

Node.js 소개

Node.js는 구글 크롬의 V8 JavaScript 엔진을 기반으로 한 오픈 소스 JavaScript 런타임 환경입니다. 서버 측 프로그래밍, 실시간 웹 애플리케이션, 네트워크 도구 등 다양한 용도로 사용되며, 비동기 이벤트 기반 I/O 모델을 사용하여 효율적인 애플리케이션 개발이 가능합니다.

 **작성자: 세현 박**



Express.js

특징

Express.js는 Node.js를 위한 가장 널리 사용되는 웹 프레임워크입니다. 간단하고 유연한 API를 제공하며, 라우팅, 미들웨어, 템플릿 엔진 등의 다양한 기능을 포함하고 있습니다.

장단점

Express.js는 빠르고 간단한 개발을 가능하게 하지만, 복잡한 프로젝트에서는 구조화와 확장성에 어려움이 있을 수 있습니다.

사용 예

Express.js는 웹 API 개발, 단일 페이지 애플리케이션(SPA), 모바일 앱 백엔드 등 다양한 웹 애플리케이션 개발에 사용됩니다.

Koa.js

1

특징

Koa.js는 Express.js의 제작자들이 만든 차세대 Node.js 웹 프레임워크로, 보다 가벼운 크기와 강력한 미들웨어 시스템을 특징으로 합니다.

2

장단점

Koa.js는 학습 곡선이 다소 높지만, 깨끗한 코드와 유연성을 제공합니다. 단, 확장성이 상대적으로 부족할 수 있습니다.

3

사용 예

Koa.js는 REST API, 실시간 웹 애플리케이션, 마이크로서비스 아키텍처 등에 적합합니다.



teerue



Hapi.js

특징

Hapi.js는 Node.js를 위한 강력한 프레임워크로, 보안, 테스트, 구조화 등 다양한 기능을 제공합니다.

장단점

Hapi.js는 구조화된 코드와 명확한 API를 제공하지만, 학습 곡선이 다소 높은 편입니다.

사용 예

Hapi.js는 보안이 중요한 엔터프라이즈 애플리케이션, API 서버, 마이크로 서비스 등에 적합합니다.

주요 기능

플러그인 시스템, 인증, 캐싱, 테스트 등 웹 애플리케이션 개발에 필요한 다양한 기능을 제공합니다.

Nest.js



TypeScript

Nest.js는 TypeScript를 기반으로 하며, 엔터프라이즈급 애플리케이션 개발에 적합합니다.



확장성

Nest.js는 마이크로서비스 아키텍처 구현에 적합하며, 대규모 애플리케이션 개발을 지원합니다.



모듈성

Nest.js는 모듈 시스템을 통해 코드의 재사용성과 유지보수성을 높입니다.



테스팅

Nest.js는 단위 테스트, 통합 테스트, E2E 테스트 등 다양한 테스트 기능을 제공합니다.





LoopBack

1

API 개발

LoopBack은 API 개발을 위한 강력한 도구를 제공합니다. 코드 생성, 문서화, 테스트 등의 기능을 포함합니다.

2

데이터 통합

LoopBack은 SQL, NoSQL, REST 등 다양한 데이터 소스와 연동할 수 있는 기능을 제공합니다.

3

확장성

LoopBack은 마이크로서비스 아키텍처 구현에 적합하며, 대규모 프로젝트에서 확장성을 보장합니다.

Sails.js

특징

Sails.js는 MVC 패턴을 기반으로 하는 Node.js 웹 프레임워크로, 데이터베이스 연동, 실시간 통신,

