

# 교육 과정 소개서.

---

10개 프로젝트로 시작하는 백엔드 웹개발 : 신입 개발자 취업  
집중반 (w. Java/Spring Boot)



## 강의정보

강의장	온라인 강의   데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	<a href="https://fastcampus.co.kr/dev_online_javaspring">https://fastcampus.co.kr/dev_online_javaspring</a>
강의시간	122시간 51분
문의	<a href="#">고객센터</a>

## 강의특징

나만의 속도로	낮이나 새벽이나 <b>내가 원하는 시간대</b> 에 나의 스케줄대로 수강
------------	---

원하는 곳 어디서나	시간을 쪼개 먼 거리를 오가며 오프라인 강의장을 찾을 필요 없이 <b>어디서든 수강</b>
---------------	---

무제한 복습	무엇이든 반복적으로 학습해야 내것이 되기에 이해가 안가는 구간 <b>몇번이고 재생</b>
-----------	--



## 강의목표

- 백엔드 개발의 기초인 Git & Github 과 Java 복습부터 SpringBoot 핵심 개념과 기능, 문제 해결 능력까지 모두 키울 수 있는 프로젝트로만 준비했습니다.
- 단순히 코드를 따라 치는 것이 아닌 실무 프로젝트를 완성하는 방식으로 한 단계씩 쌓아가는 과정을 함께 따라가다보면 채용에서 가장 중요한 역량인 문제 해결 역량을 갖춘 개발자로 레벨업 할 수 있어요!
- 프론트엔드 개발의 핵심 4가지- 성능최적화,재사용성(생산성), 유지보수 & 풍부한 기능 구현까지 한 번에! 대규모 프로젝트 구현으로 학습하여 바로 실무에 적용하세요.

## 강의요약

- 어디서부터 시작해야 할지, 다음에 어떤 것을 공부해야 할지 막막하다면, 4명의 현직 강사님들과 10개 프로젝트로 한 단계 나아가 보세요.
- 테스트에 대한 개념과 활용 범위를 점진적으로 넓혀가는 과정을 진행하며 프로젝트를 보다 제대로 이해하고 만들 수 있게 됩니다.
- 백엔드 개발자가 꼭 알아야 하는 핵심적인 내용을 프로젝트에 적용해보며 빠르게 학습할 수 있습니다.
- 프로젝트를 통해 성능 향상을 위한 기술 도입과 그 장단점에 대해 충분히 고려해보는 경험을 만들어 드립니다.



## 강사

김도현

과목

- Part 1-4

약력

- 현) 카카오뱅크 백엔드 개발자
- 전) 트렌비
- 전) 쿠팡
- 백엔드 시스템 개발 강의 다수 진행 (코멘토)
- 대학생 대상 개발자 취업 관련 특강 진행 (광운대학교)

개발자 J

과목

- 똑똑똑똑 만들어보는 Thread 스타일 SNS

약력

- Part 5-7
- 비전공자 대상 온/오프라인 강의 다수 진행 (수강자 100여명 이상)



## 강사

김은호

과목

- Part 8

약력

- 현) 스타트업 기업
- 전) 카카오페이
- 전) 카카오커머스
- 전) NHN
- 한 번에 끝내는 Spring 완.전.판 초격차 패키지 Online.
- 10개 프로젝트로 완성하는 백엔드 웹개발 (Java/Spring)
- 일할맛 2기 멘토

최병현

과목

- Part 9

약력

- 현) 뽀빠이
- 전) 컬리페이
- 전) 마켓보로
- 국비지원 멘토링 - 멀티캠퍼스 백엔드 개발자 취업캠프
- 프로그래밍 강의 강사 - 프로그래머스, 코멘토
- 부트캠프 코드 리뷰어 - 코드스쿼드, 코드스테이츠
- 인프런 멘토링



CURRICULUM

01.

Git을 활용한  
형상 관리

파트별 수강시간 05:35:35

<b>CH01. 개발 환경 설정하기</b>
01. iTerm2 설정
02. Java 와 IntelliJ 설정
03. H2 데이터베이스 설정
<b>CH02. Git 시작하기</b>
01. Git 의 역사와 버전 관리란?
02. Git 기초
03. Git 설치와 설정
04. GitHub 소개 및 활용법
<b>CH03. Git 기초 따라하기</b>
01. 저장
02. 조회
03. 되돌리기
04. 리모트 저장소
<b>CH04. Git 브랜치 이해하기</b>
01. 브랜치 기초와 관리
02. 머지
03. 컨플릭트 해결
04. 리모트 브랜치
05. 리베이스
<b>CH05. 좀 더 효과적으로 Git 활용하기</b>
01. 리비전 조회
02. 스테시와 클리닝
03. 커밋 히스토리
04. 리셋
<b>CH06. 협업하기</b>
01. 커밋 가이드라인
02. Git Flow 전략
<b>CH07. 마크다운 문법</b>
01. 마크다운 문법 배워보기
02. 마크다운 활용하기
<b>CH08. 이력서 만들기</b>
01. 이력서에 들어가야 할 내용 정리하기
02. 이력서 템플릿 소개 및 Fork
03. <a href="https://github.io">Github.io</a> 이력서 제작 및 Push



CURRICULUM

02.

**일정 관리  
프로그램**

파트별 수강시간 08:42:26

<b>CH01. Java 기초</b>
01. 자바 둘러보기
02. 자바 코드 구조 살펴보기
03. 자바 변수와 자료형
04. 네이밍 컨벤션
05. 주석 달기 팁
<b>CH02. Java 의 제어문</b>
01. If 문
02. Switch / Case 문
03. While / Do While 문
04. For / For Each 문
05. [실습] 별 만들기
<b>CH03. 객체 지향 프로그래밍 (OOP)</b>
01. 객체 지향 프로그래밍이란?
02. 클래스
03. 메소드
04. 인터페이스
05. 상속
06. 다형성
07. 추상 클래스
08. 컬렉션 (Collection)
09. 제네릭 (Generic)
10. SOLID 원칙 (1)
11. SOLID 원칙 (2)
<b>CH04. JUnit5 와 Spock 을 활용하여 테스트 코드 작성하기</b>
01. 단위 테스트란?
02. 단위 테스트의 구조
03. 좋은 단위 테스트의 특징
04. 단위 테스트 프레임워크 알아보기 JUnit5
05. 단위 테스트 프레임워크 알아보기 - Spock
06. [실습] JUnit5 와 Spock 으로 테스트 코드 작성하기
<b>CH05. 일정 관리 프로그램 만들기</b>
01. 요구사항 파악하기
02. 시스템 디자인 및 설계
03. 구현
04. 단위 테스트 작성하기
05. 기능 테스트



CURRICULUM

03.

온라인 커피 주문 서비스 만들기

파트별 수강시간 08:38:10

<b>CH01. 스프링 프레임워크</b>
01. 스프링 프레임워크 개요와 역사
02. 스프링 관련 프로젝트 소개
03. 의존성 주입 (DI)
04. [실습] DI
05. AOP 개념 소개
06. [실습] AOP
07. 프로퍼티 관리
08. [실습] 프로퍼티 관리
<b>CH02. JDBC</b>
01. 데이터 소스
02. 스프링과 JDBC
03. [실습] JDBC 사용해보기
04. 쿼리 결과를 Java 객체로 변환하기
05. [실습] 쿼리 결과를 Java 객체로 변환하기
06. 데이터 일괄 처리
07. [실습] 데이터 일괄 처리
<b>CH03. 트랜잭션</b>
01. 트랜잭션이란?
02. 트랜잭션 매니저
03. 선언적 트랜잭션과 명시적 트랜잭션
04. [실습] 트랜잭션
<b>CH04. 웹 어플리케이션 개발</b>
01. 웹 어플리케이션의 종류
02. 어플리케이션 설정
03. Handler 와 Controller
04. PathVariable, RequestParam
05. [실습] PathVariable, RequestParam
06. 입력값 검사
07. [실습] Spring Validation
08. 예외 처리 (ExceptionHandler)
09. [실습] 예외 처리
<b>CH05. [프로젝트] 온라인 커피 주문 시스템 만들기</b>
01. 요구사항 파악하기
02. 시스템 디자인 및 설계
03. 구현
04. 단위 테스트 작성하기
05. 기능 테스트



CURRICULUM

04.

가상화폐 데이터  
알림 서비스

파트별 수강시간 08:54:56

<b>CH01. RESTful 웹 서비스 개발</b>
01. REST API 아키텍처
02. 어플리케이션 설정
03. RestController 구현
04. [실습] RestController 와 RESTful API
05. 리소스 클래스
06. [실습] 리소스 클래스
07. 예외 처리
08. [실습] 예외 처리
<b>CH02. 외부 통신하기</b>
01. 외부 통신하는 방법 소개
02. RestTemplate 사용하기
03. [실습] RestTemplate
04. FeignClient 사용하기
05. [실습] FeignClient
06. 오류 처리 및 타임아웃 설정
07. [실습] 오류 처리 및 타임아웃 설정
<b>CH03. 스프링 환경에서의 단위 테스트</b>
01. 스프링 테스트란?
02. DI 컨테이너와 빈 테스트
03. 데이터베이스 테스트
04. MockMvc
05. [실습] 테스트 코드 작성
<b>CH04. 스프링과 MySQL</b>
01. MySQL 이란?
02. 스프링 + MySQL 설정
03. [실습] 스프링 MySQL 설정
04. 기본적인 CRUD 조작
05. [실습] 기본적인 CRUD 조작
06. SQL
07. [실습] SQL

## CURRICULUM

## 04.

가상화폐 데이터  
알림 서비스

파트별 수강시간 08:54:56

## CH05. 스프링 JPA

01. JPA 란?
02. JPA를 이용하여 DB 접근
03. QueryDSL
04. [실습] JPA를 이용하여 DB 접근
05. 영속성 컨텍스트
06. 엔티티 매핑
07. 연관관계 매핑
08. [실습] 연관관계 매핑
09. N 1 이슈 소개 및 해결방안

## CH06. 스프링 배치

01. 일반적인 배치 시스템
02. 스프링 배치의 아키텍처
03. 스프링배치 활용하기
04. [실습] 스프링 배치 사용해보기 (1)
05. [실습] 스프링 배치 사용해보기 (2)

## CH07. 가상화폐 실시간 데이터를 슬랙으로 알람 보내기

01. GitHub Action 과 Slack 소개
02. [실습] GitHub Action 활용하기
03. [실습] Slack 연결하기
04. 요구사항 파악하기
05. 구현
06. 단위 테스트 작성하기
07. 기능 테스트



CURRICULUM

05.

Thread 스타일  
SNS

파트별 수강시간 12:43:31

<b>CH01. 프로젝트 개요 및 웹의 이해</b>
01. 강사 및 프로젝트 소개
02. WEB 이란 무엇일까?
03. Client & Server
04. HTTP 의 이해
05. 모던 웹 아키텍처(Modern WEB Architecture)의 이해
06. Spring Boot 프로젝트 세팅
<b>CH02. 일단 만들어보는 Spring Boot REST API</b>
01. REST API 란?
02. Postman 소개 및 환경설정
03. 게시물을 위한 CRUD API(이론)
04. 게시물을 위한 CRUD API(실습) - READ
05. 게시물을 위한 CRUD API(실습) - CREATE
06. 게시물을 위한 CRUD API(실습) - UPDATE
07. 게시물을 위한 CRUD API(실습) - DELETE
08. 데이터베이스 연동 - Docker 소개 및 설치
09. 데이터베이스 연동 - Postgresql 소개 및 설치
10. TablePlus 소개 및 설치
11. 데이터베이스 연동 - JPA(Hibernate) 소개 및 연동
12. 게시물 CRUD API 수정 - 게시물 Entity 개발
<b>CH03. React FE 프로젝트 연동하기</b>
01. Spring Security 설정
02. CSRF 소개 및 관련 설정
03. FE 프로젝트 세팅 및 게시물 REST API 연동
04. CORS 소개 및 관련 설정
05. 로깅 추가하기
06. 예외처리 추가하기

## CURRICULUM

## 05.

Thread 스타일  
SNS

파트별 수강시간 12:43:31

<b>CH04. JWT 인증 추가하기</b>
01. JWT 인증에 대한 이해
02. 유저 모델 개발
03. JWT 인증 관련 필터개발 및 설정 추가
04. 회원가입 API 개발
05. 유효성 검사 추가
06. 로그인 API 개발
<b>CH05. 게시물과 유저 연동하기</b>
01. 게시물 <=> 유저 관계 설정
02. 게시물 API 수정
03. 유저 API 개발
04. FE 프로젝트 세팅 및 REST API 연동
<b>CH06. 진짜 SNS처럼 만들기</b>
01. 댓글 기능 개발
02. FE 프로젝트 세팅 및 댓글 API 연동
03. 좋아요 기능 개발
04. FE 프로젝트 세팅 및 좋아요 API 연동
05. 팔로우 기능 개발
06. 팔로워 및 팔로잉 목록 조회 API 개발
07. FE 프로젝트 세팅 및 팔로우 API 연동
08. 추가 기능 개발 - 게시물 API 좋아요 상태 추가
09. 추가 기능 개발 - 유저 API 팔로우 상태 추가
10. 추가 기능 개발 - 잔여 기능 개발 및 FE 프로젝트 연동
<b>CH07. 마무리</b>
01. 개선할 부분 생각해보기
02. 마무리



CURRICULUM

06.

컨퍼런스 신청 서비스

파트별 수강시간 14:17:01

<b>CH01. 프로젝트 개요 및 현업 맛보기</b>
01. 강사 및 프로젝트 소개
02. 현업에서는 어떻게 개발할까
03. 요구사항 정의서 살펴보기
04. 기획과 개발의 괴리
05. 프로젝트 설계하기
06. API 명세 작성 및 공유하기
<b>CH02. 컨퍼런스 페이지를 위한 REST API 개발하기</b>
01. Spring Boot 프로젝트 세팅 및 데이터베이스 연동
02. 개발환경 분리하기
03. 유저 모델 개발
04. JWT 인증 - JWT 서비스 개발
05. JWT 인증 - JWT 인증 필터 개발
06. JWT 인증 - Spring Security 설정
07. 회원가입 및 로그인 API 개발
08. 유효성 검사 추가
09. 세션 발표자 CRUD API 개발
10. 사용자 역할 별 API 권한 설정
11. Faker 기반 테스트 데이터 세팅
12. 컨퍼런스 세션 CRUD API 개발
13. 컨퍼런스 등록 및 등록 상태 조회 API 개발
<b>CH03. React FE 프로젝트 연동하기</b>
01. React 프로젝트 세팅 및 컨퍼런스 서비스 REST API 연동

## CURRICULUM

## 06.

컨퍼런스 신청  
서비스

파트별 수강시간 14:17:01

<b>CH04. 온 세상이 API다</b>
01. 외부 API를 결들일 때의 장점
02. 외부 API에 의존할 때의 문제
03. 스타트업과 대기업의 차이
04. API 실습 및 API 연동 시 주의할 점
<b>CH05. Slack API 연동하기</b>
01. Slack 소개 및 Slack 환경 세팅
02. Slack API 실습 및 컨퍼런스 API 연동
<b>CH06. 더 나은 서비스를 향해</b>
01. 현업 개발자의 입장에서 상상해보기
02. 서버가 터지면 어떻게 될까?
03. 중요한 것은 꺾이지 않는 서버
04. Redis 소개 및 설치
05. Redis 기본 명령어 실습
06. Sprint Boot 서비스에 Redis 적용하기
07. 추가로 고민해보기
08. 마무리



CURRICULUM

07.

가상자산  
조회/환율 서비스

파트별 수강시간 08:24:17

<b>CH01. 프로젝트 개요</b>
01. 강사 소개 및 프로젝트 소개
<b>CH02. Docker 살펴보기</b>
01. 도커 소개
02. 도커가 없던 세상, 그리고 지금
03. 도커 기본 개념 학습
<b>CH03. Docker 실습하기</b>
01. 도커 설치
02. 도커 기본 명령어 실습
03. 도커 이미지 실습
<b>CH04. Spring Boot REST API 개발</b>
01. 프로젝트 소개
02. 가상자산 가격 조회 API 개발
03. 환율 조회 API 개발
04. 가상자산 대시보드 API 개발
05. 가상자산 대시보드 FE 연동
<b>CH05. Docker 이미지 만들기</b>
01. 리눅스 기초 명령어 학습
02. Spring Boot 어플리케이션 빌드
03. Spring Boot 어플리케이션 도커 이미지 빌드
04. 도커 이미지 빌드 개선
<b>CH06. Docker 이미지 간 연동하기</b>
01. 도커 네트워크
02. 도커 네트워크 실습
03. REST API 이미지 수정
04. 도커 이미지 연동
<b>CH07. Docker Compose 로 정리하기</b>
01. Docker Compose 소개
02. Docker Compose 구성하기
03. Docker Compose 기본 명령어 실습
<b>CH08. 마무리</b>
01. 추가로 고민해보기
02. 마무리



CURRICULUM

08.

한글 테스트  
데이터 자동  
생성기 만들기

파트별 수강시간 40:24:31

CH01. 이론 설명
01-1. 강사 소개, 강의 목표 잡기 (1)
01-2. 강사 소개, 강의 목표 잡기 (2)
02. mock data란
03. mock이란 무엇일까
04. 테스트 더블이란
05. 테스트를 해야 할까
06. 테스트의 종류 개요
07-1. 테스트의 종류 단위 테스트 (Unit Test) 1
07-2. 테스트의 종류 단위 테스트 (Unit Test) 2
08-1. 테스트의 종류 통합 테스트 (Integration Test) + 스프링 1
08-2. 테스트의 종류 통합 테스트 (Integration Test) + 스프링 2
09. 테스트의 종류, 기능 테스트 (Functional Test) + BDD
10. 테스트의 종류 E2E 테스트 (End-to-End Test) + 스프링
11. 테스트의 종류 어디까지 해야 할까
12. 테스트 자동화 CI
13. CI/CD 딱 정리하기
14-1. TDD란?1
14-2. TDD란?2
15. Red-Green-Refactor
16. TDD를 해야 할까?
17. 스프링 부트 최신 동향 돌아보기 스프링 부트 3
18. 스프링 부트 최신 동향 돌아보기 스프링 부트 3.1
19. 스프링 부트 최신 동향 돌아보기 스프링 부트 3.2, 그리고 미래
20. 스프링 부트 2 - 3.2 마이그레이션 1
20. 스프링 부트 2 - 3.2 마이그레이션 2
21. 다른 스프링 부트 3 마이그레이션 참고 사례 1
21. 다른 스프링 부트 3 마이그레이션 참고 사례 2
22. Java 17 무엇이 바뀌었을까 1
22. Java 17 무엇이 바뀌었을까 2
23. 스프링 시큐리티 6 무엇이 바뀌었을까
24. 스프링 Data JPA 3 무엇이 바뀌었을까



CURRICULUM

08.

한글 테스트  
데이터 자동  
생성기 만들기

파트별 수강시간 40:24:31

<b>CH02. 설계와 테스트</b>
01. 우리는 무엇을 만들 것인가 Mockaroo 둘러보기
02. 프로젝트 시작 깃헙 세팅하기 1
03. 프로젝트 시작 깃헙 세팅하기 2
04. 필요한 기능 정리하기
05. 스프링 부트 프로젝트 초기화하기 1
05. 스프링 부트 프로젝트 초기화하기 2
05. 스프링 부트 프로젝트 초기화하기 3
06. 도메인 설계 1_기본 회원 정보와 개인 스키마 저장 테이블 1
06. 도메인 설계 1_기본 회원 정보와 개인 스키마 저장 테이블 2
07. 도메인 설계 2_mock 데이터의 저장 테이블 1
07. 도메인 설계 2_mock 데이터의 저장 테이블 2
07. 도메인 설계 2_mock 데이터의 저장 테이블 3
08. 데이터베이스_DB 서버 준비 1
08. 데이터베이스_DB 서버 준비 2
09. 데이터베이스_DB 접근 세팅
10. 데이터베이스_DB를 표현할 entity 코드의 작성 1
10. 데이터베이스_DB를 표현할 entity 코드의 작성 2
10. 데이터베이스_DB를 표현할 entity 코드의 작성 3
11. 데이터베이스_DB 접근용 repository 코드의 작성
12. 데이터베이스_repository 코드의 테스트 1
12. 데이터베이스_repository 코드의 테스트 2
12. 데이터베이스_repository 코드의 테스트 3
12. 데이터베이스_repository 코드의 테스트 4
13. 필요 기능을 제공할 api의 테스트 작성하기 1
13. 필요 기능을 제공할 api의 테스트 작성하기 2
13. 필요 기능을 제공할 api의 테스트 작성하기 3
13. 필요 기능을 제공할 api의 테스트 작성하기 4
14. 필요 기능을 제공할 api 와 dto 만들기 1-1
14. 필요 기능을 제공할 api 와 dto 만들기 1-2
15. 필요 기능을 제공할 api 와 dto 만들기 2



CURRICULUM

08.

한글 테스트  
데이터 자동  
생성기 만들기

파트별 수강시간 40:24:31

CH03. 구현과 배포
01. 뷰 만들기- 메인 페이지 1
01. 뷰 만들기- 메인 페이지 2
01. 뷰 만들기- 메인 페이지 3
01. 뷰 만들기- 메인 페이지 4
02. 뷰 만들기 내 스키마 페이지 1
02. 뷰 만들기 내 스키마 페이지 2
03. 뷰 만들기 자료형 선택 화면
04. 인증 기능- OAuth 인증이란 무엇일까
05. 인증 기능_github OAuth 인증 구현
06. 인증 기능_인증 기능의 컨트롤러 테스트
07. 인증 기능_개인 페이지는 인증을 거치도록 만들기
08. 로직 구현_개인 스키마 정보 읽어들이기 1
08. 로직 구현_개인 스키마 정보 읽어들이기 2
09. 로직 구현_개인 스키마 정보 저장하기
10. 로직 구현_개인 스키마 정보 수정하기
11. 로직 구현 개인 스키마 정보 삭제하기
12. 로직 구현 스키마 데이터를 원하는 유형으로 변환하는 기능 1
12. 로직 구현 스키마 데이터를 원하는 유형으로 변환하는 기능 2
12. 로직 구현_스키마 데이터를 원하는 유형으로 변환하는 기능 3
12. 로직 구현_스키마 데이터를 원하는 유형으로 변환하는 기능 4
13. 로직 구현_개인 스키마 정보를 원하는 유형으로 출력하기 1
13. 로직 구현_개인 스키마 정보를 원하는 유형으로 출력하기 2
14. 서비스 개선_ 메인 페이지 스키마 컬럼 순서를 드래그 앤 드롭으로 바꾸기
15. 서비스 개선_ 대량의 스키마 정보를 출력하려면 어떤 것을 고민해야 할까_
16. 배포_ Github Action 소개
17. 배포_ Github Action과 CI_CD
18. 배포_ 헤로쿠 소개
19. 배포_ 헤로쿠 연동
20. 마무리_ 배포, 최종 시연



CURRICULUM

09.

상품권 발권  
시스템 만들기

파트별 수강시간 06:09:36

<b>CH01. 강의 개요</b>
01. 강의 개요
02. 강의 목표
03. 다루는 기술에 대해
<b>CH02. 프로젝트 생성</b>
01. 개발 환경 세팅
02. Spring Boot 프로젝트 생성 & Github 연결
03. Employee API 구현
04. Employee 영속화 구현 1
05. Employee 영속화 구현 2
06. 리팩토링
07. 정리 & 실무라면
<b>CH03. 기본 구현 1차</b>
01. 업무 요건 이해 & 분석
02. 라이프사이클 구현
03. 리팩토링
04. 정리 & 실무라면
<b>CH04. 기본 구현 1차</b>
01. 업무 요건 이해 & 분석
02. API 구현
03. API 요청자에 대한 고민
04. API 에 요청자 정보 구현
05. 리팩토링
06. 정리 & 실무라면

## CURRICULUM

## 09.

# 상품권 발권 시스템 만들기

파트별 수강시간 06:09:36

<b>CH05. 이력 관리</b>
01. 업무 요건 이해 & 분석
02. 이력 관리 구현
03. 표준 예외 처리
04. 리팩토링
05. 정리 & 실무라면
<b>CH06. 계약 적용</b>
01. 업무 요건 이해 & 분석
02. 계약 구현 & 상품권 발행에 적용
03. 계약 데이터 생성 & API 구현
04. 리팩토링
05. 정리 & 실무라면
<b>CH07. 검증 로직 추가</b>
01. 업무 요건 이해 & 분석
02. 계약 유효기간 검증 로직 구현
03. 상품권 사용 불가 검증 로직 구현
04. 리팩토링
05. 정리 & 실무라면
<b>CH08. 마치며</b>
01. 구현 기능 정리
02. 그외 고려할 점

---

CURRICULUM

10.

Part 1. 강의 소개

파트별 수강시간 00:03:08

---

CH01. 강의 소개
-------------

01. 강의 소개
-----------



---

CURRICULUM

# 10.

## Part 2. Web Socket에 대해 이해하기

파트별 수강시간 00:21:50

---

CH01. 웹소켓 이론
01. 웹소켓이란?
02. 웹소켓과 스프링부트

## CURRICULUM

## 10.

Part 3.  
채팅서비스  
구현하기

파트별 수강시간 05:31:55

<b>CH01. 간단한 채팅 서비스 구현하기</b>
01. 프로젝트 생성하기
02. 웹 소켓을 사용하여 구현하기
03. STOMP를 사용하여 구현하기
<b>CH02. 로그인 기능 적용하기</b>
01. 로그인 기능 적용하기
<b>CH03. 로그인 기능 고도화하기</b>
01. 로그인 기능 고도화하기
<b>CH04. 회원가입 기능 구현하기</b>
01. docker와 스프링데이터 JPA에 대해 이해하기
02. docker 및 mysql, 스프링데이터 JPA 적용하기
03. 회원정보를 가져오기 위한 카카오 설정하기
04. 로그인 시 회원가입 기능 구현하기
<b>CH05. 채팅방 기능 구현하기</b>
01. JPA의 연관관계 이해하기
02. 채팅방 기능 구현하기1
03. 채팅방 기능 구현하기2
04. 채팅방 기능 구현하기3
<b>CH06. 과거메세지 표시하기</b>
01. 과거메세지 표시하기
<b>CH07. 신규메세지알림 기능 구현하기</b>
01. 신규메세지 알림기능 구현하기
<b>CH08. 상담사 기능 구현하기</b>
01. 상담사 기능 구현하기1
02. 상담사 기능 구현하기2



---

CURRICULUM

10.

Part 4. 서비스  
확장을 고려하기

파트별 수강시간 00:51:39

---

CH01. OAuth2 로그인 추가 연동하기
01. 구글로그인 연동하기
CH02. 서비스 확장 고려하기
01. 서비스 확장 고려하기 - 웹서버
02. 서비스 확장 고려하기 - 데이터베이스



---

CURRICULUM

CH01. 마무리
-----------

01. 마무리
---------

10.

## Part 5. 마무리

파트별 수강시간 00:02:29

---

---

CURRICULUM

**부록.**

**기술면접 TOP 50**

파트별 수강시간 01:36:10

---

CH01. 컴퓨터구조 & 운영체제
--------------------

01. 문제 1~10
-------------

02. 문제 11~20
--------------

CH02. 네트워크
------------

01. 문제 21~30
--------------

02. 문제 31~40
--------------

CH03. 데이터베이스
--------------

01. 문제 41~50
--------------

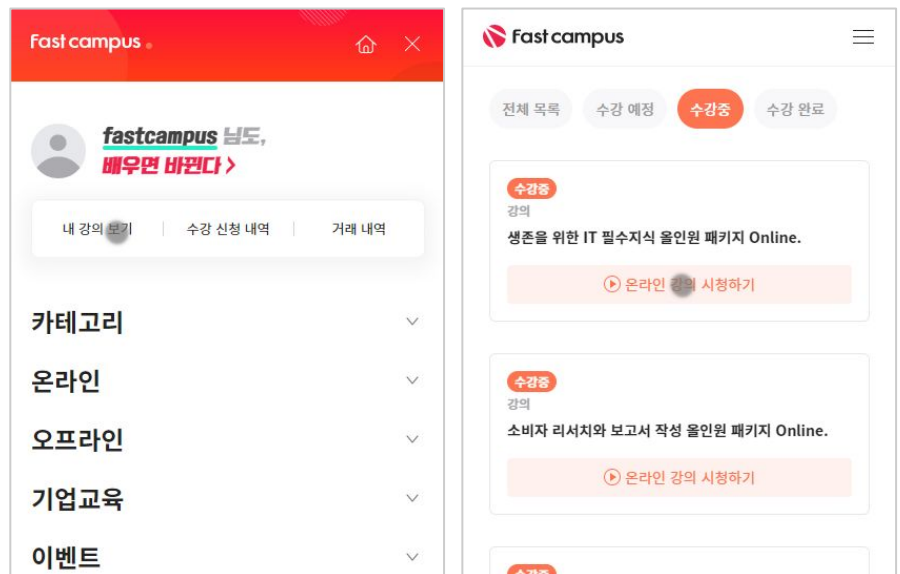


## 주의 사항

- 상황에 따라 사전 공지 없이 할인이 조기 마감되거나 연장될 수 있습니다.
- 패스트캠퍼스의 모든 온라인 강의는 아이디 공유를 금지하고 있으며 1개의 아이디로 여러 명이 수강하실 수 없습니다.
- 별도의 주의사항은 각 강의 상세페이지에서 확인하실 수 있습니다.

## 수강 방법

- 패스트캠퍼스는 크롬 브라우저에 최적화 되어있습니다.
- 사전 예약 판매 중인 강의의 경우 1차 공개일정에 맞춰 '온라인 강의 시청하기'가 활성화됩니다.



## 환불 규정

- 온라인 강의는 각 과정 별 '정상 수강기간(유료수강기간)'과 정상 수강기간 이후의 '복습 수강기간(무료수강기간)'으로 구성됩니다.
- 환불금액은 실제 결제금액을 기준으로 계산됩니다.

수강 시작 후 7일 이내	100% 환불 가능 (단, 수강하셨다면 수강 분량만큼 차감)
수강 시작 후 7일 경과	정상(유료) 수강기간 대비 잔여일에 대해 환불규정에 따라 환불 가능

※ 강의별 환불규정이 상이할 수 있으므로 각 강의 상세페이지를 확인해 주세요.