

강의 구성

NEST JS 소개

CRUD 앱 구성하며
NEST JS 기본 구성 배우기

PG DB & TypeORM이용해서 앱 구성

인증 처리 추가(JWT)

권한 처리 추가(PASSPORT)

로그 남기기

Nest JS란 ?

Nest JS 란?

Nest (NestJS)는 효율적이고 확장 가능한 Node.js 서버 측 애플리케이션을 구축하기 위한 프레임 워크입니다. 프로그래시브 JavaScript를 사용하고 TypeScript로 빌드되고 완벽하게 지원하며 (개발자가 순수 JavaScript로 코딩할 수 있음) OOP (Object Oriented Programming), FP (Functional Programming) 및 FRP (Functional Reactive Programming) 요소를 사용할 수 있게 해줍니다.

Nest JS 는 내부적으로 어떻게 구성되었나요?

내부적으로 Nest는 Express (기본값)와 같은 강력한 HTTP 서버 프레임 워크를 사용하며 선택적으로 Fastify를 사용하도록 구성 할 수도 있습니다!

Nest는 이러한 공통 Node.js 프레임 워크 (Express / Fastify) 위에 추상화 수준을 제공하지만 API를 개발자에게 직접 노출합니다. 이를 통해 개발자는 기본 플랫폼에서 사용할 수 있는 수많은 타사 모듈을 자유롭게 사용할 수 있습니다.

Nest JS 의 철학

Node (및 서버 측 JavaScript)를 위한 훌륭한 라이브러리, 도우미 및 도구가 많이 존재하지만 이들 중 어느 것도 아키텍처의 주요 문제를 효과적으로 해결하지 못합니다.

Nest는 개발자와 팀이 고도로 테스트 가능하고 확장 가능하며 느슨하게 결합되고 유지 관리가 쉬운 애플리케이션을 만들 수 있는 즉시 사용 가능한 애플리케이션 아키텍처를 제공합니다. 이 아키텍처는 Angular에서 크게 영감을 받았습니

Nest JS 공식 문서

<https://docs.nestjs.com/>

Node js 설치하기

Nodejs 설치하기

<https://nodejs.org/en/>

Nest JS CLI 설치하기

Nest JS CLI로 Nest JS 시작하기

NestJS를 이용해서 프로젝트를 시작할 때 Nest CLI를 이용하면 간단히 프로젝트를 시작할 수 있습니다.

Nest CLI를 이용해서 아래의 명령어를 작성하면

새 프로젝트 디렉터리가 생성되고 초기 핵심 Nest 파일 및 지원 모듈로 디렉터리가 채워져 프로젝트의 기존 기본 구조가 생성됩니다.

```
$ npm i -g @nestjs/cli  
$ nest new project-name
```

이렇게 한 후 NestJS가 잘 설치되어있는지 보려면 "nest --version" 명령어를 이용해주시면 됩니다.