

# Intro to node.js

Volodya Valko

October 2015

## JavaScript

### Using JS on a front-end and back-end

Використання JS як на фронт-енді, так і на бек-енді зменшує невідповідності між середовищами, а також дозволяє розподілити деяку логіку між клієнтом та сервером (наприклад перевірку форм).

## JSON

Не потрібно кастити об'єкти з бек-енду щоб відправити на фронт і навпаки. Все має єдиний формат JSON.

### People already know it

JavaScript — це безперебійно найбільш популярна мова програмування, навіть люди, які не є програмістами можуть знати JS. А якщо навіть котрийсь брутальний бекендщик все життя писав на PHP чи Ruby, то node.js стане поштовхом для вивчення JS.

## Event-driven development

### Async code

Все побудовано на принципах асинхронності, ми нічого не чекаємо, а йдемо напролом

### No multithreading, deadlocks

Немає жодних проблем зв'язаних з багатопоточністю (але з'являються нові, та про це трохи пізніше)

### Single thread for your code

node.js виконує весь ваш код в одному потоці, а іншу роботу в багато потоків (тут має бути наочний приклад, який довго описувати)

## **Extremily Fast**

### **V8 Engine**

Побудований на швидкому V8 як і Chrome

### **Optimized for high concurrent environments**

Легко масштабувати до тисяч активих з'єднань. Дуже швидкий та ефективний. Може в десятки разів зменшити кількість серверних ресурсів в порівнянні з Ruby чи PHP

### **npm**

#### **Huge community with tons of great libraries**

Коли ви бачите ці три маленькі букви — уявляйте собі лавину — приблизно так розвивається дана спільнота. Все у вільному доступі і лежить на відстані однієї команди в терміналі

#### **Five versions of everything**

Жодних стандартів тут немає. Ваш досвід та враження від написання коду в більшості визначаються вибраними вами бібліотеками, фреймами та модулями

### **npm must-have modules**

#### **Asynchronous**

Всі модулі даного розділу покликані звільнити вас від 20 рівнів асинхронних викликів

#### **Web**

Фреймворки даного розділу спрощують життя і включають в себе template engine

#### **Testing**

Без фреймворків даного типу в JS далеко не заїдеш. Наша улюблена асинхронність та специфіка самого JS робить перевірки дебагом практично повністю неефективними

## Why I write this

Це доповідь для презентації під назвою 'Intro to node.js' яку ви можете знайти за посиланням. Також, ви можете знайти сирцевий код презентації та цього документу за посиланням.