1. 노드 시작하기

1.1 핵심 개념 이해하기

1.1.1 서버

- 서버 : 네트워크를 통해 클라이언트에 정보나 서비스를 제공하는 컴퓨터 또는 프로그램

- 클라이언트 : 요청을 보내는 주체로 브라우저일 수도 있고, 데스크톱 프로그램일 수도 있고,   
모바일 앱일 수도 있고, 다른 서버에 요청을 보내는 서버일 수도 있음.

- 서버라고 해서 요청에 대한 응답만 하는 것은 아님. 다른 서버에 요청을 보낼 수도 있으며,   
이때는 요청을 보낸 서버가 클라이언트 역할을 함.

1.1.2 자바스크립트 런타임

- 노드는 자바스크립트 런타임이다. (자바스크립트 실행기)

- 런타임은 특정 언어로 만든 프로그램들을 실행할 수 있는 환경

- 노드는 V8과 더불어 libuv라는 라이브러리를 사용  
(libuv 라이브러리는 노드의 특성인 이벤트 기반, 논 블로킹 I/O 모델을 구현하고 있음)

1.1.3 이벤트 기반

- 이벤트 기반(event-driven)이란, 이벤트가 발생할 때 미리 저장해둔 작업을 수행하는 방식

- 이벤트 기반 시스템에서는 특정 이벤트가 발생할 때 무엇을 할지 미리 등록해두어야 함.  
이를 이벤트 리스너(event listener)에 콜백(callback) 함수를 등록한다고 표현

- 노드도 이벤트 기반 방식으로 동작하므로, 이벤트가 발생하면 이벤트 리스너에 등록해둔 콜백 함수를 호출. 발생한 이벤트가 없거나 발생했던 이벤트를 다 처리하면, 노드는 다음 이벤트가 발생할 때까지 대기

- 이벤트 루프 : 이벤트 발생 시 호출할 콜백 함수들을 관리, 호출된 콜백 함수의 실행 순서 결정

- 백그라운드 : setTimeout 같은 타이머나 이벤트 리스너들이 대기하는 곳

- 테스크 큐 (콜백 큐) : 이벤트 발생 후, 백그라운드에서는 테스크 큐로 타이머나 이벤트 리스너들의 콜백 함수들을 보냄

1.1.4 논 블로킹 I/O

- 논 블로킹 : 이전 작업이 완료될 때까지 대기하지 않고 다음 작업을 수행함  
(논 블로킹과 동시가 같은 의미가 아님)

블로킹 : 이전 작업이 끝나야만 다음 작업을 수행함

- 노드에서는 동기와 블로킹이 유사, 비동기와 논 블로킹이 유사

1.1.5 싱글 스레드

- 노드가 채택하고 있는 방식 : 싱글 스레드, 논 블로킹 모델

- 노드가 싱글 스레드로 동작하지 않는 경우 : 스레드풀, 워커 스레드