

Введение в Nest.js

Интерактивный workshop

Игорь Антонов, 17.12.2024

Привет 🙌

Игорь Антонов

- TeamLead в Т-Банк;
- Помогаю делать конференции: Podlodka React/Java/TechLead Crew;
- Веду канал на YouTube и Telegram – https://youtube.com/@antonov_i и <https://t.me/antonovjs>
- Спикер, лектор (HTML Academy, ИТМО, Политех)



JavaScript и TypeScript

О чём пойдёт речь

План лекции

О чём пойдёт речь

План лекции

1. Backend на JavaScript;

О чём пойдёт речь

План лекции

1. Backend на JavaScript;
2. Зачем нужны фреймворки;

О чём пойдёт речь

План лекции

1. Backend на JavaScript;
2. Зачем нужны фреймворки;
3. Express, Nest.js и Fastify;

О чём пойдёт речь

План лекции

1. Backend на JavaScript;
2. Зачем нужны фреймворки;
3. Express, Nest.js и Fastify;
4. Основные преимущества и недостатки Nest.js;

О чём пойдёт речь

План лекции

1. Backend на JavaScript;
2. Зачем нужны фреймворки;
3. Express, Nest.js и Fastify;
4. Основные преимущества и недостатки Nest.js;
5. Практика (CLI, Layers, CRUD, Validation, Interceptors, JWT)

Backend на JS



Backend на JS



- Backend и Node.js. Практики, инструменты;



Backend на JS

- Backend и Node.js. Практики, инструменты;
- Низкоуровневый API от Node.js;



Backend на JS

- Backend и Node.js. Практики, инструменты;
- Низкоуровневый API от Node.js;
- Бойлерплейт. Сложно поддерживать, ещё сложней написать;



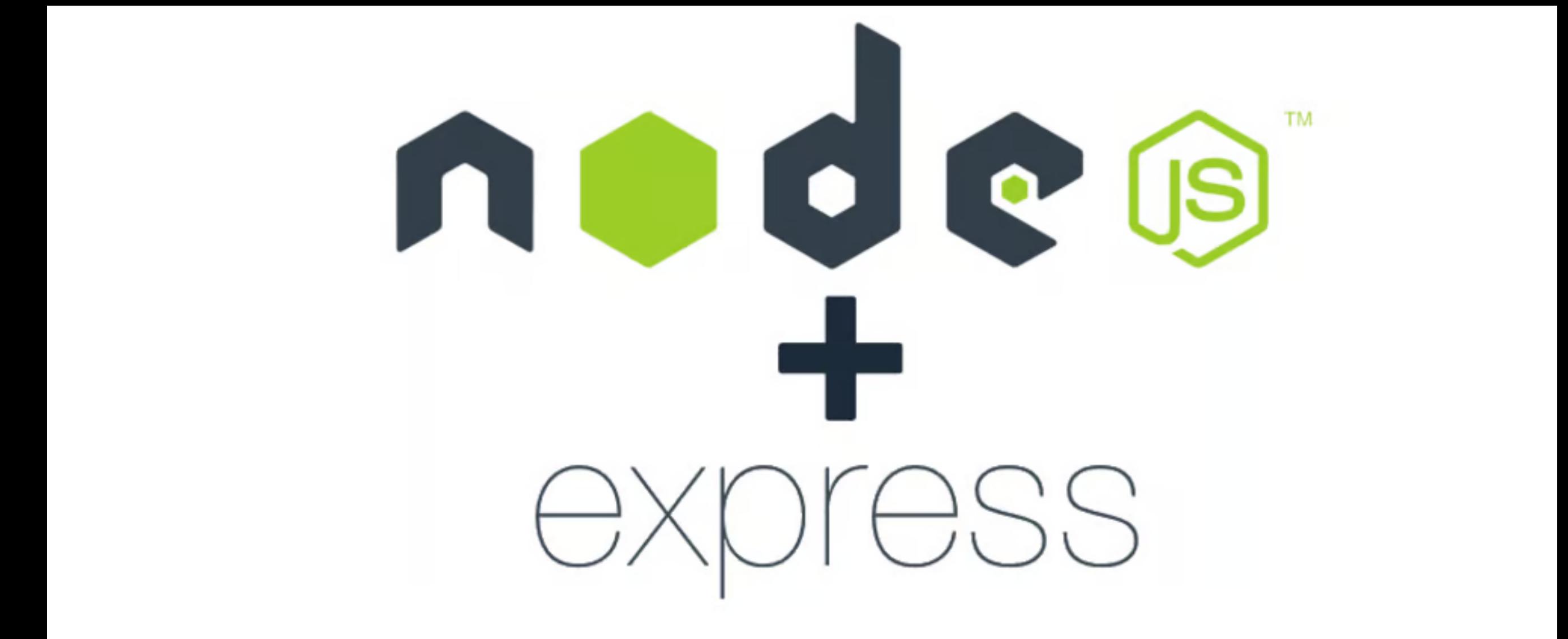
Backend на JS

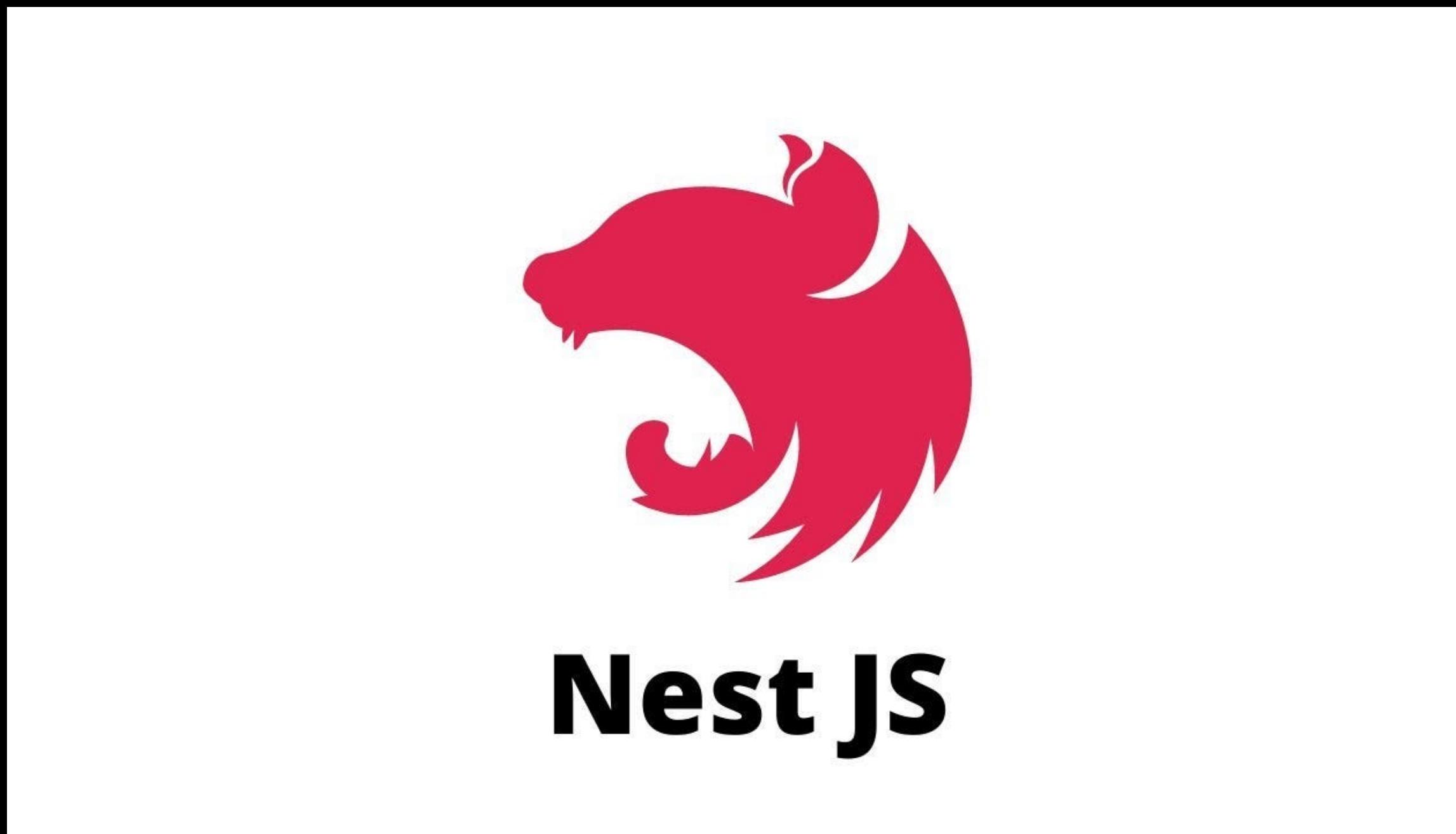
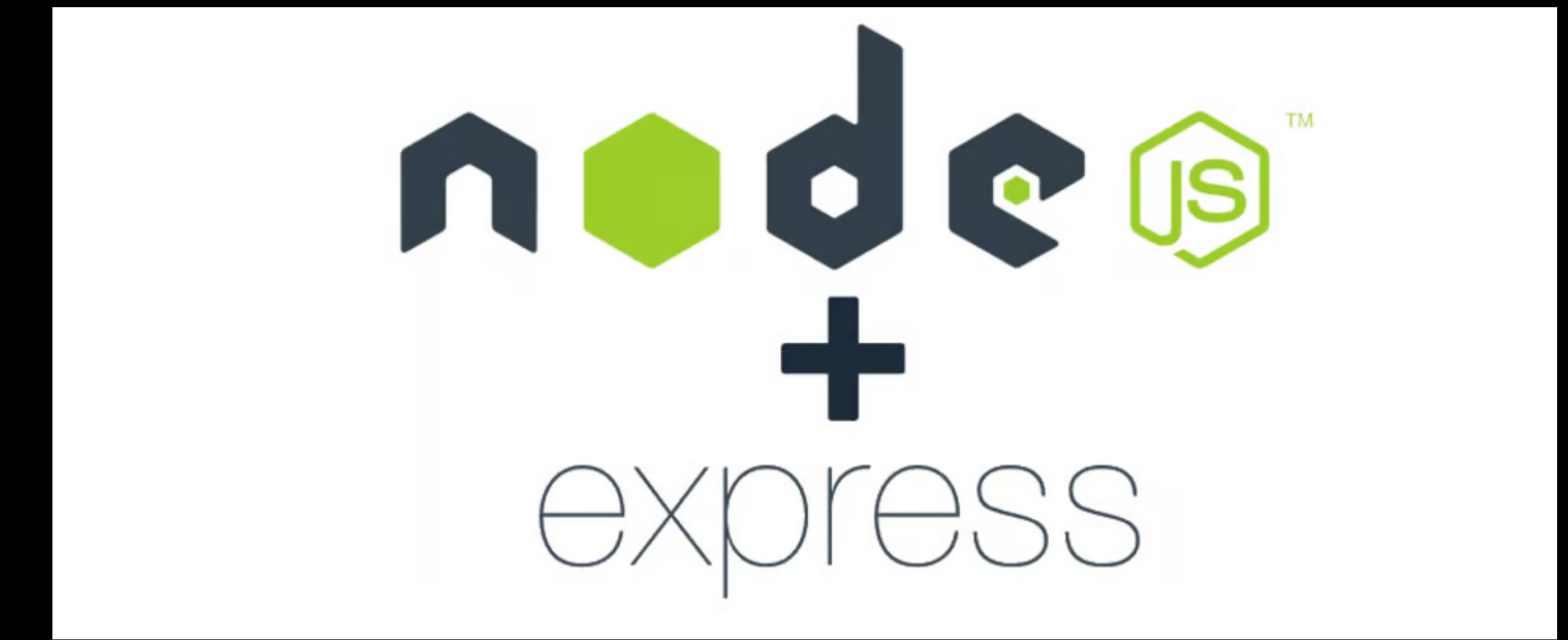


- Backend и Node.js. Практики, инструменты;
- Низкоуровневый API от Node.js;
- Бойлерплейт. Сложно поддерживать, ещё сложней написать;
- Фреймворки;









Express

Фреймворки для Node.js

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;
- Минималистичный;

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;
- Минималистичный;
- Чертовски популярный;

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;
- Минималистичный;
- Чертовски популярный;
- Хорошая производительность;

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;
- Минималистичный;
- Чертовски популярный;
- Хорошая производительность;
- Полная свобода действий;

Express

Фреймворки для Node.js

- Один из первых фреймворков для Node.js;
- Минималистичный;
- Чертовски популярный;
- Хорошая производительность;
- Полная свобода действий;
- Прост в обучении.

Fastify

Фреймворки для Node.js

Fastify

Фреймворки для Node.js

- Высокопроизводительный фреймворк, ориентирован на скорость;

Fastify

Фреймворки для Node.js

- Высокопроизводительный фреймворк, ориентирован на скорость;
- Прямая замена Express;

Fastify

Фреймворки для Node.js

- Высокопроизводительный фреймворк, ориентирован на скорость;
- Прямая замена Express;
- Современная кодовая база;

Fastify

Фреймворки для Node.js

- Высокопроизводительный фреймворк, ориентирован на скорость;
- Прямая замена Express;
- Современная кодовая база;
- Развитая экосистема инструментов;

Fastify

Фреймворки для Node.js

- Высокопроизводительный фреймворк, ориентирован на скорость;
- Прямая замена Express;
- Современная кодовая база;
- Развитая экосистема инструментов;
- Не навязывает архитектуру.

Nest

Фреймворки для Node.js

Nest

Фреймворки для Node.js

- Прогрессивный фреймворк;

Nest

Фреймворки для Node.js

- Прогрессивный фреймворк;
- Ориентирован на сложные и большие приложения;

Nest

Фреймворки для Node.js

- Прогрессивный фреймворк;
- Ориентирован на сложные и большие приложения;
- Архитектура;

Nest

Фреймворки для Node.js

- Прогрессивный фреймворк;
- Ориентирован на сложные и большие приложения;
- Архитектура;
- Более высокий порог входа.

Nest.js

Nest

Введение

Nest

Введение

- Первая версия вышла в 2016 году;

Nest

Введение

- Первая версия вышла в 2016 году;
- Популярность. 68K+ звёзд на GitHub 

Nest

Введение

- Первая версия вышла в 2016 году;
- Популярность. 68K+ звёзд на GitHub 
- Фреймворк для корпоративных приложений

Nest

Введение

- Первая версия вышла в 2016 году;
- Популярность. 68K+ звёзд на GitHub 
- Фреймворк для корпоративных приложений
 - Базовая архитектура;

Nest

Введение

- Первая версия вышла в 2016 году;
- Популярность. 68K+ звёзд на GitHub 
- Фреймворк для корпоративных приложений
 - Базовая архитектура;
 - Поддержка;

Nest

Введение

- Первая версия вышла в 2016 году;
- Популярность. 68K+ звёзд на GitHub 
- Фреймворк для корпоративных приложений
 - Базовая архитектура;
 - Поддержка;
 - Best Practices.

Nest

Плюсы

Nest

Плюсы

- **Правила и ограничения из коробки.** Следуй документации;

Nest

Плюсы

- **Правила и ограничения из коробки.** Следуй документации;
- **Экосистема.** Готовые решения. Встроенные инструменты;

Nest

Плюсы

- **Правила и ограничения из коробки.** Следуй документации;
- **Экосистема.** Готовые решения. Встроенные инструменты;
- **Архитектурные практики** (Laravel, Symfony, ASP .NET Core);

Nest

Плюсы

- **Правила и ограничения из коробки.** Следуй документации;
- **Экосистема.** Готовые решения. Встроенные инструменты;
- **Архитектурные практики** (Laravel, Symfony, ASP .NET Core);
- **Готовые абстракции.** Абстракции доступны из коробки;

Nest

Плюсы

- **Правила и ограничения из коробки.** Следуй документации;
- **Экосистема.** Готовые решения. Встроенные инструменты;
- **Архитектурные практики** (Laravel, Symfony, ASP .NET Core);
- **Готовые абстракции.** Абстракции доступны из коробки;
- **Не пытается заменить Express / Fastify.** Инкапсулирует более низкоуровневые фреймворки.

Node.js

TypeScript

Express

Абстракции

Контроллеры

Сервисы

Обработка
ошибок

Inversion of Control

Nest

Минусы

Nest

Минусы

- **Порог входа выше.** По сравнению с Express / Fastify. Требуется опыт применения паттернов проектирования, ООП подход. Много времени на чтение документации.

Nest

Минусы

- **Порог входа выше.** По сравнению с Express / Fastify. Требуется опыт применения паттернов проектирования, ООП подход. Много времени на чтение документации.
- **Кастомизация.** Чем больше требуется кастомизировать поведение, тем сложней.

Практика



Step 01. Создаём проект

Step 02. CLI

Step 03. CRUD

Step 04. Тесты

Step 05. Валидация

Step 06. Интерсепторы

Step 07. Фильтр-исключений

Step 08. JWT

The End

