

NestJS

Framework for building efficient, scalable Node.js server-side applications

Contents

1. Philosophy.
2. Fundamentals.
3. Built-in functionalities.
4. Comparison.

Philosophy

Thanks to Node.js, Javascript became the most popular language in web development and plenty of frameworks have developed.

None of them effectively solve the main problem of - Architecture

Fundamentals: Angular-like Architecture

01

Module

02

Controller

03

Provider

04

Middleware

Caching

```
$ npm install --save cache-manager
```

```
import { CacheModule, Module } from '@nestjs/common';  
import { AppController } from './app.controller';
```

```
@Module({  
  imports: [CacheModule.register()],  
  controllers: [AppController],  
})  
export class ApplicationModule {}
```

```
import { CacheInterceptor } from '@nestjs/common';  
  
@Controller()  
@UseInterceptors(CacheInterceptor)  
export class AppController {  
  @Get()  
  findAll(): string[] {  
    return [];  
  }  
}
```

Performance

```
$ npm i --save @nestjs/platform-fastify
```

```
import {  
  FastifyAdapter,  
  NestFastifyApplication,  
} from '@nestjs/platform-fastify';  
  
const app = await NestFactory.create<NestFastifyApplication>(  
  AppModule,  
  new FastifyAdapter()  
);  
  
await app.listen(3000);
```

Microservice

```
$ npm i --save @nestjs/microservices
```

```
import { Transport, MicroserviceOptions } from '@nestjs/microservices';

const app = await NestFactory.createMicroservice<MicroserviceOptions>(
  AppModule,
  {
    transport: Transport.TCP,
  },
);

app.listen(() => console.log('Microservice is listening'));
```

Swagger

```
$ npm install --save @nestjs/swagger fastify-swagger
```

```
const app = await NestFactory.create(AppModule);

const options = new DocumentBuilder()
  .setTitle('Cats example')
  .setDescription('The cats API description')
  .setVersion('1.0')
  .addTag('cats')
  .build();

const document = SwaggerModule.createDocument(app, options);

SwaggerModule.setup('api', app, document);
```


NestJS vs Loopback.io

- Typescript support
- OpenAPI specification
- GraphQL compatible
- Similar concepts (Controller, Interceptor, ...)
- Multiple connectors for various databases.
- CLI-generated code
- Community still young

NestJS vs Loopback.io

Efficient, scalable oriented

Faster than express.js

Free to use external libs

Lack of CLI support

29.2k star, 2.6k fork

Fast development oriented

Slower than express.js

Restricted to use external libs

Strong CLI support

13.2k star, 1.2k fork

The background features a dark blue gradient with several overlapping, semi-transparent geometric shapes. These include diagonal bands of a lighter blue and purple hue, and a series of parallel white lines that create a sense of depth and movement. The overall aesthetic is modern and tech-oriented.

Thank you