

프론트엔드 개발자가 알아야 할 기술

1 지금 당장 알지 못해도 괜찮아

프론트엔드 개발자가 **알아야 할** 기술

HTML / JS / CSS

React

Next.js

뭘 더 알아야 하는가?

- 좋은 프론트엔드 **개발자가 되기 위해서는** 어떤 부분을 신경써야 할까요?
- 좋은 회사에 **취업**하려면 어떤 부분을 준비하면 될까요?
- 프론트엔드는 **기술 트렌드**가 정말 빠르게 바뀌는데 무엇을 어떻게 공부 해야하나요?(이부분은 다음 강의에서..)

배워야 할 게 너무 많다고 느끼는 경우

- 프론트엔드 개발자가 퍼블리싱도 해야하나요?
- 프론트엔드 개발자가 배포까지 알아야 하나요?
- 프론트엔드 개발자가 백엔드에 대한 지식도 있어야 하나요?

배워야 할 내용이 너무 많다고 생각해서 스스로 선을 긋고자 할수도..

Front end Developer Road Map

(<https://roadmap.sh/frontend>)

(<https://roadmap.sh/react>)

React 만 해도 배울게 많아보이는데...

그래도 한번 살펴보자

(<https://roadmap.sh/frontend>)

이미 많은 부분에 대해 들어봤거나 알고 있다.

HTML / CSS / JS / Git / Github / HTTPS / CORS / npm / yarn

React / Styled Components / CSS Modules / Styled JSX / Bootstrap

TailwindCSS / TypeScript / Storage / SSR / Next.js / SSG / Gatsby...

하지만,

프론트엔드 개발자가 알아야 할 기술은 여기에 **한정되지 않는다.**

(안타까울 수도 있지만)

한편으로는 프론트엔드 **개발자로서** 좋은 사실이다.

진입 장벽은 낮지만,

알아야 할 기술의 영역이 넓고,

결국, 다양한 기술들에 대한 **열린 태도**와 **학습**으로 격차가 생긴다.

이미 잘하고 있는 사람도 학습하지 않으면 도태될 수 있다.(뉴비에게는 기회)

해야할 것으로 보지말고 나의 성장 + 나의 몸값 상승으로 접근해보면..

성능 최적화: 프로파일 / Lighthouse / 여러 성능 측정 툴

스트레스 테스트: 부하 트래픽 등

보안: XSS 공격 등 대응

배포 환경: AWS(Lambda / CDN / S3) / Terraform / Docker / k8s

프로그래밍 방식: 함수형 프로그래밍 / RxJS

모바일 앱을 직접 개발: React Native

컴파일러: SWC

새로운 언어: Dart / Rust

에러 모니터링: Sentry

로깅: Elastic Search

데이터 조회: SQL

Interaction: Framer Motion 등 다양한 인터렉션 도구

애니메이션: Gasp 등 애니메이션 정의 도구

웹 서버: Nginx / Express / Koa

Chrome Extension 개발 / 데스크탑 앱 개발

그 외에도 계속 생겨나는 라이브러리와 패러다임 변화

React에 Hooks이 도입 된 것도 어찌보면 패러다임의 변화
클래스 컴포넌트에서 함수형 컴포넌트로 세대 교체
이제는 componentDidCatch 를 사용하는 ErrorBoundary 에만 남아있
음

그런 패러다임을 Vue도 따라갔다.

아예 극단적으로 번들 사이즈와 성능을 고려하는 Svelte 라는 새로운
프레임워크가 나오기도 했다.

로드맵을 대하는 우리의 자세

로드맵과 알아야 할 기술들에 대해서는 **인지하고 있되**,
압도되지 않아도 된다.

그때 그때 만나는 문제를 해결하기 위해 필요한 기술들을 찾고,
쫓아가다보면 어느새 성장해 있을 것이다.

PHP 기반의 기술스택에 React를 도입하기 시작했고,
SEO를 하기 위해서 Next.js를 선택
타입 정의로 코드 베이스 효과적으로 관리 하려고 Typescript 도입
다국어 기능을 제공하기 위해 i18n 적용
Client side에서 발생하는 error를 트래킹 하기 위해 Sentry 도입
Vercel이 없었기에 AWS EC2에 직접 node 서버를 띄움
트래픽 대응을 위해 node 서버들 process manager PM2 활용
확장을 위해서 Docker로 서비스를 맡고 동적으로 확장 가능하게 대응

지금 내가 고민하고 있는 문제는

이미 많은 사람들이 미리 고민했을 것이다.

미리 고민한 것에서 더 나아가 좋은 해결책까지 마련해뒀을 것이다.

내가 하는 고민에 대한 해결책을 검색(구글링)하고 찾아서 적용하고
익히자.

프론트엔드 개발자 기술 습득 사이클



문제 발견은 결국 사이드 프로젝트 혹은 업무

방법 찾기는 문제를 명확화 해야 함 (찾아지지 않는다면, 문제가 아닐 수 있음)

검색이 안된다면 문제를 해결하려는 접근 방식이 잘못 되었을 수 있다.

적용을 넘어 숙달을 계속 시도해야한다.

정리

프론트엔드 개발자가 알아야 할 기술

뭘 더 알아야 하는가?

선을 그으려고 하지 말자

프론트엔드 로드맵

roadmap.sh/frontend

다양한 기술 및 영역

프론트엔드 / 브라우저 / 배포 / 모니터링

사이클

문제 발견 - 방법 찾기 - 적용 - 익히기