

УДК 004.738.5:004.4Node.js

ББК 32.973.202-018.2

С91

**Сухов, Кирилл Константинович.**

С91 Node.js. Путеводитель по технологии / К. К. Сухов. — 2-е изд., эл. — 1 файл pdf : 418 с. — Москва : ДМК Пресс, 2023. — Систем. требования: Adobe Reader XI либо Adobe Digital Editions 4.5 ; экран 10". — Текст : электронный.

ISBN 978-5-89818-328-8

За последние несколько лет платформа Node.js стремительно повысила свой статус от экспериментальной технологии до основы для серьезных промышленных проектов. Тысячи программистов оценили возможность построения достаточно сложных, высоко нагруженных приложений на простом, элегантном и, самое главное, легковесном механизме.

Все эти скучные слова правдивы, но на самом деле не это главное. Прежде всего Node.js — это совершенно увлекательная и захватывающая вещь, с которой по-настоящему интересно работать!

Есть одна проблема — невозможно рассказывать про использование Node.js в отрыве от остальных технологий современной веб-разработки (и Highload-разработки). Я и не стал этого делать, дав обзор инструментов, без которых сегодня трудно обойтись. Прежде всего это чудесный язык JavaScript, и в книге рассказано о новинках в его последней и будущей спецификациях (EcmaScript 5 и 6). Кроме того, дается краткое введение в большинство связанных веб-технологий — от NoSQL-хранилищ данных (Memcached, MongoDB, Redis) до CSS-препроцессоров и MVC JavaScript-фреймворков. Конечно, эту книгу нельзя рассматривать как полноценный учебник по MongoDB, LESS или EcmaScript 6, Dart или CoffeeScript, но в ней дано основное представление об этих довольно интересных вещах, вполне достаточное для начала работы.

УДК 004.738.5:004.4Node.js

ББК 32.973.202-018.2

**Электронное издание на основе печатного издания:** Node.js. Путеводитель по технологии / К. К. Сухов. — Москва : ДМК Пресс, 2015. — 416 с. — ISBN 978-5-97060-164-8. — Текст : непосредственный.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

В соответствии со ст. 1299 и 1301 ГК РФ при устранении ограничений, установленных техническими средствами защиты авторских прав, правообладатель вправе требовать от нарушителя возмещения убытков или выплаты компенсации.

ISBN 978-5-89818-328-8

© Сухов К. К., 2015

© Оформление, издание, ДМК Пресс, 2015

# Содержание

<b>Благодарности</b> .....	<b>9</b>
<b>Вступление</b> .....	<b>10</b>
О Node.js.....	10
Об этой книге .....	10
<b>Предыстория</b> .....	<b>12</b>
Веб-страница, веб-сервис, веб-приложение.....	12
Асинхронность как необходимость .....	12
Решения – Twisted, Tornado и другие .....	14
<b>Странный язык – JavaScript</b> .....	<b>16</b>
Волшебные замыкания .....	17
Что такое замыкание? .....	17
Применение замыканий .....	19
ООП в JavaScript.....	21
Явление Node.....	26
<b>Хватит теории! Начало работы с Node.js</b> .....	<b>27</b>
Установка Node .....	27
Веб-сервер из пяти строк.....	29
Сайт на Node.js.....	32
<b>Node Core</b> .....	<b>37</b>
Как все работает? Event loop в Node.js .....	37
Глобальные объекты (Globals) .....	37
Global .....	38
Объект Console .....	38
Require и псевдоглобальные объекты.....	39
Процессы.....	40
Свойства и методы объекта Process .....	40
Метод process.nextTick().....	42
Процессы ввода/вывода.....	43
Signal Events .....	44
Child Process – параллелизм в Node.js.....	45
Понятие буфера .....	49
Таймеры .....	52
<b>События</b> .....	<b>54</b>
Слушаем!.....	54
Объект EventEmitter.....	56

## 4 ❖ Содержание

---

<b>Модули .....</b>	<b>58</b>
Пакетный менеджер npm.....	59
Создаем собственный модуль.....	63
А теперь по-взрослому – пишем Си++ Addons.....	70
<b>Работа с файлами.....</b>	<b>75</b>
В дебри файловой системы .....	75
Маленький полезный модуль – Path .....	81
Бродим по папкам.....	84
Работа с файлами – вот она, асинхронность .....	85
Читаем файлы.....	85
Watching Files.....	88
Потоки – унифицируем работу с источниками данных .....	90
ReadStream/WriteStream .....	90
Веб-сервер на потоках .....	93
<b>Сервер HTTP, и не только .....</b>	<b>95</b>
Создаем TCP-сервер .....	95
UDP – тоже полезный протокол .....	104
Переходим на прикладной уровень – реализация HTTP.....	106
IncomingMessage – входящий HTTP-запрос.....	107
ServerResponse .....	108
HTTP-клиент (грабим Центробанк) .....	110
HTTPS – шифруемся!.....	114
Запускаем HTTPS-сервер .....	115
И секретный клиент.....	115
<b>WebSockets – стандарт современного веба.....</b>	<b>118</b>
Браузер – веб-сервер. Надо что-то менять.....	118
WebSockets – окончательное решение? .....	120
Простой способ – модуль ws.....	121
Начинаем работу с ws.....	121
Реализация WebSocket-чата .....	123
Обмен бинарными данными .....	127
Socket.io – webSockets для пролетариата .....	129
Реальное время для всех! .....	129
Начинаем работать с socket.io .....	130
Простой чат на socket.io.....	132
Усложняем.....	134
Совершенствуем приложение – дополнительные возможности socket.io.....	136
Пространства имен.....	140
«Летучие» сообщения .....	141
Извещения (acknowledgements).....	142
Конфигурация .....	142

<b>Пирамиды судьбы – асинхронный поток выполнения и как с ним бороться .....</b>	<b>145</b>
Начинаем строить пирамиды.....	146
Долой анонимность! .....	150
Node.js control-flow .....	151
Async – берем поток исполнения в свои руки .....	153
Инструменты async .....	153
Control Flow средствами async.....	157
Живительный водопад (async.waterfall).....	161
<b>Node.js и данные. Базы данных .....</b>	<b>165</b>
<b>MySQL и Node.js .....</b>	<b>166</b>
Четыре буквы – CRUD .....	167
Preparing Queries .....	168
Чтение, обновление и удаление данных .....	169
Работа с пулом соединений .....	171
<b>ORM-система Sequelize .....</b>	<b>173</b>
Начинаем работать с Sequelize .....	173
CRUD на Sequelize .....	176
Связи .....	179
<b>NoSQL.....</b>	<b>182</b>
<b>NodeJS и Memcached .....</b>	<b>183</b>
Основы Memcached.....	183
Реализация .....	184
Создаем приложение .....	186
MemcacheDB – все-таки DB .....	190
<b>Redis – очень полезный овощ .....</b>	<b>192</b>
Redis – что он умеет? .....	192
Основы работы с Redis.....	193
Модуль Redis для Node.js .....	194
Хэши (Hashes) .....	196
Множества (Sets).....	199
Упорядоченные множества (Sorted Sets).....	201
Механизм Publish/Subscribe.....	202
Очередь сообщений с помощью Redis.....	205
<b>MongoDB: JavaScript – он везде! .....</b>	<b>208</b>
Для чего? .....	209
Основы работы с MongoDB .....	210
Native MongoDB.....	214

## 6 ❖ Содержание

Рики-тики-тави: Mongoose для MongoDB.....	219
Основы работы с Mongoose.....	220
CRUD по-мангустски.....	221
Сеттеры, геттеры и прочие приятные вещи .....	224
<b>Переходим на сторону клиента .....</b>	<b>226</b>
<b>Мыслим шаблонами .....</b>	<b>227</b>
Mustache – уса́тый и нелогичный .....	227
EJS – пришелец из мира RoR .....	232
Синтаксис EJS-шаблонов .....	233
Помощники (Helpers) .....	233
EJS и Node.js .....	234
Фильтры.....	237
Jade – нечто нефритовое.....	239
Начинаем работу с Jade .....	240
Include – собираем шаблон по частям.....	247
Примеси .....	248
<b>CSS-препроцессоры – решение проблем стиля .....</b>	<b>250</b>
LESS – больше, чем Sass.....	250
Вложенные блоки .....	251
Переменные .....	252
Операции и функции .....	253
Примеси .....	255
Расширения .....	257
Работаем с LESS в Node.js .....	257
Stylus.....	260
Возьмем CSS и отсечем лишнее .....	260
И тут примеси .....	262
Функции и переменные.....	263
Stylus и Node.js.....	267
<b>Поднимаем разработку на новый уровень .....</b>	<b>270</b>
Чего нам не хватает? .....	270
<b>Connect – middleware framework для node.js .....</b>	<b>272</b>
Что такое middleware? (и зачем все это?).....	272
Connect на практике.....	273
Статический сайт одной строчкой (почти) .....	275
Совершенствуем сайт .....	275
Пишем свое СПО .....	278
Еще немного Connect.....	280
<b>Веб-каркас для node (node.js web-framework'и) .....</b>	<b>282</b>
Что такое web-framework? .....	282

Express.....	283
Начало работы с Express .....	283
Закат солнца вручную.....	285
Подключаем шаблонизатор .....	289
Задействуем статику.....	291
Подключаем CSS .....	291
Разработка RESTful-приложения на основе Express.....	293
Немного о REST-архитектуре.....	293
Приступаем к реализации RESTful API .....	294
Подключаем источник данных .....	299
А теперь – на три буквы (на MVC).....	306
Немного об архитектуре MVC .....	306
Структурируем код .....	307
Добавляем новую сущность.....	313

## **Практика разработки приложений Node.js ..... 316**

Nodemon – друг разработчика.....	316
Отладка Node.js-приложений (debug-режим).....	320
Node Inspector – отладка на стороне клиента .....	324

## **Тестирование Node.js-приложений ..... 325**

Что такое Unit-тестирование?.....	325
TDD/BDD .....	325
Assert – утвердительный модуль .....	326
Should – BDD-style тестовый фреймворк.....	330
Цепочки утверждений.....	330
Chai – BDD/TDD-библиотека утверждений .....	335
Chai TDD-стиль .....	335
Chai BDD.....	335
Mocha – JavaScript тест-фреймворк.....	336
Начинаем работать с Mocha.....	337
Exports .....	340
QUnit.....	341
Асинхронный код .....	341
Jasmine – ароматный BDD-фреймворк.....	342
Основы работы с Jasmine .....	342
Jasmine и асинхронный код.....	345
Spies – шпионим и эмулируем.....	346

## **Grunt – The JavaScript Task Runner ..... 349**

Grunt – начало работы.....	350
Инструменты Grunt .....	352
Grunt watch – задача-наблюдатель.....	359
Grunt connect web server.....	361

## **Альтернативы JavaScript и Node.js ..... 363**

CoffeeScript – зависимость с первой чашки .....	363
---	-----

## 8 ❖ Содержание

Краткий курс по кофе .....	364
Классы, наследование, полиморфизм, генераторы! .....	367
CoffeScript и Node.js.....	368
Пишем сервер на CoffeScript .....	370
TypeScript – типа JavaScript от Microsoft.....	372
Node.js как TypeScript-компилятор.....	373
Аннотации типов .....	373
Классы! настоящие классы! .....	374
Интерфейсы.....	376
Модули .....	377
Что еще?.....	378
Dart – дротик в спину JavaScript от Google .....	378
Экосистема Dart.....	378
Знакомство с Dart.....	382
ООП – чтим традиции!.....	383
Область видимости и библиотеки .....	385
Изоляторы.....	386
А в общем-то.....	387
<b>Будущее уже сейчас – ECMAScript.next и Node .....</b>	<b>388</b>
ECMAScript 5 – уже стандарт .....	388
Всякие строгости – Strict Mode в JavaScript .....	388
JSON.....	390
Массивы .....	392
Объекты .....	394
Who's Next? ECMAScript 6 .....	399
ECMAScript 6 в браузерах и в Node.js.....	400
Вот теперь настоящие классы!.....	400
Модульная структура.....	402
Цикл for-of.....	403
Функции – let it block!.....	403
Arrow function .....	404
Обещания .....	406
Проху – займемся метапрограммированием на JavaScript .....	407
Константы .....	409
Генераторы.....	409
<b>Заключение – что дальше?.....</b>	<b>412</b>
<b>Приложение – полезные ресурсы по платформе</b>	
<b>Node.js .....</b>	<b>413</b>
<b>Список литературы .....</b>	<b>414</b>
<b>Предметный указатель .....</b>	<b>415</b>