

БЕР БИБО
ИГУДА КАЦ

второе
издание

jQuery

Подробное руководство
по продвинутому JavaScript



НИГНТЕНС

jQuery in Action

Second edition

Bear Bibeault, Yehuda Katz



H I G H T E C H

jQuery

Подробное руководство
по продвинутому JavaScript

Второе издание

Бер Бибо, Иегуда Кац



Санкт-Петербург — Москва
2011

Серия «High tech»
Бер Бибо, Иегуда Кац
**jQuery. Подробное руководство
по продвинутому JavaScript, 2-е издание**

Перевод А. Киселева

Главный редактор	<i>А. Галунов</i>
Зав. редакцией	<i>Н. Макарова</i>
Выпускающий редактор	<i>П. Щеголев</i>
Редактор	<i>Ю. Бочина</i>
Корректор	<i>С. Минин</i>
Верстка	<i>К. Чубаров</i>

Бер Бибо, Иегуда Кац

jQuery. Подробное руководство по продвинутому JavaScript, 2-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 624 с., ил.

ISBN 978-5-93286-201-8

Издание представляет собой введение и справочное руководство по jQuery – мощной платформе для разработки веб-приложений. Подробно описывается, как выполнять обход документов HTML, обрабатывать события, добавлять поддержку технологии Ajax в свои веб-страницы, воспроизводить анимацию и визуальные эффекты. Уникальные «лабораторные страницы» помогут закрепить изучение каждой новой концепции на практических примерах. Рассмотрены вопросы взаимодействия jQuery с другими инструментами и платформами и методы создания модулей расширения для этой библиотеки.

Книга предназначена для разработчиков, знакомых с языком JavaScript и технологией Ajax и стремящихся создавать краткий и понятный программный код. Уникальная способность jQuery составлять «цепочки» из методов позволяет выполнять несколько последовательных операций над элементами страницы, в результате чего код сокращается втрое.

Второе издание подверглось обширной переработке, чтобы продемонстрировать новые возможности версии jQuery 1.4: новые нестандартные события, пространства имен событий, функции и эффекты и другие полезные методы и функции. Кроме того, в книге появилась совершенно новая часть, полностью посвященная библиотеке jQuery UI, которая охватывает обширнейшие изменения, внесенные в jQuery UI с момента прошлой публикации.

ISBN 978-5-93286-201-8

ISBN 978-1-935182-32-0 (англ)

© Издательство Символ-Плюс, 2011
Authorized translation of the English edition © 2010 Manning Publications Co. This translation is published and sold by permission of Manning Publications Co., the owner of all rights to publish and sell the same.

Все права на данное издание защищены Законодательством РФ, включая право на полное или частичное воспроизведение в любой форме. Все товарные знаки или зарегистрированные товарные знаки, упоминаемые в настоящем издании, являются собственностью соответствующих фирм.

Издательство «Символ-Плюс». 199034, Санкт-Петербург, 16 линия, 7, тел. (812) 380-5007, www.symbol.ru. Лицензия ЛП N 000054 от 25.12.98.

Подписано в печать 28.02.2011. Формат 70×100^{1/16}. Печать офсетная.

Объем 39 печ. л. Тираж 1200 экз. Заказ №

Отпечатано с готовых диапозитивов в ГУП «Типография «Наука»
199034, Санкт-Петербург, 9 линия, 12.

Оглавление

Отзывы на первое издание.....	13
Предисловие.....	15
Введение ко второму изданию	17
Введение к первому изданию.....	19
Благодарности	22
Об этой книге.....	25
Об авторах.....	31
Часть I. Ядро jQuery	33
Глава 1. Введение в jQuery	35
1.1. Больше возможностей, меньше кода.....	36
1.2. Ненавязчивый JavaScript	38
1.2.1. Отделение поведения от разметки.....	39
1.2.2. Отделение сценария	40
1.3. Основы jQuery	41
1.3.1. Обертка jQuery	41
1.3.2. Вспомогательные функции.....	44
1.3.3. Обработчик готовности документа	45
1.3.4. Создание элементов DOM	47
1.3.5. Расширение jQuery.....	48
1.3.6. Сочетание jQuery с другими библиотеками	50
1.4. Итоги.....	51
Глава 2. Выбор элементов для дальнейшей работы	53
2.1. Отбор элементов для манипуляций.....	54
2.1.1. Управление контекстом	56
2.1.2. Базовые селекторы CSS.....	58
2.1.3. Селекторы выбора потомков, контейнеров и атрибутов	59
2.1.4. Выбор элементов по позиции	64
2.1.5. CSS и нестандартные селекторы jQuery	67
2.2. Создание новых элементов HTML.....	71
2.3. Манипулирование обернутым набором элементов.....	74
2.3.1. Определение размера обернутого набора элементов	76

2.3.2. Получение элементов из перевернутого набора	77
2.3.3. Получение срезов перевернутого набора элементов	81
2.3.4. Получение перевернутого набора с учетом взаимоотношений	92
2.3.5. Дополнительные способы использования перевернутого набора	94
2.3.6. Управление цепочками методов jQuery	95
2.4. Итоги	98
Глава 3. Оживляем страницы с помощью jQuery	99
3.1. Манипулирование свойствами и атрибутами элементов	100
3.1.1. Манипулирование свойствами элементов	102
3.1.2. Извлечение значений атрибутов	103
3.1.3. Установка значений атрибутов	105
3.1.4. Удаление атрибутов	107
3.1.5. Игры с атрибутами	108
3.1.6. Сохранение собственных данных в элементах	110
3.2. Изменение стиля отображения элемента	112
3.2.1. Добавление и удаление имен классов	113
3.2.2. Получение и установка стилей	118
3.3. Установка содержимого элемента	127
3.3.1. Замена HTML-разметки или текста	127
3.3.2. Перемещение и копирование элементов	129
3.3.3. Обертывание элементов	137
3.3.4. Удаление элементов	139
3.3.5. Копирование элементов	140
3.3.6. Замена элементов	142
3.4. Обработка значений элементов форм	144
3.5. Итоги	147
Глава 4. События – место, где все происходит	148
4.1. Модель событий браузера	150
4.1.1. Модель событий DOM уровня 0	151
4.1.2. Модель событий DOM уровня 2	158
4.1.3. Модель событий Internet Explorer	164
4.2. Модель событий jQuery	164
4.2.1. Подключение обработчиков событий с помощью jQuery	165
4.2.2. Удаление обработчиков событий	171
4.2.3. Исследование экземпляра Event	172
4.2.4. Упреждающая установка обработчиков событий	175
4.2.5. Запуск обработчиков событий	179
4.2.6. Прочие методы для работы с событиями	181
4.3. Запуск событий (и не только) в работу	187
4.3.1. Фильтрация больших наборов данных	188

4.3.2. Создание элементов по шаблону	190
4.3.3. Основная разметка	194
4.3.4. Добавление новых фильтров	195
4.3.5. Добавление элементов для ввода параметров фильтра	199
4.3.6. Удаление ненужных фильтров и другие операции.....	201
4.3.7. Всегда можно что-то улучшить	202
4.4. Итоги	204
Глава 5. Заряжаем страницы анимацией и эффектами	206
5.1. Скрытие и отображение элементов	207
5.1.1. Реализация сворачиваемого «модуля»	208
5.1.2. Переключение состояния отображения элементов	212
5.2. Анимационные эффекты при изменении визуального состояния элементов	213
5.2.1. Постепенное отображение и скрытие элементов.....	213
5.2.2. Плавное растворение и проявление элементов	220
5.2.3. Закатывание и выкатывание элементов.....	224
5.2.4. Остановка анимационных эффектов	226
5.3. Создание собственных анимационных эффектов	227
5.3.1. Эффект масштабирования	230
5.3.2. Эффект падения	231
5.3.3. Эффект рассеивания	232
5.4. Анимационные эффекты и очереди	234
5.4.1. Одновременное воспроизведение анимационных эффектов	234
5.4.2. Поочередное выполнение функций	237
5.4.3. Добавление функций в очередь анимационных эффектов	244
5.5. Итоги	245
Глава 6. За пределы DOM с помощью вспомогательных функций jQuery	247
6.1. Флаги jQuery.....	248
6.1.1. Запрет воспроизведения анимационных эффектов.....	249
6.1.2. Определение типа браузера.....	249
6.1.3. Флаги, определяющие тип браузера.....	254
6.2. Применение других библиотек совместно с jQuery	257
6.3. Управление объектами и коллекциями JavaScript.....	261
6.3.1. Усечение строк	261
6.3.2. Итерации по свойствам и элементам коллекций	262
6.3.3. Фильтрация массивов.....	264
6.3.4. Преобразование массивов	266
6.3.5. Другие полезные функции для работы с массивами JavaScript	269

6.3.6. Расширение объектов	271
6.3.7. Сериализация значений параметров	274
6.3.8. Проверка типов объектов	278
6.4. Различные вспомогательные функции	279
6.4.1. Пустая операция	279
6.4.2. Проверка на вхождение	280
6.4.3. Присоединение данных к элементам	280
6.4.4. Предварительная установка контекста функции	281
6.4.5. Синтаксический анализ строк в формате JSON	285
6.4.6. Вычисление выражений	286
6.4.7. Динамическая загрузка сценариев	287
6.5. Итоги	290
Глава 7. Расширение jQuery с помощью собственных модулей	292
7.1. Зачем нужны расширения?	293
7.2. Основные правила создания модулей расширения jQuery	294
7.2.1. Именованые функции и файлы	294
7.2.2. Остерегайтесь \$	296
7.2.3. Укращение сложных списков параметров	296
7.3. Создание собственных вспомогательных функций	299
7.3.1. Создание вспомогательной функции для манипулирования данными	301
7.3.2. Создание функции форматирования даты	303
7.4. Добавление новых методов обертки	307
7.4.1. Применение нескольких операций в методах обертки	310
7.4.2. Сохранение состояния внутри метода обертки	315
7.5. Итоги	329
Глава 8. Взаимодействие с сервером по технологии Ajax	330
8.1. Знакомство с Ajax	331
8.1.1. Создание экземпляра XMLHttpRequest	332
8.1.2. Инициализация запроса	334
8.1.3. Слежение за ходом выполнения запроса	336
8.1.4. Получение ответа	336
8.2. Загрузка содержимого в элемент	338
8.2.1. Загрузка содержимого с помощью jQuery	340
8.2.2. Загрузка динамических данных	343
8.3. Выполнение запросов GET и POST	348
8.3.1. Получение данных методом GET	351
8.3.2. Получение данных в формате JSON	353
8.3.3. Выполнение запросов POST	354
8.3.4. Каскады раскрывающихся списков	356

8.4. Полное управление запросами Ajax	362
8.4.1. Выполнение запросов Ajax со всеми настройками.....	363
8.4.2. Настройка запросов, используемых по умолчанию	366
8.4.3. Обработка событий Ajax	368
8.5. Соединяем все вместе	372
8.5.1. Реализация всплывающей подсказки.....	374
8.5.2. Опробование расширения Termifier	380
8.5.3. Место для усовершенствований	384
8.6. Итоги	385
Часть II. jQuery UI.....	387
Глава 9. Введение в jQuery UI: оформление и эффекты	389
9.1. Настройка и загрузка библиотеки jQuery UI	391
9.1.1. Настройка и загрузка	391
9.1.2. Использование библиотеки jQuery UI.....	392
9.2. Темы оформления в jQuery	394
9.2.1. Обзор.....	395
9.2.2. Использование инструмента ThemeRoller	399
9.3. Эффекты jQuery UI	401
9.3.1. Эффекты jQuery UI.....	402
9.3.2. Расширенные возможности базовых анимационных эффектов	409
9.3.3. Расширения методов управления видимостью.....	410
9.3.4. Расширения методов управления классами	411
9.3.5. Функции перехода	413
9.4. Улучшенный механизм позиционирования	416
9.5. Итоги	420
Глава 10. Механизмы взаимодействий jQuery UI с мышью	421
10.1. Перетаскивание объектов	423
10.1.1. Добавление способности к перетаскиванию	425
10.1.2. События, возникающие в процессе перетаскивания	433
10.1.3. Управление способностью к перетаскиванию.....	434
10.2. Отпускание перетаскиваемых элементов	436
10.2.1. Добавление способности к приему перетаскиваемых элементов	437
10.2.2. События отпускания элементов	441
10.3. Переупорядочение элементов.....	446
10.3.1. Добавление способности к переупорядочению.....	448
10.3.2. Подключение сортируемых списков.....	453
10.3.3. События переупорядочения	454
10.3.4. Получение информации о порядке следования элементов	456

10.4. Изменение размеров элементов	458
10.4.1. Добавление способности к изменению размеров	460
10.4.2. События, возникающие при изменении размеров элементов	465
10.4.3. Визуальное оформление вспомогательных элементов	466
10.5. Выделение элементов	467
10.5.1. Добавление способности к выделению	472
10.5.2. События, возникающие при выделении элементов	476
10.5.3. Поиск выделенных элементов	478
10.6. Итоги	479
Глава 11. Виджеты jQuery UI: за пределами элементов управления HTML	481
11.1. Кнопки и группы кнопок	483
11.1.1. Оформление внешнего вида кнопок с помощью тем.....	483
11.1.2. Добавление оформления к кнопкам.....	485
11.1.3. Значки для кнопок	488
11.1.4. События, порождаемые кнопками	489
11.1.5. Оформление кнопок	489
11.2. Ползунки.....	490
11.2.1. Создание ползунков	491
11.2.2. События, порождаемые ползунками	496
11.2.3. Советы по изменению оформления ползунков.....	498
11.3. Индикаторы хода выполнения операции	499
11.3.1. Создание индикаторов хода выполнения операции	501
11.3.2. События, порождаемые индикаторами хода выполнения операции	503
11.3.3. Расширение автоматического обновления индикатора хода выполнения операции	503
11.3.4. Оформление индикаторов хода выполнения операции	511
11.4. Виджеты с функцией автодополнения.....	512
11.4.1. Создание виджетов с функцией автодополнения	513
11.4.2. Источники данных для функции автодополнения....	517
11.4.3. События, порождаемые виджетом с функцией автодополнения	520
11.4.4. Оформление виджета с функцией автодополнения ...	521
11.5. Виджеты выбора даты.....	522
11.5.1. Создание виджетов выбора даты	523
11.5.2. Форматы представления дат, используемые виджетом	535

11.5.3. События, возбуждаемые виджетом выбора даты	537
11.5.4. Вспомогательные функции виджета выбора даты	537
11.6. Виджеты с вкладками.....	541
11.6.1. Создание многостраничного виджета с вкладками	541
11.6.2. События, порождаемые виджетами с вкладками.....	551
11.6.3. Оформление виджетов с вкладками.....	552
11.7. Многостраничные виджеты Accordion.....	553
11.7.1. Создание виджетов Accordion.....	554
11.7.2. События, порождаемые виджетом Accordion	561
11.7.3. Классы CSS, добавляемые к элементам виджета Accordion	562
11.7.4. Загрузка содержимого для панелей виджета Accordion с использованием Ajax.....	563
11.8. Диалоги.....	564
11.8.1. Создание диалогов	565
11.8.2. События, порождаемые диалогами.....	572
11.8.3. Имена классов CSS для диалогов.....	573
11.8.4. Некоторые приемы работы с диалогами	574
11.9. Итоги	576
11.10. Конец?.....	577
Приложение А.	
JavaScript: возможно, вы этого не знаете, а стоило бы!.....	578
А.1. Основные сведения об объекте Object языка JavaScript	579
А.1.1. Как создаются объекты	579
А.1.2. Свойства объектов	579
А.1.3. Литералы объектов	582
А.1.4. Объекты как свойства объекта window.....	584
А.2. Функции как обычные объекты	585
А.2.1. Что есть имя?.....	586
А.2.2. Функции обратного вызова	588
А.2.3. К чему это все?	589
А.2.4. Замыкания	594
А.3. Итоги	597
Алфавитный указатель	598

Список лабораторных работ

Селекторы	54
Операции.....	74
Перемещение/копирование.....	132
Эффекты.....	218
Функция \$.Param().....	277
Закругление углов	398
Эффекты jQuery UI.....	403
Влияние функций перехода на анимационные эффекты	414
Позиционирование	417
Способность к перетаскиванию	427
Прием перетаскиваемых объектов.....	438
Переупорядочение	453
Изменение размеров.....	461
Выделение элементов	470
Оформление кнопок	487
Параметры ползунков	493
Автодополнение	514
Выбор даты	525
Вкладки	546
Панель-гармошка	556
Диалог	570

Отзывы на первое издание

Отличная книга, которая лишний раз подтверждает высокое качество книг из серии «in Action», выпускаемых издательством Manning. Она очень легко читается и до краев наполнена примерами действующего программного кода. Лабораторные работы предоставляют увлекательный способ исследования библиотеки, которая обязательно должна войти в арсенал инструментов каждого веб-разработчика. Пять звезд – достойная оценка для этой книги, с какой стороны ни посмотри!

Дэвид Силлс (David Sills), JavaLobby, Dzone

Настоятельно рекомендую приобрести эту книгу – с ее помощью вы изучите основные принципы работы с библиотекой jQuery, а затем сможете использовать ее как отличный справочник, помогающий все шире и шире использовать jQuery в своих проектах.

Дэвид Хайден (David Hayden), MVP C#, Codebetter.com

«Азбука» для JavaScript.

Джошуа Хейер (Joshua Heyer), Trane Inc.

Тем, кто только приступает к изучению jQuery, эта книга послужит хорошим учебником, охватывающим полный спектр приемов использования этой библиотеки... Все примеры в этой книге имеют практическую ценность и являются важным дополнением к обсуждениям. Фрагменты программного кода визуально легко отличимы от остального текста, а сам текст читается легко и просто.

Грант Палин (Grant Palin), блогер

У авторов получилась очень интересная книга; вы прочтаете ее на одном дыхании и усвоите массу полезной информации.

Рич Страл (Rich Strahl), блогер

Благодаря усилиям авторов, Бера Бибо (Bear Bibeault) и Иегуды Каца (Yehuda Katz), а также их стилю подачи информации, достойному подражания, эту исчерпывающую книгу, или руководство по применению, как ее можно было бы назвать, можно использовать в качестве учебни-

ка для изучающих jQuery или в качестве справочника для тех, кто уже владеет навыками работы с библиотекой и ищет наиболее оптимальные решения.

*Мэтью Маккалоу (Matthew McCullough),
Denver Open Source Users Group*

Благодаря технической точности, широкому использованию примеров и доступному стилю изложения книга «jQuery in Action» является неоценимым источником информации для любого веб-разработчика, стремящегося использовать всю мощь JavaScript, и совершенно необходима для всех, кто желает освоить jQuery.

*Майкл Дж. Росс (Michael J. Ross),
веб-разработчик, участник проекта Slashdot*

8 из 10 за то, чтобы приобрести эту книгу! Если вы желаете освоить jQuery, эта книга послужит вам отличным учебником...

*Джон Виш (John Whish),
основатель Adobe ColdFusion User Group for Devon*

Настоятельно рекомендую приобрести эту книгу как начинающим, так и опытным разработчикам на JavaScript, желающим всерьез использовать JavaScript и писать оптимальный и изящный программный код, не испытывая сложностей, традиционных при разработке сценариев на JavaScript.

Val's Blog

Книга «jQuery in Action» представляет собой детальное исследование перспективной библиотеки jQuery, предназначенной для разработки клиентских сценариев на JavaScript.

www.DZone.com

Я считаю «jQuery in Action» отличной книгой, которая поможет вам освоить библиотеку jQuery. Я получил массу удовольствия от чтения этой книги.

*Гуннар Хиллерт (Gunnar Hillert),
Atlanta Java User Group*

Предисловие

Эта книга о простоте. Зачем веб-разработчику писать сложный программный код объемом с хорошую книгу, если все, что требуется, – это реализовать достаточно простые взаимодействия? Нигде не сказано, что программный код веб-приложений непременно должен быть сложным.

Еще только обдумывая создание jQuery, я решил, что сосредоточусь на компактном и простом программном коде, обслуживающем все практические приложения, с которыми веб-разработчики имеют дело каждый день. Прочитав книгу «jQuery in Action», я был приятно удивлен, поскольку увидел в ней прекрасную демонстрацию принципов, заложенных в библиотеке jQuery.

Благодаря своей практической направленности, выразительности представленного программного кода и удачной структуре книга «jQuery in Action» послужит идеальным источником информации для тех, кто стремится познакомиться с этой библиотекой.

Больше всего в этой книге мне понравилось то внимание, которое проявили Бер и Иегуда к деталям внутренних механизмов библиотеки. Они очень тщательно исследовали и описали jQuery API. Ежедневно я бывал удостоен электронного письма или мгновенного сообщения с просьбой пояснить что-нибудь, с сообщением о новой обнаруженной в библиотеке ошибке или с рекомендациями по ее улучшению. Вы можете быть уверены, что эта книга – один из самых продуманных и проработанных источников знаний о библиотеке jQuery.

Меня приятно удивило то обстоятельство, что в книге описаны подключаемые модули (plugins), а также тактика и теория, лежащие в основе разработки модулей jQuery. Именно модульная архитектура позволяет jQuery оставаться такой простой. Библиотека предоставляет множество документированных точек расширения, позволяющих наращивать ее функциональность путем подключения к ним модулей. Добавляемые функциональные возможности, будучи полезными, зачастую недостаточно универсальны, чтобы включить их в саму библиотеку jQuery, – это и делает необходимой поддержку модульной архитектуры. Некоторые из подключаемых модулей, например Forms, Dimension и LiveQuery, получили очень широкое распространение по вполне очевидным причинам: они грамотно написаны, хорошо документированы

и обеспечены технической поддержкой. Обязательно обратите особое внимание на то, как создаются и как используются модули, поскольку на них базируется jQuery.

С такими источниками информации, как эта книга, проект jQuery будет и дальше успешно развиваться. Раз уж вы решились изучать и применять jQuery, надеюсь, эта книга будет полезна и вам.

*Джон Ресиг (John Resig),
создатель библиотеки jQuery*

Введение ко второму изданию

Прошло два года с момента выхода первого издания этой книги. Действительно ли была необходимость выпустить следующее издание так скоро?

Безусловно!

По сравнению с устойчивым миром языков программирования, таких как Java, используемых для разработки компонентов веб-приложений, выполняемых на стороне сервера, клиентские веб-технологии развиваются чрезвычайно быстро. И jQuery не плетется в хвосте, а скорее находится на гребне этой волны!

Коллектив разработчиков jQuery выпускает новые основные версии библиотеки примерно раз в год (в последнее время они стараются выпускать новые версии каждый год в январе), кроме того, в течение года выпускается множество обновлений. Это означает, что с момента выхода первого издания книги, описывающего версию jQuery 1.2, на свет появились две новые основные версии библиотеки, jQuery 1.3 и jQuery 1.4, а также бесчисленное множество их обновлений!

С выходом каждой новой основной версии библиотека jQuery приобретает все больше и больше новых возможностей. Это и новые нестандартные события, пространства имен событий, функции и эффекты, и просто масса действительно полезных методов и функций. Как бы то ни было, но с момента выхода первого издания книги диапазон возможностей jQuery расширился весьма существенно.

И это даже без учета jQuery UI! В первом издании описанию зарождавшейся тогда библиотеки jQuery UI было посвящено всего несколько разделов в одной главе. С тех пор библиотека jQuery UI увеличилась в объеме и достигла определенной зрелости, поэтому в этом издании под ее описание была выделена заключительная часть книги, содержащая три полноценных главы.

Учитывая все сказанное выше, было принято решение выпустить второе издание книги, охватывающее все новшества, появившиеся в jQuery и jQuery UI за последние два года.

Что нового во втором издании?

Когда мы решили приступить к работе над вторым изданием нашей книги, кто-то сказал мне: «Это будет всего лишь кусок пирога. В конце концов, вам нужно просто обновить первое издание».

Как же неправ он оказался! Фактически работа над вторым изданием заняла больше времени, чем над первым. Дело в том, что мы не хотели «халтурить», заниматься простым добавлением изменений и называть это работой. Мы хотели, чтобы второе издание было больше, чем просто дополненная версия первого издания.

Любой, кто попытается сравнить оглавление первого и второго изданий этой книги, без труда заметит, что структура глав с 1 по 8 не претерпела существенных изменений. Но на этом сходство этих двух изданий и заканчивается.

Второе издание не следует считать слегка переделанной версией первого издания, в которую местами были добавлены новые сведения. Каждый абзац текста и каждая строка в примерах были тщательно пересмотрены. Мало того что мы приняли во внимание все изменения, внесенные в jQuery между версиями 1.2 и 1.4, мы также обновили информацию в главах и примеры, чтобы отразить последние достижения в области веб-разработки и использования jQuery. В конце концов, в течение более чем двух лет сообщество продолжало приобретать опыт разработки интерактивных веб-страниц с применением библиотеки jQuery.

Каждый пример в книге был пересмотрен и либо изменен с целью продемонстрировать новые возможности версии jQuery 1.4, либо заменен новым примером, более наглядно демонстрирующим обсуждаемые понятия. Например, читатели первого издания могут вспомнить объемистый пример формы заказа для ресторана азиатской кухни «*Bamboo Asian Grille*» в конце главы 4, демонстрирующий возможности обработки событий, предусмотренные в jQuery. Если бы мы оставили все как есть, нам не удалось бы блеснуть новейшими возможностями jQuery, связанными с обработкой событий, такими как «оперативные» и нестандартные события. Поэтому нам пришлось полностью заменить его примером «*DVD Ambassador*», представляющим собой интерфейс к базе данных с дисками DVD, который лучше подходит для демонстрации вновь появившихся возможностей обработки событий.

Вторая, совершенно новая часть книги полностью посвящена библиотеке jQuery UI и охватывает обширнейшие изменения, которые были внесены в библиотеку jQuery UI с момента публикации первого издания.

По нашим оценкам, второе издание, по крайней мере наполовину, является совершенно новой книгой, если учесть дополнения, изменения и обновления в первой части, а также принять во внимание совершенно новую вторую часть книги. Вторая половина подверглась обширной переделке, чтобы обеспечить ее соответствие современным достижениям в веб-разработке. Довольно много для «всего лишь куска пирога»!

Введение к первому изданию

Один из авторов книги – седой ветеран, начавший программировать, когда появление языка FORTRAN было подобно взрыву бомбы, а второй – не по годам сообразительный специалист в данной области, не заставший еще то время, когда не было Интернета. Что объединило этих двух человек с таким разным жизненным опытом для совместной работы над книгой?

Ответ очевиден – библиотека *jQuery*.

Мы пришли к этому, одному из наиболее интересных инструментов создания клиентских приложений путями разными, как день и ночь.

Я (Бер) впервые услышал о *jQuery*, когда работал над книгой «Ajax in Practice»¹. Заключительный и самый лихорадочный этап работы над книгой называется *редактированием*, – когда, кроме всего прочего, технический редактор проверяет грамматическую правильность текста и ясность изложения информации. Это самый напряженный, по крайней мере для меня, период, когда меньше всего мне хотелось бы услышать: «Определенно, здесь нужен еще один раздел».

Одна из глав, написанных мною для книги «Ajax in Practice», рассматривает несколько библиотек поддержки технологии Ajax на стороне клиента, с одной из которых я уже был очень близко знаком (Prototype), а с другими (Dojo Toolkit и DWR) мне пришлось познакомиться очень быстро.

Жонглируя множеством задач (все время оставаясь в курсе дел, руководя своим бизнесом и решая семейные проблемы), технический редактор Валентин Креттаз (Valentin Crettaz) случайно обронил: «А почему у вас нет раздела о *jQuery*?»

«О джей чём?», – спросил я.

И тут же прослушал лекцию о том, как прекрасна эта совершенно новая библиотека и что ее непременно следует включать в любое современное исследование библиотек, обеспечивающих поддержку технологии Ajax на стороне клиента. Я поспрашивал вокруг, не слыхал ли кто о библиотеке *jQuery*?

¹ Дейв Крейн, Бер Бибо, Джордон Сонневельд «Ajax на практике». – К.: Вильямс, 2008.

Многие ответили положительно, с восторгом подтвердив, что jQuery – действительно превосходная библиотека. Дождливым воскресным днем я провел четыре часа на сайте jQuery, изучая документацию и пробуя писать маленькие тестовые программы. Затем я написал новый раздел и отправил его техническому редактору, чтобы убедиться в том, что правильно понял суть библиотеки.

Раздел был принят на ура, и заключительный этап работы над книгой «Ajax in Practice» продолжился. (Добавлю, что раздел о jQuery был опубликован в электронном журнале «Dr. Dobb's Journal».)

Когда пыль улеглась, на задворках сознания пустила ростки моя суматошная попытка понять суть jQuery. Мне понравилось то, что я увидел во время своего стремительного погружения, и захотелось познакомиться с библиотекой поближе. Я стал использовать jQuery в веб-проектах. То, что получилось, мне тоже понравилось. Я пробовал заменять старый программный код предыдущих проектов, чтобы увидеть, насколько проще становятся веб-страницы благодаря jQuery. И мне действительно понравился полученный результат.

Восхищенный своим новым открытием и подгоняемый желанием поделиться им с другими, я послал в издательство Manning предложение написать книгу «jQuery in Action». Разумеется, я должен был их убедить. (Из чувства мести за переполох я предложил своему техническому редактору, с которого все началось, стать техническим редактором и этой книги. Клянусь, это послужило ему хорошим уроком!)

И тут редактор Майк Стефенс (Mike Stephens) спросил: «А не хотели бы вы работать над книгой вместе с Иегудой Кацем (Yehuda Katz)?»

«С Иеентой кем?», – спросил я...

Иегуда пришел в этот проект совершенно другим путем; он был знаком с библиотекой jQuery еще в те дни, когда она даже не имела номера версии. Наткнувшись на модуль Selectables, он заинтересовался основной библиотекой jQuery. Несколько разочарованный недостатком (в то время) электронной документации, он обыскал различные wiki-ресурсы¹ и основал сайт Visual jQuery (visualjquery.com).

В скором времени он дал толчок появлению отличной документации, стал оказывать помощь проекту jQuery и следить за модульной архитектурой и экосистемой, попутно пропагандируя jQuery в сообществе пользователей Ruby.

¹ Те, кого интересует происхождение и история wiki-проектов, могут заглянуть, например, сюда: wiki.org/wiki.cgi?WhatIsWiki – что такое Wiki, c2.com/cgi/wiki?WikiWikiWeb – распространенный термин, c2.com/cgi/wiki?InformalHistoryOfProgrammingIdeas – «ты помнишь, как все начиналось...». – *Прим. науч. ред.*

И вот настал день, когда ему позвонили из издательства Manning (издателю его порекомендовал друг) и предложили поработать над книгой о jQuery вместе с парнем по имени Бер...

Несмотря на разницу в возрасте, опыте и путях, которыми мы пришли в этот проект, из нас получилась отличная команда и мы прекрасно провели время, работая над книгой. Даже географическая разобщенность (я живу в сердце Техаса, а Йегуда – на побережье Калифорнии) не стала для нас препятствием благодаря электронной почте и системе обмена мгновенными сообщениями!

Думаем, что композиция наших знаний и талантов позволила сделать добротную и информативную книгу для вас. Надеемся, что, читая ее, вы получите столько же удовольствия, сколько и мы, работая над ней.

Рекомендуем только выбрать наиболее подходящее для чтения время.

Благодарности

Вас не удивляет длинный список имен, пробегающих по экрану в конце фильма? Неужели для создания фильма действительно требуется столько людей?

В работе над книгой тоже участвует столько людей, что большинство из вас удивится. Многие должны вложить в книгу свой труд и талант, чтобы у вас в руках оказался этот томик (или электронная книга, которую вы читаете с экрана).

Специалисты издательства Manning усердно трудились вместе с нами, чтобы книга получилась хорошей, и мы благодарны им за это. Без них эта книга не состоялась бы. В «заключительные титры» этой книги попали имена не только нашего издателя Марьян Бэйс (Marjan Base) и редактора Майка Стефенса, но и следующих сотрудников: Лианна Власюк (Lianna Wlasiuk), Карен Тегтмайер (Karen Tegtmaier), Энди Кэрролл (Andy Carroll), Дипак Вохра (Deepak Vohra), Барбара Мирецки (Barbara Mirecki), Меган Йоки (Megan Yockey), Дотти Марсико (Dottie Marsico), Мэри Пиргис (Mary Piergies), Габриель Добреску (Gabriel Dobrescu) и Стивен Хонг (Steven Hong).

Трудно переоценить труд наших рецензентов, которые помогали довести книгу до совершенства, отыскивали опечатки, исправляли ошибки в терминологии и в программном коде и даже помогли организовать структуру глав в книге. Каждый цикл рецензирования существенно улучшал книгу. За время, потраченное на рецензирование книги, мы хотим поблагодарить Тони Ниманна (Tony Niemann), Скотта Сьюета (Scott Sauyet), Рича Фрийдмана (Rich Freedman), Филиппа Халлстрема (Philip Hallstrom), Майкла Смоляка (Michael Smolyak), Мэрион Стюртвант (Marion Sturtevant), Йонаса Банди (Jonas Bandi), Джея Бланчарда (Jay Blanchard), Никандра Брюггман (Nikander Bruggeman), Маргрит Брюггман (Margriet Bruggeman), Грега Дональда (Greg Donald), Френка Ванга (Frank Wang), Кертиса Миллера (Curtis Miller), Кристофера Хаупта (Christopher Haupt), Черил Джерозал (Cheryl Jerozal), Чарльза Е. Логстона (Charles E. Logston), Эндрю Симера (Andrew Siemer), Эрика Реймонда (Eric Raymond), Кристиана Маркуардта (Christian Marquardt), Робби О'Коннора (Robby O'Connor), Марка Гравелла (Marc Gravell), Эндрю Гроса (Andrew Grothe), Анила Радхакришна (Anil Radhakrishna), Дэниела Бретой (Daniel Bretoi) и Массимо Перга (Massimo Perga).

Отдельное спасибо Валентину Креттазу, техническому редактору книги. Помимо проверки всех примеров программного кода в разных окружениях, он предложил массу ценных рекомендаций по повышению технической точности текста, добавлял сведения, которые первоначально отсутствовали в книге, следил за развитием библиотек, пока мы были заняты работой над книгой, благодаря чему в книгу попали самые свежие сведения о jQuery и jQuery UI, и даже написал серверные сценарии на языке PHP, необходимые для опробования примеров из книги.

Бер Бибо (Bear Bibeault)

Это моя четвертая публикация, и список тех, кого я хотел бы поблагодарить, включая всех членов и персонал *jav ranch.com*, довольно велик. Без моей причастности к проекту JavaRanch я никогда не смог бы начать писать, и поэтому я искренне благодарен Полу Ветону (Pol Wheaton) и Кэти Сиерра (Kathy Sierra), с которых все это началось, а также другим штатным сотрудникам, которые подбадривали и поддерживали меня, включая (как минимум) Эрика Паскарелло (Eric Pascarello), Бена Соузера (Ben Souther), Эрнеста Фридмана Хилла (Ernest Friedman Hill), Марка Хершберга (Mark Herschberg), Эндрю Манкхауза (Andrew Munkhouse), Джинни Боярски (Jeanne Boyarski), Берта Бейтса (Bert Bates) и Макса Хаббиби (Max Habbibi).

Выражаю свою благодарность Валентину Креттазу: он не только был техническим редактором, но и первым ввел меня в мир jQuery (как уже отмечалось выше).

Мой партнер Джей (Jay) и собаки – Литл Бер и Козмо (чьи изображения украшают страницы этой книги), – спасибо вам за ваше незаметное присутствие и за то, что делили со мной крышу над головой, почти не задевая клавиатуру MacBook Pro в течение всех месяцев, пока я работал над этой книгой.

Наконец, я хочу поблагодарить моего соавтора Иегуду Каца, без которого этот проект не смог бы воплотиться, а также Джона Ресига (John Resig) и других разработчиков jQuery и jQuery UI.

Иегуда Кац (Yehuda Katz)

Для начала я хотел бы выразить свою благодарность моему соавтору Бери Бибо за то, что нам дал его огромный опыт в работе над книгами. Рождению этой книги в огромной мере способствовали его писательский талант и удивительная способность преодолевать препятствия, возникающие на пути профессиональной публикации.

Заговорив о тех, благодаря кому все стало возможным, считаю своим долгом поблагодарить мою любимую жену Леа (Leah), которой при-

шлось мириться с моим отсутствием в течение стольких ночей и стольких выходных, что заранее я не решился бы попросить об этом. Ее вклад в эту книгу сравним с моим собственным; как всегда, благодаря ей я смог перенести самый сложный этап проекта. Я люблю тебя, Леа.

Совершенно очевидно, что без библиотеки jQuery не было бы и книги «jQuery in Action». Я хочу выразить свою благодарность создателю jQuery Джону Ресигу за то, что он изменил процесс разработки клиентских сценариев, облегчив работу веб-разработчиков всей планеты (может, в это трудно поверить, но у нас есть довольно большие группы разработчиков в Китае, Японии, Франции и многих других странах). Я считаю его своим другом, который, будучи сам талантливым автором, помог мне подготовиться к этому нелегкому предприятию.

Никакой библиотеки jQuery не было бы без огромного сообщества пользователей и членов основной команды, включая разработчиков Брендона Аарона (Brandon Aaron) и Йорна Зафферера (Jörn Zaefferer), популяризаторов Рея Банго (Rey Bango) и Карла Шведберга (Rarl Swedberg), Пауля Бакауса (Paul Bakaus), возглавляющего проект jQuery UI, а также Клауса Хартла (Klaus Hartl) и Майка Алсупа (Mike Alsup), работающих в команде разработки модулей вместе со мной. Эта большая группа программистов помогала продвигать платформу jQuery от простого набора базовых операций до библиотеки JavaScript мирового уровня, содержащую обеспеченную усилиями пользователей (и модульную) поддержку практически всего, что только можно пожелать. Я не привожу полный список тех, кто участвовал в работе, – вас так много! Достаточно сказать, что я не проделал бы все это без уникального сообщества, сплотившегося вокруг этой библиотеки, и у меня не хватает слов, чтобы выразить свою благодарность.

Наконец, я хотел бы поблагодарить свою семью, которую я почти не вижу с тех пор, как начались мои поездки по стране. По мере взросления я и мои братья впитали дух товарищества, а вера семьи в меня всегда придавала мне сил и уверенности. Мама, Никки, Эби и Иаков, спасибо вам, я люблю вас.

Об этой книге

Делай больше меньшими усилиями.

Эти слова просто и понятно выражают цель этой книги: помочь вам узнать, как наполнить веб-страницы большей функциональностью при меньшем объеме сценариев. Авторы книги, один из которых сотрудник и популяризатор (evangelist) jQuery, а другой – увлеченно-восторженный пользователь, полагают, что jQuery – лучшая из доступных на сегодняшний день библиотек, которая позволит вам сделать это.

Эта книга поможет вам быстро и эффективно овладеть jQuery и, надеюсь, сделает это занятие увлекательным. Здесь рассматривается собственно ядро библиотеки jQuery и ее спутницы – библиотеки jQuery UI. Каждый метод прикладного программного интерфейса описан в удобном для восприятия формате, включающем описание входных параметров и возвращаемых значений. В книге присутствует множество небольших примеров эффективного использования библиотеки, а для представления важных концепций мы предусмотрели *лабораторные работы*. Эти забавные проверочные страницы представляют собой прекрасный способ увидеть нюансы использования методов jQuery без необходимости писать много своего программного кода.

Все исходные тексты примеров и лабораторных работ можно загрузить по адресу: <http://www.manning.com/jqueryinActionSecondEdition> или <http://www.manning.com/bibeault2>.

Мы могли бы еще долго расписывать, как хороша эта книга, с применением разных маркетинговых ходов, но кому интересно читать эту чепуху, не так ли? Ведь в действительности вам хочется погрузиться в биты и байты, ведь так? Именно в этом и есть назначение данной книги!

Так что же вас держит? Вперед!

Для кого эта книга

Книга адресована как начинающим, так и опытным веб-разработчикам, желающим задействовать сценарии JavaScript на своих веб-страницах и создавать настоящие, интерактивные веб-приложения без необходимости писать весь клиентский программный код, что обычно требуется при создании приложений на пустом месте.

Пользу из этой книги извлекут все веб-разработчики, стремящиеся создавать с помощью библиотеки jQuery веб-приложения, которые бы восхищали, а не раздражали пользователей.

При чтении некоторых разделов начинающий веб-разработчик может почувствовать себя несмышленишем, но это не должно его останавливать. Мы включили в книгу приложение с описанием основных понятий JavaScript, помогающих освоить весь потенциал jQuery. Поняв основные концепции, такой читатель обнаружит, что сама библиотека jQuery очень дружелюбна к новичкам, будучи при этом достаточно мощной для более опытных веб-разработчиков.

Как новички, так и ветераны разработки веб-приложений выиграют от включения jQuery в свой арсенал. Мы ручаемся, что эта книга поможет вам быстро изучить все, что для этого нужно.

Структура книги

Книга организована так, чтобы вы смогли овладеть jQuery и jQuery UI максимально быстро и эффективно. Она начинается с введения в основы jQuery и сразу переходит к фундаментальным концепциям прикладного программного интерфейса jQuery. После этого мы продемонстрируем вам различные области применения, где jQuery позволяет писать невероятно продуктивный клиентский программный код – от обработки событий до выполнения запросов к серверу с применением технологии Ajax. Затем вашему вниманию будет представлен обзор наиболее популярных расширений jQuery.

Книга делится на две части: первая охватывает ядро библиотеки jQuery, а вторая описывает библиотеку jQuery UI. Часть 1 содержит восемь глав.

В главе 1 мы рассмотрим философию jQuery и то, как она согласуется с такими современными принципами программирования, как «ненавязчивый JavaScript». Мы исследуем причины, по которым было бы желательно использовать jQuery, вкратце рассмотрим принцип ее действия и такие базовые концепции, как обработчики события готовности документа, вспомогательные функции, создание элементов объектной модели документа (Document Object Model, DOM) и расширений для jQuery.

В главе 2 представлена концепция обернутых наборов элементов – базовая концепция jQuery, определяющая принцип ее действия. Вы узнаете, как создавать обернутые наборы элементов – коллекции элементов DOM, участвующие в операции как единое целое, – за счет выбора элементов документа страницы с помощью богатой коллекции мощных селекторов jQuery. При определении этих селекторов используется стандартный синтаксис CSS, что делает их достаточно мощным инструментом и расширяет возможности, предлагаемые стандартными способами применения каскадных таблиц стилей (CSS).

В главе 3 мы узнаем, как можно использовать обернутые наборы jQuery для управления структурой DOM страницы. Мы рассмотрим вопро-

сы изменения стилей и атрибутов элементов, содержимого элементов, а также перемещения элементов в пределах страницы и изучим принципы работы с элементами формы.

В главе 4 показано, как с помощью jQuery существенно упростить обработку событий в веб-страницах. В конце концов, именно обработка пользовательских событий делает веб-приложения полнофункциональными, и все, кому уже приходилось иметь дело с запутанными механизмами обработки событий в разных браузерах, порадуются простоте данной задачи с использованием jQuery. Дополнительные концепции, связанные с обработкой событий, такие как определение пространств имен событий, возбуждение и обработка собственных событий и даже предварительная установка обработчиков «живых» событий, будут детально рассмотрены в обширном примере в конце главы.

Мир анимации и визуальных эффектов станет предметом обсуждения главы 5. Здесь мы увидим, что jQuery позволяет создавать анимационные эффекты не только просто, но и эффективно, и даже забавно. В этой главе во всех подробностях будут рассматриваться функции создания очередей эффектов для последовательного их выполнения, а также ряд функций общего назначения.

В главе 6 мы познакомимся со вспомогательными функциями и флагами, которые jQuery предоставляет в распоряжение не только авторов страниц, но и всех, кто пишет расширения и модули для jQuery.

О том, как пишутся расширения и модули, будет рассказано в главе 7. Мы увидим, насколько просто пишутся расширения для jQuery; для этого не потребуется писать хитроумный программный код JavaScript или знать устройство jQuery. Поэтому есть смысл всякий программный код многократного использования оформлять в виде расширения для jQuery.

Глава 8 заинтересует вас одной из наиболее важных областей в разработке современных веб-приложений: выполнение запросов Ajax. Здесь мы увидим, насколько просто применять технологию Ajax с помощью jQuery и как эта библиотека ограждает нас от ловушек, которыми чревато введение поддержки технологии Ajax в страницы, существенно упрощая реализацию наиболее распространенных видов взаимодействий Ajax (таких как возврат данных в формате JSON¹). В этой главе приводится еще один объемный пример, демонстрирующий применение всего того, что было изучено к этому моменту.

Во второй части, состоящей из трех глав, мы займемся исследованием сопутствующей библиотеки: jQuery UI.

¹ JSON (JavaScript Object Notation) – формат обмена данными; основан на подмножестве языка программирования JavaScript, определенного в стандарте ECMA-262 3rd Edition, December 1999 (json.org/json-ru.html). – *Прим. науч. ред.*

В главе 9 мы познакомимся с библиотекой jQuery UI и узнаем, как настраивать и создавать свои ограниченные версии этой библиотеки, а также темы визуального оформления, используемые для оформления элементов пользовательского интерфейса из библиотеки. Темы визуального оформления будут рассматриваться так, чтобы показать вам не только, как они конструируются, но и как их можно изменить, чтобы они соответствовали нашим потребностям. В конце этой главы рассматриваются дополнительные анимационные эффекты, добавляемые библиотекой jQuery UI, а также способы применения базовых методов, позволяющие использовать эти расширенные особенности.

В главе 10 мы исследуем возможности организации взаимодействий пользователя с веб-страницей с помощью мыши, предоставляемые библиотекой jQuery UI. Здесь будет рассматриваться широкий диапазон приемов, от перетаскивания элементов с помощью мыши до сортировки, выделения и изменения размеров элементов.

Наконец, в главе 11 мы исследуем набор виджетов (визуальных элементов управления пользовательского интерфейса), предоставляемый библиотекой jQuery UI, расширяющий доступный набор механизмов ввода, которые мы можем задействовать в своих страницах. Сюда входят и простые элементы управления, такие как кнопки, и более сложные, такие как элементы выбора даты, поля ввода с функцией автодополнения, панели с вкладками и диалоговые окна.

В довершение всего в приложении описаны такие ключевые концепции JavaScript, как контексты функции и замыкания (closures), составляющие основу для максимально эффективного использования библиотеки jQuery в наших страницах. Это приложение предназначено для тех, кто хочет освежить свои знания об этих концепциях.

Лабораторные работы и упражнения

На протяжении всей книги вам будут встречаться особые лабораторные работы, которые добавлены с целью проиллюстрировать обсуждаемые понятия jQuery и jQuery UI. Они представляют собой интерактивные веб-страницы, реализованные в виде загружаемых файлов примеров, которые вы можете запускать на своем компьютере.

Лабораторная работа

Каждая лабораторная работа будет обведена рамкой с соответствующим заголовком, что позволит вам легко отыскивать ее в тексте. Кроме того, для удобства список всех лабораторных работ приведен сразу вслед за оглавлением. Ссылки, которые приводятся в описаниях, указывают на основную страницу загружаемого примера.

Можно также получить удаленный доступ к лабораторным работам и остальным примерам на сайте одного из авторов <http://www.bibeault.org/jqia2> или на сайте издательства <http://www.manning.com/jQueryinActionSecondEdition>.

Кроме того, в книге часто будут встречаться рамки, указывающие на упражнения для самостоятельного выполнения.

Упражнение

Чаще всего упражнения будут связаны с какой-либо лабораторной работой, но иногда будут логическим продолжением примеров, описываемых в книге, или простыми самостоятельными упражнениями, которые следует выполнить, чтобы убедиться, что все обсуждаемые понятия усвоены правильно.

Соглашения по оформлению примеров программного кода

Программный код в листингах примеров или в тексте выполнен моноширинным шрифтом, как, например, этот текст, чтобы отделить его от обычного текста. Имена методов и функций, свойств, элементов XML и атрибутов также выделяются этим шрифтом.

В некоторых случаях первоначальный программный код может быть отформатирован с целью уместить его на книжной странице. Вообще, программный код изначально был написан с учетом ограниченной ширины книжной страницы, но иногда вы можете заметить незначительные различия между оформлением листингов и исходных программных кодов примеров, доступных для загрузки. В нескольких редких случаях, когда длинная строка не может быть отформатирована без изменения ее смысла, в листинге появится маркер продолжения строки.

Большая часть листингов сопровождается описанием, в котором подчеркиваются наиболее важные концепции. Часто в листингах присутствуют нумерованные маркеры для ссылок из пояснительного текста.

Загрузка программного кода примеров

Исходный программный код всех приведенных в книге примеров (а также ряд дополнительных примеров, не попавших в книгу) доступен для загрузки на странице <http://www.manning.com/jQueryinActionSecondEdition> или <http://www.bibeault.org/jqia2/>.

Примеры программного кода организованы в виде единого веб-приложения с отдельными разделами для каждой главы, которое легко может быть размещено на локальном веб-сервере, таком как Apache HTTP Server. Распакуйте загруженный архив с примерами в любую папку и назначьте ее корневой папкой документов веб-сервера. Начальная страница, откуда можно запускать примеры, находится в корневой папке в файле *index.html*.

За исключением примеров для главы 8 и части примеров из глав, описывающих библиотеку jQuery UI, основная часть примеров может быть опробована без использования локального веб-сервера – их можно загружать непосредственно в браузер. Примеры, иллюстрирующие применение технологии Ajax, требуют чуть больше возможностей, чем может предоставить веб-сервер Apache, поэтому для их опробования вам потребуется либо установить поддержку сценариев на языке PHP в Apache, либо настроить веб-сервер с поддержкой возможности выполнения сервлетов Java и страниц JavaServer Pages (JSP), такой как Tomcat. Инструкции по установке Tomcat – веб-сервера для примеров из главы 8 – приведены в файле *chapter8/tomcat.pdf*.

Все примеры были опробованы в разных типах браузеров, включая Internet Explorer 7 и 8, Firefox 3, Safari 3 и 4 и Google Chrome.

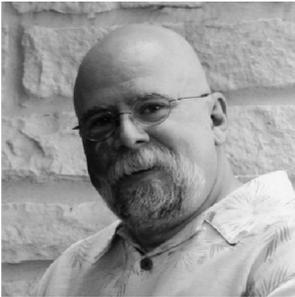
Форум Author Online

Покупка книги «jQuery in Action» дает право свободного доступа к частному форуму, который поддерживается издательством Manning Publications, где можно оставлять свои комментарии о книге, задавать вопросы технического характера и получать помощь от авторов книги и от других пользователей. Чтобы получить доступ к форуму, откройте в браузере страницу <http://www.manning.com/jqueryinAction-SecondEdition> и щелкните на ссылке Online. Здесь вы найдете информацию о том, как попасть на форум после регистрации, какого рода помощь будет доступна и о правилах поведения на форуме.

Издательство Manning обязуется предоставить своим читателям место встречи, где может завязаться диалог между читателями и авторами книги. Но издательство не гарантирует присутствие всех или части авторов на форуме – их содействие форуму книги остается добровольным (и неоплачиваемым). Рекомендуем задавать такие вопросы, которые могли бы вызвать интерес у авторов!

Форум Author Online и архивы предыдущих обсуждений доступны на веб-сайте издателя.

Об авторах



Бер Бибо разрабатывает программное обеспечение уже более тридцати лет, начав с создания программы Tic-Tac-Toe на суперкомпьютере Control Data Cyber, где в качестве устройства ввода применялся теле-тайп со скоростью обмена 100 бод. Поскольку у Бера два электротехнических диплома, он должен был бы проектировать антенны или что-то вроде того, но, начав работать в Digital Equipment Corporation, все больше тянулся к программированию.

Бер успел поработать на Lightbridge Inc., BMC Software, Dragon Systems и даже служил в вооруженных силах США, где обучал солдат-пехотинцев взрывать танки, – эти навыки он теперь с успехом использует на ежедневных производственных совещаниях. В настоящее время Бер – разработчик архитектур программных комплексов в ведущей компании, предоставляющей услуги распределенных вычислений.

Кроме повседневной работы Бер пишет книги, руководит небольшой собственной фирмой, которая создает веб-приложения и предлагает другие электронные услуги (только не видеосъемку свадеб, никакой свадебной видеосъемки!), а также помогает поддерживать порядок на сайте *JavaRanch.com* под псевдонимом «sheriff» (старший модератор). Когда не сидит за компьютером, Бер любит приготовить *много* еды (чем и объясняется размер его джинсов), увлекается фото- и видеосъемкой, катается на своем мотоцикле Yamaha V-Star и носит футболки с тропическими картинками.

Живет и работает в городе Остин (Austin), штат Техас, в котором нежно любит все, за исключением абсолютно ненормальных водителей.



Иегуда Кац за последние несколько лет поучаствовал в разработке нескольких проектов с открытым исходным кодом. Он не только член основной команды проекта jQuery, но также участник проекта Merb, альтернативы Ruby on Rails (также реализованной на языке Ruby).

Иегуда родился в штате Миннесота, рос в Нью-Йорке, а теперь живет в солнечной Калифорнии, в городе Санта-Барбара. Он занимался разработкой веб-сайтов для *New-York Times*, *Allure Magazine*, *Architectural Digest*, *Yoga Journal* и других известных клиентов. Программирует на многих языках, включая Java, Ruby, PHP и JavaScript.

В свободное от работы время помогает поддерживать сайт *VisualjQuery.com* и отвечает на вопросы начинающих пользователей jQuery на канале IRC и в официальной почтовой рассылке jQuery.

I

Ядро jQuery

Когда люди слышат название jQuery, они, как правило, полагают, что речь идет о ядре библиотеки jQuery. Однако под этим названием также подразумевается целая экосистема, возникшая вокруг ядра, которая содержит дополнительные, сопутствующие библиотеки, такие как jQuery UI (которой посвящена вторая часть этой книги), официальные расширения (<http://plugins.jquery.com/>) и бесчисленное множество расширений, созданных сторонними разработчиками, которые легко можно отыскать в Интернете с помощью любой поисковой системы. (Поиск в Google по фразе «jQuery plugin» принесет вам более 4 миллионов ссылок!)

Как бурное развитие рынка расширений сторонних разработчиков для iPod компании Apple обусловлено существованием самого iPod, так и причиной всего этого изобилия вокруг jQuery можно достоверно считать достоинства ядра библиотеки jQuery.

В следующих восьми главах первой части этой книги мы исследуем ядро библиотеки от основания и до самого верха. Закончив читать эти главы, вы будете иметь полное представление о возможностях библиотеки jQuery и сможете приступить к разработке любых веб-приложений, задействовав один из самых мощных инструментов разработки клиентских сценариев. Кроме того, вы будете готовы к использованию библиотек, сопутствующих jQuery, которые, подобно различным принадлежностям для iPod, совершенно бесполезны без основной библиотеки.

Итак, переверните страницу, и мы предоставим вам возможность узнать, что процесс создания интерактивных веб-приложений – это не только простое, но и увлекательное занятие!

1

Введение в jQuery

В этой главе:

- Для чего нужна библиотека jQuery
- Что такое «ненавязчивый JavaScript»
- Основные элементы и понятия jQuery
- jQuery и другие библиотеки JavaScript

Язык JavaScript, большую часть своей жизни считавшийся среди серьезных веб-разработчиков «игрушечным», обрел нынешний авторитет на волне интереса к полнофункциональным веб-приложениям и технологиям Ajax. Языку пришлось очень быстро повзрослеть, так как разработчики клиентских сценариев отказались от приема копирования и вставки программного кода JavaScript в пользу полноценных библиотек, которые позволяют решать сложные проблемы, связанные с различиями между браузерами разных типов, и реализуют новые и улучшенные парадигмы разработки веб-приложений.

jQuery, появившаяся в мире библиотек JavaScript с некоторым опозданием, покорила сообщество веб-разработчиков, быстро завоевав поддержку крупных компаний, использующих веб-приложения для решения важных задач. В число наиболее заметных пользователей jQuery входят такие компании, как IBM, Netflix, Amazon, Dell, Best Buy, Twitter, Bank of America и множество других известных компаний. Компания Microsoft даже включила библиотеку jQuery в состав инструментов, поставляемых вместе со средой разработки Visual Studio, а компания Nokia использует jQuery во всех своих телефонах, включающих их компонент Web Runtime.

И это вовсе *не* случайно!

В отличие от других инструментов, сконцентрированных на применении сложных методик JavaScript, jQuery стремится изменить представление веб-разработчиков о принципах создания полнофункциональных веб-приложений. Вместо того чтобы тратить время на жонглирование непростыми возможностями языка JavaScript, разработчики получили возможность с помощью каскадных таблиц стилей (Cascading Style Sheets, CSS), расширенного языка разметки гипертекста (eXtensible Hypertext Markup Language, XHTML) и старого доброго JavaScript напрямую манипулировать элементами страницы и воплотить мечту о быстрой разработке веб-приложений.

В этой книге мы во всех подробностях рассмотрим, что нам может предложить jQuery как авторам полнофункциональных интерактивных веб-приложений. Для начала узнаем, что в действительности может дать jQuery при разработке веб-страниц.

Последнюю версию jQuery можно получить на сайте проекта jQuery: <http://jquery.com/>. Установка jQuery заключается в простом сохранении файлов библиотеки в пределах досягаемости веб-приложения и использовании HTML-тега `<script>`, подключающего библиотеку к вашей странице, например:

```
<script type="text/javascript"
      src="scripts/jquery-1.4.js"></script>
```

Вместе с загружаемыми примерами программного кода распространяется версия библиотеки jQuery, с помощью которой выполнялось тестирование этих примеров (доступна по адресу <http://www.manning.com/bibeault2>).

1.1. Больше возможностей, меньше кода

Потратив некоторое время на попытки привести динамическую функциональность в ваши страницы, вы обнаружите, что постоянно следуете одному и тому же шаблону: сначала отбирается элемент или группа элементов, а затем над ними выполняются некоторые действия. Вы можете скрывать или показывать элементы, добавлять к ним класс CSS, создавать анимационные эффекты или изменять атрибуты.

С обычным JavaScript для решения каждой из задач потребуются десятки строк программного кода. Создатели jQuery разработали свою библиотеку именно для того, чтобы сделать наиболее общие задачи тривиальными. Например, любой, кому приходилось иметь дело с группами радиокнопок на JavaScript, знает, насколько это утомительно — определить, какая из радиокнопок в группе отмечена в настоящий момент времени, и получить значение ее атрибута `value`. Для этого необходимо отыскать радиогруппу, последовательно обойти все радиокнопки в группе, отыскать элемент с установленным атрибутом `checked` и затем получить значение атрибута `value` этого элемента.

Ниже приводится вариант реализации такой задачи на JavaScript:

```
var checkedValue;
var elements = document.getElementsByTagName('input');
for (var n = 0; n < elements.length; n++) {
    if (elements[n].type == 'radio' &&
        elements[n].name == 'someRadioGroup' &&
        elements[n].checked) {
        checkedValue = elements[n].value;
    }
}
```

Для сравнения ниже показано, как ту же задачу можно решить, используя библиотеку jQuery:

```
var checkedValue = $('[name="someRadioGroup"]:checked').val();
```

Не волнуйтесь, если сейчас эта строка кажется вам странной. Вскоре вы поймете, как она работает, и сами будете использовать короткие, но мощные инструкции jQuery, чтобы добавить живости своим страницам. Давайте коротко рассмотрим, как работает этот фрагмент кода.

Эта инструкция отыскивает все элементы, атрибут `name` которых имеет значение `someRadioGroup` (напомним, что радиогруппа формируется за счет присваивания всем ее элементам одного и того же имени), затем она оставляет только элементы с установленными атрибутами `checked` и извлекает значение атрибута `value` найденного элемента. (Это может быть только один элемент, потому что браузеры допускают возможность появления только одного отмеченного элемента в радиогруппе.)

Истинная мощь этой инструкции jQuery заложена в *селекторе* – выражении идентификации элементов, позволяющем идентифицировать и отбирать нужные элементы страницы. В данном случае – все отмеченные элементы в радиогруппе.

Если вы еще не загрузили исходные тексты примеров для этой книги, сейчас самое время сделать это. Вы можете получить их, обратившись на сайт книги по адресу <http://www.manning.com/bibeault2> (не забудьте добавить 2 в конце). Загрузив архив, распакуйте его и перейдите к требуемому файлу самостоятельно либо воспользуйтесь замечательной начальной страницей *index.html*, расположенной в каталоге, куда был распакован архив.

Откройте в браузере ссылку *chapter1/radio.group.html*. На рис. 1.1 показано, как выглядит страница, использующая только что исследованную нами инструкцию jQuery, которая определяет, какая из радиокнопок была отмечена.

Даже такой простой пример должен убедить вас, что библиотека jQuery обеспечивает простой способ создания высокоинтерактивных веб-приложений следующего поколения. В последующих главах мы покажем вам еще более мощные возможности, которые предлагает библиотека jQuery разработчикам веб-страниц.

Позднее мы узнаем, как легко создавать эти селекторы, но сначала посмотрим, как изобретатели jQuery представляют себе эффективное использование JavaScript в наших страницах.

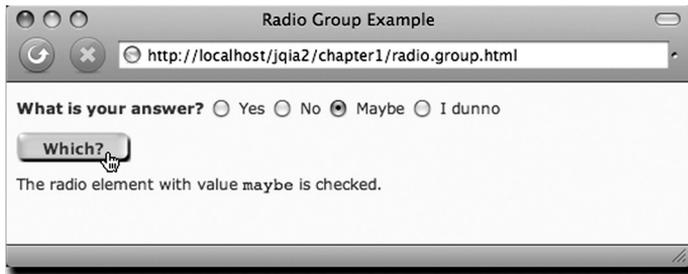


Рис. 1.1. Библиотека jQuery позволяет определить, какая из радиокнопок была выбрана, с помощью единственной инструкции!

1.2. Ненавязчивый JavaScript

Возможно, вы помните суровые времена до появления CSS, когда мы были вынуждены смешивать в HTML-страницах стилевую разметку со структурой документа. Эти воспоминания почти наверняка заставят содрогнуться любого, кто создавал тогда страницы.

Добавление CSS в арсенал инструментов веб-разработчика позволяет отделить стилистическую информацию от структуры документа и проводить на заслуженный отдых такие теги, как ``. Отделение стиля от структуры не только упрощает управление документами, но и придает им гибкость, позволяя добиться полного изменения стиля отображения страницы простой заменой таблицы стилей.

Немногие из нас захотели бы вернуться в прошлое, когда стили отображения применялись непосредственно к HTML-элементам. И все же до сих пор обычной остается разметка вроде этой:

```
<button
  type="button"
  onclick="document.getElementById('xyz').style.color='red';">
  Click Me
</button>
```

Здесь видно, что стиль элемента-кнопки, включая шрифт надписи на ней, определяется не тегом `` или другой нежелательной разметкой стиля, а задается правилами CSS, действующими в пределах страницы. В этом объявлении не смешивается разметка стиля и структура, однако здесь налицо смешение *поведения* и структуры за счет включения кода JavaScript, выполняемого по щелчку по кнопке, в код разметки элемента `<button>` (в данном случае щелчок по кнопке вызывает окрашивание в красный цвет некоторого элемента объектной модели доку-

мента (Document Object Model, DOM), атрибут `id` которого имеет значение `xyz`).

Рассмотрим, как можно было бы улучшить эту ситуацию.

1.2.1. Отделение поведения от разметки

По тем же причинам, по которым желательно отделять стиль от разметки HTML-документа, также (если не более) желательно было бы отделить *поведение* элементов от их разметки.

В идеале HTML-страница должна иметь структуру, как показано на рис. 1.2, где разметка, информация о стилях и реализация поведения находятся в своих отдельных нишах.

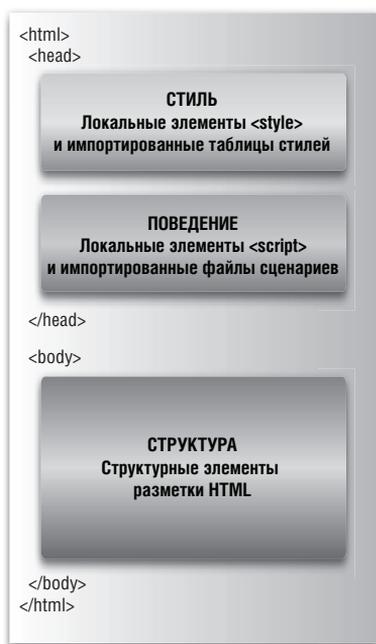


Рис. 1.2. Отделение элементов, определяющих структуру, стиль и поведение компонентов страницы, повышает удобочитаемость исходного кода и упрощает его сопровождение

Эта стратегия, известная под названием *ненавязчивый JavaScript*, была предложена разработчиками jQuery и теперь поддерживается всеми большими библиотеками JavaScript, помогая авторам страниц достигать этого отделения при разработке страниц. Библиотека jQuery сделала эту стратегию популярной, и ее ядро выстроено так, что ненавязчивый JavaScript поддерживается очень органично. Ненавязчивый JavaScript предполагает, что ошибочно присутствие *любых* выражений или инструкций на языке JavaScript в теге `<body>` HTML-страниц – как в атри-

бутах HTML-элементов (например, `onclick`), так и в блоках сценариев в теле страницы.

Вы можете спросить: «Как же пользоваться кнопкой без атрибута `onclick`?» Рассмотрим такое определение кнопки:

```
<button type="button" id="testButton">Click Me</button>
```

Оно гораздо проще! Но теперь кнопка просто *ничего не делает*. Мы можем щелкать на ней весь день и не получим никакого результата.

Давайте исправим этот недостаток.

1.2.2. Отделение сценария

Вместо того чтобы встраивать определение поведения кнопки в ее разметку, мы поместим его в блок сценария в разделе `<head>` страницы *за пределами* тела документа, как показано ниже:

```
<script type="text/javascript">
  window.onload = function() {
    document.getElementById('testButton').onclick = function() {
      document.getElementById('xyz').style.color = 'red';
    };
  };
</script>
```

В обработчике события страницы `onload` мы связываем функцию с атрибутом `onclick` элемента-кнопки.

Обработчик события `onload` (а не встроенный код) используется потому, что нам нужно, чтобы элемент-кнопка уже *существовал* к тому моменту, когда мы попытаемся манипулировать им. (В разделе 1.3.3 будет показано, что jQuery предоставляет лучший способ размещения такого программного кода.)

Примечание

Для достижения более высокой производительности блоки сценариев можно поместить в конец тела документа, однако для современных браузеров полученный прирост производительности будет совсем незначительным. Здесь важно понять, что не следует смешивать реализацию поведения со структурной разметкой элементов.

Если в этом примере что-то покажется вам непонятным (например, объявления встроенных функций), не пугайтесь! В приложении приведен обзор понятий JavaScript, которые помогут вам эффективно использовать jQuery. Кроме того, в оставшейся части главы мы узнаем, как jQuery позволяет писать код, аналогичный предыдущему, более короткий, более простой и одновременно более гибкий.

Ненавязчивый JavaScript – это мощная методика, позволяющая разделить обязанности в веб-приложениях, но за это приходится платить.