

라이브러리란?

1 라이브러리는 도구다

라이브러리란?

라이브러리 vs 프레임워크

- 라이브러리는 개발 편의를 위한 **도구의 모음**
- 프레임워크는 **기반 구조까지** 잡혀있음
- 라이브러리는 **공구**
- 프레임워크는 **공장**

리엑트는 **도구** 모음

JSX / 컴포넌트의 상태 관리 / 렌더링 최적화 / 성능 확인

그 외에도 **필요한** 것들이 많음

이미 잘 만들어둔 도구들을 가져다가 쓰면서 리엑트 **생태계**의 이점을 누리자

라이브러리 적용 과정

필요 > 검색 > 사용법 파악 > 적용

필요 > 검색 > 사용법 파악 > 적용

- 특정 기능이 필요해짐(직접 구현하기에는 조금 벅참)
- 더 나은 도구가 필요해짐
- 트렌드에 따른 새로운 기술에 대한 호기심

필요 > **검색** > 사용법 파악 > 적용

- 필요 그대로를 검색
- github / npm trends / 프론트엔드 커뮤니티 / 기술 블로그
- 유사한 라이브러리들을 후보로 비교(기능 / 성능 / 사이즈 / 인기 / 최신 ...)

필요 > 검색 > **사용법 파악** > 적용

- 공식 문서
- github 코드 검색(실제 용례)
- 기술 블로그의 적용하기 예제

필요 > 검색 > 사용법 파악 > 적용

- POC(Proof of concept)
- 프로젝트 일부에 적용
- 프로젝트 전체에 적용

다루게 될 라이브러리들

2 강의를 통해 얻어갈 수 있는 것들

Date 관련(날짜 표기 / 계산 도구)

스타일링 관련(스타일링을 조금 더 편하게 도와주는 도구)

디자인 UI 관련(미리 해둔 스타일링을 제공하는 도구)

애니메이션 관련(애니메이션 기능을 미리 구현해두고 제공하는 도구)

모킹 관련(데이터 페칭을 대체해주는 도구)

상태 관리 관련(컴포넌트의 상태 관리를 도와주는 도구)

데이터 페칭 관련(네트워크 통신을 도와주는 도구)

GraphQL(REST api를 대체하는 도구)

Next.js(SSR 등 최신 프론트엔드 기술을 제공해주는 도구)

리액트와 밀접하게 관련된 라이브러리도 있고,
그렇지 않은 라이브러리도 있음

웬만하면 최근 트렌드를 담으려고 했음(2021년 기준)

라이브러리의 사용법을 익히기 보다
라이브러리를 **찾고** / **적용하는 과정**을 익혀 혼자 할 수 있는 힘을 기르
자

정리

라이브러리란?

라이브러리

공구 / 도구

라이브러리 도입 과정

필요 > 검색 > 사용법 파악 > 적용

사용법을 익히기 보다

찾고 / 적용하는 과정을 익히자