Section 01 - Next.js 기본개념정리

Next.js는 React 기반 프레임워크로, 서버 사이드 렌더링(SSR)과 정적 사이트 생성(SSG) 구현의 손쉬움이 특징. 이를 통해 빠른 로딩 속도와 SEO 최적화의 동시 달성이 가능. 주요 기능으로는 라우팅, 이미지 최적화, 스타일링 등이 있음.

설치하기:Next.js 설치 과정의 간단함. Node.js 환경에서 npx create-next-app명령어 사용 시 기본적인 프로젝트 구조의 자동 생성. 이 명령어는 프로젝트 초기 설정의 간편함을 제공하여 개발자의 빠른 프로젝트 시작을 돕음. 필요한 폴더 구조와 초기 파일들이 준비되어 바로 개발 가능.

스타일 적용하기:Next.js는 다양한 스타일링 방법의 지원으로 프로젝트 스타일의 체계적인 관리가 가능. 기본적으로 CSS 모듈을 제공하여 각 컴포넌트 스타일의 독립적 관리 가능하며, Sass, Styled-components와 같은 최신 스타일링 방법도 지원. 이를 통해 스타일 재사용성과 유지보수성의 향상 가능.

폰트 및 이미지 최적화:Next.js는 폰트와 이미지의 자동 최적화로 페이지 로드 속도의 개선 가능. 사용자가 웹사이트를 더 빠르게 이용할 수 있도록 도움. Next.js의 이미지 최적화 기능은 이미지 크기와 포맷을 자동 조정하여 최적화된 이미지 제공. 이로 인해 페이지 로딩 시간 단축 및 사용자 경험 향상. 폰트 최적화 기능을 통해 웹폰트 로딩의 최적화 및 필요한 폰트만 로드하여 성능 극대화 가능.

라우팅 이해하기:Next.js는 파일 기반 라우팅을 사용하여 페이지와 URL의 간편한 매핑 가능. pages디렉토리에 파일을 추가하는 것만으로 새로운 페이지의 생성 가능. 예를 들어, about.js파일을 pages디렉토리에 추가하면 /aboutURL로 접근 가능한 페이지가 자동 생성됨. 동적 라우팅을 통해 유연한 URL 구조의 구현 가능하며, URL 파라미터를 사용하여 다양한 페이지의 효율적 관리 가능. 이를 통해 복잡한 라우팅 설정 없이도 손쉬운 페이지 추가 및 관리 가능.



Next.js를 처음 접하면서 React의 장점을 극대화할 수 있는 프레임워크라는 점이 인상적이었음. 서버 사이드 렌더링과 정적 사이트 생성 기능이 매우 유용했음. 스타일링과 폰트/이미지 최적화 기능도 프로젝트의 유지 보수성을 높이는 데 큰 도움이 되었음.

팀 프로젝트에 배정되었고, 팀원들과 협력하여 좋은 성과를 내고자 함. 서로의 강점을 활용하고 부족한 부분을 보완하며 프로젝트를 성공적으로 마무리하는 것이 목표임. Next.js를 활용한 첫 프로젝트이지만 앞으로 더 많은 기능들을 활용해보고 싶음. 사전 학습을 통해 웹 개발의 전반적인 흐름을 이해하는 데 큰 도움이 되었음.

Next.js의 다양한 기능을 더 깊이 있게 학습하여 복잡한 프로젝트에서도 효율적으로 활용할 수 있도록 해야겠다고 느꼈음. 스타일링 및 최적화 기법을 더 연습하여 유지 보수성을 높일 계획임. 공식 문서와 다양한 리소스를 통해 Next.js의 여러 기능을 체계적으로 학습할 필요가 있음.



#유데미 #udemy #웅진씽크빅 #스나이퍼팩토리 #인사이드아웃 #미래내일일경험 #프로젝트캠프 #부트캠프 #Next.js #프론트엔드개발자양성과정 #개발자교육과정