있는 내용을 이해할 수 있습니다.

**4. 활용편 2** - 실전에서 JPA를 개발하고, JPA 성능 최적화를 어떻게 하는지 이해합니다.

**5. 스프링 데이터 JPA** - 실무에서 필요한 기능 위주로 스프링 데이터 JPA를 학습합니다.

**6. Querydsl** - 복잡한 쿼리와 동적쿼리 문제를 해결합니다.

- 야생형 코스를 완료한 다음에 본인에게 부족한 스프링 기술을 더 깊이있게 학습해도 좋고, 또는 야생형 코스 중간에 필요한 스프링 기술을 학습해도 좋다.