

O'REILLY®

3-е издание

JavaScript

рецепты
для разработчиков



Адам Д. Скотт
Мэтью Макдоналд
Шелли Пауэрс

THIRD EDITION

JavaScript Cookbook

*Adam D. Scott, Matthew MacDonald,
and Shelley Powers*

Beijing • Boston • Farnham • Sebastopol • Tokyo

O'REILLY®

JavaScript

Рецепты для разработчиков

3-е издание

Адам Д. Скотт, Мэтью Макдоналд,
Шелли Пауэрс



Санкт-Петербург • Москва • Минск

2023

ББК 32.988.02-018
УДК 004.738.5
С44

Скотт Адам Д., Макдоналд Мэтью, Пауэрс Шелли

С44 JavaScript. Рецепты для разработчиков. 3-е изд. — СПб.: Питер, 2023. — 528 с.: ил. — (Серия «Бестселлеры O'Reilly»).

ISBN 978-5-4461-2001-7

Зачем изобретать велосипед, сталкиваясь с очередной проблемой в JavaScript? Здесь вы найдете множество рецептов кода для типовых задач программирования, а также познакомитесь с методами создания приложений, работающих в любом браузере. Адаптируемые примеры кода можно вставить практически в любой проект, а заодно поглубже изучить JS.

С помощью этой книги вы научитесь настраивать эффективную среду разработки с редактором кода, статическим анализатором и тестовым сервером; станете лучше понимать функции JS, включая замыкания и генераторы; узнаете, как использовать классы и наследование — основные концепции ООП; освоите работу с мультимедиа, включая аудио, видео и SVG; научитесь управлять HTML и CSS; благодаря Node.js сможете использовать JavaScript где угодно; узнаете, как получать доступ к удаленным данным и управлять ими с помощью технологий REST, GraphQL и Fetch; освоите популярную среду разработки приложений Express; научитесь выполнять асинхронные операции с промисами, `async/await` и веб-процессами.

16+ (В соответствии с Федеральным законом от 29 декабря 2010 г. № 436-ФЗ.)

ББК 32.988.02-018
УДК 004.738.5

Права на издание получены по соглашению с O'Reilly. Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельцев авторских прав.

Информация, содержащаяся в данной книге, получена из источников, рассматриваемых издательством как надежные. Тем не менее, имея в виду возможные человеческие или технические ошибки, издательство не может гарантировать абсолютную точность и полноту приводимых сведений и не несет ответственности за возможные ошибки, связанные с использованием книги. В книге возможны упоминания организаций, деятельность которых запрещена на территории Российской Федерации, таких как Meta Platforms Inc., Facebook, Instagram и др. Издательство не несет ответственности за доступность материалов, ссылки на которые вы можете найти в этой книге. На момент подготовки книги к изданию все ссылки на интернет-ресурсы были действующими.

ISBN 978-1492055754 англ.

Authorized Russian translation of the English edition JavaScript Cookbook,
3E ISBN 9781492055754 © 2021
Adam D. Scott and Matthew MacDonald

This translation is published and sold by permission of O'Reilly Media, Inc.,
which owns or controls all rights to publish and sell the same.

ISBN 978-5-4461-2001-7

© Перевод на русский язык ООО «Прогресс книга», 2022

© Издание на русском языке, оформление ООО «Прогресс книга», 2023

© Серия «Бестселлеры O'Reilly», 2023

Краткое содержание

Предисловие	15
-------------------	----

Часть I ЯЗЫК JAVASCRIPT

Глава 1. Настройка среды разработки	20
Глава 2. Строки и регулярные выражения	56
Глава 3. Числа.....	84
Глава 4. Даты.....	98
Глава 5. Массивы	110
Глава 6. Функции	141
Глава 7. Объекты.....	169
Глава 8. Классы	198
Глава 9. Асинхронное программирование	227
Глава 10. Ошибки и тестирование.....	254

Часть II JAVASCRIPT В БРАУЗЕРЕ

Глава 11. Инструментарий браузера.....	280
Глава 12. Работа с HTML	289
Глава 13. Получение удаленных данных.....	327
Глава 14. Сохранение данных.....	351
Глава 15. Работа с мультимедиа.....	371
Глава 16. Создание веб-приложений.....	387

Часть III
NODE.JS

Глава 17. Основы Node	420
Глава 18. Модули Node	442
Глава 19. Управление экосистемой Node	462
Глава 20. Удаленные данные.....	481
Глава 21. Построение веб-приложений с помощью Express.....	488
Об авторах	524
Иллюстрация на обложке.....	525

Оглавление

Предисловие	15
Аудитория этой книги	15
Структура издания.....	16
Условные обозначения.....	16
Примеры кода.....	17
Благодарности	18
От издательства.....	18

Часть I ЯЗЫК JAVASCRIPT

Глава 1. Настройка среды разработки	20
1.1. Выбираем редактор кода	21
1.2. Использование консоли разработки в браузере	23
1.3. Выполнение фрагментов кода в консоли разработчика	27
1.4. Использование строгого режима для выявления типичных ошибок.....	29
1.5. Вставка блоков HTML с помощью сокращенных команд Emmet	31
1.6. Установка менеджера пакетов npm (с Node.js).....	33
1.7. Загрузка пакета с помощью npm	36
1.8. Обновление пакета с помощью npm	40
1.9. Настройка локального тестового сервера	42
1.10. Соблюдение стандартов кодирования с помощью статического анализатора	45
1.11. Согласованное оформление кода с помощью форматировщика.....	50
1.12. Эксперименты в интерактивной среде JavaScript	52
Глава 2. Строки и регулярные выражения	56
2.1. Проверка того, что строка существует и она не пустая.....	56
2.2. Преобразование числового значения в форматированную строку	59

2.3. Вставка специальных символов	61
2.4. Использование эмодзи.....	63
2.5. Использование шаблонных литералов для более наглядной конкатенации строк	65
2.6. Сравнение строк без учета регистра	67
2.7. Проверка того, содержит ли строка заданную подстроку.....	68
2.8. Замена всех вхождений строки	69
2.9. Замена тегов HTML на именованные сущности	70
2.10. Использование регулярных выражений для создания шаблонов при замене строк.....	71
2.11. Извлечение списка из строки	74
2.12. Поиск по шаблону.....	76
2.13. Удаление пробелов в начале и в конце строки	80
2.14. Замена первой буквы строки на прописную	81
2.15. Валидация адреса электронной почты.....	82
Глава 3. Числа	84
3.1. Генерирование случайных чисел	84
3.2. Генерирование криптографически надежных случайных чисел.....	86
3.3. Округление до заданного десятичного разряда	88
3.4. Сохранение точности в дробных числах.....	89
3.5. Преобразование строки в число	91
3.6. Преобразование десятичных значений в шестнадцатеричные.....	93
3.7. Преобразование градусов в радианы	94
3.8. Вычисление длины дуги окружности.....	94
3.9. Манипуляции с очень большими целыми числами в формате BigInt	95
Глава 4. Даты.....	98
4.1. Получение текущих даты и времени.....	98
4.2. Преобразование строки в дату.....	100
4.3. Добавляем дни к дате.....	102
4.4. Сравнение дат и проверка двух дат на равенство.....	103
4.5. Вычисление времени, прошедшего между двумя датами	105
4.6. Представление даты в виде форматированной строки	107

Глава 5. Массивы	110
5.1. Проверка того, является ли объект массивом.....	110
5.2. Перебор всех элементов массива	111
5.3. Проверка равенства двух массивов	113
5.4. Разбиение массива на отдельные переменные	116
5.5. Передача массива в функцию, которая принимает список значений	117
5.6. Клонирование массива.....	118
5.7. Слияние двух массивов	120
5.8. Копирование части массива, выбранной по положению элемента	122
5.9. Извлечение из массива элементов, удовлетворяющих заданному условию	123
5.10. Очистка массива.....	124
5.11. Удаление дубликатов	125
5.12. Сведение двумерного массива.....	126
5.13. Точный поиск элементов массива.....	128
5.14. Поиск элементов массива, удовлетворяющих заданному критерию.....	129
5.15. Удаление и замена элементов массива.....	131
5.16. Сортировка массива объектов по заданному свойству	132
5.17. Преобразование элементов массива	134
5.18. Использование всех элементов массива в одном вычислении	134
5.19. Проверка содержимого массива	136
5.20. Построение коллекции недублирующихся значений	137
5.21. Создание коллекции элементов, индексируемой по ключу.....	139
Глава 6. Функции	141
6.1. Передача одной функции в другую в качестве аргумента.....	141
6.2. Использование стрелочных функций.....	145
6.3. Предоставление значения параметра по умолчанию	148
6.4. Создание функции, принимающей неограниченное число аргументов ...	149
6.5. Использование именованных параметров функции	150
6.6. Создание функции с сохранением состояния посредством замыкания....	153
6.7. Создание функции-генератора, которая возвращает несколько значений.....	155
6.8. Уменьшение избыточности за счет частичного применения.....	160

6.9. Фиксация this посредством привязки функций	163
6.10. Реализация рекурсивного алгоритма	166

Глава 7. Объекты.....169

7.1. Проверка того, относится ли объект к заданному типу	169
7.2. Объединение данных с помощью объектных литералов	171
7.3. Проверка существования свойства у объекта	174
7.4. Перебор всех свойств объекта	176
7.5. Проверка того, является ли объект пустым.....	178
7.6. Объединение свойств двух объектов.....	180
7.7. Выбор способа определения свойств	181
7.8. Запрет любых изменений объекта	184
7.9. Перехват и изменение объектов с помощью прокси-объектов.....	185
7.10. Клонирование объектов	189
7.11. Создание глубокой копии объекта.....	191
7.12. Создание абсолютно уникальных ключей для свойств объекта	193
7.13. Создание перечислений с помощью Symbol.....	195

Глава 8. Классы198

8.1. Создание класса для многократного использования	198
8.2. Добавление в класс новых свойств	202
8.3. Улучшенное строковое представление класса	207
8.4. Создание произвольного класса посредством шаблона «Конструктор»	208
8.5. Создание возможности для объединения методов класса в цепочку	211
8.6. Создание статических методов класса.....	213
8.7. Создание объектов посредством статических методов	216
8.8. Наследование функционала другого класса	218
8.9. Объединение классов JavaScript в модули.....	223

Глава 9. Асинхронное программирование.....227

9.1. Обновление страницы в цикле	228
9.2. Использование функции, которая возвращает промис.....	230
9.3. Замена асинхронной функции с обратным вызовом на промис	234
9.4. Конкурентное выполнение нескольких промисов	237
9.5. Ожидание выполнения промиса с помощью await и async.....	240

9.6. Создание асинхронной функции-генератора	244
9.7. Выполнение фоновых задач с помощью Web Worker	247
9.8. Поддержка сообщений о ходе выполнения задач в Web Worker	251
Глава 10. Ошибки и тестирование	254
10.1. Обнаружение и обезвреживание ошибок	254
10.2. Перехват различных типов ошибок	257
10.3. Перехват асинхронных ошибок	259
10.4. Обнаружение необработанных ошибок	261
10.5. Выдача обычной ошибки	264
10.6. Выдача нестандартных ошибок	266
10.7. Написание модульных тестов для кода	268
10.8. Отслеживание покрытия кода тестами	276

Часть II

JAVASCRIPT В БРАУЗЕРЕ

Глава 11. Инструментарий браузера	280
11.1. Отладка кода JavaScript	280
11.2. Анализ производительности во время выполнения кода	282
11.3. Обнаружение неиспользуемого кода JavaScript	284
11.4. Выдача наилучших рекомендаций посредством Lighthouse	286
Глава 12. Работа с HTML	289
12.1. Доступ к определенному элементу, поиск его родительского и дочерних элементов	289
12.2. Перебор результатов, полученных от <code>querySelectorAll()</code> , с помощью <code>forEach()</code>	292
12.3. Привязка к элементу действия в ответ на щелчок	293
12.4. Поиск всех элементов с данным атрибутом	295
12.5. Выбор всех элементов определенного типа	296
12.6. Исследование дочерних элементов с помощью Selectors API	298
12.7. Изменение класса элемента	300
12.8. Присвоение элементу атрибута <code>style</code>	300
12.9. Создание абзаца и вставка в него текста	303
12.10. Вставка нового элемента в определенной точке DOM	304

12.11. Проверка того, установлен ли флажок.....	305
12.12. Вставка значений в таблицу HTML.....	306
12.13. Удаление строк из таблицы HTML.....	309
12.14. Скрытие частей страницы.....	312
12.15. Создание окон, всплывающих по наведению указателя мыши.....	313
12.16. Валидация данных формы.....	315
12.17. Выделение ошибочно заполненных полей форм и реализация специальных возможностей	319
12.18. Создание автоматически обновляемой области, доступной на специализированных устройствах	325
Глава 13. Получение удаленных данных	327
13.1. Запрос удаленных данных с помощью Fetch API.....	327
13.2. Использование метода XMLHttpRequest	331
13.3. Отправка данных формы.....	332
13.4. Заполнение списка выбора данными, полученными с сервера	336
13.5. Синтаксический анализ данных, полученных в формате JSON.....	340
13.6. Получение и синтаксический анализ данных в формате XML	342
13.7. Передача двоичных данных и загрузка изображения	344
13.8. Обмен HTTP cookies между несколькими доменами	345
13.9. Двухнаправленный обмен данными между клиентом и сервером посредством WebSockets.....	346
13.10. Длинный опрос удаленного источника данных.....	348
Глава 14. Сохранение данных	351
14.1. Сохранение информации в cookies.....	351
14.2. Хранение данных на стороне клиента с помощью sessionStorage	354
14.3. Создание хранилища данных на стороне клиента на основе localStorage.....	360
14.4. Сохранение больших объемов данных на стороне клиента с помощью IndexedDB.....	364
14.5. Упрощение IndexedDB с помощью библиотеки	367
Глава 15. Работа с мультимедиа	371
15.1. JavaScript для SVG.....	371
15.2. Доступ к SVG из скрипта веб-страницы.....	374
15.3. Построение столбчатой диаграммы в формате SVG с помощью библиотеки D3	376

15.4. Интеграция элементов SVG и Canvas в HTML	379
15.5. Выполнение процедуры в начале воспроизведения аудиофайла	382
15.6. Управление отображением видео с помощью элемента video и JavaScript	383
Глава 16. Создание веб-приложений.....	387
16.1. Создание пакетов JavaScript	387
16.2. JavaScript для мобильного интернета	389
16.3. Создание прогрессивного веб-приложения.....	392
16.4. Тестирование и профилирование прогрессивных веб-приложений	399
16.5. Получение текущего URL.....	403
16.6. Перенаправление на другой URL	404
16.7. Копирование текста в буфер обмена.....	405
16.8. Вывод на стационарном компьютере таких же уведомлений, как на мобильном устройстве.....	407
16.9. Открытие в браузере файла с локального устройства.....	410
16.10. Расширение возможностей с помощью Web Components	413
16.11. Выбор фреймворка для разработки на стороне клиента	416

Часть III

NODE.JS

Глава 17. Основы Node	420
17.1. Управление версиями Node с помощью Node Version Manager	420
17.2. Ответ на простой запрос браузера.....	423
17.3. Интерактивная проверка кода Node с помощью REPL.....	425
17.4. Чтение данных из файла и запись данных в файл	428
17.5. Получение данных из терминала	433
17.6. Получение пути к выполняемому скрипту	435
17.7. Работа с таймерами и циклом событий Node	436
Глава 18. Модули Node	442
18.1. Поиск нужного модуля Node через npm	442
18.2. Преобразование библиотеки в модуль Node.....	444
18.3. Перенос кода в модульную среду	445
18.4. Создание устанавливаемого модуля Node	448

18.5. Создание мультиплатформенных библиотек	454
18.6. Тестирование модулей	458
Глава 19. Управление экосистемой Node.....	462
19.1. Использование переменных среды	462
19.2. Что делать с адом обратных вызовов.....	464
19.3. Доступ к функциям командной строки из приложений Node	467
19.4. Передача аргументов в командную строку	470
19.5. Создание утилиты командной строки с подсказкой с помощью Commander	471
19.6. Обеспечение работоспособности экземпляра Node.....	474
19.7. Отслеживание изменений и перезапуск приложения в процессе разработки на локальном компьютере	475
19.8. Многократное выполнение задач по расписанию.....	477
19.9. Тестирование производительности и возможностей приложения WebSockets.....	478
Глава 20. Удаленные данные	481
20.1. Получение удаленных данных.....	481
20.2. Анализ экранных данных	483
20.3. Доступ к данным в формате JSON посредством RESTful API.....	485
Глава 21. Построение веб-приложений с помощью Express	488
21.1. Использование Express для ответов на запросы.....	488
21.2. Использование Express-Generator	492
21.3. Задача маршрутизации.....	497
21.4. Работа с OAuth.....	499
21.5. Аутентификация пользователей в OAuth 2 с помощью Passport.js.....	510
21.6. Обработка форматированных данных	514
21.7. Построение RESTful API.....	516
21.8. Построение API GraphQL	519
Об авторах	524
Иллюстрация на обложке.....	525

Предисловие

Сядясь за работу над последним изданием книги «JavaScript. Рецепты для разработчиков», я решил, что «рецепты» — точная метафора. Что отличает хороший сборник кулинарных рецептов? Перебрав несколько кулинарных справочников, стоящих на полке в кухне, я обнаружил, что мои любимые — это те, в которых не только описаны самые вкусные блюда, но и дается много советов, основанных на богатом опыте знающих людей. Хорошая кулинарная книга — не та, где вы найдете все варианты приготовления мяса по-бургундски, а та, где описаны наилучшие, по мнению автора, методики и рецепты, к которым обычно прилагается несколько советов. Именно такой концепцией мы и решили руководствоваться, собирая коллекцию рецептов по JavaScript. Советы, представленные в этой книге, дают трое маститых профессионалов, но это кульминация главным образом *нашего* уникального опыта. Окажись на нашем месте другие разработчики, они бы написали похожую, но другую книгу.

JavaScript вырос в невероятный, высокоэффективный, многоцелевой язык программирования. С нашей подборкой рецептов в руках вы сможете не только решать всевозможные проблемы, которые вам встретятся, но и придумывать собственные рецепты.

Аудитория этой книги

Предупреждаем сразу: стремясь охватить все разнообразие тем и областей, в которых сегодня используется JavaScript, мы не рассчитывали на новичков в программировании. Для тех, кто начинает изучать программирование на JavaScript, есть так много хороших книг и учебных пособий, что мы со спокойной совестью посвятили свою *практикующим разработчикам* — тем, кто ищет решение конкретных проблем и задач на JavaScript.

Если вы успели попрактиковаться с JavaScript лишь в течение пары месяцев — возможно, попробовали силы в разработке для Node или веб-разработке, то изучение материала этой книги не должно вызвать у вас затруднений. Издание станет полезным руководством и для тех разработчиков, кто в основном имеет дело с другим языком программирования, но время от времени испытывает потребность в JavaScript. Наконец, если вы действующий разработчик на JavaScript, но вас иногда ставят в тупик некоторые особенности языка, то эта книга также станет для вас полезным ресурсом.

Структура издания

У книги есть два типа читателей. Первые — те, кто прочитает ее от корки до корки, подобрав по дороге каждую крошку полезных знаний. Вторые будут углубляться в чтение лишь по мере необходимости в поисках решения определенной задачи или категории стоящих перед ними проблем. Мы постарались структурировать книгу таким образом, чтобы она была полезной обеим категориям читателей. Для этого разбили ее на три части.

- Часть I. Язык JavaScript. Сюда вошли рецепты для JavaScript как языка программирования.
- Часть II. JavaScript в браузере. Здесь описан JavaScript в его естественной среде обитания — в браузере.
- Часть III. Node.js. Здесь JavaScript рассматривается сквозь призму Node.js.

Все главы книги разделены на несколько самостоятельных рецептов. Каждый из них состоит из следующих частей.

- *Задача* — описание типичного сценария разработки, в котором может быть использован JavaScript.
- *Решение* — решение задачи с примером кода и минимальным описанием.
- *Обсуждение* — подробное обсуждение примера кода и применяемых методик.

Кроме того, в рецепте могут присутствовать разделы «Читайте также» со списком литературы, рекомендованной для дальнейшего чтения, и «Дополнительно» с описанием других методик.

Условные обозначения

В этой книге использованы следующие условные обозначения.

Курсив

Курсивом выделены новые термины.

Рубленый шрифт

Этим шрифтом выделены элементы пользовательского интерфейса, URL, адреса электронной почты и названия клавиш.

Моноширинный шрифт

Любой компьютерный код — команды, массивы, элементы, операторы, опции, переключатели, переменные, атрибуты, ключи, функции, типы, классы, пространства имен, методы, модули, свойства, параметры, значения, объекты,

события, обработчики событий, XML- и HTML-теги, макросы, названия и содержимое файлов, результаты работы команд.

Моноширинный жирный шрифт

Команды и другой текст, который должен вводить пользователь.



Этот значок обозначает обычное примечание.



Этот значок обозначает совет или рекомендацию.



Этот значок обозначает предупреждение или предостережение.

Вам также могут пригодиться упоминаемые в книге сайты и веб-страницы — это упростит поиск информации онлайн. Обычно приводится адрес (URL) и название (заглавие или другой подходящий заголовок) этих ресурсов. Некоторые адреса довольно длинные, так что вам, возможно, будет проще найти такие страницы, введя их название в той поисковой системе, которой вы обычно пользуетесь. Названия помогут и в том случае, если страницу не удастся найти по адресу: URL может измениться, но название страницы останется прежним.

Примеры кода

Вспомогательные материалы (примеры кода, упражнения и т. п.) доступны для скачивания по адресу <https://github.com/javascripteverywhere/cookbook>.

В общем случае все примеры кода из книги вы можете использовать в своих программах и в документации. Вам не нужно обращаться в издательство за разрешением, если вы не собираетесь воспроизводить существенные части программного кода. Если вы разрабатываете программу и используете в ней несколько фрагментов кода из книги, вам не нужно обращаться за разрешением. Но для продажи или распространения примеров из книги вам потребуется разрешение от издательства O'Reilly. Вы можете отвечать на вопросы, цитируя данную книгу или примеры из нее, но для включения существенных объемов программного кода из книги в документацию вашего продукта потребуется разрешение.

Мы рекомендуем, но не требуем добавлять ссылку на первоисточник при цитировании. Под ссылкой на первоисточник мы подразумеваем указание авторов, издательства и ISBN.

За получением разрешения на использование значительных объемов программного кода из книги обращайтесь по адресу permissions@oreilly.com.

Благодарности

Это третье издание книги «JavaScript. Рецепты для разработчиков». Первые два были написаны Шелли Пауэрс. Третье издание написали и обновили Адам Скотт и Мэтью Макдоналд. Адам и Мэтью благодарят своих редакторов Анджелу Руфино и Дженнифер Поллок, которые сопровождали проект на протяжении всего периода его трудного взросления; Сару Вакс, Шайка Нитлинга и Элизабет Робсон, которые сделали множество тонких замечаний и полезных предложений. Адам также благодарен Джону Пакстону за поддержку и общение на ранних стадиях работы над книгой.

Шелли благодарит своих редакторов Симона Сен-Лорана и Брайана Макдоналда, а также доктора Аксея Раушмайера и Семми Пьюроул.

Все вместе мы благодарим производственный персонал O'Reilly за постоянную помощь и поддержку.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу comp@piter.com (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

На веб-сайте издательства www.piter.com вы найдете подробную информацию о наших книгах.