

01. React와 Next.js란 무엇인가?

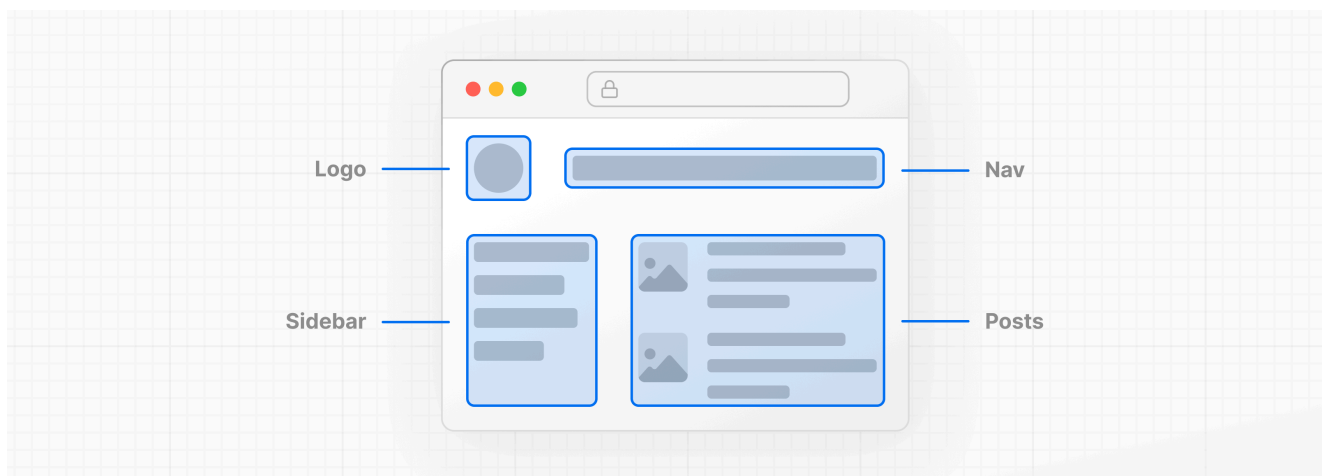
React와 Next.js란 무엇인가?

이 과정을 시작하기 전에, React와 Next.js의 개념을 살펴보고 웹 개발에서 이들의 역할에 대해 이해해 봅시다.

React란 무엇인가?

React는 인터랙티브한 사용자 인터페이스를 구축하기 위한 JavaScript 라이브러리입니다.

사용자 인터페이스(UI)란 사용자가 화면에서 보고 상호작용하는 요소들을 말합니다. 여기에는 버튼, 텍스트, 이미지 등이 포함됩니다.



사용자 인터페이스를 구축하려면 먼저 개별적인 요소들을 만든 다음, 이들을 서로 작동하도록 해야 합니다. 이는 복잡한 로직과 상태 관리가 필요할 수 있습니다.

React는 이런 UI 로직을 간소화하고, 인터페이스를 더 작고 재사용 가능한 조각으로 나누기 쉽게 해줍니다.

Next.js란 무엇인가?

Next.js는 풀스택 React 애플리케이션을 구축하기 위한 프레임워크입니다.

React는 라이브러리로, 기본적으로 React로 애플리케이션을 구축할 때 많은 결정을 내려야 합니다. 예를 들어:

- 애플리케이션을 번들링하고 변환하는 방법
- 코드에 대한 최적화 구현
- 성능과 SEO를 위한 정적 및 서버 렌더링 지원
- 클라이언트와 서버 코드 작성
- 애플리케이션에 대한 라우팅 설정

Next.js는 React에 추가적인 구조, 기능 및 최적화를 제공하는 React 프레임워크입니다. 또한 개발자와 최종 사용자 경험을 향상시키기 위한 추가 기능들을 갖추고 있습니다.

UI 빌딩 블록

웹 애플리케이션의 UI를 구축하기 위해 브라우저는 HTML, CSS 및 JavaScript를 함께 사용합니다:

- **HTML:** HyperText Markup Language는 웹 콘텐츠의 구조와 의미를 표현합니다.
- **CSS:** Cascading Style Sheets는 콘텐츠의 형태와 모양을 지정합니다.
- **JavaScript:** JavaScript는 상호작용을 추가하고 동적인 콘텐츠를 만듭니다.

React를 사용하면 이러한 기술들을 자체 JSX 구문으로 결합하여 UI 컴포넌트를 만들 수 있습니다. 예를 들어, 이 코드를 살펴보세요:

```
function Header() {  
  return <h1>Develop. Preview. Ship.</h1>;  
}
```

위의 JSX 코드는 <h1> HTML 요소를 반환하는 JavaScript 함수로 변환됩니다.

React와 Next.js로 무엇을 구축할 수 있나요?

React와 Next.js는 랜딩 페이지부터 풀스택 웹 애플리케이션까지 다양한 것들을 구축하는 데 사용할 수 있습니다:

- 정적 웹사이트: 블로그, 포트폴리오, 문서 사이트
- 전자상거래 사이트: 제품 목록, 장바구니, 결제 시스템을 갖춘 온라인 상점
- 대시보드: 데이터 시각화와 분석 도구
- 실시간 앱: 채팅, 협업 도구, 소셜 미디어 피드
- 인터랙티브 웹 앱: 일기 앱, 노트 작성 도구, 프로젝트 관리 시스템
- 멀티미디어 플랫폼: 비디오 또는 오디오 스트리밍 서비스

Next.js가 제공하는 것

Next.js는 React 애플리케이션을 구축하는 데 필요한 모든 도구와 함께 다음과 같은 기능을 제공합니다:

- 최적화: 코드 분할, 이미지 최적화, 폰트 최적화 등을 통한 성능 향상
- 라우팅: 클라이언트 측 및 서버 측 라우팅을 위한 직관적인 파일 기반 시스템
- 렌더링 옵션: 서버 사이드 렌더링, 정적 사이트 생성, 증분 정적 재생성
- 데이터 가져오기: 다양한 소스에서 데이터를 가져오고 사전 렌더링할 수 있는 간소화된 방법
- 스타일링: CSS 모듈, Tailwind CSS, styled-components 등 다양한 스타일링 옵션 지원
- 개발 경험: 빠른 새로고침, 타입스크립트 지원, 에러 리포팅
- 국제화: 다국어 콘텐츠 및 라우팅 지원
- 미들웨어: 요청 및 응답을 맞춤화하는 유연한 방법

요약

React는 UI를 구축하기 위한 JavaScript 라이브러리입니다. 그리고 Next.js는 React 애플리케이션을 구축하기 위한 프레임워크입니다.

이제 React의 기본 개념과 Next.js가 어떻게 도움이 되는지 이해했으니, React의 핵심 개념인 컴포넌트에 대해 자세히 살펴보겠습니다.

연습 문제

1. React 라이브러리와 Next.js 프레임워크의 차이점을 설명해보세요.
2. React를 사용하여 웹 인터페이스를 구축할 때의 주요 이점은 무엇인가요?
3. 다음 중 Next.js의 기능이 아닌 것은 무엇인가요?
 - 코드 분할
 - 이미지 최적화
 - 브라우저 엔진 개발
 - 서버 사이드 렌더링