교육
과정
소개서.

MSA 환경에서 REST API, gRPC, GraphQL로 마스터하는
백엔드 API 개발

Fast campus

강의정보

강의장	온라인 강의	데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	https://fastcampus.co.kr/dev_online_api3
강의시간	47시간 12분
문의	고객센터

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

Copyright ⓒ FAST CAMPUS Corp. All Rights Reserved.

강의목표

- 실무 MSA 환경에서 필수 사용되는 TOP 3 API를 한번에 배울 수 있습니다.
- 3개 API를 유기적으로 연계한 실전 MSA 환경 프로젝트 - TOP3 API들의 특장점을 활용해 고도화 된 기능을 개발하는 방법을 학습합니다.
- API Gateway 활용 트래픽 분산, 디버깅, 모니터링까지 학습합니다.
- MSA 환경에서 다양한 API 서비스 간 통신에 필수인 API Gateway 개념부터 응용까지 확실하게 배워봅니다.

강의요약

- MSA 환경에서 가장 널리 쓰이는 API인 REST, GraphQL, gRPC를 강의 하나로 모두 배웁니다
- MSA를 연결하는 API Gateway를 통해 다양한 실무 환경을 운영하는 방법을 직접 경험합니다.
- 얕게 배워 알고 있다고 착각하는 강의가 아니라, 실무 적용시 유의해야할 포인트까지 모두 해결한 진짜! 실무에 가까운 강의로 API 개발의 마스터가 되세요!
- 기술 스택별 기본 사용법이 아닌 실질적인 실무 적용 방법을 학습하세요!

Copyright ⓒ FAST CAMPUS Corp. All Rights Reserved.

강사

Doddy	**과목**	-	Part 1~5
	약력	-	헌) 대기업 Data Platform Engineer - 전) 대기업 Backend Developer
김태현	**과목**	-	Part 6
	약력	-	Amazon AWS - Senior SDE - Sendbird - Software Engineer - Bllizzard - Senior SWE II

Copyright ⓒ FAST CAMPUS Corp. All Rights Reserved.

CURRICULUM

01.

API 이론 : REST API

파트별 수강시간 04:31:18

CH01. REST API 개요
01. REST API 개념 소개
02. RESTful 아키텍처의 특징
03. REST API의 장점 및 활용 사례
CH02. HTTP 기본 개념
01. HTTP 프로토콜 소개
02. HTTP 요청과 응답 구조
03. HTTP 상태 코드
CH03. REST API 디자인
01. REST API 디자인 원칙과 URI 설계
02. HTTP 메서드를 활용한 리소스 조작
CH04. HTTP 메서드 활용
01. HTTP 메서드 설명
02. HTTP 메서드 활용 예제
03. HTTP 상태 코드와 오류 처리
CH05. REST API 보안
01. REST API 보안 이슈 소개
02. 인증과 권한 관리
03. API 보안 모범 사례
CH06. REST API 문서화
01. API 문서화의 중요성
02. Swagger와 같은 도구 소개
03. API 문서 작성 가이드라인
CH07. REST API 고급주제
01. 웹 소켓과 실시간 통신
02. GraphQL과 REST API 비교
03. 모바일 애플리케이션 및 웹 앱에 대한 최적화
04. API 테스팅 및 품질 보증
CH08. REST API 실습
01. Spring 기반 개발 환경 세팅
02. 기본 REST API 구현 실습
03. Swagger와 RestDocs를 사용한 API 문서화 실습
04. HATEOAS 구현 실습

CURRICULUM

02.

API 이론 : gRPC

파트별 수강시간 03:47:22

CH01. gRPC 개요
01. gRPC 소개 및 기본 개념 소개
CH02. Protocol Buffers (ProtoBuf) 기초
01. Protocol Buffers의 기본 개념과 구조 소개
02. 메시지 정의, 필드, Enum, 기본 데이터 타입 등
03. 실제 ProtoBuf 파일 작성 및 사용 예제
CH03. gRPC 기본 개념
01. RPC (Remote Procedure Call) 개념 소개
02. gRPC 서비스 요청 방식 이해
03. 클라이언트 및 서버에서 gRPC 사용 예제
CH04. gRPC 웹 및 REST 브리지
01. gRPC 웹 서비스 소개
02. RESTful API와 gRPC 간의 변환 방법 소개
CH05. gRPC 스트리밍
01. gRPC 스트리밍의 이해
02. gRPC 스트리밍을 사용하는 예제
CH06. gRPC 에러 처리와 상태 코드
01. gRPC 상태 코드 소개
02. gRPC 예외 처리 및 에러 핸들링
03. gRPC 상태 코드 사용 예제
CH07. gRPC 보안과 인증
01. gRPC 보안 계층 소개
02. TLS 기반의 보안 설정
03. 사용자 인증 및 권한 부여 방법
CH08. gRPC 서버와 클라이언트 구현
01. gRPC 서버 구현
02. gRPC 서버 Interceptor 구현
03. gRPC 클라이언트 구현
04. gRPC 서버-클라이언트 통신 및 에러 핸들링

CURRICULUM

03.

API 이론 : GraphQL

파트별 수강시간 03:32:52

CH01. GraphQL 개요
01. GraphQL 소개 및 배경 설명
02. GraphQL의 주요 특징과 장점 소개
CH02. GraphQL 스키마와 타입
01. GraphQL 스키마 정의 및 타입 시스템 소개
02. Scalar, Object, Enum, Interface, Union 타입 설명
03. 스키마 확장과 사용자 정의 스칼라 타입 소개
CH03. GraphQL 쿼리 언어
01. GraphQL 쿼리 작성 및 구문 설명
02. 다양한 GraphQL 쿼리 구성 요소
03. 내부 타입 및 리졸버(resolver) 개념 소개
CH04. GraphQL 쿼리와 뮤테이션
01. 쿼리와 뮤테이션의 차이 설명
02. 쿼리와 뮤테이션 작성 예제
03. 복잡한 쿼리와 중첩된 필드 예제 소개
CH05. GraphQL 보안 및 인증
01. GraphQL 보안 개요
02. 인증과 권한 부여(Authentication & Authorization) 방법 설명
03. 보안 관련 예제 및 팁
CH06. GraphQL 서버 개발 실습
01. GraphQL 서버 구축 및 설정
02. 스키마 정의 및 데이터베이스 연동
03. GraphQL Directive 구현
04. GraphQL Instrumentation 구현
05. GraphQL 문서화

CURRICULUM

04.

API Gateway

파트별 수강시간 06:10:01

CH01. API 게이트웨이 개념
01. API Gateway 개요 및 도입
02. API Gateway 역할과 특징
03. API Gateway 여러가지 사례
CH02. API 게이트웨이 프록시와 라우팅
01. 프록시 서버 개념과 역할
02. 라우팅의 중요성
CH03. 인증(Authentication) 및 인가(Authorization)
01. API Gateway에서 인증 및 인가 처리 방법
CH04. 트래픽 제어
01. 트래픽 제어의 중요성
02. 트래픽 제어 구현 사례
CH05. 로깅 및 모니터링
01. 로깅과 모니터링의 중요성
02. 분산 로깅 및 추적
03. 모니터링 도구 Prometheus와 Grafana 연동
CH06. BFF 패턴
01. BFF 패턴 소개 및 특징
02. BFF 패턴의 장점과 단점
CH07. 마이크로서비스 아키텍처와의 연계
01. 마이크로서비스 아키텍처 개요
02. API 게이트웨이와 마이크로서비스 아키텍처의 상호 작용
CH08. API 게이트웨이 서버 개발 실습
01. Spring Cloud Gateway 프로젝트 구축
02. Spring Cloud Gateway를 사용한 서비스 디스커버리 및 라우팅 구현
03. Spring Security를 활용한 인증 및 인가 구현
04. Spring Cloud Gateway를 사용한 Rate Limiting 구현
05. Spring Cloud Gateway를 활용한 BFF 분리
06. Zipkin을 사용한 분산 로깅 구현
07. Gateway와 Prometheus 연동
08. Grafana 연동 및 간단한 대시보드 구축

CURRICULUM

05.

Next Campus 프로젝트

파트별 수강시간 20:01:12

CH01. 프로젝트 소개 및 환경 설정
01. 프로젝트 소개 및 목표
02. 개발 환경 설정 및 필요 도구 설치
CH02. 프로젝트 요구 사항 분석 및 설계
01. 요구 사항 분석 및 기능 명세 작성
02. MSA 아키텍처 설계 및 컴포넌트 정의
03. 데이터베이스 스키마 설계
CH03. 프로젝트 요구 사항 분석 및 설계
01. Spring Cloud Eureka를 사용한 Service Discovery 구축
CH04. REST API 기반 마이크로서비스 구축
01. 강의 정보 서비스 컴포넌트 설계
02. 강의 정보 서비스 컴포넌트 개발
03. 강의 정보 서비스 컴포넌트 테스트
04. 인증 서비스 컴포넌트 설계
05. 인증 서비스 컴포넌트 개발
06. 인증 서비스 컴포넌트 테스트
07. 컨텐츠 파일 서비스 컴포넌트 설계
08. 컨텐츠 파일 서비스 컴포넌트 개발
09. 컨텐츠 파일 서비스 컴포넌트 테스트
CH05. gRPC 기반 마이크로서비스 구축
01. 컨텐츠 재생 정보 서비스 컴포넌트 설계
02. 컨텐츠 재생 정보 서비스 컴포넌트 개발
03. 컨텐츠 재생 정보 서비스 컴포넌트 테스트
04. 이용권 구매 서비스 컴포넌트 설계
CH06. gRPC 기반 마이크로서비스 구축 : 이용권 구매/권한 서버 만들기
01. GraphQL 스키마 설계
02. GraphQL 프로젝트 설정
03. GraphQL 데이터 Aggregation 및 리졸버 구현 1
04. GraphQL 데이터 Aggregation 및 리졸버 구현 2
05. GraphQL 서버와 gRPC 연동
06. GraphQL 서버와 REST API 연동
07. GraphQL 에러 핸들링과 보안 기능 추가
08. GraphQL 최적화 1 DataLoader 적용
09. GraphQL 최적화 2 쿼리 제한 적용
10. GraphQL 최적화 3 캐싱 적용

CURRICULUM

05.

Next Campus 프로젝트

파트별 수강시간 20:01:12

CH07. API Gateway 설계 및 서비스 구축
01. API Gateway 서버 설계
02. API Gateway 프로젝트 설정 및 라우팅 설정
03. API Gateway Filter 적용 1 JWT 인증 서버와 연동
04. API Gateway Filter 적용 2 Circuit Breaker, Rate Limiting 설정
05. API Gateway Exception Handler 구현
CH08. 로깅 및 모니터링 기능 추가
01. Zipkin, ElasticSearch를 사용한 분산 로깅 구현
02. Kibana 로 Zipkin Span 데이터 분석 및 모니터링
03. Spring Actuator와 Prometheus 연동
04. Grafana 대시보드 구축 및 사용
CH09. Docker 이미지 빌드 및 Docker Compose로 배포하기
01. 컴포넌트 별 Dockerfile 생성 및 배포 환경 구축
02. Docker-Compose로 컨테이너 배포
CH10. K8S Chart로 어플리케이션 배포하기
01. Kubernetes 소개 및 기본 개념
02. 클러스터 설정 및 어드민 앱 배포
03. Helm 소개 및 차트 개요
04. Helm 차트 작성 실습 및 배포
05. Kubernetes 관련 알아두면 좋은 것들
CH11. API 문서화
01. gRPC 컴포넌트 문서화 도구 도입 및 문서 작성
02. REST API 문서화 도구 도입 및 문서 작성
03. GraphQL 문서화 도구 도입 및 문서 작성
CH12. 웹 클라이언트 구축
01. Vue3 프로젝트 환경 설정 및 Apollo Client 설정
02. Vue3 로그인 페이지 구현 및 JWT 인증
03. Vue3 주요 페이지 구현
CH13. 프로젝트 완성 및 리뷰
01. 전체 프로젝트 검토 및 리뷰

CURRICULUM

부록.

고성능 / 유연한 백엔드 서버를 위한 API 전략

파트별 수강시간 04:09:10

CH01. API 바로알기
01. API 라는 말의 본질 이해하기
02. API 의 발전과 SDK
03. 분산 Network와 리모트 API
CH02. API를 만드는 기술들
01. SOAP의 등장과 핵심원리
02. REST 의 등장과 핵심원리
03. SOAP 과 REST 무엇을 써야하나
04. gRPC와 protobuf의 등장과 핵심원리
05. GraphQL의 등장과 핵심원리
06. SOAP, REST, gRPC, GraphQL 중 무엇을 써야하나
07. Open API와 Async API 알아보기
CH03. MSA 기반의 API 설계 개론
01. MSA 소개 및 MSA에서 API의 역할 이해
02. 누가 내 API의 고객인가
03. 요구되는 응답속도와 성능 한계
04. API 호출 형식과 배포 형식
05. API 인증과 보안
06. API 비용구조가 설계에 미치는 영향
07. API 표기법과 호환성의 고려
08. API 테스트
09. 개발 에코시스템의 고려

CURRICULUM

부록.

WebSocket과 GraphQL Subscription

파트별 수강시간 03:45:43

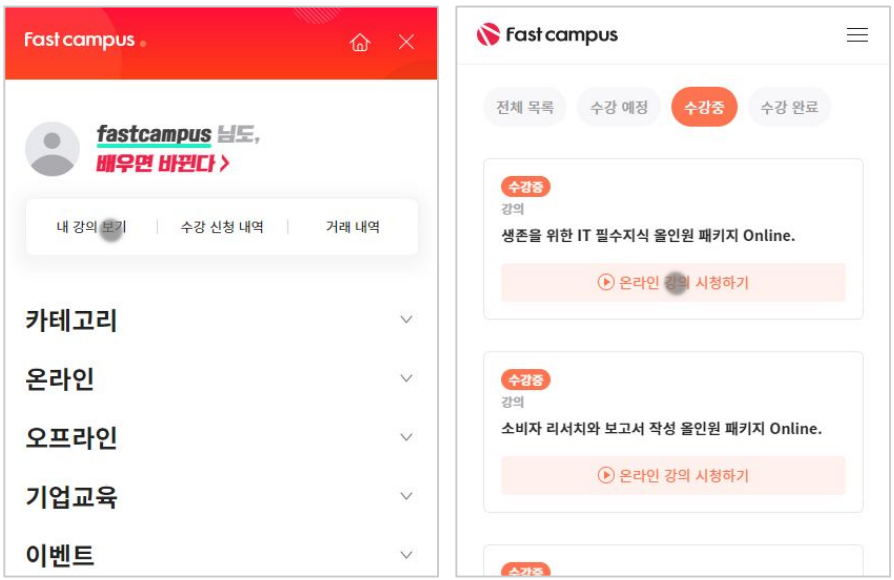
01. WebSocket의 기본 개념과 필요성
02. Spring WebSocket 설정 및 기본 사용법
03. Spring WebSocket 보안 및 인증
04. GraphQL Subscription과 WebSocket의 연계
05. Next Campus 실시간 채팅 시스템 설계
06. gRPC 강의 채팅 컴포넌트 구현
07. gRPC 강의 채팅 컴포넌트 테스트
08. GraphQL Subscription 구현 및 Gateway 연동
09. Vue3 + Apollo Client를 이용한 GraphQL Subscription 클라이언트 구현

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.

Copyright ⓒ FAST CAMPUS Corp. All Rights Reserved.