

교육 과정 소개서.

내 생애 마지막 JavaScript :
기초 문법부터 실무 웹 개발까지 한 번에 끝내기

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 01월 17일
- 2차 : 2024년 02월 14일
- 3차 : 2024년 03월 13일
- 4차 : 2024년 04월 17일
- 전체공개 : 2024년 05월 29일

최근 수정일자 2023년 12월 11일



강의정보

강의장	온라인 강의 데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	https://fastcampus.co.kr/dev_online_deepjs
강의시간	78시간 예정 (* 사전 판매 중인 강의는 시간이 변경될 수 있습니다.)
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
-----------	---



강의목표

- JavaScript의 핵심 문법을 학습하고 예제 18가지 및 프로젝트 16가지를 통해 완벽하게 이해할 수 있습니다.
- JavaScript를 통해 프론트엔드의 구조와 동작원리를 이해를 이해할 수 있습니다.
- JavaScript를 통해 프레임워크 4가지와 상태관리 3가지를 한 번에 학습할 수 있습니다.
- 기술적 설명만 있는 문법 강의는 어렵고, 이해가 안 돼 배우다 지치기만 하죠. 본 강의에서는 개념의 어원, 기술의 유래 등 배경지식과 스토리를 바탕으로 설명해 추상적 개념을 더 쉽게 이해할 수 있습니다.

강의요약

- 변수와 데이터타입으로 가장 기본적인 구성 요소부터 시작하여, 웹 페이지와의 상호작용을 가능하게 하는 DOM API까지 순차적으로 학습합니다.
- 비동기, 이벤트루프 등의 고급 주제로 나아가 학습자가 단계적으로 지식을 쌓고, 각 개념의 연결성을 이해하며 체계적으로 JavaScript 심화 문법을 학습합니다.
- 알고리즘 핵심 유형부터 문제 풀이를 위한 문법 예제까지 제공합니다.
- 프론트엔드 취업 & 커리어 Up을 위한 시니어 개발자들의 꿀팁 3종 부록 제공합니다.



강사

김의중	과목	- Part 1
약력		<ul style="list-style-type: none"> - 프론트엔드 개발 경력 총 8년 이상 - 현) 위대한 상상 시니어 프론트엔드 개발자 - 전) 우아한형제들 프론트엔드 개발자 [경력] - C사 개발자 커뮤니티 6,600명 이상의 팔로워를 보유한 인플루언서
박현범	과목	- Part 2/3 + 부록 3
약력		<ul style="list-style-type: none"> - 프론트엔드 개발 경력 총 15년 이상 - 현) 알서포트 프론트엔드 팀 리드 개발자
Teo	과목	- Part 4 + 부록 1
약력		<ul style="list-style-type: none"> - 프론트엔드 개발 경력 총 14년 이상 - 현) Teo의 프론트엔드 운영 - 현) 카카오 프론트엔드 시니어 개발자 - 전) 카카오 엔터프라이즈 프론트엔드 파트장 [경력] - Teo의 프론트엔드 등 유명 프론트엔드 블로그 및 디스코드 채널 운영 - Teo의 스프린트 1~15기 진행 - TeoConf 2회차 진행 - C사 개발자 커뮤니티 5,000명 이상의 팔로워를 보유한 인플루언서
이규덕	과목	- Part 5 + 부록 2
약력		<ul style="list-style-type: none"> - 프론트엔드 개발 경력 총 11년 이상 - 현) 마키나락스 프론트엔드 개발자 - 전) 딥네츄럴 프론트엔드 팀 리드 개발자



CURRICULUM

01.
JavaScript
언어의 이해

CH01. FE 개념 훑아보기
01. HTML 둘러보기 (웹의 기등)
02. HTML 의 구조(이정표와 지침)
03. HTML 주요 태그들
04. HTML 태그의 추가 정보를 제공하는 속성
05. CSS 둘러보기 (역할과 목적)
06. CSS 특징 (Cascading)
07. CSS 핵심요소 (1) 선택자
08. CSS 핵심요소 (2) 속성
09. JavaScript 필수 기초 개념 1 (변수와 데이터타입)
10. JavaScript 필수 기초 개념 2 (조건문, 반복문)
11. JavaScript 필수 기초 개념 3 (함수)
12. JavaScript 필수 기초 개념 4 (객체)
13. JavaScript 필수 기초 개념 5 (표준 내장 객체 와 DOM API)
14. HTML 의 구조(이정표와 지침)
CH02. 어려운 개념 Top 10 필수 도장 깨기
01. 실행컨텍스트
02. 클로저 - 람다 에서 왔어요
03. 자바스크립트는 왜 기본형, 참조형 데이터로 나뉘까(메모리 할당 방식의 차이)
04. 호이스팅 - 끌어올려지는거 인듯, 끌어올려지는게 아닌, 끌어올려지는거 같은 너
05. 비동기 다루기(콜백헬, Promise, Async, Await)
06. 이벤트루프 - 싱글스레드 언어인 자바스크립트가 비동기가 가능한 이유
07. 프로토타입과 비트겐슈타인
08. this - 3가지로 정리해보자
09. 이벤트 버블링과 캡처링
10. ESM, CJS - 모듈시스템

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

CURRICULUM

02.

5개 VanillaJS 예제로 실무능력 키우기

CH01. 지겹지만 팔다리 묶고 TO-DO 리스트 만들기

01. 오리엔테이션 & 환경 설정
02. Javascript로 DOM & CSS 다루기
03. 이벤트와 상태변화
04. 정리 하기

CH02. 굳이 carousel을 자바스크립트로 만들기

01. 재료 준비
02. Javascript에서 CSS 애니메이션 처리
03. 디테일 더하기
04. 일반화 하기
05. 모듈화와 인터페이스 구성

CH03. 갤러리앱 만들기

01. Media 다루기
02. Canvas?
03. Canvas에 그리기
04. Canvas에서 움직이기
05. audio 그리기?
06. 갤러리 화면 구성
07. 이벤트 처리
09. 예외 처리

CH04. 애플 st 스크롤 인터랙션 웹 페이지 만들기

01. 인터랙티브 웹?
02. 애니메이션 처리 - 1
03. 애니메이션 처리 - 2
04. 인터랙션 처리 - 1
05. 인터랙션 처리 - 2
06. 인터랙티브 페이지 만들기 - 1
07. 인터랙티브 페이지 만들기 - 2
09. 인터랙티브 페이지 만들기 - 3

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

CURRICULUM

02.

5개 VanillaJS 예제로 실무능력 키우기

CH05. 신기한거 만들어 보기

01. Web Speech API 둘러보기
02. STT 샘플 만들기
03. TTS 샘플 만들기
04. 기획 하기
05. 구현하고 테스트
06. 기능 확장 해보기 - 1
07. 기능 확장 해보기 - 2
08. 기능 확장 해보기 - 3

CH06. OOP 간보기

01. 객체지향이란?
02. 디자인 패턴?
03. 패턴 몇가지 구경하기 - 1
04. 패턴 몇가지 구경하기 - 2
05. 무지성 코드를 OOP로 바꿔 보기 - 1
06. 무지성 코드를 OOP로 바꿔 보기 - 2
07. 무지성 코드를 OOP로 바꿔 보기 - 3

CH07. FP도 짚먹

01. 함수형 프로그래밍?
02. 일급함수?
03. 순수한 함수?
04. 함수형을 대하는 마음가짐
05. 함수형 프로그래밍을 위한 도구 만들기 - 1
06. 함수형 프로그래밍을 위한 도구 만들기 - 2
07. 함수형 프로그래밍을 위한 도구 만들기 - 3
08. 무지성 코드를 함수형으로 바꿔보기 - 1
09. 무지성 코드를 함수형으로 바꿔보기 - 2
10. 무지성 코드를 함수형으로 바꿔보기 - 3

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



CURRICULUM

03.

Typescript
문법학습에서
실무 적용까지

CH01. 1인 개발할래? Typescript 개발할래?
01. Typescript란?
02. 환경설정
03. any, void, never, option
04. tuple, enum
05. readonly
06. type & interface
07. union & intersection
08. keyof
09. generic
10. overloading
11. index signature
12. partial & required
13. record
14. pick & omit
15. exclude & extract
16. awaited
17. index signature
18. using
CH02. Typescript로 마이그레이션 해보자
01. 에러 확인 및 제거 - 1
02. 에러 확인 및 제거 - 2
03. 갤러리 앱을 활용한 TS 마이그레이션 (Feat.OOP) - 1
04. 갤러리 앱을 활용한 TS 마이그레이션 (Feat.OOP) - 2
05. 갤러리 앱을 활용한 TS 마이그레이션 (Feat.OOP) - 3
06. 캐러셀을 활용한 TS 마이그레이션 (Feat.FP)_1
07. 캐러셀을 활용한 TS 마이그레이션 (Feat.FP)_2
08. 캐러셀을 활용한 TS 마이그레이션 (Feat.FP)_3

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

CURRICULUM

04.

JS 베이스의 프론트엔드 프레임워크와 라이브러리 이해하기

CH01. Intro

01. 프레임워크와 상태관리를 배워야 하는 이유
02. 강의 로드맵
03. 이 강의를 통해 알려주고 싶은 것
04. TodoApp 요구사항 정리해보기

CH02. Vanilla javascript: DOM 조작과 상호작용

01. 좋아요 버튼을 만들면서 정리하는 웹프론트엔드 개발 리마인드 : 선택, 상호작용, 화면변경
02. 자바스크립트의 기능을 확장하는 API, 그리고 DOM
03. DOM API의 3가지 분류: Selector, Event Listener, Dom Manipulation
04. (실습) Vanilla javascript로 Todo App을 만들어 보며 이해해보자
05. Summary: DOM을 이용한 웹 프론트엔드 개발 멘탈 모델 이해하기

CH03. jQuery : Ajax, DOM API의 간소화와 크로스 브라우징

01. jQuery는 무엇을 해결하고자 했나? : Ajax, DOM API의 간소화와 크로스 브라우징
02. jQuery 설치와 기본기능 소개
03. (실습) jQuery로 Todo App 만들기
04. Summary : jQuery의 역사적 의의

CH04. BackboneJS : 화면과 로직의 분리, 구조화된 개발의 시작

01. Model과 View: 관심사의 분리
02. BackboneJS 코드로 살펴보는 MV 아키텍처

CH05. KnockoutJS : 웹 개발의 진화, MVVM 패러다임의 등장

01. MVVM 개념의 등장과 웹 개발 패러다임의 변화 이해하기: 템플릿과 데이터 바인딩
02. MVVM: Controller에서 ViewModel로의 진화
03. KnockoutJS 익히기: data-bind와 observable
04. (실습) KnockoutJS로 Todo App을 만들어 보며 MVVM을 이해해보자.
05. Summary: MVVM이 만들어낸 선언적 프론트엔드 개발 멘탈 모델의 진화

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



CURRICULUM

04.

JS 베이스의
프론트엔드
프레임워크와
라이브러리
이해하기

CH06. Angular.js에서 Vue까지 : MVVM의 진화와 웹 프레임워크 개념의 정립
01. MVVM 아키텍처의 진화 : 웹 프레임워크의 탄생
1. KnockoutJS에서 Angular.js까지: MVVM의 초기 개념과 발전과 SPA의 등장
2. Vue2.0의 등장: MVVM 아키텍처의 현대적 해석
3. Vue와 MVVM의 계보: MVVM 아키텍처가 보편적인 웹 프레임워크가 되는 과정
4. 잠깐! : Vue2.0과 Vue3.0의 차이: Options vs Composition
02. 현대 웹 프레임워크의 정립: 반응형 모델, 선언적 렌더링, 컴포넌트 기반 설계
03. Vue로 알아보는 현대 웹 프레임워크
04. (실습) Vue(Options 방식)로 Todo App을 만들어보기
05. Summary : Vue를 통한 현대적 웹 프레임워크 개념 이해하기
CH07. React : Virtual DOM과 JSX, 주류 프레임워크가 되기 까지
01. React의 특징: Virtual DOM과 JSX
02. 클래스 컴포넌트에서 함수형 컴포넌트로의 전환
03. React 기본 구성 요소 알아보기 : JSX Template, Component, Props
04. React Hook 이해하기
1. 훅이란?
2. 훅의 구현 원리로 알아보는 훅의 특징과 제약사항 이해하기
3. 기본 Hook 이해하기
05. (실습) React 기반 To-Do 리스트 만들기
06. Summary : React를 통한 최신 웹 개발방식 이해하기
CH08. Preact, Vue3, Svelte, Solid, Qwik 등 최신 프레임워크의 방향성
01. Post React를 꿈꾸며
02. No Virtual DOM과 Signal
03. 저마다의 차별화 포인트
04. 앞으로의 방향성: MPA, 최적화, 개발경험 향상
05. Summary : 프레임워크 진화의 방향성과 흐름을 통한 프레임워크 이해하기

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



CURRICULUM

04.

JS 베이스의
프론트엔드
프레임워크와
라이브러리
이해하기

CH09. 상태관리란 무엇인가?
01. 상태란 무엇일까?
02. 왜 상태를 관리 해야겠다는 생각이 들었을까? : 예측 가능성과 컴포넌트간 데이터 교환문제
03. MVI 아키텍처
04. 상태관리 모델 등장 발전 흐름: Top-down, Bottom-up, Server State
CH10. Redux(Top-down 상태관리)로 To-Do앱에 상태관리 적용하기
01. Top-down 방식의 상태관리 및 Redux 소개와 설치: 예측가능성과 구조화
02. Redux의 주요 특징 학습 1 - Action, ActionCreator
03. Redux의 주요 특징 학습 2 - Reducer
04. Redux의 주요 특징 학습 3 - Store
05. Redux의 주요 특징 학습 4 - Dispatch
06. To-Do 리스트에 Redux 적용하기
07. Summary : VanilaJS To-Do와 Redux To-Do를 비교하며 JS 특징 요약 정리
CH11. Recoil(Bottom-up 상태관리)로 To-Do앱에 상태관리 적용하기
01. Bottom-up 방식의 상태관리 및 Recoil 소개와 설치: 컴포넌트간 데이터 교환
02. Recoil의 주요 특징 학습 1 - atom
03. Recoil의 주요 특징 학습 2 - selector
04. Recoil의 주요 특징 학습 3 - effect
05. To-Do 리스트에 Recoil 적용하기
06. Summary : VanilaJS To-Do와 Recoil To-Do를 비교하며 JS 특징 요약 정리
CH12. React-Query(서버 상태관리)로 To-Do앱 적용하기
01. 서버 상태관리 및 React-Query 소개 : 서버 데이터 관리
02. React-Query의 주요 특징 학습 1 - Query
03. React-Query의 주요 특징 학습 2 - Mutation
04. React-Query의 주요 특징 학습 3 - Cache
05. Summary : VanilaJS To-Do와 React-Query To-Do를 비교하며 JS 특징 요약 정리
CH13. 프레임워크와 상태관리 정리
01. Summay: 웹의 발전, 그리고 프레임워크와 상태관리

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

CURRICULUM

05.

자주 발생하는 문제 예방부터 해결까지

CH01. 문제는 왜 발생하는가?
01. 테스트 코드의 부재
02. 예제 1) 여러가지 경우의 수를 고려하지 못했다
03. 예제 2) 원래는 잘 됐던 기능인데, 추가 기능을 넣었더니 문제가 발생했다
04. 유사 패턴 관리
05. 예제 1) 문제를 해결 했는데 다른 곳에서 같은 문제가 발생 했다
06. 예제 2) 문제가 발생했던 코드와 유사한 패턴이 없었던 것 같은데 동일한 문제가 발생했다
CH02. 왜 우리는 원인을 “잘” 찾지 못하는 걸까?
01. 무심코 지나치는 에러 메시지
02. 에러 메시지 중 자주 보이는 메시지들을 알아보자
03. 충분히 개발자 도구를 활용하고 있는가
04. 개발자 도구를 활용해 디버깅을 해보자
CH03. 해결 방법과 예방 방법
01. 문제가 발생한 상황과 동일한 환경의 테스트 환경을 구축해보자
02. 추가/수정한 테스트 케이스를 통과하도록 기존 코드를 수정하자
03. 문제가 됐던 코드와 유사한 패턴을 가진 코드를 찾고 수정하여 문제를 예방하기
CH04. 실전에서 일어날 법한 문제들 해결하기
01. 예제 1) 게시글 수정이 정상적으로 이루어진 이 후 수정 된 게시글이 목록에서 갱신 되어있지 않았다
02. 예제 2) 로그인 성공 후 메인 페이지로 넘어가야 하는데 로그인 화면에서 멈춰 있다
03. 예제 3) 서버에서 Todo 목록을 받아와 렌더링 하려고 하는데 아무런 동작을 하지 않는다
04. 예제 4) 게시글 목록 화면과 게시글 상세 화면에서 날짜 표기가 일치하지 않는다
05. 예제 5) 제품을 생성, 조회하는 화면과 제품의 카테고리를 생성, 조회하는 화면이 있는데, 유사한 동작을 하는 두 기능을 하나의 패턴으로 묶어보자
06. 예제 6) 유사한 패턴으로 묶은 코드가 유연성과 확장성이 필요해졌다
CH05. JS를 TS로 변환하는 과정에서 발생하는 문제들
01. 단순히 타입만 붙이면 해결 될 줄 알았는데 아니었다.
02. 유니언 타입(!)과 인터섹션 타입(&)을 적용했는데 빨간 줄이 뜬다
03. 제너릭(Generic) 타입을 사용했을 때 발생한 문제
04. 외부 라이브러리를 가져와서 사용하는데 @types 가 없어 전부 빨간 줄이 뜬다

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

CURRICULUM

Special 1.

JavaScript로 끝내는 자료구조/알고리 즘 (코딩 테스트)

CH01. 코딩 테스트 개요 및 문제 풀이를 위한 JavaScript 문법

01. 코딩 테스트 알아보기
02. 문제 풀이를 위한 JavaScript 핵심 문법 알아보기
03. JavaScript 입출력 문제 풀이 4문제
04. JavaScript 조건문 문제 풀이 4문제
05. JavaScript 반복문 문제 풀이 4문제
06. JavaScript 배열 문제 풀이 4문제
07. JavaScript 문자열 문제 풀이 4문제

CH02. JavaScript 핵심 자료구조 알아보기

01. 자료구조(Data Structure)
02. 배열(Array)과 리스트(List)
03. JavaScript 스택(stack)
04. JavaScript 큐(queue)
05. 트리(Tree)와 우선순위 큐(Priority Queue)
06. 그래프(Graph)의 표현

CH03. JavaScript 정렬(sorting) 알고리즘

01. 선택 정렬
02. 버블 정렬
03. 삽입 정렬
04. 병합 정렬
05. JavaScript 정렬 라이브러리
06. 정렬 문제 풀이 1
07. 정렬 문제 풀이 2
08. 정렬 문제 풀이 3

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

CURRICULUM

Special 1.

JavaScript로 끝내는 자료구조/알고리 즘 (코딩 테스트)

CH04. JavaScript 탐욕법(greedy) 알고리즘

01. 그리디 알고리즘 이해하기
02. 그리디 문제 풀이 1
03. 그리디 문제 풀이 2
04. 그리디 문제 풀이 3
05. 그리디 문제 풀이 4

CH05. JavaScript 이진 탐색 알고리즘

01. 이진 탐색 알고리즘 이해하기
02. 정렬된 배열에서 특정한 값을 가지는 원소의 개수 구하기
03. 파라메트릭 서치 이해하기
04. 이진 탐색 문제 풀이 1
05. 이진 탐색 문제 풀이 2

CH06. JavaScript 백트래킹 알고리즘

01. 백트래킹 알고리즘 이해하기
02. 백트래킹 문제 풀이 1
03. 백트래킹 문제 풀이 2
04. 백트래킹 문제 풀이 3
05. 백트래킹 문제 풀이 4

CH07. JavaScript DFS 알고리즘

01. DFS 알고리즘 이해하기
02. DFS 문제 풀이 1
03. DFS 문제 풀이 2
04. DFS 문제 풀이 3
05. DFS 문제 풀이 4
06. DFS 문제 풀이 5
07. DFS 문제 풀이 6

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



CURRICULUM

Special 1.
JavaScript로
끝내는
자료구조/알고리
즘 (코딩 테스트)

CH08. JavaScript BFS 알고리즘
01. BFS 알고리즘 이해하기
02. BFS 문제 풀이 1
03. BFS 문제 풀이 2
04. BFS 문제 풀이 3
05. BFS 문제 풀이 4
06. BFS 문제 풀이 5
07. BFS 문제 풀이 6
CH09. JavaScript 다이내믹 프로그래밍
01. 다이내믹 프로그래밍 이해하기
02. 다이내믹 프로그래밍 문제 풀이 1
03. 다이내믹 프로그래밍 문제 풀이 2
04. 다이내믹 프로그래밍 문제 풀이 3
05. 다이내믹 프로그래밍 문제 풀이 4
06. 다이내믹 프로그래밍 문제 풀이 5
CH10. JavaScript 최단 경로
01. 다익스트라 알고리즘 이해하기 1단계
02. 다익스트라 알고리즘 이해하기 2단계
03. 플로이드 워셜 알고리즘 이해하기
04. 벨만 포드 알고리즘 이해하기
05. 최단 경로 문제 풀이 1
06. 최단 경로 문제 풀이 2
07. 최단 경로 문제 풀이 3
CH11. JavaScript 투 포인터 알고리즘
01. 투 포인터 알고리즘 이해하기
02. 투 포인터 알고리즘 문제 풀이 3문제
03. 투 포인터 알고리즘 문제 풀이 3문제
CH12. JavaScript 누적합 알고리즘
01. 누적합 알고리즘 이해하기
02. 누적합 알고리즘 문제 풀이 3문제
03. 누적합 알고리즘 문제 풀이 3문제

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

CURRICULUM

Special 2.

프론트엔드 개발자가 알아야 하는 CS 지식

CH01. 웹(Web)

01. 웹 브라우저의 동작 방식
02. 쿠키(Cookie)와 세션(Session)
03. HTTP
04. REST API
05. OAuth
06. JWT(JSON Web Token)

CH02. 컴퓨터 구조

01. 컴퓨터 구조 기초
02. 컴퓨터의 구성요소
03. 중앙처리장치(CPU) 동작 방식
04. 캐시 메모리

CH03. 데이터베이스

01. 데이터베이스 기초
02. 키(Key)
03. SQL 알아보기
04. 정규화(Normalization)
05. 트랜잭션(Transaction)

CH04. 컴퓨터 네트워크

01. OSI 7 계층
02. TCP 3-Way / 4-Way Handshake
03. TCP/IP 흐름제어 & 혼잡제어
04. UDP

CH05. 소프트웨어 공학(SW Engineering)

01. 코드 리팩토링
02. 클린코드
03. TDD
04. 애자일(Agile)
05. 객체 지향 프로그래밍
06. 함수형 프로그래밍

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

CURRICULUM

CH01. 기술면접 모음집

01. 최신성 반영을 위해 1월 중 공개

부록1.

기술면접 모음집

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

CURRICULUM

부록2.

FE 개발자가
알려주는
클린코드 작성법

CH01. 좋은 코드를 쓰기 위해 갖춰야 할 개발자의 자세
01. 코드를 객관적으로 바라보는 시선이 필요하다
02. 이론으로 습득한 지식을 적용할 줄 알아야 한다
CH02. 클린코드의 개념 정의
01. 가독성과 간결성
02. 재사용성과 테스트 가능성
CH03. 클린코드 작성 예시
01. 의미 있는 이름을 지어보자
02. 작은 함수를 만들어보자
03. 단위 테스트가 가능하도록 코드를 나눠보자
CH04. 클린코드 작성법
01. 예제) 간단한 기능을 가진 Todo List 를 작성하고 클린코드를 적용해보자

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



CURRICULUM

부록3.

워크홀릭 팀장이 같이 일하고 싶은 개발자?

CH01. 누구냐 넌?
01. 이 사람 알고싶다...
02. 함께하면 편안한 사람
03. 난 나쁜 사람이 되고싶지 않아요
04. 진짜 이직 사유?
05. 핑계와 이유사이
06. 생각하는건 쉽고 행동하는건 귀찮다

본 과정은 현재 촬영 및 편집이 진행되고 있는 **사전 판매 중인 강의**입니다.

해당 교육과정 소개서는 변경되거나 추가될 수 있습니다.

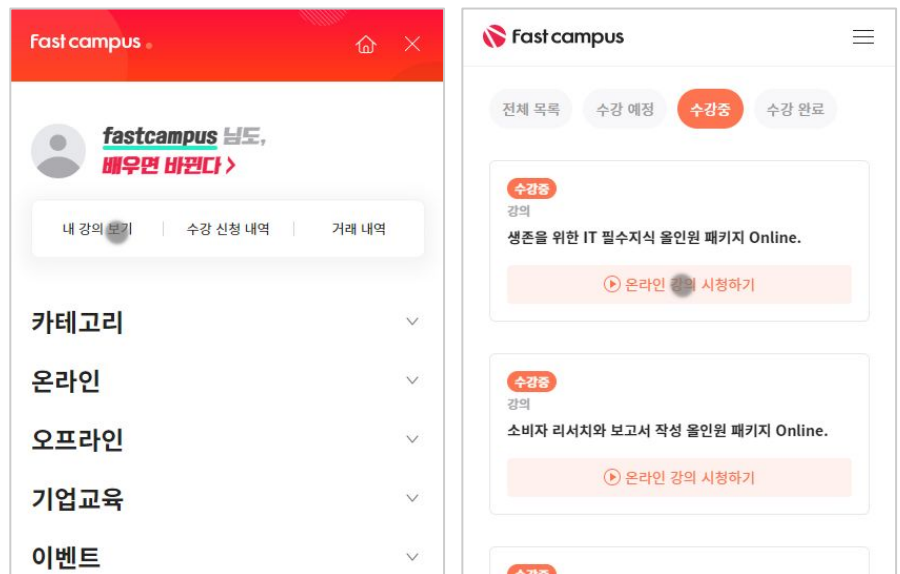


주의 사항

- 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.