



Node.js

2020575009 김규태

Node.js란 무엇인가

- 서버 사이드 JavaScript 플랫폼

- - Node.js는 Chrome V8 JavaScript 엔진을 기반으로 서버에서 실행되는 JavaScript 런타임입니다.
- - 비동기 이벤트 기반 구조를 사용하여 효율적으로 네트워크 애플리케이션을 처리합니다.

- 1. 비동기 및 이벤트 기반
 - - 작업을 동시에 처리하며, 하나의 스레드로 많은 요청을 처리할 수 있습니다.
- 2. V8 엔진
 - - 빠른 JavaScript 코드 실행을 위해 Google 의 V8 엔진을 사용합니다.
- 3. 확장성
 - 큰 규모의 네트워크 애플리케이션을

Node.js의 장점

- - 높은 성능: 빠른 처리 속도
- - 단일 언어로의 개발: 프론트엔드와 백엔드 모두 JavaScript로 작성 가능
- - 큰 생태계: npm(Node Package Manager)을 통한 다양한 라이브러리 제공

Node.js의 단점

- - 단일 스레드의 한계: CPU 집중 작업에 어려움이 있음
- - 비동기 프로그래밍의 복잡성: 콜백 헬 문제 등이 발생할 수 있음
- - 미성숙한 API: 일부 서버 사이드 작업에서는 다른 언어에 비해 성숙하지 않음

주요 사용 사례

- 1. 실시간 애플리케이션
 - - 예: 채팅, 실시간 피드 업데이트
- 2. API 서버
 - - 예: RESTful API, GraphQL 서버
- 3. 데이터 스트리밍 애플리케이션
 - - 예: 미디어 스트리밍 서비스

주요 도구 및 라이브러리

- - Express.js: 간단하고 강력한 웹 프레임워크
- - Socket.io: 실시간 통신을 위한 라이브러리
- - PM2: 프로덕션 환경에서 프로세스를 관리하는 도구

- - 이벤트 루프: 단일 스레드로 비동기 작업을 처리하는 핵심 메커니즘
- - 콜백 함수와 프로미스: 비동기 프로그래밍을 가능하게 하는 주요 방식

- 간단한 서버 생성 코드:
- `const h1p = require('h1p');`
- `const server = h1p.createServer((req, res) => {`
- `res.statusCode = 200;`
- `res.setHeader('Content-Type', 'text/plain');`
- `res.end('Hello World\n');`
- `});`
- `server.listen(3000, () => {`

Node.js의 발전과 커뮤니티

- - npm을 통해 전 세계 개발자들이 만든 다양한 패키지에 접근 가능
- - 활발한 커뮤니티가 있어 다양한 리소스와 지원을 받을 수 있음

A close-up photograph of several ripe raspberries, showing their characteristic bumpy texture and deep red color. The raspberries are piled together, with some in sharp focus and others slightly blurred in the background. The lighting is soft, highlighting the individual drupelets.

Q&A
