



renderToStaticMarkup

`renderToStaticMarkup` 은 상호작용하지 않는 **React** 트리를 **HTML** 문자열로 렌더링합니다.

```
const html = renderToStaticMarkup(reactNode, options?)
```

- [레퍼런스](#)

- `renderToStaticMarkup(reactNode, options?)`

- [사용법](#)

- 상호작용하지 않는 **React** 트리를 **HTML** 문자열로 렌더링하기

레퍼런스

`renderToStaticMarkup(reactNode, options?)`

서버에서 `renderToStaticMarkup` 을 호출하여 앱을 **HTML**로 렌더링합니다.

```
import { renderToStaticMarkup } from 'react-dom/server';
```

```
const html = renderToStaticMarkup(<Page />);
```

이는 **React** 컴포넌트의 상호작용하지 않는 **HTML** 출력을 생성합니다.

[아래 예시를 참고하세요.](#)

매개변수

- `reactNode`: HTML로 렌더링할 React 노드입니다. 예를 들어, `<Page />` 와 같은 JSX 노드입니다.
- **optional** `options`: 서버 렌더링을 위한 객체입니다.
 - **optional** `identifierPrefix`: `useId`에 의해 생성된 ID에 대해 React가 사용하는 문자열 접두사입니다. 같은 페이지에서 여러 루트를 사용할 때 충돌을 피하기 위해 유용합니다.

반환값

HTML 문자열을 반환합니다.

주의 사항

- `renderToStaticMarkup`의 출력값은 Hydrate 될 수 없습니다.
- `renderToStaticMarkup`은 `Suspense`를 제한적으로 지원합니다. 만약 `Suspense` 컴포넌트라면, `renderToStaticMarkup`은 즉시 HTML을 Fallback으로 보냅니다.
- `renderToStaticMarkup`은 브라우저에서 동작하지만, 클라이언트 코드에서 사용하는 것은 권장하지 않습니다. 브라우저에서 컴포넌트를 HTML로 렌더링해야 하는 경우, [HTML을 DOM 노드로 렌더링해서 가져오세요](#).

사용법

상호작용하지 않는 React 트리를 HTML 문자열로 렌더링하기

`renderToStaticMarkup`을 서버 응답과 함께 보낼 수 있는 HTML 문자열로 앱에 렌더링하기 위해 호출하세요.

```
import { renderToStaticMarkup } from 'react-dom/server';

// The route handler syntax depends on your backend framework
app.use('/', (request, response) => {
  const html = renderToStaticMarkup(<Page />);
  response.send(html);
});
```

이것은 React 컴포넌트의 상호작용하지 않는 초기 HTML 출력을 생성합니다.

⚠️ 주의하세요!

이 메서드는 **Hydrate 될 수 없는 상호작용하지 않는 HTML**을 렌더링합니다. 이 메서드는 React를 간단한 정적 페이지 생성기로 사용하거나, 이메일과 같은 완전히 정적인 콘텐츠를 렌더링할 때 유용합니다.

상호작용을 위한 앱은 서버에서 `renderToString`, 클라이언트에서는 `hydrateRoot`를 사용해야 합니다.

이전
[renderToReadableStream](#)

다음
[renderToString](#)

Meta Open Source

Copyright © Meta Platforms, Inc

uwu?

React 학습하기

- [빠르게 시작하기](#)
- [설치하기](#)
- [UI 표현하기](#)
- [상호작용성 더하기](#)
- [State 관리하기](#)
- [탈출구](#)

커뮤니티

- [행동 강령](#)
- [팀 소개](#)
- [문서 기여자](#)

API 참고서

- [React APIs](#)
- [React DOM APIs](#)

더 보기

- [블로그](#)
- [React Native](#)
- [개인 정보 보호](#)

감사의 말

약관

