

# 교육 과정 소개서.

초격차 패키지 : 9개 도메인 프로젝트로 끝내는 백엔드 웹 개발 (Java/Spring)

## 안내.

해당 교육 과정 소개서는 모든 강의 영상이 촬영하기 전 작성되었습니다.

\* 커리큘럼은 촬영 및 편집을 거치며 일부 변경될 수 있으나, 전반적인 강의 내용에는 변동이 없습니다.

아래 각 오픈 일정에 따라 공개됩니다.

- 1차 : 2024년 08월 19일 (월)
- 2차 : 2024년 09월 19일 (목)
- 3차 : 2024년 10월 14일 (월)
- 4차 : 2024년 11월 18일 (월)
- 전체 : 2024년 12월 16일 (월)

최근 수정일자 2024년 07월 18일



## 강의정보

강의장	온라인 강의   데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	<a href="https://fastcampus.co.kr/dev_online_backend">https://fastcampus.co.kr/dev_online_backend</a>
강의시간	85시간 예정 (* 사전 판매 중인 강의는 시간이 변경될 수 있습니다.)
문의	<a href="#">고객센터</a>

## 강의특징

나만의 속도로	낮이나 새벽이나 내가 원하는 시간대에 나의 스케줄대로 수강
------------	-------------------------------------

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

무제한 복습	무엇이든 반복적으로 학습해야 내것이 되기에 이해가 안가는 구간 몇번이고 재생
-----------	---



## 강의목표

- 기본적인 Java/Spring부터 Kafka, Redis, Kubernetes까지 프로젝트에 따라 적합한 기술 스택 38가지를 다양하게 조합하여 사용합니다.
- 알림 생성 서버와 알림 조회 서버를 멀티 모듈 프로젝트로 분리하여, 알림 생성 서버의 부하가 알림 조회 서버에 영향을 미치지 않도록 설계합니다.
- DDD 방법론을 이용해서 실무에서는 어떤 방식으로 요구사항을 분석하고 도메인 모델링 설계하는지에 대한 방법과 도구에 대해서 학습합니다.
- 유지보수성을 위하여 문서화와 로깅을 중요시하고 하드웨어 이슈가 장애로 이어지지 않도록 하기 위해서 필수적인 클라우드 클러스터 배포를 진행합니다.

## 강의요약

- 바로 투입 가능한 경력직으로 거듭나는 실무 최적화된 9개 비즈니스적 + 기술적 도메인 특화 프로젝트
- 백엔드 개발자가 갖춰야 할 모든 스킬을 다각도로 향상시킬 122개 수준별 실습과 36가지 기술 스택
- Spring batch를 활용하여 일별 거래 데이터를 처리하는 시스템 (\_배치시스템) 개발
- 실시간 비동기 플랫폼 Kafka 활용하여 빌링/결제 시스템 개발



## 강사

이강호	과목	<ul style="list-style-type: none"> <li>- Spring JPA를 활용한 커뮤니티 피드 서비스 구현</li> <li>- Spring JPA와 Mysql을 활용한 커뮤니티 피드 서비스 고도화</li> </ul>
	약력	<ul style="list-style-type: none"> <li>- 현) 스마일게이트 게임 플랫폼 개발</li> <li>- 빌링 개발 - 유지보수, 운영, 배포 경험</li> <li>- 배치 시스템, 포인트 적립 시스템 경험</li> </ul>
홍민형	과목	<ul style="list-style-type: none"> <li>- Redis를 활용하여 서비스 속도를 개선하는 틱X/유X브 서비스 개발</li> </ul>
	약력	<ul style="list-style-type: none"> <li>- 현) 배달의민족</li> <li>- 전) SK planet, 네이버, 싸이월드</li> </ul>
김태엽	과목	<ul style="list-style-type: none"> <li>- Websocket을 활용한 실시간 메시징 도메인 서버 개발</li> </ul>
	약력	<ul style="list-style-type: none"> <li>- 현) 핀다 백엔드 Lead</li> <li>- 전) SK, 쿠팡, 야놀자</li> </ul>
이동준	과목	<ul style="list-style-type: none"> <li>- Spring Batch를 활용한 이커머스 플랫폼 거래 데이터 처리</li> </ul>
	약력	<ul style="list-style-type: none"> <li>- 현) 카카오</li> </ul>
유진호	과목	<ul style="list-style-type: none"> <li>- Github Action와 Jenkins를 활용한 PG사 CI/CD 자동화 시스템 개발</li> <li>- Kafka를 활용한 대량의 트랜잭션을 처리하는 이커머스 주문 및 결제 시스템 개발</li> </ul>
	약력	<ul style="list-style-type: none"> <li>- 현) NHN KCP</li> <li>- 전) Korbit, NHN 커머스</li> </ul>



## 강사

조민주

과목

- 대규모 트래픽을 고려한 확장성 있는 실시간 알림센터 시스템

약력

- 현) 카카오

문혜림

과목

- k8s를 활용한 클라우드 기반 인증 서비스 도메인

약력

- 현) 카카오
- 전) 아마존, SAP

배정호

과목

- 백엔드 개발자를 위한 데이터베이스

약력

- 현) SK

김남윤

과목

- 실무밀착 프로젝트:  
③유지보수하기 좋은 코드 디자인

약력

- 현)카카오계열사
- 전)스타트업기업



CURRICULUM

01.

Spring JPA를  
활용한 커뮤니티  
피드 서비스 구현

CH01. 강사 소개 및 강의 소개
01. 강사 소개 및 강의 소개
CH02. 강의 100% 습득 하기
01. 왜 우리는 이 강의를 듣고 있을까?
02. 챗터 구성 방식 및 100% 활용하기
CH03. 객체 지향 프로그래밍을 활용한 커뮤니티 피드 서비스 설계
01. 객체 지향의 사실과 오해
02. 객체 지향 설계 실습 - 수강생 관리 프로그램 과제 실습
03. 객체 지향 설계 실습 - 커뮤니티 피드 서비스 초안 설계 하기 - 유저
04. 객체 지향 설계 실습 - 커뮤니티 피드 서비스 초안 설계 하기 - 게시글 및 댓글
CH04. 클린 아키텍처를 활용한 커뮤니티 피드 서비스 설계
01. 도메인과 클린아키텍처
02. 클린아키텍처 설계 실습 - 커뮤니티 피드 서비스 아키텍처 설계 하기
CH05. JUnit5를 활용한 단위 테스트와 커뮤니티 피드 서비스 구현
01. 테스트 더블, 단위 테스트
02. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 테스트 작성 - 유저 도메인 테스트
03. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 구현 - 유저 서비스 구현
04. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 구현 - 유저 서비스 테스트
05. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 테스트 작성 - 게시글 및 댓글 도메인 테스트
06. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 구현 - 게시글 및 댓글 서비스 구현
07. 비즈니스 로직 구현 실습 - 커뮤니티 피드 서비스 구현 - 게시글 및 댓글 서비스 테스트

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

## 01.

## Spring JPA를 활용한 커뮤니티 피드 서비스 구현

### CH06. Spring JPA 를 활용한 커뮤니티 피드 서비스 구현

01. Spring 과 객체 지향
02. Spring 실습 - 커뮤니티 피드 서비스에 Spring 적용하기
03. JPA 기본 개념과 영속성 컨텍스트
04. 영속성 컨텍스트 특징
05. JPA 실습 - Mysql 연동 및 환경 구축하기
06. JPA 실습 - 객체 지향 엔티티 작성하기 - 유저
07. JPA 실습 - 객체 지향 엔티티 작성하기 - 게시물 및 댓글
08. JPA 실습 - Spring data jpa, JPQL, QueryDsl 이용한 쿼리 작성 연습 하기
09. JPA 실습 - 커뮤니티 피드 서비스 API 만들기 - 유저 관계 기능
10. JPA 실습 - 커뮤니티 피드 서비스 API 만들기 - 게시물 및 댓글 기능
11. JPA 실습 - 커뮤니티 피드 서비스 API 만들기 - 피드 조회 기능
12. JPA 트랜잭션과 db 커넥션 스레드 풀
13. JPA 실습 - 동시성 문제 해결, 좋아요를 동시에 눌렀을 때의 해결책
14. JPA 실습 - 데이터가 많아졌을 때의 성능 최적화

### CH07. 인수 테스트를 활용한 커뮤니티 피드 서비스 리팩토링

01. 확장성을 위한 시스템 재설계
02. 리팩토링 실습 - 유저수 증가에 따른 시스템 재설계
03. 리팩토링을 위한 인수테스트
04. 인수 테스트 실습 - 인수테스트 템플릿 작성
05. 인수 테스트 실습 - 커뮤니티 서비스에 인수 테스트 적용하기
06. 리팩토링 실습 - 커뮤니티 서비스에 변경된 설계 적용하기

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

## 02.

## Spring JPA와 Mysql을 활용한 커뮤니티 피드 서비스 고도화

<b>CH01. 커뮤니티 피드 서비스 추가 기획 내용</b>
01. 기능 개요
02. 인증 기능 개요
03. 어드민 기능 개요
<b>CH02. JWT를 활용한 회원 가입과 로그인 기능 구현</b>
01. 인증과 인가, OAuth 개념
02. 회원가입 테스트 구현하기
03. 회원가입 기능 구현하기
04. 로그인 테스트 구현하기
05. 로그인 기능 구현하기
06. 소셜 로그인 기능 구현하기
07. 권한 구별 기능 구현하기
<b>CH03. Mysql 쿼리 튜닝을 활용한 어드민 기능 구현</b>
01. 기획 확인 및 대시 보드 설계
02. admin 템플릿 적용 및 기본 설정
03. 대시 보드 기능 구현하기
04. 회원 관리 페이지 구현하기
05. 게시글 관리 페이지 구현하기
06. 댓글 관리 페이지 구현하기
<b>CH04. Mysql 쿼리 튜닝을 활용한 어드민 기능 구현</b>
01. 멱등성, 멱등키
02. 짧은 시간에 많은 요청 대응 방법 설계하기
03. Mysql를 활용한 멱등키 확인하고 응답하기
04. 알림 기능 개요
05. 알림 기능 환경 세팅
06. 알림 기능 테스트 추가
07. 알림 기능 구현하기

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.





CURRICULUM

03.

Redis를 활용하여  
서비스 속도를  
개선하는  
틱X/유X브  
서비스 개발

<b>CH01. 프로젝트 소개 및 요구 사항 분석과 개발 설정</b>
01. 강사 소개 및 수업 커리큘럼 설명
02. 개발 요구 사항 분석
03. 프로젝트 구조 설명
04. 필요한 기술 설명과 의존성 설정
05. Docker 기반의 로컬 mysql,redis,mongodb 실행 환경 설정
<b>CH02. 동영상 콘텐츠 개발</b>
01. 채널, 비디오 도메인 설계
02. 채널 기능 개발 : 생성, 수정
03. 채널 기능 개발 : 조회
04. 비디오 기능 개발 : 조회, 목록
05. Redis 캐시를 통한 조회 개선
<b>CH03. 사용자별 동작 개발</b>
01. 사용자 도메인 설계
02. 인증된 사용자 API 공통 요소 개발
03. 사용자 액션 개발 : 좋아요
04. 사용자 액션 개발 : 구독
<b>CH04. MongoDB를 이용한 댓글 개발</b>
01. 댓글 도메인 설계
02. 댓글 기능 개발 1 : 생성, 수정, 삭제
03. 댓글 기능 개발 2 : 대댓글
04. 댓글 기능 개발 3 : 대댓글로 인한 댓글 기능 변경
<b>CH05. 어드민 기능 개발</b>
01. 어드민 공통 요소 개발
02. 채널 관리
03. 비디오 관리
04. 댓글 관리

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

## 04.

## Websocket을 활용한 실시간 메시징 도메인 서버 개발

CH01. 웹소켓 이론
01. 강의 소개
02. 웹소켓이란?
03. 웹소켓과 스프링부트
CH02. 간단한 채팅 서비스 구현하기
01. 프로젝트 생성하기
02. 웹 소켓을 사용하여 구현하기
03. STOMP를 사용하여 구현하기
CH03. 로그인 기능 적용하기
01. 로그인 기능 적용하기
02. 로그인 기능 고도화하기
CH04. 회원가입 기능 구현하기
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
CH05. 채팅방 기능 구현하기
01. JPA의 연관관계 이해하기
02. 채팅방 기능 구현하기1
03. 채팅방 기능 구현하기2
04. 채팅방 기능 구현하기3
CH06. 과거메세지 표시하기
01. 과거메세지 표시하기
02. 신규메세지알림 기능 구현하기
03. 상담사 기능 구현하기
CH07. OAuth2 로그인 추가 연동하기 및 서비스 확장 고려하기
01. 구글로그인 추가 연동하기
02. 서비스 확장 고려하기 - 웹서버
03. 서비스 확장 고려하기 - 데이터베이스

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

# 05.

## Spring Batch를 활용한 전자상거래 플랫폼의 거래 데이터 처리

CH01. 개요
01. 감사 및 강의 소개
CH02. 도메인 소개
01. 일별 거래 데이터 처리
02. 전자상거래 플랫폼의 도메인 이해
CH03. Spring Batch의 이해
01. Spring Batch 소개
02. Spring Batch 아키텍처
CH04. 전자상거래 플랫폼의 거래 데이터 배치 처리 개발
01. 프로젝트 개요
02. 요구사항 분석
03. 아키텍처 설계
04. 개발환경 구성
05. 전자상거래 플랫폼의 주요 로직 구현
06. 일별 거래 데이터 처리 및 보고서 생성 배치 구현
07. 배치 작업 등록 및 실행
CH05. 강의 정리 및 회고
01. 강의 정리 및 회고

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.



CURRICULUM

06.

Github  
Action와  
Jenkins를 활용한  
PG사 CI/CD  
자동화 시스템  
개발

CH01. 과거 개발 세상 vs 현재 개발 세상
01. 강사 소개 및 수업 커리큘럼 설명
02. 폭포식 vs 애자일 개발방법론
03. CI/CD란 무엇이고 왜 필요한가?
04. 부록: DevOps란?
CH02. CI/CD를 위한 사전 지식 배우기
01. SW 형상관리의 중요성
02. Git; 기초 명령어
03. Git; 고급 명령어
04. Docker; 기초 학습
05. 부록; Git 브랜치 전략이란?
CH03. 개발 환경 구성
01. CI/CD를 구축을 위한 사전 준비
03. 온프레미스 기반 CI/CD 구성도 그려 보기
04. 클라우드 기반 CI/CD 구성도 그려 보기
05. 부록: 어떤 녀석(CI/CD)가 캡일까?
CH04. Github Action으로 CI/CD 구축하기 - part.1
01. Github Action의 구성 요소 배우기
02. Github Action의 흐름 배우기
03. Github Action으로 CI/CD 구축 실습1 - 구축
04. Github Action으로 CI/CD 구축실습2 - 배포 테스트

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

## 06.

# Github Action와 Jenkins를 활용한 PG사 CI/CD 자동화 시스템 개발

## CH05. Github Action으로 CI/CD 구축하기 part.2

- 01. Container 기반 CI/CD 구축 실습1 - 구축
- 02. Container 기반 CI/CD 구축 실습2 - 배포 테스트
- 03. 부록: 백엔드 개발자의 업무 롤은 어디까지?

## CH06. Jenkins으로 CI/CD 구축하기

- 01. Jenkins의 구성 요소 배우기
- 02. Jenkins 구축 실습1 - 초기 설정
- 03. Jenkins 구축 실습2 - 배포
- 04. 부록: Jenkins 파이프라인이란?

## CH07. Jenkins으로 CI/CD 구축하기

- 01. 클라우드 기반의 CI/CD 구축 사례 살펴보기
- 02. 실전 Github의 브랜치 전략
- 03. CI/CD의 자동화 테스트와 시큐어 코딩에 대해서...

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.



CURRICULUM

07.

**대규모 트래픽을 고려한 확장성 있는 실시간 알림센터 시스템**

<b>CH01. 강사 및 커리큘럼 소개</b>
01. 강사 소개 및 수업 커리큘럼 설명
<b>CH02. 알림센터 요구사항 분석 및 설계</b>
01. 알림센터 요구사항 분석
02. 대규모 트래픽을 고려한 알림센터 설계
03. 단일 모듈 멀티 프로젝트 소개
04. 단일 프로젝트 멀티 모듈 소개
05. 알림센터 API 설계
06. Springdoc 을 활용하여 API 문서 제공하기
<b>CH03. 알림센터 프로젝트 구성</b>
01. IntelliJ 개발 환경 설정
02. Docker 소개 및 설치
03. 프로젝트 구성
04. 멀티 모듈 프로젝트로 구성
05. 기본 스프링 의존성 설정
<b>CH04. 알림 도메인 생성 및 저장 구현</b>
01. Notification 클래스 추가
02. Docker를 이용한 MongoDB 설정
03. 테스트용 알림 데이터 저장
<b>CH05. Kafka Event Consumer 구현</b>
01. Docker를 이용한 Kafka 설정
02. Kafka Topic 생성
03. Kafka Event 전송 테스트
04. Spring Cloud Stream 소개
05. Kafka Consumer 구현

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

## 07.

## 대규모 트래픽을 고려한 확장성 있는 실시간 알림센터 시스템

### CH06. 알림 종류에 따른 알림 생성 구현

01. 좋아요 알림 생성 구현
02. 댓글 알림 생성 구현
03. 다형성을 활용한 리팩토링

### CH07. 알림 목록 조회 API 구현

01. 알림 목록 조회 API 초기 구현
02. 알림 목록 조회 API에 Paging 방식 적용
03. 알림 목록 조회 API에 Pivot 방식 적용
04. 알림 목록 조회 API에 SpringDoc 적용
05. Docker를 이용한 Redis 설정
06. 알림 읽음 처리 API 구현

### CH08. 신규 알림 여부 조회 API 구현

01. 새로운 알림 여부 조회 API 구현

### CH09. 알림센터 성능 최적화

01. MongoDB TTL 인덱스를 이용한 보관 기간 만료 알림 삭제
02. MongoDB 인덱스 설정
03. 캐싱 필요성 분석
04. Redis 기반 캐싱 구현
05. 팔로우 알림 / 번역

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.



## CURRICULUM

## 08.

## k8s를 활용한 클라우드 기반 인증 서비스 도메인

**CH01. 요구사항 분석 및 도메인 설명**

01. 프로젝트 소개
02. 인증 도메인 이해하기
03. 아이디어 도출과 요구사항 분석
04. 요구사항 명세서 작성하기

**CH02. 아키텍처와 DB 설계**

01. 아키텍처 설계
02. DB 설계

**CH03. 기본 기능: Spring, JPA, CRUD API**

01. Spring 기본 세팅
02. JPA를 활용한 기본 CRUD 구현 및 스웨서 문서화
03. Junit 유닛테스트 작성

**CH04. Spring security 와 App2app 인증**

01. Spring security 설정과 filter
02. app2app jwt token 생성
03. app2app token validation
04. app2app token 관리 고도화

**CH05. 기능 고도화 : threshold 와 redis cache**

01. threshold 기능 요구사항 정리
02. 레디스 설정하기
03. 설계변경 구현 반영하기

**CH06. 기능 고도화: k8s 배포**

01. k8s 도메인 이해하기
02. 클러스터 구축하기
03. 배포하기

**CH07. multithreading**

01. 멀티스레딩 이해하기
02. 멀티스레드 캐시 적용된 threshold 기능 적용하기

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.





CURRICULUM

09.

**Kafka를 활용한  
대량의  
트랜잭션을  
처리하는  
이커머스 주문 및  
결제 시스템 개발**

<b>CH01. 이커머스 기반 결제 및 빌링 시스템 요구 사항 분석 및 설계</b>
01. 강사 소개 및 수업 커리큘럼 설명
02. DDD 아키텍처; c c c
03. DDD 기반 이커머스 요구사항 분석; Event Storming
04. DDD 기술적 핵심 개념; DIP와 프로젝트 구조
05. Hexagonal 아키텍처; Domain, Application, Infrastructure
06. 부록: CQRS
<b>CH02. 이커머스 기반 결제 및 빌링 시스템 구조 설계 - 1</b>
01. 요구사항 분석 및 이벤트 스토밍; Entity와 Aggregate PG사 API 연동 준비
02. 주문 및 결제 API 설계 및 문서 자동화
03. DB ERD 설계; 주문 및 결제 서비스
04. 부록: RestDocs 설정 자세히 알아보기
<b>CH03. 개발 환경 구성</b>
01. IntelliJ 개발 환경 Setting 해보기
02. 프로젝트 구조 분석; Port and Adapter 패턴
03. Docker 기반의 MySQL, Kafka Cluster node 구성 가이드
04. 부록; 백엔드 개발자 도커 어디까지 알아야 하나?
<b>CH04. 이커머스 기반 결제 및 빌링 기능 개발 - 1</b>
01. 공통 Request, Validation 설계 및 개발
02. 공통 Response, Error Handling 설계 및 개발; AOP · 변경이 쉬운 외부 API 연동 설계 및 개발; Retrofit
03. TDD 기반 카드 승인 기능 개발; 주문
04. TDD 기반 카드 승인 기능 개발; 결제
05. TDD 기반 카드 취소 기능 개발; 취소
<b>CH05. 이커머스 기반 결제 및 빌링 기능 개발 - 2</b>
01. TDD 기반 정산 기능 개발
02. 정산 데이터가 많아지는 경우, 발생하는 문제와 해결방법 Produce와 Consumer 아키텍처 구조 알아보기
03. Kafka를 이용한 대용량 데이터 처리 개발 part1
04. Kafka를 이용한 대용량 데이터 처리 개발 part2
<b>CH06. 배포 및 성능 테스트</b>
01. 도커 이미지로 로컬 서버에 배포해보기
02. AWS EC2에 도커 이미지로 배포해보기
03. 부록; AWS Prettier 계정 만들어보기

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

# 부록.

## 백엔드 개발자를 위한 데이터베이스

CH01. 요구사항 분석 및 도메인 설명
01. 오리엔테이션
CH02. 데이터베이스 개요
01. 데이터베이스 이야기
02. RDBMS 개요
03. NoSQL 개요
04. DBMS 유형별 특징 및 선택기준
CH03. 데이터 모델링
01. 데이터 모델링 개요
02. RDBMS 데이터 모델링
03. NoSQL 데이터 모델링
04. 데이터 표준
CH04. 효과적인 SQL 활용
01. SQL 개요
02. Hard-parsing, Literal SQL
03. 테이블 접근 횟수 줄이기
04. 데이터 구조를 바꿔야하는 경우
05. 데이터 연결 방법 (JOIN)

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

## CURRICULUM

# 부록.

## 백엔드 개발자를 위한 데이터베이스

## CH05. 트랜잭션

01. 트랜잭션 개요
02. RDBMS 데이터 모델링
03. NoSQL 데이터 모델링
04. 데이터 표준

## CH06. 파티셔닝

01. 파티셔닝 개요
02. 파티셔닝 전략
03. Kafka 파티셔닝
04. NoSQL 샤딩 (MongoDB)

## CH07. 복제

01. 복제 개요
02. 단일 리더 복제 (MongoDB)
03. 멀티 리더 복제
04. 리더 없는 복제

## CH08. 분산 시스템 개요

01. 분산 시스템 개요
02. 분산 시스템 문제
03. 분산 시스템 아키텍처
04. 분산 시스템 트랜잭션 처리
05. 분산 시스템의 합의 알고리즘
06. 분산 원장(블록체인) DBMS 일까?
07. 분산 원장은 언제 사용해야 할까?

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.  
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

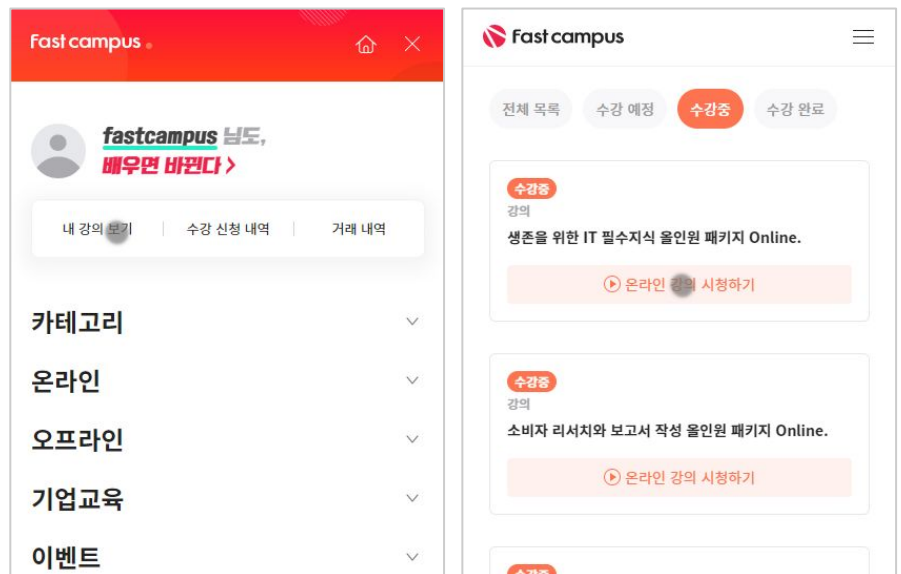


## 주의 사항

- 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.