



Node.js란 무엇인가

Node.js는 Chrome V8 JavaScript 엔진을 기반으로 하는 오픈 소스 JavaScript 런타임 환경입니다. 웹 브라우저 밖에서도 JavaScript를 실행할 수 있게 해주며, 빠르고 확장 가능한 네트워크 애플리케이션 개발에 사용됩니다.

Node.js의 특징 및 장점

1

비동기 I/O

I/O 작업을 처리하는 동안 다른 요청을 처리할 수 있어 높은 처리량을 제공합니다.

2

단일 스레드 이벤트 루프

이벤트 기반 아키텍처를 사용하여 스레드 관리 없이도 확장성 있는 애플리케이션을 만들 수 있습니다.

3

풍부한 에코시스템

수많은 오픈 소스 라이브러리와 프레임워크를 제공하여 개발 속도를 높일 수 있습니다.



Node.js 아키텍처 이해하기

1

이벤트 루프

비동기 I/O 작업을 처리하고 이벤트를 관리하는 단일 스레드 기반의 메인 루프입니다.

2

워커 스레드

CPU 집약적인 작업을 병렬로 처리하여 응답성을 높이는 역할을 합니다.

3

비동기 I/O

시스템 호출을 분리하여 I/O 작업과 계산 작업을 동시에 처리할 수 있습니다.



Node.js의 비동기 처리방식

콜백 함수

비동기 작업이 완료되면 실행되는 함수입니다. 비동기 처리의 핵심 메커니즘입니다.

이벤트 발생기

이벤트를 발생시키고 이벤트 리스너를 등록하여 비동기 처리를 구현합니다.

Promises/Async-Await

비동기 작업의 흐름을 더 잘 제어할 수 있는 최신 기능입니다.



Node.js와 JavaScript 연계성

단일 언어 사용

Node.js를 통해 브라우저와 서버에서 모두 JavaScript를 사용할 수 있습니다.

풍부한 라이브러리

기존 JavaScript 기반 라이브러리를 재사용할 수 있어 개발 속도가 빨라집니다.

동일한 개발 스킬

서버 측과 클라이언트 측 모두에서 동일한 스킬을 활용할 수 있습니다.

생산성 향상

풍부한 에코시스템과 익숙한 언어로 인해 생산성이 크게 향상됩니다.

Node.js 활용 사례 살펴보기

1

웹 서버

Node.js는 Express.js와 같은 웹 프레임워크를 통해 강력한 웹 서버를 구축할 수 있습니다.

2

실시간 애플리케이션

Socket.IO와 같은 실시간 통신 라이브러리를 이용해 채팅, 게임 등 실시간 애플리케이션을 개발할 수 있습니다.

3

마이크로서비스

Node.js의 비동기 처리 능력을 활용해 확장성 높은 마이크로서비스 아키텍처를 구축할 수 있습니다.



Node.js의 확장성과 생태계

글로벌 확산

Node.js는 전 세계적으로 널리 채택되고 있는 인기 있는 기술입니다.

강력한 생태계

수많은 오픈 소스 라이브러리와 도구로 구성된 풍부한 생태계가 있습니다.

지속적인 발전

Node.js는 지속적으로 발전하고 있으며 새로운 기능과 성능 향상이 이루어지고 있습니다.





Node.js 개발 환경 구축하기

| | |
|------------|--|
| Node.js 설치 | 공식 웹사이트에서 운영체제에 맞는 Node.js 설치 패키지를 다운로드하여 설치합니다. |
| IDE 사용 | Visual Studio Code, WebStorm 등의 통합 개발 환경(IDE)을 활용하여 코딩을 진행합니다. |
| npm 패키지 관리 | NPM(Node Package Manager)을 사용하여 필요한 라이브러리와 도구를 설치하고 관리합니다. |
| 디버깅 및 테스트 | 브라우저 내장 개발자 도구 및 디버거, Jest 등의 테스트 프레임워크를 활용합니다. |