



# Node.js 소개

Node.js는 Chrome V8 자바스크립트 엔진을 기반으로 한 오픈 소스 런타임 환경입니다. 서버 측 웹 애플리케이션 개발에 널리 사용되며, 비동기 I/O, 이벤트 기반 아키텍처, 그리고 광범위한 모듈 생태계로 인해 빠르고 효율적인 웹 서비스 구축이 가능합니다.

 작성자: 유진 최



# Node.js의 특징

## 비동기 I/O

Node.js는 입출력 작업을 비동기적으로 처리하여 병목 현상을 방지하고 효율성을 높입니다.

## 단일 스레드

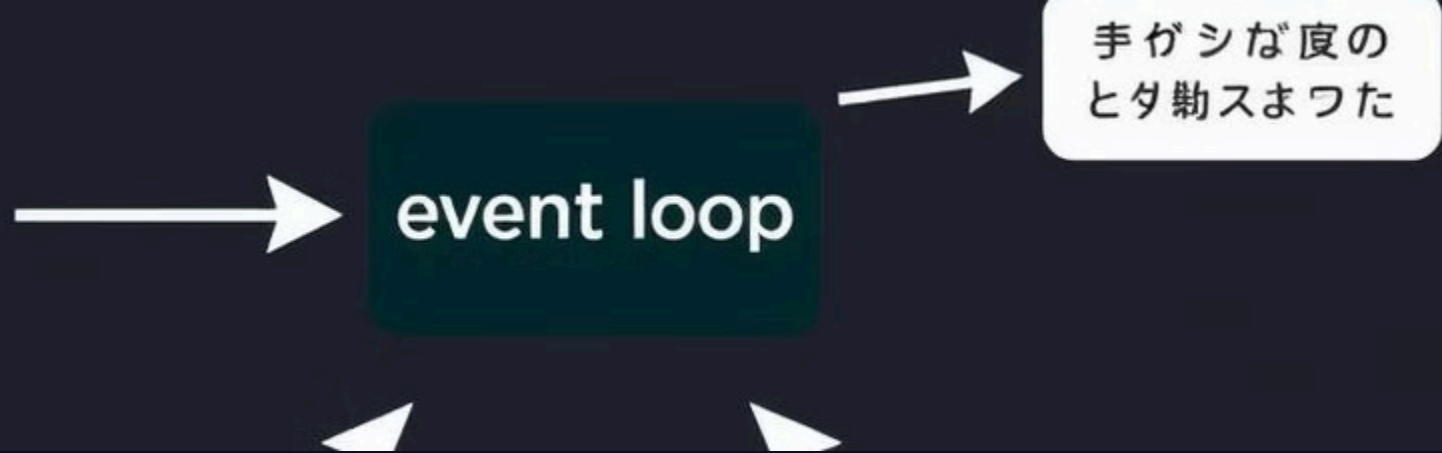
이벤트 루프를 통해 하나의 스레드로 운영되어 경량 및 확장성이 뛰어납니다.

## 확장성

단일 프로세스로 동작하여 수직/수평 확장이 쉽고, 웹 서버, API 서버, 실시간 채팅 등에 적합합니다.

## 빠른 속도

Google의 V8 엔진을 사용하여 빠른 자바스크립트 실행 속도를 보장합니다.



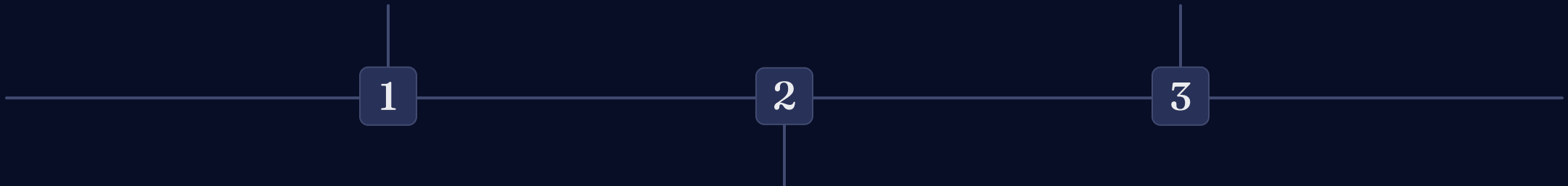
# 비동기 I/O 모델

## 입력 요청

클라이언트로부터 비동기 입력 요청이 들어옵니다.

## 비동기 처리

입출력 작업은 별도의 스레드에서 비동기적으로 처리됩니다.



## 이벤트 루프

이벤트 루프가 요청을 처리하고 비동기 작업을 스케줄링합니다.

# 이벤트 기반 아키텍처

## 이벤트 발생

클라이언트로부터 이벤트가 발생하면 이벤트 발생기에 의해 처리됩니다.

## 이벤트 큐

이벤트는 이벤트 큐에 저장되어 순차적으로 처리됩니다.

## 이벤트 리스너

이벤트 리스너가 이벤트 큐에서 이벤트를 가져와 처리합니다.



# Node.js의 주요 모듈



## HTTP 모듈

웹 서버 기능을 제공합니다.



## File 모듈

파일 시스템 작업을 수행합니다.



## Database 모듈

데이터베이스 연결 및 쿼리를 지원합니다.



## Event 모듈

이벤트 기반 프로그래밍을 가능하게 합니다.



1

Node.js 프로젝트에 필요한 패키지를 설치, 업데이트, 삭제할 수 있습니다.

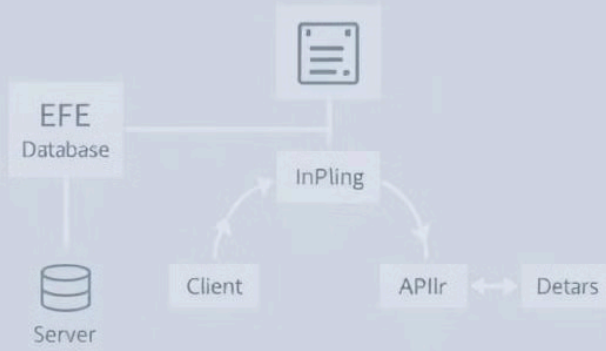
2

모듈의 버전을 관리하여 프로젝트의 안정성을 보장합니다.

## 3

수많은 오픈 소스 모듈을 제공하여 개발 생산성을 높입니다.

# Node.js 웹 프레임워크 (Express)



1

라우팅

URL 경로에 따라 적절한 처리 함수를 실행합니다.

2

미들웨어

요청/응답 처리 과정에 필요한 기능을 독립적인 모듈로 제공합니다.

3

템플릿 엔진

동적 HTML 페이지 생성을 지원합니다.

# Node.js 활용 사례 및 미래 전망

## 웹 애플리케이션

확장성과 실시간 처리 능력으로 웹 애플리케이션 개발에 널리 사용됩니다.

## IoT 및 실시간 데이터

IoT 기기와 실시간 데이터 처리에 적합한 기술 스택을 제공합니다.

## 마이크로서비스

경량성과 확장성으로 마이크로서비스 아키텍처 구축에 유용합니다.