

12. 다음으로

#1.인강/5. 스프링 MVC 2/강의#

학습 내용 정리

전체 목차

- 1. 타임리프 - 기본 기능
- 2. 타임리프 - 스프링 통합과 폼
- 3. 메시지, 국제화
- 4. 검증1 - Validation
- 5. 검증2 - Bean Validation
- 6. 로그인 처리1 - 쿠키, 세션
- 7. 로그인 처리2 - 필터, 인터셉터
- 8. 예외 처리와 오류 페이지
- 9. API 예외 처리
- 10. 스프링 타입 컨버터
- 11. 파일 업로드

1. 타임리프 - 기본 기능

- 1. 타임리프 - 기본 기능 - 프로젝트 생성
- 1. 타임리프 - 기본 기능 - 타임리프 소개
- 1. 타임리프 - 기본 기능 - 텍스트 - text, utext
- 1. 타임리프 - 기본 기능 - 변수 - SpringEL
- 1. 타임리프 - 기본 기능 - 기본 객체들
- 1. 타임리프 - 기본 기능 - 유틸리티 객체와 날짜
- 1. 타임리프 - 기본 기능 - URL 링크
- 1. 타임리프 - 기본 기능 - 리터럴
- 1. 타임리프 - 기본 기능 - 연산
- 1. 타임리프 - 기본 기능 - 속성 값 설정
- 1. 타임리프 - 기본 기능 - 반복
- 1. 타임리프 - 기본 기능 - 조건부 평가

- 1. 타임리프 - 기본 기능 - 주식
- 1. 타임리프 - 기본 기능 - 블록
- 1. 타임리프 - 기본 기능 - 자바스크립트 인라인
- 1. 타임리프 - 기본 기능 - 템플릿 조각
- 1. 타임리프 - 기본 기능 - 템플릿 레이아웃1
- 1. 타임리프 - 기본 기능 - 템플릿 레이아웃2
- 1. 타임리프 - 기본 기능 - 정리

2. 타임리프 - 스프링 통합과 폼

- 2. 타임리프 - 스프링 통합과 폼 - 프로젝트 설정
- 2. 타임리프 - 스프링 통합과 폼 - 타임리프 스프링 통합
- 2. 타임리프 - 스프링 통합과 폼 - 입력 폼 처리
- 2. 타임리프 - 스프링 통합과 폼 - 요구사항 추가
- 2. 타임리프 - 스프링 통합과 폼 - 체크 박스 - 단일1
- 2. 타임리프 - 스프링 통합과 폼 - 체크 박스 - 단일2
- 2. 타임리프 - 스프링 통합과 폼 - 체크 박스 - 멀티
- 2. 타임리프 - 스프링 통합과 폼 - 라디오 버튼
- 2. 타임리프 - 스프링 통합과 폼 - 셀렉트 박스
- 2. 타임리프 - 스프링 통합과 폼 - 정리

3. 메시지, 국제화

- 3. 메시지, 국제화 - 프로젝트 설정
- 3. 메시지, 국제화 - 메시지, 국제화 소개
- 3. 메시지, 국제화 - 스프링 메시지 소스 설정
- 3. 메시지, 국제화 - 스프링 메시지 소스 사용
- 3. 메시지, 국제화 - 웹 애플리케이션에 메시지 적용하기
- 3. 메시지, 국제화 - 웹 애플리케이션에 국제화 적용하기
- 3. 메시지, 국제화 - 정리

4. 검증1 - Validation

- 4. 검증1 - Validation - 검증 요구사항
- 4. 검증1 - Validation - 프로젝트 설정 V1
- 4. 검증1 - Validation - 검증 직접 처리 - 소개
- 4. 검증1 - Validation - 검증 직접 처리 - 개발

- 4. 검증1 - Validation - 프로젝트 준비 V2
- 4. 검증1 - Validation - BindingResult1
- 4. 검증1 - Validation - BindingResult2
- 4. 검증1 - Validation - FieldError, ObjectError
- 4. 검증1 - Validation - 오류 코드와 메시지 처리1
- 4. 검증1 - Validation - 오류 코드와 메시지 처리2
- 4. 검증1 - Validation - 오류 코드와 메시지 처리3
- 4. 검증1 - Validation - 오류 코드와 메시지 처리4
- 4. 검증1 - Validation - 오류 코드와 메시지 처리5
- 4. 검증1 - Validation - 오류 코드와 메시지 처리6
- 4. 검증1 - Validation - Validator 분리1
- 4. 검증1 - Validation - Validator 분리2
- 4. 검증1 - Validation - 정리

5. 검증2 - Bean Validation

- 5. 검증2 - Bean Validation - Bean Validation - 소개
- 5. 검증2 - Bean Validation - Bean Validation - 시작
- 5. 검증2 - Bean Validation - Bean Validation - 프로젝트 준비 V3
- 5. 검증2 - Bean Validation - Bean Validation - 스프링 적용
- 5. 검증2 - Bean Validation - Bean Validation - 에러 코드
- 5. 검증2 - Bean Validation - Bean Validation - 오브젝트 오류
- 5. 검증2 - Bean Validation - Bean Validation - 수정에 적용
- 5. 검증2 - Bean Validation - Bean Validation - 한계
- 5. 검증2 - Bean Validation - Bean Validation - groups
- 5. 검증2 - Bean Validation - Form 전송 객체 분리 - 프로젝트 준비 V4
- 5. 검증2 - Bean Validation - Form 전송 객체 분리 - 소개
- 5. 검증2 - Bean Validation - Form 전송 객체 분리 - 개발
- 5. 검증2 - Bean Validation - Bean Validation - HTTP 메시지 컨버터
- 5. 검증2 - Bean Validation - 정리

6. 로그인 처리1 - 쿠키, 세션

- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 요구사항
- 6. 로그인 처리1 - 쿠키, 세션 - 프로젝트 생성
- 6. 로그인 처리1 - 쿠키, 세션 - 홈 화면

- 6. 로그인 처리1 - 쿠키, 세션 - 회원 가입
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 기능
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 쿠키 사용
- 6. 로그인 처리1 - 쿠키, 세션 - 쿠키와 보안 문제
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 세션 동작 방식
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 세션 직접 만들기
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 직접 만든 세션 적용
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 서블릿 HTTP 세션1
- 6. 로그인 처리1 - 쿠키, 세션 - 로그인 처리하기 - 서블릿 HTTP 세션2
- 6. 로그인 처리1 - 쿠키, 세션 - 세션 정보와 타임아웃 설정
- 6. 로그인 처리1 - 쿠키, 세션 - 정리

7. 로그인 처리2 - 필터, 인터셉터

- 7. 로그인 처리2 - 필터, 인터셉터 - 서블릿 필터 - 소개
- 7. 로그인 처리2 - 필터, 인터셉터 - 서블릿 필터 - 요청 로그
- 7. 로그인 처리2 - 필터, 인터셉터 - 서블릿 필터 - 인증 체크
- 7. 로그인 처리2 - 필터, 인터셉터 - 스프링 인터셉터 - 소개
- 7. 로그인 처리2 - 필터, 인터셉터 - 스프링 인터셉터 - 요청 로그
- 7. 로그인 처리2 - 필터, 인터셉터 - 스프링 인터셉터 - 인증 체크
- 7. 로그인 처리2 - 필터, 인터셉터 - ArgumentResolver 활용
- 7. 로그인 처리2 - 필터, 인터셉터 - 정리

8. 예외 처리와 오류 페이지

- 8. 예외 처리와 오류 페이지 - 프로젝트 생성
- 8. 예외 처리와 오류 페이지 - 서블릿 예외 처리 - 시작
- 8. 예외 처리와 오류 페이지 - 서블릿 예외 처리 - 오류 화면 제공
- 8. 예외 처리와 오류 페이지 - 서블릿 예외 처리 - 오류 페이지 작동 원리
- 8. 예외 처리와 오류 페이지 - 서블릿 예외 처리 - 필터
- 8. 예외 처리와 오류 페이지 - 서블릿 예외 처리 - 인터셉터
- 8. 예외 처리와 오류 페이지 - 스프링 부트 - 오류 페이지1
- 8. 예외 처리와 오류 페이지 - 스프링 부트 - 오류 페이지2
- 8. 예외 처리와 오류 페이지 - 정리

9. API 예외 처리

- 9. API 예외 처리 - API 예외 처리 - 시작
- 9. API 예외 처리 - API 예외 처리 - 스프링 부트 기본 오류 처리
- 9. API 예외 처리 - API 예외 처리 - HandlerExceptionResolver 시작
- 9. API 예외 처리 - API 예외 처리 - HandlerExceptionResolver 활용
- 9. API 예외 처리 - API 예외 처리 - 스프링이 제공하는 ExceptionResolver1
- 9. API 예외 처리 - API 예외 처리 - 스프링이 제공하는 ExceptionResolver2
- 9. API 예외 처리 - API 예외 처리 - @ExceptionHandler
- 9. API 예외 처리 - API 예외 처리 - @ControllerAdvice
- 9. API 예외 처리 - 정리

10. 스프링 타입 컨버터

- 10. 스프링 타입 컨버터 - 프로젝트 생성
- 10. 스프링 타입 컨버터 - 스프링 타입 컨버터 소개
- 10. 스프링 타입 컨버터 - 타입 컨버터 - Converter
- 10. 스프링 타입 컨버터 - 컨버전 서비스 - ConversionService
- 10. 스프링 타입 컨버터 - 스프링에 Converter 적용하기
- 10. 스프링 타입 컨버터 - 뷰 템플릿에 컨버터 적용하기
- 10. 스프링 타입 컨버터 - 포맷터 - Formatter
- 10. 스프링 타입 컨버터 - 포맷터를 지원하는 컨버전 서비스
- 10. 스프링 타입 컨버터 - 포맷터 적용하기
- 10. 스프링 타입 컨버터 - 스프링이 제공하는 기본 포맷터
- 10. 스프링 타입 컨버터 - 정리

11. 파일 업로드

- 11. 파일 업로드 - 파일 업로드 소개
- 11. 파일 업로드 - 프로젝트 생성
- 11. 파일 업로드 - 서블릿과 파일 업로드1
- 11. 파일 업로드 - 서블릿과 파일 업로드2
- 11. 파일 업로드 - 스프링과 파일 업로드
- 11. 파일 업로드 - 예제로 구현하는 파일 업로드, 다운로드
- 11. 파일 업로드 - 정리

다음으로

스프링 데이터(DB) 접근 기술 안내

예제를 통해 단계적으로 기능을 발전시키며, 각 기능을 코드로 개발하면서 자연스럽게 학습

- 데이터베이스 커넥션
 - JDBC 커넥션
 - 커넥션 풀
- 스프링 트랜잭션
 - 트랜잭션 내부 원리
 - 자세한 사용법
 - 실무 주의 사항
- 다양한 데이터 접근 기술 이해
 - JDBC
 - JdbcTemplate
 - 마이바티스 - MyBatis
 - JPA
 - 스프링 데이터 JPA
 - QueryDSL
- 데이터베이스 예외 처리
 - 스프링 데이터베이스 예외 추상화
 - 올바른 예외 처리 사용법

로드맵 소개

스프링 완전 정복 시리즈(진행중)

스프링을 완전히 마스터 할 수 있는 로드맵

URL: <https://www.infllearn.com/roadmaps/373>

강의 목록

- 스프링 입문 - 코드로 배우는 스프링 부트, 웹 MVC, DB 접근 기술

- 스프링 핵심 원리 - 기본편
- 모든 개발자를 위한 HTTP 웹 기본 지식
- 스프링 웹 MVC 1편
- 스프링 웹 MVC 2편
- 스프링 데이터(DB) 접근 기술 (예정)
- 스프링 부트 (예정)

실전 REST API 설계, 개발 강의 요청이 많아서, 고민중 → 스프링 시리즈를 일단 마무리 짓고 진행

스프링 부트와 JPA 실무 완전 정복 로드맵

최신 실무 기술로 웹 애플리케이션을 만들어보면서 학습하고 싶으면 **스프링 부트와 JPA 실무 완전 정복 로드맵**을 추천

URL: <https://www.inflern.com/roadmaps/149>

강의 목록

- 자바 ORM 표준 JPA 프로그래밍
- 실전! 스프링 부트와 JPA 활용1 - 웹 애플리케이션 개발
- 실전! 스프링 부트와 JPA 활용2 - API 개발과 성능 최적화
- 실전! 스프링 데이터 JPA
- 실전! Querydsl

Q: 전체 코스 순서는 어떻게 듣는게 좋나요?

목표: 최신 기술의 자바 백엔드 개발자

- **스프링 완전 정복 로드맵**
 - 스프링 입문 - 코드로 배우는 스프링 부트, 웹 MVC, DB 접근 기술
 - 스프링 핵심 원리 - 기본편
 - 모든 개발자를 위한 HTTP 웹 기본 지식
 - 스프링 웹 MVC 1편
 - 스프링 웹 MVC 2편
- **스프링 부트와 JPA 실무 완전 정복 로드맵**

- **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

추천 학습 방법

스프링 입문과 스프링 핵심 원리를 듣고 나면 스프링으로 개발하는 가장 중요한 기본 지식을 쌓은 상태가 됩니다. 그래서 바로 **스프링 부트와 JPA 실무 완전 정복 로드맵**에 들어가는 것도 좋은 선택입니다. 스프링을 실무에서 어떤 식으로 활용해서 개발하는지 먼저 배워두고, 이후에 **스프링 완전정복 로드맵**를 통해 스프링 MVC나 스프링 데이터 접근 기술 같은 부분은 더 깊이있게 학습하시면 됩니다.

문제 해결 방안과 야생형

야생형 코스가 맞는 사람도 있고, 학자형 코스가 맞는 사람도 있음