

Next.js 심화 2

1 Dynamic Import / Static Export

Dynamic Import

(<https://nextjs.org/docs/advanced-features/dynamic-import>)

컴포넌트를 Lazy load하는 방법

```
dynamic(() => import('../components/Button'), {suspense: true})
```

React 18 미만에서는

suspense: true 대신 loading: () => 컴포넌트로

로딩 중에 노출할 컴포넌트를 노출할 수 있다.

ssr: false를 주면

client-side 에서만 dynamically load 하게 된다.

window 등 존재 여부 판단 하지 않아도 됨

외부 라이브러리도

import 함수를 사용해서 dynamic 하게 load해서 사용할 수 있다.

실제로 한번 해보자

blog 안에 있는 [id].js 의 Button을 component로 빼보자.

Automatic Static Optimization

정적인 페이지는 .html 으로

요청에 맞춰 동작하는 페이지는 .js 로

getInitialProps 나 getServerSideProps 가 있다면 .js

router의 query..

client-side 페이지의 경우,

hydration 이후에 query 값을 읽을 수 있다.

Static HTML Export

(<https://nextjs.org/docs/advanced-features/static-html-export>)

Next.js 프로젝트를 정적인 파일들만으로 Build 하는 것

CDN 등에 올려서 서비스를 제공 가능

단, Node.js 서버가 있어야지만 동작하는 기능들은 포기해야함

정리

Dynamic Import / Static Export

Dynamic Import

Lazy load로 초기 청크 사이즈 줄이기

Suspense

Promise resolved 되어야 렌더

Automatic Static Optimize

알아서 정적 파일과 동적 파일 구분

Static HTML Export

의도적으로 정적 파일로 export 가능