

---

# Оглавление

Предисловие .....	16
Об этой книге.....	18
Условные обозначения .....	19
Использование примеров кода.....	19
Благодарности .....	20
От издательства .....	22
<b>Глава 1. Hello World</b> .....	<b>23</b>
Установка.....	23
Hello React World .....	24
Так что же сейчас произошло? .....	26
Метод <code>React.createElement()</code> .....	28
Синтаксис JSX .....	30
Установка Babel .....	31
Привет, мир JSX .....	32
О транспиляции .....	33
Далее: настраиваемые компоненты .....	35

---

<b>Глава 2. Жизнь компонента</b> .....	36
Настраиваемый функциональный компонент.....	36
JSX-версия .....	37
Настраиваемый классовый компонент .....	38
Какой синтаксис использовать .....	39
Свойства .....	40
Свойства в функциональных компонентах .....	42
Свойства по умолчанию .....	43
Состояние .....	44
Компонент текстовой области .....	45
Компонент <code>textarea</code> , отслеживающий свое состояние.....	47
Немного о DOM-событиях .....	50
Обработка событий в прежние времена.....	50
Обработка событий в React.....	52
Синтаксис обработки событий.....	53
Сравнение свойств и состояния.....	55
Свойства в исходном состоянии: антипаттерн.....	55
Доступ к компоненту извне .....	56
Методы управления жизненным циклом .....	58
Примеры управления жизненным циклом: тотальная регистрация .....	59
Параноидальная защита состояния.....	62
Пример жизненного цикла: применение дочернего компонента .....	63

Выигрыш в производительности: предотвращение обновлений компонентов.....	66
Что случилось с функциональными компонентами.....	68
<b>Глава 3. Excel: причудливый компонент таблицы.....</b>	<b>69</b>
Начнем с данных.....	69
Цикл создания заголовков таблицы.....	70
Цикл заголовков таблиц, упрощенная версия.....	72
Отладка для избавления от предупреждения в консоли.....	75
Добавление содержимого <code>&lt;td&gt;</code> .....	76
Типы свойств.....	79
Можете ли вы улучшить компонент.....	81
Сортировка.....	82
Можете ли вы улучшить компонент.....	85
Сортировка подсказок пользовательского интерфейса.....	85
Редактирование данных.....	87
Редактируемая ячейка.....	88
Поле ввода ячейки.....	91
Сохранение.....	91
Выводы и определение различий в виртуальной DOM.....	92
Поиск.....	94
Состояние и пользовательский интерфейс.....	96
Фильтрация содержимого.....	99

---

Обновление метода save() .....	102
Как усовершенствовать поиск.....	103
Мгновенное воспроизведение .....	103
Очистка обработчиков событий.....	106
Решение задачи очистки .....	108
Как усовершенствовать воспроизведение .....	109
Возможна ли альтернативная реализация .....	110
Скачивание данных таблицы .....	110
Получение данных.....	113
<b>Глава 4. Функциональный Excel</b> .....	<b>115</b>
Кратко освежим знания: функциональные и классовые компоненты.....	115
Отображение данных .....	116
Хук состояния.....	118
Сортировка таблицы.....	120
Редактирование данных .....	123
Поиск.....	125
Жизненные циклы в мире хуков .....	126
Проблемы с методами жизненного цикла.....	126
Хук useEffect() .....	127
Устранение побочных эффектов .....	128
Безаварийный жизненный цикл.....	130
Хук useLayoutEffect().....	131

Пользовательский хук.....	134
Завершение воспроизведения.....	137
Хук useReducer.....	139
Функции редюсера.....	139
Действия.....	140
Пример редюсера.....	141
Модульное тестирование редюсера.....	144
Компонент Excel с редюсером.....	145
<b>Глава 5. JSX.....</b>	<b>149</b>
Несколько инструментов.....	150
Пробельные символы в JSX.....	151
Комментарии в JSX.....	153
Элементы HTML.....	154
Anti-XSS.....	156
Распространяемые атрибуты.....	157
Атрибуты, распространяемые от родительского компонента к дочернему.....	158
Возвращение в JSX нескольких узлов.....	161
Оболочка.....	161
Фрагмент.....	162
Массив.....	162
Отличия JSX от HTML.....	163
Просто class использовать нельзя, а как насчет for?.....	164

---

Атрибут style — объект.....	164
Закрывающие теги .....	165
Атрибуты в верблюжьем регистре .....	166
Компоненты с пространством имен .....	166
JSX и формы .....	168
Обработчик события onChange.....	168
Сравнение value и defaultValue .....	170
Значение компонента <textarea> .....	171
Значение компонента <select> .....	172
Управляемые и неуправляемые компоненты.....	173
<b>Глава 6. Настройки, необходимые для разработки приложения .....</b>	<b>181</b>
Создание React-приложения .....	182
Node.js.....	182
Привет, CRA .....	183
Сборка и развертывание .....	184
Были допущены ошибки .....	186
Файл package.json и папка node_modules.....	186
Рассмотрим код подробнее.....	187
Индексы .....	187
Модернизированный JavaScript.....	188
CSS .....	189
Идем дальше.....	190

---

<b>Глава 7. Создание компонентов приложения.....</b>	<b>191</b>
Настройка .....	192
Приступим к программированию .....	192
Рефакторинг компонента Excel .....	193
Версия 0.0.1 нового приложения.....	196
CSS .....	197
Локальное хранилище .....	198
Компоненты .....	199
Исследование.....	201
Логотип и тело .....	203
Логотип.....	203
Тело .....	204
Исследование компонентов.....	204
Компонент <button>.....	205
Файл Button.js.....	206
Пакет classNames .....	207
Формы .....	209
Компонент <Suggest> .....	209
Компонент <Rating> .....	211
Компонент <FormInput> .....	215
Компонент <Form> .....	218
Компонент <Actions>.....	224
Диалоги .....	226
Компонент <Header> .....	232

---

Конфигурация приложения.....	233
<Excel>: новый и усовершенствованный .....	235
Общая структура.....	238
Отображение .....	239
Строгий режим React и редюсеры.....	244
Небольшие вспомогательные функции Excel.....	247
<b>Глава 8. Готовое приложение .....</b>	<b>254</b>
Обновленный App.js.....	257
Компонент DataFlow .....	258
Тело компонента DataFlow .....	260
Работа выполнена .....	263
Whinerpad v2 .....	266
Контекст .....	266
Следующие шаги.....	267
Циклические данные .....	268
Предоставление контекста.....	270
Потребление контекста.....	273
Контекст в заголовке.....	274
Контекст в таблице данных .....	279
Обновление Discovery.....	281
Маршрутизация .....	284
Контекст маршрута .....	285
Использование URL фильтра.....	288



Использование контекста маршрута в заголовке .....	290
Использование контекста маршрута в таблице данных .....	292
Хук <code>useCallback()</code> .....	294
Заключение .....	298
Об авторе .....	299
Иллюстрация на обложке .....	300

---

# Предисловие

Еще одна восхитительная теплая калифорнийская ночь. Легкий океанский бриз лишь усиливает чувство абсолютного блаженства. Место — Лос-Анджелес, время — двухтысячные годы. Я готовлюсь отправить по FTP на свой сервер мое новое небольшое веб-приложение под названием CSSsprites.com и выпустить его в мир. Последние несколько вечеров, работая над приложением, я размышлял над одной проблемой: ну почему 20 % усилий ушло на отладку основной логики работы приложения, а 80 % — на усовершенствование пользовательского интерфейса? Сколько других инструментов я мог бы создать, если бы мне не приходилось все время пользоваться методом `getElementById()` и беспокоиться о состоянии приложения? (Подгрузил ли пользователь что-то на страницу? Что, произошла ошибка? А это диалоговое окно все еще на экране?) Почему разработка пользовательского интерфейса отнимает так много времени? И что происходит со всеми этими разными браузерами? Постепенно мое блаженство таяло, превращаясь в раздражение.

Перенесемся в март 2015 года, на Facebook-конференцию F8. Моя команда готовится сообщить о полной переработке двух наших веб-приложений: средства работы со сторонними комментариями и связанного с ним инструмента их модерации. По сравнению с моим небольшим приложением CSSsprites.com это были полноценные веб-приложения, имеющие куда более существенные возможности, гораздо бóльшую эффективность и нереальные объемы трафика. Несмотря на это, разработка доставляла огром-

ное удовольствие. Члены команды, незнакомые с приложением (а некоторые даже с JavaScript и CSS), вносили свои предложения по поводу возможностей и каких-либо улучшений его различных частей, и мы без промедления и не прилагая особых усилий внедряли их. Как сказал один из членов команды, «теперь я понимаю, что значит любить свою работу».

Что же произошло за это время? Появилась технология React.

React представляет собой библиотеку для создания пользовательского интерфейса, которая помогает вам определить его раз и навсегда. Затем при изменении состояния приложения интерфейс перестраивается, *реагируя* на изменения, и вам не нужно ничего делать дополнительно. В конце концов, вы уже определили интерфейс. Определили? Скорее, вы его *объявили*. А для создания большого полнофункционального приложения вы воспользовались небольшими управляемыми *компонентами*. Теперь уже половина внутреннего кода ваших функций не занимается поиском DOM-узлов, и вам остается лишь отслеживать состояние (state) вашего приложения (с помощью обычных объектов JavaScript), а все остальное происходит в соответствии с этим состоянием.

Изучение React стоит того — вы изучаете всего одну библиотеку и сможете использовать ее, чтобы в дальнейшем создавать:

- веб-приложения;
- приложения для работы под управлением iOS и Android;
- TV-приложения;
- обычные приложения для настольных компьютеров.

Используя ту же идею создания компонентов и пользовательских интерфейсов, вы сможете конструировать приложения, приспособленные под исходную систему с присущими ей производительностью и элементами управления (*реальными* элементами управления, а не похожими на них копиями). Речь идет не о «напиши

один раз, запусти везде» (наша индустрия постоянно терпит в этом неудачу), а о «научиться один раз, использовать везде».

Одним словом, изучение React позволяет сэкономить 80 % вашего времени и сосредоточиться на том, что важно (то есть на реальных задачах, для решения которых предназначено ваше приложение).

## Об этой книге

Книга посвящена изучению React с точки зрения веб-разработки. В первых трех главах изучение начинается с использования пустого HTML-файла, на основе которого выстраивается весь остальной код. Это позволяет сосредоточиться на изучении React, не отвлекаясь на новый синтаксис или на применение дополнительных инструментальных средств.

В главе 5 больше внимания уделяется JSX — дополнительной технологии, обычно используемой в сочетании с React.

Далее мы перейдем к изучению приемов разработки реального приложения и освоению дополнительных инструментов, которые могут быть полезны. Чтобы быстро начать работу и свести обсуждение вспомогательных технологий к минимуму, в книге используется приложение `create-react-app`. Цель состоит в том, чтобы сосредоточиться прежде всего на React.

Спорным решением стало включение компонентов классов в дополнение к функциональным компонентам. Вероятно, за функциональными компонентами будущее, однако читатель может столкнуться с существующим кодом и руководствами, в которых говорится только о компонентах классов. Знание обоих синтаксисов удваивает шансы на чтение и понимание кода в реальных условиях.

Успехов вам в изучении React — пусть оно будет приятным и плодотворным!

## Условные обозначения

В данной книге используются следующие обозначения.

### *Курсив*

Курсивом выделены новые термины.

### Моноширинный шрифт

Используется для листингов программ, а также внутри абзацев для обозначения таких элементов, как переменные и функции, базы данных, типы данных, переменные среды, операторы и ключевые слова, имена файлов и их расширений.

### **Жирный моноширинный шрифт**

Показывает команды или другой текст, который пользователь должен ввести самостоятельно.

### Шрифт без засечек

Используется для обозначения URL, адресов электронной почты, названий кнопок, каталогов.



Этот рисунок указывает на совет или предложение.



Этот рисунок указывает на общее примечание.

## Использование примеров кода

Дополнительные материалы (примеры кода, упражнения и т. д.) доступны для скачивания по адресу <https://github.com/stoyan/reactbook2>.

Эта книга призвана помочь вам выполнять свою работу. В общем случае все примеры кода вы можете использовать в своих программах и в документации. Вам не нужно обращаться в издательство за разрешением, если вы не собираетесь воспроизводить существенные части кода, или если разрабатываете программу и используете в ней несколько фрагментов кода из книги, или если будете цитировать эту книгу или примеры из нее, отвечая на вопросы. Вам потребуется разрешение от издательства O'Reilly, если вы хотите продавать или распространять примеры из книги либо хотите включить существенные объемы программного кода из книги в документацию вашего продукта.

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](mailto:permissions@oreilly.com).

## Благодарности

Хочу поблагодарить всех, кто вычитывал черновые варианты этой книги и присылал отзывы и исправления.

Для первого издания: Андреа Маноле (Andrea Manole), Ильян Пейчев (Iliyan Peychev), Костадин Илов (Kostadin Illov), Марк Дуппенталер (Mark Duppenenthaler), Стефан Албер (Stephan Alber), Асен Божилов (Asen Bozhilov).

Для второго издания: Адам Ракис (Adam Rackis), Максимилиано Фиртман (Maximiliano Firtman), Четан Каранде (Chetan Karande), Кирилл Христов (Kiril Christov) и Скотт Сатоши Ивако (Scott Satoshi Iwako).

Спасибо всем сотрудникам Facebook, которые работают над (или с) React и каждый день отвечают на мои вопросы. Я также благодарен всему сообществу React, которое продолжает создавать великолепные инструментальные средства, библиотеки, публикует статьи и создает примеры применения этой технологии.

Я чрезвычайно благодарен Джордану Уалке (Jordan Walke).

Спасибо всем сотрудникам издательства O'Reilly, содействовавшим выходу этой книги: Анджеле Руфино (Angela Rufino), Дженнифер Поллок (Jennifer Pollock), Мег Фоли (Meg Foley), Киму Коферу (Kim Cofer), Джастину Биллингу (Justin Billing), Николь Шелби (Nicole Shelby), Кристен Браун (Kristen Brown) и многим другим.

Спасибо Явору Вачкову (Javor Vatchkov), дизайнеру пользовательского интерфейса примера приложения, разрабатываемого в данной книге (работу приложения можно оценить на [whinepad.com](http://whinepad.com)).