교육
과정
소개서.

Spring Webflux 완전 정복 : 코루틴부터 리액티브 MSA 프로젝트까지

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 : 2023년 05월 04일**
- **2차 : 2023년 06월 02일**
- **3차 : 2023년 07월 03일**
- **최종 오픈 : 2023년 08월 07일**

최근 수정일자 2023년 6월 07일

Fast campus

강의정보

강의장	온라인 강의	데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	https://fastcampus.co.kr/dev_online_webflux
강의시간	52시간 예정 **(* 사전 판매 중인 강의는 시간이 변경될 수 있습니다.)**
문의	고객센터

강의특징

나만의 속도로	낮이나 새벽이나 **내가 원하는 시간대**에 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

- 체계적인 커리큘럼을 통해 Webflux를 통한 리액티브 웹 개발을 마스터 할 수 있습니다.
- 기본인 스프링과 비동기 코드의 가독성을 높여주는 코루틴을 한 강의에서 만날 수 있습니다.
- 기획부터 부하 테스트까지 실무 프로세스로 고성능 리액티브 MSA를 구축할 수 있습니다.

강의요약

- Webflux 완벽 마스터 탄탄한 커리큘럼
- Java와 Kotlin(Coroutine)기반으로 학습
- 리액티브 MSA 실무 프로젝트
- rR2DBC와 Spring Data로 고효율 데이터베이스 구축
- 유지보수 학습으로 Webflux 난점 완벽 커버
- IT 대기업 출신 강사님 질의응답 커뮤니티

강사

김태우

약력

- 현) 당근마켓
  로컬 커머스 아키텍처 설계 및 구축, Webflux 등 Reactive Stack을 활용하여 리소스 효율과 응답 속도 향상
- 전) 카카오
  Spring mvc, JPA 등 Servlet stack을 활용하여 챗봇 빌더 개발, 리팩토링과 클린 아키텍처 도입으로 구조적 문제 해결
- 전) 쿠팡
  Spring boot, memcashed, react, redux를 활용한 풀스택 개발

강사의 한 마디

안녕하세요. 수강생 여러분, 김태우입니다.
지역 기반의 커머스 서비스를 만들며, 설계부터 개발, 운영에 이르기까지 다양한 업무에 참여하고 있습니다.
제가 작업하던 프로젝트에 Webflux를 도입했더니 훨씬 적은 서버로 적은 리소스를 활용하여 서비스를 제공할 수 있었고, 요청량이 늘어나도 안정적으로 그리고 탄력적으로 요청을 처리하는 좋은 성능에 놀랐습니다.
Webflux에 대한 자료 부족과 비동기 개발의 복잡성 때문에 걱정이시라면 이 번 강의를 통해 새로운 패러다임을 도전해보시고,
저와 함께 Webflux를 통해서 서비스의 성능을 향상시키고 개발자로서 크게 성장한 경험을 얻어가세요!

CURRICULUM

01.

Reactive programming

Ch 01. 전체 커리큘럼과 강의진행 소개
01. 전체 커리큘럼과 강의 진행 소개
Ch 02. Reactive Programming
01. 챕터 소개
Ch 03. 비동기 Programming
01. 함수 관점에서 동기와 비동기의 차이
02. 함수 관점에서 blocking과 non-blocking의 차이
03. IO 관점에서 blocking과 non-blocking의 차이
Ch 04. CompletableFuture
01. Future 인터페이스
02. CompletionStage 인터페이스
03. CompletableFuture 클래스
04. CompletableFuture 사용해서 비동기 로직 처리하기
05. CompletableFuture 정리
Ch 05. Reactive Streams
01. Reactive streams의 역사
02. Reactive manifesto
03. Reactive programming
04. Reactive streams
05. Cold & Hot Publisher 구현
06. Reactive streams 정리
Ch 06. Reactive Streams 구현 라이브러리 소개
01. Reactor 소개, Publisher 중심
02. RxJava 소개, Publisher 중심
03. Mutiny 소개, Publisher 중심

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

CURRICULUM

01.

Reactive programming

Ch 07. Java NIO
01. Java IO - InputStream
02. Java IO - OutputStream
03. Java NIO - Buffer
04. Java NIO - DirectByteBuffer와 Channel
05. Java IO, NIO 사용해서 소켓 서버 구현
06. Java AIO(NIO2)
Ch 08. Reactor 패턴
01. Selector 소개
02. Selector 사용해서 소켓 서버 고도화
03. epoll 소개
04. Reactor pattern 소개
05. Reactor pattern 사용해서 http 서버 구현
06. Reactor pattern 정리
Ch 09. 부록 A. Proactor pattern

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

CURRICULUM

02.

Spring Webflux

Ch 01. Spring Reactive Stack
01. 챕터 소개
02. spring servlet stack
03. spring reactive stack
Ch 02. Netty
01. eventLoop
02. channelHandlerContext
03. channelHandler
04. netty로 Echo server 구현 예제
05. bootstrap
06. netty 서버 구현
07. netty 정리
Ch 03. Reactor
01. reactor 복습
02. subscribe
03. sequence 생성
04. scheduler와 쓰레드
05. 에러 핸들링
06. 결합
07. 유용한 연산자
08. context
09. reactor로 비동기 로직 처리_2
10. reactor 정리

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

CURRICULUM

02.

Spring Webflux

Ch 04. Spring Webflux
01. spring webflux 구조
02. httpHandler
03. webHandler
04. webHandler codec
05. webFilter
06. webExceptionHandler
05. Server Sent Event
Server sent event 개념과 동작 원리
Webflux에서 sse 구현하는 방법 알아보기
06. Spring webflux를 이용한 웹 어플리케이션 구현
Spring webflux rest api 구현하기
Spring webflux websocket 구현하기
Spring webflux server sent event 구현하기
Spring security로 보안 구현하기

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

CURRICULUM

03.

Spring Data Reactive

Mysql, MongoDB, Redis
· RDBMS와 NoSQL 비교
· Redis의 개념과 작동원리
· 각각 저장소가 적합한 상황 공유
R2dbc mysql
· R2dbc vs JPA
· R2dbc mysql 연산자 소개
· R2dbc mysql 심화
Reactive MongoDB
· Reactive mongoDB 개념 소개
· Reactive mongoDB 연산자 소개
· Reactive mongoDB 심화
Reactive Redis
· Reactive Redis 개념 소개
· Reactive Redis 연산자 소개
· Reactive Redis 심화
Spring data reactive를 이용한 웹 어플리케이션 구현
· R2dbc mysql로 repository와 entity 만들기
· Reactive mongodb로 repository와 document 만들기
· Reactive redis로 repository와 entity 만들기
· Repository들을 합쳐서 서비스 구성하기
Spring data reactive의 다른 라이브러리 소개
· Spring data neo4j reactive
· Spring data cassandra reactive
· Spring data elasticsearch reactive

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

CURRICULUM

04.

Kotlin Coroutine

Coroutine 기본
· Coroutine 개념 소개
· 동기, 비동기 코드들과 비교
· Coroutine 동작 원리
Structured Concurrency
· Strucuted Concurrency 소개
· job을 통한 exception과 cancel의 전파
CoroutineScope
· coroutineScope 소개
· coroutineDispatcher 소개
· coroutine 디버깅하기
Coroutine 예외 처리
· Coroutine 내부에서 발생하는 에러 처리
· CoroutineExceptionHandler를 사용한 에러 처리
Asynchronous Flow
· flow 개념 소개
· cold flow vs hot flow
· flow와 coroutineeScope
CoroutineContext
· CoroutineDispatcher에 대한 개념 소개
· CoroutineDispatcher 종류 소개
Spring webflux와 Kotlin coroutine를 이용한 웹 어플리케이션 구현
· Spring webflux와 coroutine을 이용해서 rest api 구현
· Spring mongoDB와 coroutineRepository로 db 접근
· Spring webflux와 flow를 이용해서 websocket 구현

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

CURRICULUM

05.

Spring Reactive Teste

Test 전략
· Unit test 소개
· Slice test 소개
· Integration test 소개
Junit과 reactor test
· Junit 소개
· Reactor test를 이용한 비동기 test
Mockito
· Mocktio 소개
· SpringBeans를 이용한 spring mocking
Unit test 작성하기
· Junit을 이용한 unit test 구현
· Reactor test로 비동기 test 구현
· Mockito를 이용해서 mocking 구현
Spring test
· Spring test를 지원하는 annotation 소개
· Spring test config 설정
WebTestClient
· WebTestClient로 test 구성
· MockWebServer
Slice test 작성하기
· WebFluxTest로 controller test
· DataMongoTest로 controller test
TestContainers
· TestContainers 소개
· TestContainer로 mysql test
· TestContainer로 redis test

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

CURRICULUM

05.

Spring Reactive Teste

Integration test 작성하기
· SpringBootTest로 test 세팅
· MockWebServer로 가상 api 환경 구축
· TestContainer로 저장소 환경 구축
Kotest
· Kotest 소개
· Kotest로 test 작성
Mockk
· Mockk 소개
· Mockk로 test 작성
K6와 Performance test
· performance test와 k6 소개
· k6로 test 작성

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

CURRICULUM

06.

Reactive Microservice

Reactive microservice 구성 소개
· Microservice 구성 소개
· Reactive 반영하기
Spring cloud gateway
· Spring cloud gateway 소개
· Cloud gateway에서 reactive programming 적용하기
Kafka와 Spring cloud stream
· Kafka 소개
· Spring cloud streaam 소개
· Spring cloud streaam kafka binder 소개
Spring cloud circuit breaker
· Circuit breaker 소개
· Spring cloud circuit breakeker 소개
Reactive microservice 만들기
· docker-compose를 이용한 환경 세팅
· Reactive application 서버 구현
· Corouine 기반의 reactive application 서버 구현
· Kafka와 Spring cloud function으로 큐 구현
· Spring cloud gateway로 api gateway 구현
· Sprigin cloud circuit breaker 적용
· 통합테스트 만들고 실행하기

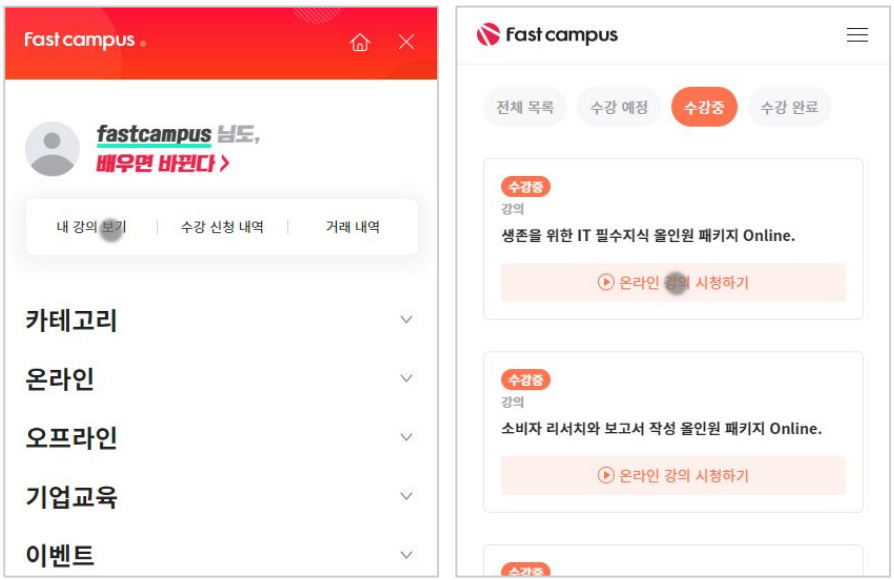
본 과정은 현재 촬영 및 편집이 진행되고 있는 사전 판매 중인 강의입니다.
해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

주의 사항

- 상황에 따라 사전 공지 없이 할인이 조기 마감되거나 연장될 수 있습니다.
- 패스트캠퍼스의 모든 온라인 강의는 **아이디 공유를 금지**하고 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.