

# 필기노트

2021년 1월 11일 월요일    오후 7:55

## 1. 노드의 정의

Node.js는 크롬 V8 자바스크립트 엔진으로 빌드된 자바스크립트 런타임이다.

옛날의 Javascript는 html 위에서만 실행, Node.js로 브라우저, html 종속성 탈출함

자바스크립트는 서버의 역할도 수행할 수 있음

### - 런타임

특정 언어로 만든 프로그램들을 실행할 수 있게 해주는 가상 머신(크롬 V8엔진)

### - 자바 스크립트 끝내기 : ko.javascript.info

위 홈페이지를 보고 사용이 어려우면 문법 문제가 아닌 프로그래밍 사고력 문제

### - 노드의 내부 구성은 V8과 libuv를 내부적으로 포함, 대부분 C, Cpp로 작성됨

Libuv : 노드 특성인 이벤트 기반, 논 블로킹 I/O 모델을 구현한 라이브러리

싱글스레드 + 비동기 I/O임

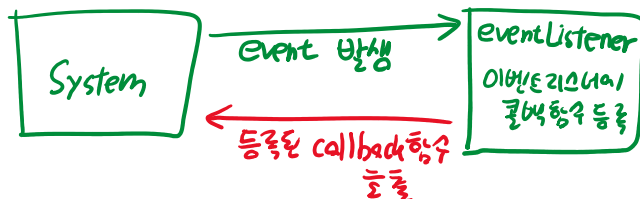
## 2. 노드의 특성

### - 이벤트 기반

이벤트 : 이벤트가 발생할 때 미리 지정해둔 작업을 수행하는 방식

이벤트 리스너 : 이벤트를 등록하는 함수

콜백 함수 : 이벤트가 발생했을 때 실행될 함수

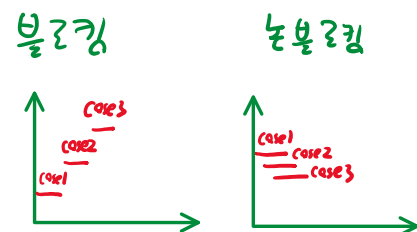


예시  
브라우저에naver.com 입력

### - 논블로킹 I/O

논블로킹 : 오래 걸리는 함수를 백그라운드로 보내 다음 코드를 먼저 실행, 나중에 오래 걸리는 함수를 실행

노드는 블로킹 & 동기 와 논블로킹 & 비동기만 사용



블로킹&동기 -> 순서대로 실행

논블로킹&비동기 -> 순서대로 실행 X

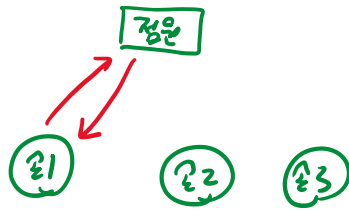
### - 멀티 스레드와 멀티 프로세싱

프로세스 : OS에서 할당하는 작업의 단위, 프로세스 간 자원공유 X

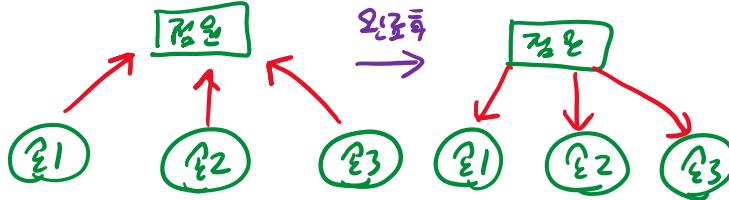
스레드 : 프로세스 내에서 실행되는 작업단위, 부모 프로세스의 자원 공유

Node V14 전까지는 멀티 스레드이지만, 직접 다룰 수 있는 스레드는 하나여서 싱글 스레드였음

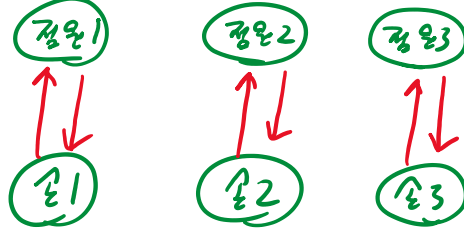
- 싱글 스레드



- 싱글 스레드 & 논블로킹



- 멀티 스레드



Node.js는 주로 싱글 스레드, 멀티 프로세싱으로 사용

- 멀티 프로세싱 : 체인점을 많이 내어 손님을 체인점에 분배하는 형식
- 멀티 스레딩 : 가게 내에 여러 점원이 여러 손님을 대응하는 형식

멀티 스레딩	멀티 프로세싱
하나의 프로세스 안에서 여러 개의 스레드 사용	여러 개의 프로세스 사용
CPU 작업이 많을 때 사용	I/O 요청이 많을 때 사용
프로그래밍이 어려움	프로그래밍이 비교적 쉬움

### 3. 노드의 역할

- 서버로서의 노드

서버 : 네트워크를 통해 클라이언트에 정보, 서비스 제공

클라이언트 : 서버에 요청을 보내는 주체

노드 != 서버, 서버 모듈을 제공한다

- 노드 서버의 장단점 = Node, Js의 장단점과 비슷

장점	단점
멀티 스레드 방식에 비해 자원을 적게 사용함	싱글 스레드라 CPU 코어를 하나만 사용
I/O 작업이 많은 서버로 적합	CPU 작업이 많은 서버로 부적합 → <b>안드로이드, iOS 등</b>
멀티스레드 방식보다 쉬움	하나뿐인 스레드가 멈추지 않도록 관리해야 함
웹 서버가 내장되어 있음	서버 규모가 커졌을 때 서버 관리가 어려움 → <b>대부분 서버</b>
자바스크립트를 사용함 <b>시장 ↑, 안정적</b>	어중간한 성능 <b>대산계, 게임 등 X</b>
JSON 형식과 호환하기 쉬움	

- 서버 외의 노드

웹, 모바일, 데스크탑 앱에서도 사용

- 웹 프레임워크 : Angular, React, Vue, Methor 등
- 모바일 앱 프레임워크 : React Native
- 데스크탑 개발 도구 : Electron(Atom, Slack, Vscode, Discord 등 제작)