

Nest

Introducción al framework Nest

CertiDevs

Índice de contenidos

1. ¿Qué es Nest?	1
2. Instalación y configuración del entorno de desarrollo	1
3. Estructura de proyectos Nest	2
4. Ejecución y desarrollo de la aplicación	2
5. Compilación y despliegue	3

1. ¿Qué es Nest?

Nest es un framework para construir aplicaciones de servidor eficientes, escalables y de nivel empresarial en **Node.js**.

[Sitio web oficial](#)

Está construido con **TypeScript** y utiliza la arquitectura de módulos para organizar el código de manera eficiente.

NestJS se inspira en Angular y adopta muchas de sus buenas prácticas, lo que facilita la integración entre ambos.

Características clave de Nest:

- Programación modular y orientada a objetos
- Inyección de dependencias y control de la inversión
- Decoradores personalizados
- Soporte para múltiples protocolos (HTTP, gRPC, WebSockets, etc.)
- Integración con bases de datos y ORMs
- Soporte para pruebas unitarias y de integración

[Documentación oficial](#)

2. Instalación y configuración del entorno de desarrollo

Necesitarás tener instalado Node.js en tu sistema.

Asegúrate de tener una versión compatible con NestJS (consulta la documentación oficial para obtener más detalles).

También necesitarás instalar el gestor de paquetes **npm** o **yarn**.

Instala el CLI de NestJS de forma global en tu máquina usando el siguiente comando:

```
npm install -g @nestjs/cli
```

Creación de un proyecto NestJS básico:

Para crear un nuevo proyecto NestJS, utiliza el siguiente comando:

```
nest new my-nestjs-project
```

Este comando crea una nueva carpeta llamada **my-nestjs-project** con la estructura de directorios y archivos necesarios para un proyecto básico.

A continuación, navega al directorio del proyecto y ejecuta el siguiente comando para instalar las dependencias:

```
cd my-nestjs-project
npm install
```

3. Estructura de proyectos Nest

Un proyecto Nest típico tiene la siguiente estructura:

```
my-nestjs-project/
|-- src/
|   |-- app.controller.ts
|   |-- app.module.ts
|   |-- app.service.ts
|   |-- main.ts
|-- test/
|-- .gitignore
|-- nest-cli.json
|-- package.json
|-- tsconfig.build.json
|-- tsconfig.json
```

- **src/**: Contiene el código fuente de la aplicación.
- **app.controller.ts**: Un controlador básico que maneja las rutas y acciones HTTP.
- **app.module.ts**: El módulo raíz que organiza las dependencias y funcionalidades de la aplicación.
- **app.service.ts**: Un servicio básico que contiene la lógica de negocio.
- **main.ts**: El punto de entrada de la aplicación.
- **test/**: Contiene los archivos de prueba.
- **nest-cli.json**: Configuración del CLI de NestJS.
- **package.json**: Contiene las dependencias del proyecto y scripts.
- **tsconfig.build.json** y **tsconfig.json**: Configuración de TypeScript.

4. Ejecución y desarrollo de la aplicación

Para ejecutar la aplicación en modo de desarrollo, usa el siguiente comando:

```
npm run start:dev
```

Esto iniciará el servidor en el puerto **3000** (por defecto) y se reiniciará automáticamente cada vez que realices cambios en el código.

5. Compilación y despliegue

Para compilar la aplicación, ejecuta el siguiente comando:

```
npm run build
```

Esto generará el código JavaScript compilado en la carpeta **dist/**.

Para ejecutar la aplicación compilada, usa el siguiente comando:

```
npm start
```