

preloadModule

📖 중요합니다!

React 기반 프레임워크는 리소스 로딩을 대신 처리하는 경우가 많으므로 이 API를 직접 호출할 필요가 없을 수도 있습니다. 자세한 내용은 해당 프레임워크의 문서를 참조하세요.

preloadModule 을 사용하면 사용할 ESM 모듈을 미리 가져올 수 있습니다.

```
preloadModule("https://example.com/module.js", {as: "script"});
```

- 레퍼런스

- `preloadModule(href, options)`

- 사용법

- 렌더링 시 미리 불러오기
- 이벤트 핸들러에서 미리 불러오기

레퍼런스

preloadModule(href, options)

ESM 모듈을 미리 불러오려면 `react-dom` 에서 `preloadModule` 함수를 호출합니다.

```
import { preloadModule } from 'react-dom';

function AppRoot() {
  preloadModule("https://example.com/module.js", {as: "script"});
  // ...
}
```

아래 예시를 참조하세요.

`preloadModule` 기능은 브라우저에 주어진 모듈 다운로드를 시작해야 한다는 힌트를 제공하여 시간을 절약할 수 있습니다.

매개변수

- `href`: 문자열입니다. 다운로드하려는 모듈의 URL입니다.
- `options`: 객체입니다. 여기에는 다음과 같은 속성들이 포함되어 있습니다.
 - `as`: 필수 문자열입니다. 'script' 여야 합니다.
 - `crossOrigin`: 문자열입니다. 사용할 [CORS 정책](#)입니다. 가능한 값은 `anonymous` 와 `use-credentials` 입니다.
 - `integrity`: 문자열입니다. 모듈의 [신뢰성 확인](#)을 위한 모듈의 암호화 해시입니다.
 - `nonce`: 문자열입니다. 엄격한 콘텐츠 보안 정책을 사용할 때 [모듈을 허용하기 위한 암호화 논스\(Nonce\)](#)입니다.

반환값

`preloadModule` 는 아무것도 반환하지 않습니다.

주의 사항

- 동일한 `href` 로 `preloadModule` 을 여러 번 호출하는 것은 한 번 호출하는 것과 동일한 효과가 있습니다.
- 브라우저에서는 컴포넌트를 렌더링하는 중이거나, `Effect`, 이벤트 핸들러 등 어떤 상황에서도 `preloadModule` 을 호출할 수 있습니다.
- 서버 측 렌더링이나 서버 컴포넌트를 렌더링할 때, `preloadModule` 은 컴포넌트를 렌더링하는 동안 호출하거나, 컴포넌트 렌더링에서 시작된 비동기 컨텍스트 내에서 호출할 때만 효과가 있습니다. 그 외의 호출은 무시됩니다.

사용법

렌더링 시 미리 불러오기

컴포넌트 또는 그 자식 컴포넌트가 특정 모듈을 사용한다는 것을 알고 있다면, 컴포넌트를 렌더링 할 때 `preloadModule` 을 호출하세요.

```
import { preloadModule } from 'react-dom';

function AppRoot() {
  preloadModule("https://example.com/module.js", {as: "script"});
  return ...;
}
```

브라우저에서 모듈을 즉시 실행하려면(단순히 다운로드하는 것 대신) `preinitModule` 을 대신 사용하세요. ESM 모듈이 아닌 스크립트를 로드하려면 `preload` 를 사용하세요.

이벤트 핸들러에서 미리 불러오기

모듈이 필요한 페이지나 State를 전환되기 전에 이벤트 핸들러에서 `preloadModule` 을 호출하세요. 이렇게 하면 새로운 페이지나 State를 렌더링할 때 호출하는 것보다 더 일찍 프로세스를 시작할 수 있습니다.

```
import { preloadModule } from 'react-dom';

function CallToAction() {
  const onClick = () => {
    preloadModule("https://example.com/module.js", {as: "script"});
    startWizard();
  }
  return (
    <button onClick={onClick}>Start Wizard</button>
  );
}
```

Meta Open Source

Copyright © Meta Platforms, Inc

uwu?

React 학습하기

빠르게 시작하기

설치하기

UI 표현하기

상호작용성 더하기

State 관리하기

탈출구

API 참고서

React APIs

React DOM APIs

커뮤니티

행동 강령

팀 소개

문서 기여자

감사의 말

더 보기

블로그

React Native

개인 정보 보호

약관

