

# 교육 과정 소개서.

---

올인원 패키지 : 만들면서 익히는 React의 모든 것 개발부터 배포까지 with 25개 스택



## 강의정보

강의장	온라인 강의   데스크탑, 노트북, 모바일 등
수강 기간	평생 소장
상세페이지	<a href="https://fastcampus.co.kr/dev_online_fereact">https://fastcampus.co.kr/dev_online_fereact</a>
강의시간	66시간 52분
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

- React의 사용성을 높이는 스택을 함께 학습합니다. 따로 배울 필요 없이 한 강의로 끝낼 수 있습니다.
- 입문부터 실무까지 한 번에! 개발부터 배포까지 한 강의로 학습하여, 웹 개발의 전체 이해도를 높일 수 있습니다.
- 대규모 프로젝트 4가지 + 실습 기반 스택 학습! 개념과 연계된 프로젝트 학습으로 실무에 바로 적용할 수 있습니다.

## 강의요약

- 기초 개념을 학습하는 공식 문서 바탕의 프로젝트로 시작해 추가 스택을 곁들여 풀스택 프로젝트까지 구현하며 실전 단계까지 학습해 봅니다.
- 개발, 배포, CI/CD 파이프라인, 최적화 등 React 기초 개념부터 실전에 필요한 개념까지 전천후로 담았습니다.
- 전체적인 웹 개발 이해가 가능하도록 서버 구현, 연동 등도 포함하여 학습합니다.
- 본 강의는 [프론트엔드 웹 개발의 모든 것] 강의의 Course 2 내용을 담고 있으며 기존 강의의 전체 분량(160시간 이상)이 부담스러우신 분을 위하여 출시되었습니다.



## 강사

안재원

과목

- 만들면서 익히는 React의 모든 것 개발부터 배포까지 with 25개 스택

약력

- 카카오 신입 공채 FE 교수자
- 다수의 강의 교육 업체에서 활발히 활동 중인 온/오프라인 강사
- '따라하며 배우는 도커와 CI환경'저자
- 리액트 및 자바스크립트를 이용한 웹 게임 개발



CURRICULUM

01.

**프로젝트로  
시작하는 React**

파트별 수강시간 06:38:48

<b>Ch 01. 리액트란 무엇이며 리액트를 배우는 이유</b>
01. 리액트란 무엇인가요?
02. 리액트를 사용하는 이유는?
03. 리액트 컴포넌트란 무엇인가요?
04. Nodejs, Visual Studio Code 설치
05. 브라우저가 그려지는 원리 및 가상돔
06. Create React App 을 이용해서 리액트 설치하기
<b>Ch 02. TicTacToe 앱 만들며 리액트 익히기</b>
01. Create React App으로 설치된 리액트 기본 구조 살펴보기
02. React App 실행해보기 (npm run start)_1
03. SPA(Single Page Application) 이란?
04. TicTacToe 앱 소개 및 JSX 알아보기
05. TicTacToe 앱 만들기 시작(App.js)
06. Board, Square 컴포넌트 생성하기
07. Props를 통해 컴포넌트 간 데이터 전달하기
08. 사용자와 상호작용하는 컴포넌트 만들기
09. State 추가하기
10. super(props) 란?
11. State와 Props
12. 부모 컴포넌트에서 State 보관하기
13. Square 컴포넌트 클릭 시 변화주기
14. 리액트 불변성 지키기
<b>Ch 03. TicTacToe 앱 함수형 컴포넌트로 변경하기</b>
01. React Hooks란 무엇인가?
02. 리액트 Hooks를 이용해서 TicTacToe 앱을 함수형 컴포넌트로 바꾸기
03. 순서를 만들어서 O를 추가해주기
04. 승자 결정하기

## CURRICULUM

## 01.

# 프로젝트로 시작하는 React

파트별 수강시간 06:38:48

## Ch 04. TicTacToe 앱 시간 여행 추가하기

01. 동작에 대한 기록 저장하기
02. 전개 연산자(Spread Operator)
03. 과거의 이동 표시하기(map 메소드)
04. JSX Key 속성 이해하기
05. JumpTo 함수 작성하기
06. TicTacToe 앱 스타일링 변경하기
07. github를 이용해서 배포하기

## Ch 05. 리액트 앱 성능 개선하기

01. 리액트 확장 프로그램 추가하기
02. React Profiler
03. Profiler 성능 측정을 위한 앱 생성하기
04. B 컴포넌트 생성하기
05. React Profiler로 성능 측정하기
06. React.memo 를 이용한 성능 최적화
07. 얕은 비교(Shallow Equal)
08. useCallback 을 이용한 함수 최적화
09. useMemo를 이용한 결과 값 최적화



CURRICULUM

02.

React로 실제 서비스 만들기

파트별 수강시간 07:48:48

<b>Ch 01. Disney Plus 앱 만들기 시작</b>
01. 앞으로 만들 Disney Plus App 살펴보기
02. Create-React-App으로 리액트 설치하기
03. The Movie DB API Key 생성하기
04. The Movie DB API 요청을 위한 Axios 인스턴스 생성 및 요청 보내기
05. Disney Plus 애플리케이션 전체 구조 생성하기
06. Styled Component 에 대해서 알아보기
07. 네비게이션 컴포넌트 생성 및 스타일링
08. 네비게이션 컴포넌트 스크롤 구현하기
09. Styled Component를 이용한 비디오 배너 생성하기
10. 이미지 배너 생성하기
11. 이미지 배너 UI 생성하기
12. 비디오 배너 생성하기
<b>Ch 02. 모달 및 이미지 및 라우팅 구현</b>
01. Category 컴포넌트 생성하기
02. Grid CSS
03. 영화 나열을 위한 Row 컴포넌트 생성하기
04. Row 컴포넌트 CSS 작성하기
05. 슬라이드 기능 추가하기
06. 영화 자세히 보기 클릭 시 모달 생성하기
07. Movie 모달 UI 생성하기
08. Movie 모달을 위한 CSS 작성하기
09. React Router Dom
10. React Router Dom APIs
11. Disney Plus 앱에 React Router Dom 적용하기
12. useLocation을 이용한 검색 페이지 구현하기
13. 검색 페이지 UI 구현하기
14. useDebounce Custom Hooks 만들기
15. useParams를 이용한 영화 상세 페이지 구현하기
16. 모달 창 외부 클릭 시 모달 닫게 만드는 Custom Hooks 생성
17. swiper 모듈을 이용한 터치 슬라이드 구현하기
18. swiper 모듈 슬라이드를 위한 CSS 작성하기

## CURRICULUM

## 02.

# React로 실제 서비스 만들기

파트별 수강시간 07:48:48

**Ch 03. Firebase를 이용한 인증처리 구현하기**

- |                              |
|------------------------------|
| 01. Firebase란 무엇인가요          |
| 02. Firebase와 애플리케이션 연결하기    |
| 03. 로그인 페이지 생성하기             |
| 04. 로그인 페이지를 위한 CSS 작성하기     |
| 05. Firebase를 이용한 구글 로그인     |
| 06. Firebase를 이용한 인증 체크      |
| 07. Firebase를 이용한 로그아웃       |
| 08. 로그아웃 버튼을 위한 CSS 작성하기     |
| 09. localStrage에 userData 담기 |
| 10. Firebase를 이용해서 배포하기      |





CURRICULUM

# 03.

## 5가지 프레임워크로 배우는 상태관리

파트별 수강시간 08:44:20

<b>Ch 01. 리액트 Context 를 이용한 간단한 앱 만들기</b>
01. 리액트 Context 란
02. Context API 살펴보기
03. useContext Hook 사용하기
04. Context를 이용해서 만들 앱 살펴보기
05. 프로젝트 시작 및 전체적인 구조 만들기
06. Summary 페이지 Form 구현하기
07. 주문 페이지 UI 생성하기
08. 상품 이미지 가져오기
09. 서버 데이터 가져올 때 에러 발생 시 처리하기
10. 옵션 정보 가져오기
11. Context를 사용해서 데이터 제공하기
12. State를 업데이트해주는 함수 작성하기
13. 상품 Count를 이용한 가격 계산
14. 계산된 가격 보여주기
15. 스텝을 이용해서 페이지 나누기
16. 주문 확인 페이지 구현하기
17. 주문 완료 페이지 구현하기
<b>Ch 02. 리덕스 를 이용한 간단한 앱 만들기</b>
01. 리덕스란
02. 미들웨어 없이 리덕스 카운터 앱 만들기
03. combineReducersCH2-3. combineReducers
04. Redux Provider
05. useSelector _ useDispatch
06. 리덕스 미들웨어
07. Redux Thunk
08. 리덕스 툴킷(redux toolkit)
09. 리덕스 툴킷 더 자세히 알아보기(APIs)
10. Disney 앱에 Redux 적용하기
11. Redux-Persist



CURRICULUM

**03.**  
**5가지**  
**프레임워크로**  
**배우는 상태관리**

파트별 수강시간 08:44:20

<b>Ch 03. MobX 를 이용한 간단한 앱 만들기</b>
01. MobX 란
02. Mobx로 카운터 앱 만들기
03. Mobx로 카운터 앱 만들기(데코레이터 사용)
04. React Context를 이용한 Observable 공유하기
05. Mobx 를 이용한 Todo 앱 만들기
06. Todo 앱 UI 생성하기
<b>Ch 04. Recoil 를 이용한 간단한 앱 만들기</b>
01. Recoil 란
02. Recoil을 이용한 간단한 앱 만들기
03. Recoil을 이용한 Todo 앱 만들기
04. Todo List 수정하는 기능 생성하기
05. Todo List Toggle 및 삭제 기능 생성하기
06. Todo List 필터링하기
07. 통계 기능 추가하기
08. 비동기 데이터 쿼리
<b>Ch 05. Zustand 를 이용한 간단한 앱 만들기</b>
01. Zustand 란?
02. Zustand로 Todo App 구현하기(Store 생성)
03. Todo App UI 구현하기
04. 저장소의 데이터 유지하기(Persist Middleware)
05. DevTools 이용하기
06. async 요청 보내기

---

CURRICULUM

[Special].

## React Version 18 대응하기

파트별 수강시간 00:19:07

---

Ch 01. 리액트 Version 18
01. React Version 18 & Automatic batching
02. Suspense on the server (간단하게)
03. Transition

## CURRICULUM

## 04.

Next.js로 블로그  
만들기

파트별 수강시간 02:59:44

## Ch 01. Next.js와 TypeScript

01. Next.js 란

02. Next.js 기본 파일 구조

03. Pre-rendering

04. Data Fetching

05. TypeScript 란

06. Next.js와 TypeScript 만들 앱 소개

07. 메인 페이지 UI 만들기(마크다운 파일 생성)

08. 마크다운 파일을 데이터로 추출하기

09. TypeScript Type

10. Typescript 추가 제공 타입

11. Type annotation, Type inference

12. Type assertion

13. getStaticProps를 이용한 포스트 리스트 나열

14. 포스트 자세히 보기 페이지로 이동(file system 기반의 라우팅)

15. 포스트 데이터를 가져와서 보여주기(remark)

16. 블로그앱 스타일링

17. Nextjs 13

## CURRICULUM

## 05.

## Next.js

## 프로젝트2:

## 중고마켓

파트별 수강시간 11:30:32

## Ch 01. 인증 및 인가 기능 구현하기

01. 리액트 앱 생성하기
02. Navbar 컴포넌트 생성하기
03. User, Admin 페이지 생성하기
04. Prisma, Postgres 설정하기
05. 윈도우에서 도커 설치[참조]
06. 맥에서 도커 설치[참조]
07. ORM에 대해서[참조]
08. pgAdmin 사용법[참조]
09. Next Auth를 위한 디비 스키마 생성하기
10. Next Auth 설정하기
11. 로그인 한 사람만 특정 경로에 접근에 접근하기
12. 세션 데이터에 토큰 데이터 넣어주기
13. 권한에 따른 인가 처리하기(authorization)
14. Input 컴포넌트 생성하기
15. Button 컴포넌트 생성하기
16. 로그인, 회원가입 UI 생성하기
17. 회원가입 기능 구현하기(Route Handlers)
18. 로그인 시 실제 유저 세션 데이터 생성하기
19. getSession

## Ch 02. 상품 업로드 페이지 생성하기

01. 상품 업로드 UI 생성하기
02. 상품을 위한 스키마 작성하기
03. ImageUpload 컴포넌트 생성하기
04. Cloudinary 연동하기
05. Category 컴포넌트 생성하기
06. Kakao API KEY 생성하기
07. Kakao Map 컴포넌트 생성하기
08. 상품 데이터베이스에 저장하기



CURRICULUM

05.

Next.js

프로젝트2:

중고마켓

파트별 수강시간 11:30:32

<b>Ch 03. 홈 페이지 생성하기</b>
01. 홈 페이지 데이터 생성하기(메인 페이지)
02. 홈 페이지 UI 생성하기
03. ProductCard 컴포넌트 생성하기
04. Favorite 기능 생성하기
05. Favorite Post Route 생성하기
06. Favorite Delete Route 생성하기
07. react toastify
08. dayjs
09. Categories 컴포넌트 생성하기
10. Pagination
11. PaginationLink 생성하기
12. Loader 컴포넌트 생성하기
<b>Ch 04. 상품 상세 페이지 생성하기</b>
01. 상품 상세 페이지 데이터 생성하기
02. 상품 상세 페이지 UI 생성하기
03. ProductHead 컴포넌트 생성하기
04. ProductInfo 컴포넌트 생성하기
<b>Ch 05. 채팅 페이지 생성하기</b>
01. 채팅 페이지 UI 생성하기
02. 채팅을 위한 스키마 작성하기
03. 채팅을 구현하는 여러 가지 방법
04. Rest vs WebSocket[참조]
05. Polling에 대해서[참조]
06. 채팅을 위해 필요한 데이터 생성하기
07. SWR
08. 채팅 유저 나열하기
09. 유저 컴포넌트 생성하기
10. 채팅 메시지를 위한 Input 컴포넌트 생성하기
11. 채팅 메시지를 위한 Route 생성하기
12. ChatHeader 컴포넌트 생성하기
13. Message 컴포넌트 생성하기
14. useSWRMutation
15. 채팅 메시지 생성 시 이미지 업로드하기



CURRICULUM

06.

**번들러 살펴보기**

파트별 수강시간 02:24:38

<b>Ch 01. Webpack</b>
01. Webpack 이란?
02. 폴더 및 파일 구조 생성
03. Import 기능 구현
04. webpack 설정 파일 생성
05. Webpack Loader
06. HTML Webpack Plugin
07. Webpack Caching
08. Webpack development server
09. gzip 압축
10. Webpack Devtool
11. Babel Loader
12. Resource Asset
13. bundle analyzer
<b>Ch 02. Vite vs Create-React-App</b>
01. Vite이란?
02. Vite을 이용한 리액트 설치
03. reate React App VS Vite



## CURRICULUM

## 07.

# CSS 프레임워크 완전정복

파트별 수강시간 12:19:06

<b>Ch 01. Material UI</b>
01. MUI 설치하기
02. MUI 기본 사용법
03. Theme 사용하기
04. BreakPoints
05. Material UI Grid
06. Stack Box
07. 현재까지 파일 및 소스 코드 정리
08. App Bar(Header)
09. Menu(Drop Down)
10. Lists
11. Cards
12. Checkbox _ Switch
13. Tooltip
14. Modal
15. Dark Mode
<b>Ch 02. Tailwind CSS</b>
01. TailWindCSS에 대해서
02. CRA 리액트 앱에 TailwindCSS 적용하기
03. PostCSS _ AutoFixer
04. Navbar 컴포넌트 생성하기
05. Home 컴포넌트 생성하기
06. About 컴포넌트 생성하기
07. Support 컴포넌트 생성하기
08. Features 컴포넌트 생성하기
09. Feedback 컴포넌트 생성하기
10. Contact 컴포넌트 생성하기
11. Footer 컴포넌트 생성하기
12. React Scroll





CURRICULUM

07.

CSS 프레임워크  
완전정복

파트별 수강시간 12:19:06

<b>Ch 03. Emotion CSS</b>
01. Emotion 이란?
02. Emotion 을 사용해서 만들 앱 살펴보기
03. 만들 앱 전체 구조 생성하기
04. React Router Dom 적용하기
05. Global Style 적용하기
06. Footer 컴포넌트 생성 및 Theme 적용하기
07. Search Page 생성하기
08. Google API를 통해 책 데이터 가져오기
09. 책 리스트 UI 작성하기
10. Book Detail Page 생성하기
<b>Ch 04. SCSS</b>
01. Sass 란?
02. Sass 특징
03. 만들게 될 Youtube 앱 살펴보기
04. 전체적인 구조 생성하기
05. Sass 파일 구조
06. Basic Styling
07. @use, @forward를 이용한 partial 구현
08. Grid를 이용한 Layout
09. 스크롤바 스타일링
10. Google Youtube API 키 발급
11. React Router Dom 적용하기
12. NavBar 생성하기
13. NavBar 스타일링
14. useWindowSize Hooks 생성하기
15. SearchBar 변경시켜주기
16. SearchContext 생성하기
17. SideBar 생성하기
18. BigSideBar 생성하기
19. BigSideBar Styling 하기
20. SmallSideBar 생성하기

## CURRICULUM

## 07.

## CSS 프레임워크 완전정복

파트별 수강시간 12:19:06

21. SmallSideBar Styling 하기
22. SideBar Toggle 기능 생성하기
23. Axios Instance 생성하기
24. 비디오 데이터 가져오기
25. 가져온 비디오 데이터 저장하기
26. 메인 페이지 UI 생성하기
27. 메인 페이지 스타일링
28. helper 함수들 생성하기
29. VideoPage 데이터 가져오기
30. VideoPage에서 사이드바 컨트롤하기
31. VideoPage UI 생성하기
32. helper 함수들 생성하기(formatNumber, formatText) 정규식 생성
33. 댓글 생성하기
34. 비디오 페이지 스타일링하기
35. 관련 비디오 생성하기
36. 관련 비디오 스타일링하기
37. SearchBar에서 검색 기능 추가하기
38. Search 기능 함수들 옮겨주기
39. 검색 페이지 생성하기
40. 검색 페이지 스타일링



CURRICULUM

08.

**Node.js로  
구현하는 백엔드**

파트별 수강시간 08:04:14

<b>Ch 01. Node.js란?</b>
01. 자바스크립트 엔진
02. Node.js History
03. Node.js Runtime
04. Node.js 설치하기
05. Node.js REPL
06. Browser APIs _ Node.js APIs
07. global 객체에 대해서
<b>Ch 02. Node.js 의 내부</b>
01. Node.js가 작업을 처리하는 방법
02. Node.js 오픈 소스 코드를 통한 이해
03. libuv
04. 비동기와 동기란?
05. 코드로 보는 async _ sync
06. Javascript는 Synchronous 언어
07. blocking, non-blocking in nodejs
08. 프로세스 및 스레드
09. 노드JS가 비동기 작업을 처리하는 방법
10. Event Loop
11. Event Loop 심화
12. setImmediate vs setTimeout vs process.NextTick
13. Node.js Event Emitter
<b>Ch 03. Node.js 모듈 시스템</b>
01. Module 이란?
02. HTTP 서버 구축하기
03. 모듈을 사용하는 이유
04. 모듈 생성하기
05. 모듈에서 exports 하는 방법
06. CommonJS와 ECMAScript 모듈의 차이
07. Node.js에서 ECMAScript 모듈 이용하기
08. 모듈 캐싱에 대해서
09. index.js 파일에 대해서

## CURRICULUM

## 08.

Node.js로  
구현하는 백엔드

파트별 수강시간 08:04:14

## Ch 04. Node.js 패키지 관리

01. npm 이란?
02. 앱에 npm package 설치하기
03. node\_modules 살펴보기
04. Semantic Versioning
05. package-lock json
06. npm audit
07. npm install -g

## Ch 05. 웹 서버 만들기

01. 웹 서버란?
02. HTTP Requests
03. Stateless Protocol
04. HTTP Request, Response 구조
05. HTTP Status Codes
06. 웹 서버 생성하기
07. HTTP Routing
08. Post 요청으로 데이터 추가하기

## CURRICULUM

## 08.

Node.js로  
구현하는 백엔드

파트별 수강시간 08:04:14

<b>Ch 06. Express.js</b>
01. Express를 사용하는 이유
02. Express js 기본 구조 코드 생성
03. req.params
04. res.send() vs res.end()
05. res.send() vs res.end()
06. 포스트맨 설치하기
07. nodemon 설치
08. middleware에 대해서
09. middleware 생성하기
10. POST 요청하기 _ express.json()
11. Model, View, Controller
12. 앱 소스 코드 MVC 패턴으로 변경해주기
13. Router, Controller
14. RESTful API
15. Express에서 파일 전송하기(res.sendFile)
16. express.static()
17. Template Engine 사용하기
18. Template Layout 생성하기
<b>Ch 07. MongoDB</b>
01. Database 란?
02. SQL vs NoSQL
03. MongoDB 사용하기
04. Mongoose에 대해서
05. Model 및 Schema 생성하기
06. CRUD 구현하기 (Create)
07. Read 구현하기
08. Update, Delete 구현하기
09. Express 에러 처리

## CURRICULUM

## 09.

N시간에 끝내는  
GraphQL

파트별 수강시간 03:56:07

<b>Ch 01. GraphQL 기본</b>
01. GraphQL
02. GraphQL 사용해보기
03. GraphQL의 장단점
04. Express GraphQL Server 생성하기
05. GraphiQL 사용하기
06. Post Comment Schema 작성하기
07. GraphQL Tools
08. Resolver
09. Resolvers 모듈화 하기
10. 필터링 기능 추가하기
11. ID로 데이터 가져오기
12. Mutation
13. Comment 생성하기
14. Apollo 란 무엇인가요_
15. Apollo Server V3
16. Migrating to Apollo Server 4
<b>Ch 02. GraphQL 을 이용한 애플리케이션 생성</b>
01. 만들게 될 Project 살펴보기
02. 앱 전체 구조 생성하기
03. TailwindCSS 설정하기
04. Apollo Client Cache
05. Fetch Policy
06. Apollo Client 사용을 위한 소스 코드 작성하기
07. useQuery를 이용한 Todo 데이터 가져오기
08. TodoItem 컴포넌트 생성하기
09. useMutation을 이용한 Todo List 생성하기
10. Todo List 삭제하기
11. Todo List 수정하기
12. 리스트 정렬하기



CURRICULUM

10.

**프론트엔드  
개발자가 알아야  
하는 Docker &  
CI/CD**

파트별 수강시간 02:06:54

<b>Ch 01. 도커로 리액트 실행하기</b>
01. 도커를 사용하는 이유
02. 도커란 무엇인가
03. 윈도우에서 도커 설치
04. 맥에서 도커 설치
05. Docker를 사용할 때의 흐름 감잡기
06. 이미지로 컨테이너를 만들기
07. 도커 이미지 생성하는 순서
08. Dockerfile 만들기
09. 도커 파일로 도커 이미지 만들기
10. 리액트를 위한 도커 파일 작성하기
11. 생성한 이미지로 어플리케이션 실행 시 접근이 안되는 이유 (포트 매핑)
12. EC2에서 도커 설치 및 실행
<b>Ch 02. Github Action을 이용한 AWS S3 자동 배포</b>
01. Github Action을 이용한 AWS S3로 앱 자동 배포하기
02. 앱 배포를 위한 AWS S3 버킷 생성하기
03. AWS S3 버킷 설정 및 어플리케이션 배포하기
04. S3로 앱 자동 배포를 위한 yml 파일 완성하기
<b>Ch 03. CloudFront</b>
01. CloudFront란
02. CloudFront 생성하기
03. 가비아를 이용한 도메인 생성하기
04. 도메인 연결하기



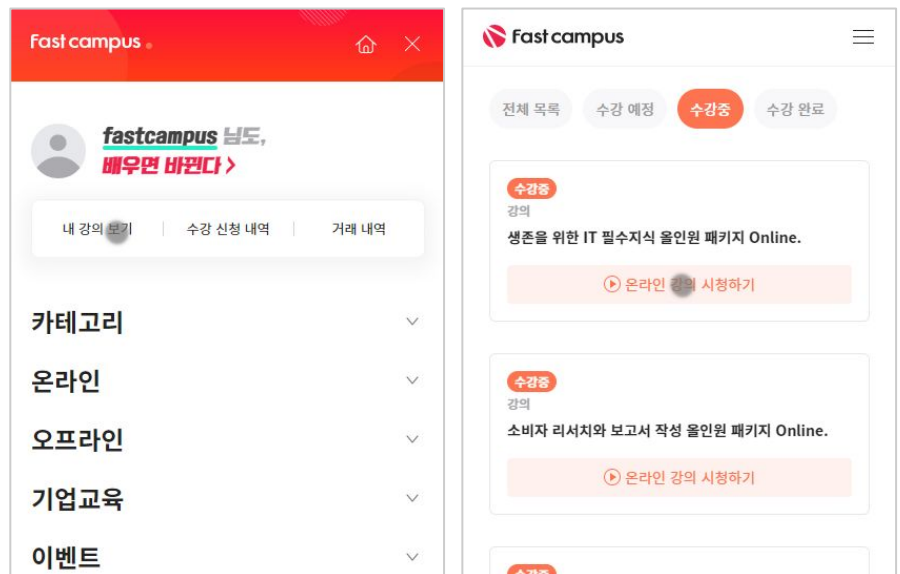


## 주의 사항

- 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.