



JAVASCRIPT[®] & JQUERY

interactive front-end
web development

JON DUCKETT

JAVASCRIPT & JQUERY

Interactive Front-End Web Development

JON DUCKETT

Design by:

EMME STONE

Additional material by:

GILLES RUPPERT & JACK MOORE

WILEY



The text stock is SFI certified

Published by
John Wiley & Sons, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

©2014 by John Wiley & Sons, Inc., Indianapolis, Indiana

ISBN: 978-1-118-53164-8; 978-1-118-87165-2

Manufactured in the United States of America

10 9 8 7 6 5 4

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Limit of Liability/Disclaimer of Warranty: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or website may provide or recommendations it may make. Further, readers should be aware that Internet websites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2013933932

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. JavaScript is a registered trademark of Oracle America, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

CREDITS

For John Wiley & Sons, Inc.

Executive Editor

Carol Long

Project Editor

Kevin Kent

Production Editor

Daniel Scribner

Editorial Manager

Mary Beth Wakefield

Associate Director of Marketing

David Mayhew

Marketing Manager

Lorna Mein

Business Manager

Amy Knies

Vice President and Executive

Group Publisher

Richard Swadley

Associate Publisher

Jim Minatel

Project Coordinator, Cover

Todd Klemme

For Wagon Ltd.

Author

Jon Duckett

Co-Authors

Jack Moore
(Chapters 11 & 12)

Gilles Ruppert
(Chapter 13)

Technical Review

Mathias Bynens

Review Team

Chris Ullman
David Lean
Harrison Thrift
Jay Bursky
Richard Eskins
Scott Robin
Stachu Korick

Thank you

Annette Loudon
Michael Tomko
Michael Vella Zarb
Pam Coca
Rishabh Pugalia

Cover Design

Emme Stone

Design

Emme Stone
Jon Duckett

Photography

John Stewardson
johnstewardson.com

Illustration

Matthew Cencich
(Hotel in Chapter 3)

Emme Stone

(Teacher in Chapter 4)

Additional Photography

Electronics in Chapters 8 & 9:
Aaron Nielsen
Arkadiusz Jan Sikorski
Matt Mets
Mirsad Dedović
Steve Lodefink

javascriptbook.com/credits

TABLE OF CONTENTS

Introduction	1
Chapter 1: The ABC of Programming	11
Chapter 2: Basic JavaScript Instructions	53
Chapter 3: Functions, Methods & Objects	85
Chapter 4: Decisions & Loops	145
Chapter 5: Document Object Model	183
Chapter 6: Events	243
Chapter 7: jQuery	293
Chapter 8: Ajax & JSON	367
Chapter 9: APIs	409
Chapter 10: Error Handling & Debugging	449
Chapter 11: Content Panels	487
Chapter 12: Filtering, Searching & Sorting	527
Chapter 13: Form Enhancement & Validation	567
Index	623



Try out & download the code in this book
www.javascriptbook.com



INTRODUCTION

This book explains how JavaScript can be used in browsers to make websites more interactive, interesting, and user-friendly. You will also learn about jQuery because it makes writing JavaScript a lot easier.

To get the most out of this book, you will need to know how to build web pages using HTML and CSS. Beyond that, no prior experience with programming is necessary. Learning to program with JavaScript involves:

1

Understanding some **basic programming concepts** and the terms that JavaScript programmers use to describe them.

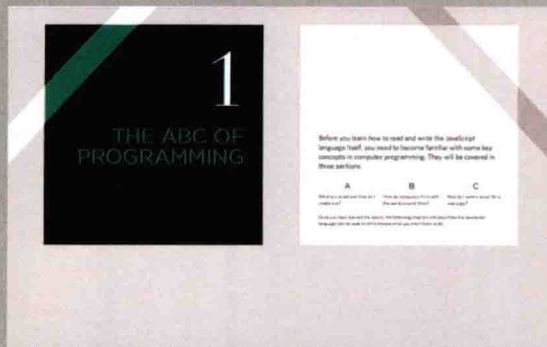
2

Learning **the language** itself, and, like all languages, you need to know its vocabulary and how to structure your sentences.

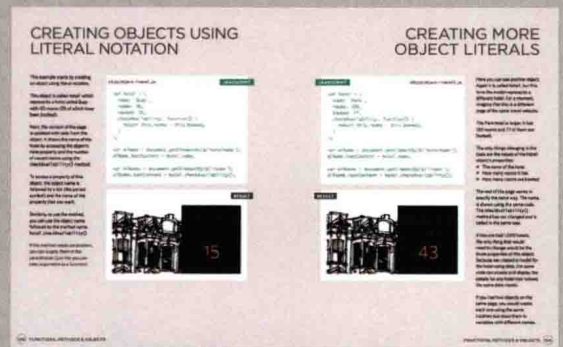
3

Becoming familiar with **how it is applied** by looking at examples of how JavaScript is commonly used in websites today.

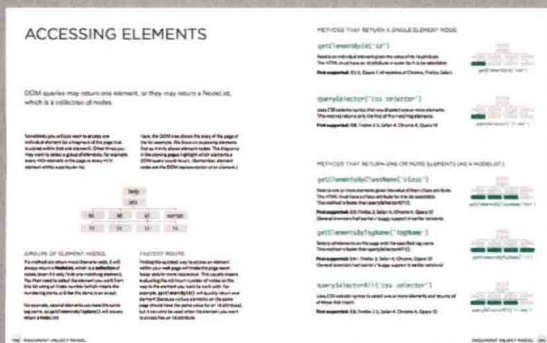
The only equipment you need to use this book are a computer with a modern web browser installed, and your favorite code editor, (e.g., Notepad, TextEdit, Sublime Text, or Coda).



Introduction pages come at the beginning of each chapter. They introduce the key topics you will learn about.



Reference pages introduce key pieces of JavaScript. HTML code is shown in blue, CSS code in pink, and JavaScript in green.



Background pages appear on white. They explain the context of the topics covered that are discussed in each chapter.

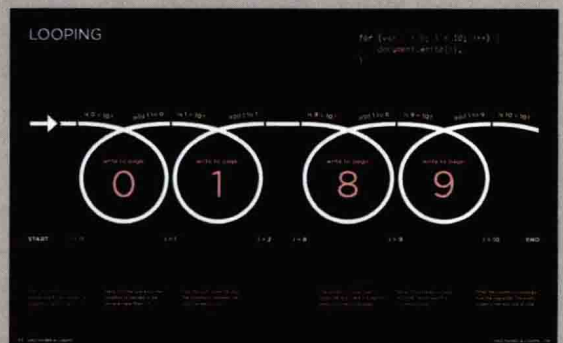
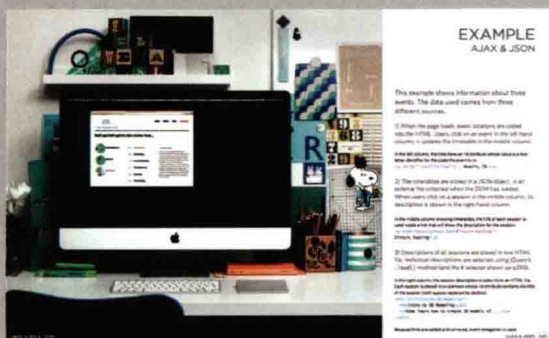
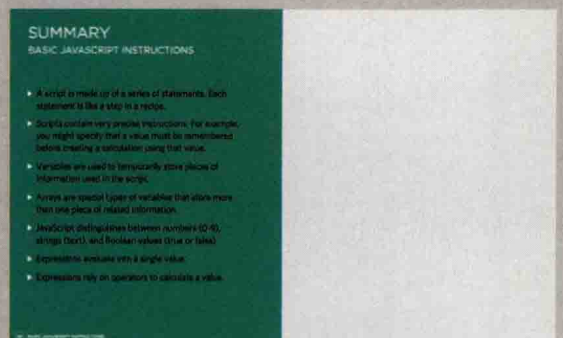


Diagram and infographic pages are shown on a dark background. They provide a simple, visual reference to topics discussed.



Example pages bring together the topics you have learned in that chapter and demonstrate how they can be applied.



Summary pages come at the end of each chapter. They remind you of the key topics that were covered in each chapter.

HOW JAVASCRIPT MAKES WEB PAGES MORE INTERACTIVE

1

ACCESS CONTENT

You can use JavaScript to select any element, attribute, or text from an HTML page. For example:

- Select the text inside all of the `<h1>` elements on a page
- Select any elements that have a `class` attribute with a value of `note`
- Find out what was entered into a text input whose `id` attribute has a value of `email`

JavaScript allows you to make web pages more interactive by accessing and modifying the content and markup used in a web page while it is being viewed in the browser.

2

MODIFY CONTENT

You can use JavaScript to add elements, attributes, and text to the page, or remove them. For example:

- Add a paragraph of text after the first `<h1>` element
- Change the value of `class` attributes to trigger new CSS rules for those elements
- Change the size or position of an `` element

3

PROGRAM RULES

You can specify a set of steps for the browser to follow (like a recipe), which allows it to access or change the content of a page. For example:

- A gallery script could check which image a user clicked on and display a larger version of that image.
- A mortgage calculator could collect values from a form, perform a calculation, and display repayments.
- An animation could check the dimensions of the browser window and move an image to the bottom of the viewable area (also known as the viewport).

JavaScript
encompasses many
of the traditional rules of
programming.

It can make the web page feel
interactive by responding
to what the user does.

4

REACT TO EVENTS

You can specify that a script should run when a specific event has occurred. For example, it could be run when:

- A button is pressed
- A link is clicked (or tapped) on
- A cursor hovers over an element
- Information is added to a form
- An interval of time has passed
- A web page has finished loading

EXAMPLES OF JAVASCRIPT IN THE BROWSER

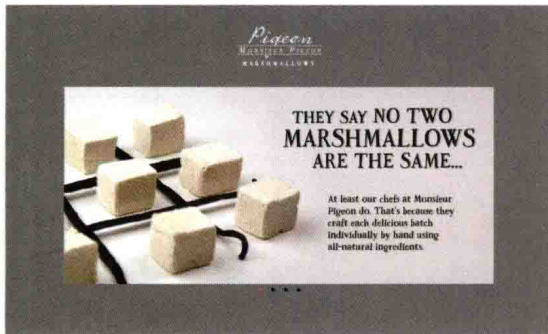
Being able to change the content of an HTML page while it is loaded in the browser is very powerful. The examples below rely on the ability to:

Access the content of the page

Modify the content of the page

Program rules or instructions the browser can follow

React to events triggered by the user or browser



SLIDESHOWS

Shown in Chapter 11

Slideshows can display a number of different images (or other HTML content) within the same space on a given page. They can play automatically as a sequence, or users can click through the slides manually. They allow more content to be displayed within a limited amount of space.

React: Script triggered when the page loads

Access: Get each slide from the slideshow

Modify: Only show the first slide (hide others)

Program: Set a timer: when to show next slide

Modify: Change which slide is shown

React: When user clicks button for different slide

Program: Determine which slide to show

Modify: Show the requested slide



FORMS

Shown in Chapter 13

Validating forms (checking whether they have been filled in correctly) is important when information is supplied by users. JavaScript lets you alert the user if mistakes have been made. It can also perform sophisticated calculations based on any data entered and reveal the results to the user.

React: User presses the submit button when they have entered their name

Access: Get value from form field

Program: Check that the name is long enough

Modify: Show a warning message if the name is not long enough

The examples on these two pages give you a taste of what JavaScript can do within a web page, and of the techniques you will be learning throughout this book.

In the coming chapters, you will learn how and when to access or modify content, add programming rules, and react to events.



RELOAD PART OF PAGE

Shown in Chapter 8

You might not want to force visitors to reload the content of an entire web page, particularly if you only need to refresh a small portion of a page. Just reloading a section of the page can make a site feel like it is faster to load and more like an application.

React: Script triggered when user clicks on link
Access: The link that they clicked on
Program: Load the new content that was requested from that link
Access: Find the element to replace in the page
Modify: Replace that content with the new content



FILTERING DATA

Shown in Chapter 12

If you have a lot of information to display on a page, you can help users find information they need by providing filters. Here, buttons are generated using data in the attributes of the HTML `` elements. When the user clicks on one of the buttons, they are only shown the images with that keyword.

React: Script triggered when page loads
Program: Collect keywords from images
Program: Turn the keywords into buttons the user can click on
React: User clicks on one of the buttons
Program: Find the relevant subset of images that should be shown
Modify: Show the subset of images that use that tag

THE STRUCTURE OF THIS BOOK

In order to teach you JavaScript, this book is divided into two sections:

CORE CONCEPTS

The first nine chapters introduce you to the basics of programming and the JavaScript language. Along the way you will learn how it is used to create more engaging, interactive, and usable websites.

Chapter 1 looks at some key concepts in computer programming, showing you how computers create models of the world using data, and how JavaScript is used to change the contents of an HTML page.

Chapters 2-4 cover the basics of the JavaScript language.

Chapter 5 explains how the Document Object Model (DOM) lets you access and change a document's contents while it is loaded into the browser.

Chapter 6 discusses how events can be used to trigger code.

Chapter 7 shows you how jQuery can make the process of writing scripts faster and easier.

Chapter 8 introduces you to Ajax, a set of techniques that allow you to just change part of a web page without reloading the entire page.

Chapter 9 covers Application Programming Interfaces (APIs), including new APIs that are part of HTML5 and those of sites like Google Maps.

PRACTICAL APPLICATIONS

By this point you will already have seen many examples of how JavaScript is used on popular websites. This section brings together all of the techniques you have learned so far, to give you practical demonstrations of how JavaScript is used by professional developers. Not only will you see a selection of in-depth examples, you will also learn more about the process of designing and writing scripts from scratch.

Chapter 10 deals with error-handling and debugging, and explains more about how JavaScript is processed.

Chapter 11 shows you techniques for creating content panels such as sliders, modal windows, tabbed panels, and accordions.

Chapter 12 demonstrates several techniques for filtering and sorting data. This includes filtering a gallery of images, and re-ordering the rows of a table by clicking on the column headings.

Chapter 13 deals with form enhancements and how to validate form entries.

Unless you are already a confident programmer, you will probably find it helpful to read the book from start to finish the first time. However, once you have grasped the basics, we hope it will continue to be a helpful reference as you create your own scripts.

HTML & CSS: A QUICK REFRESHER

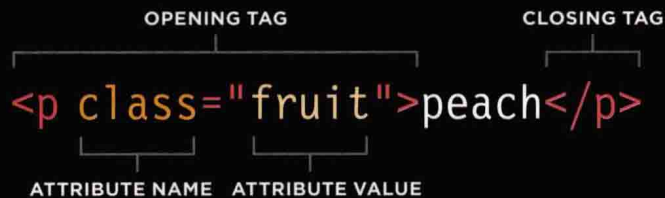
Before looking at JavaScript, let's clarify some HTML & CSS terms. Note how HTML attributes and CSS properties use name/value pairs.

HTML ELEMENTS

HTML elements are added to the content of a page to describe its structure. An element consists of the opening and closing tags, plus its content.

Tags usually come in pairs with an opening tag and a closing tag. There are a few empty elements with no content, (e.g., ``). They have one self-closing tag.

Opening tags can carry attributes, which tell us more about that element. Attributes have a name and a value. The value is usually given in quotes.



CSS RULES

CSS uses rules to indicate how the contents of one or more elements should be displayed in the browser. Each rule has a selector and a declaration block.

The CSS selector indicates which element(s) the rule applies to. The declaration block contains rules that indicate how those elements should appear.

Each declaration in the declaration block has a property (the aspect you want to control), and a value, which is the setting for that property.



BROWSER SUPPORT

Some early examples in this book do not work with Internet Explorer 8 and earlier (but alternative code samples that work in IE8 are available to download from <http://javascriptbook.com>). We explain techniques for dealing with older browsers in later chapters.

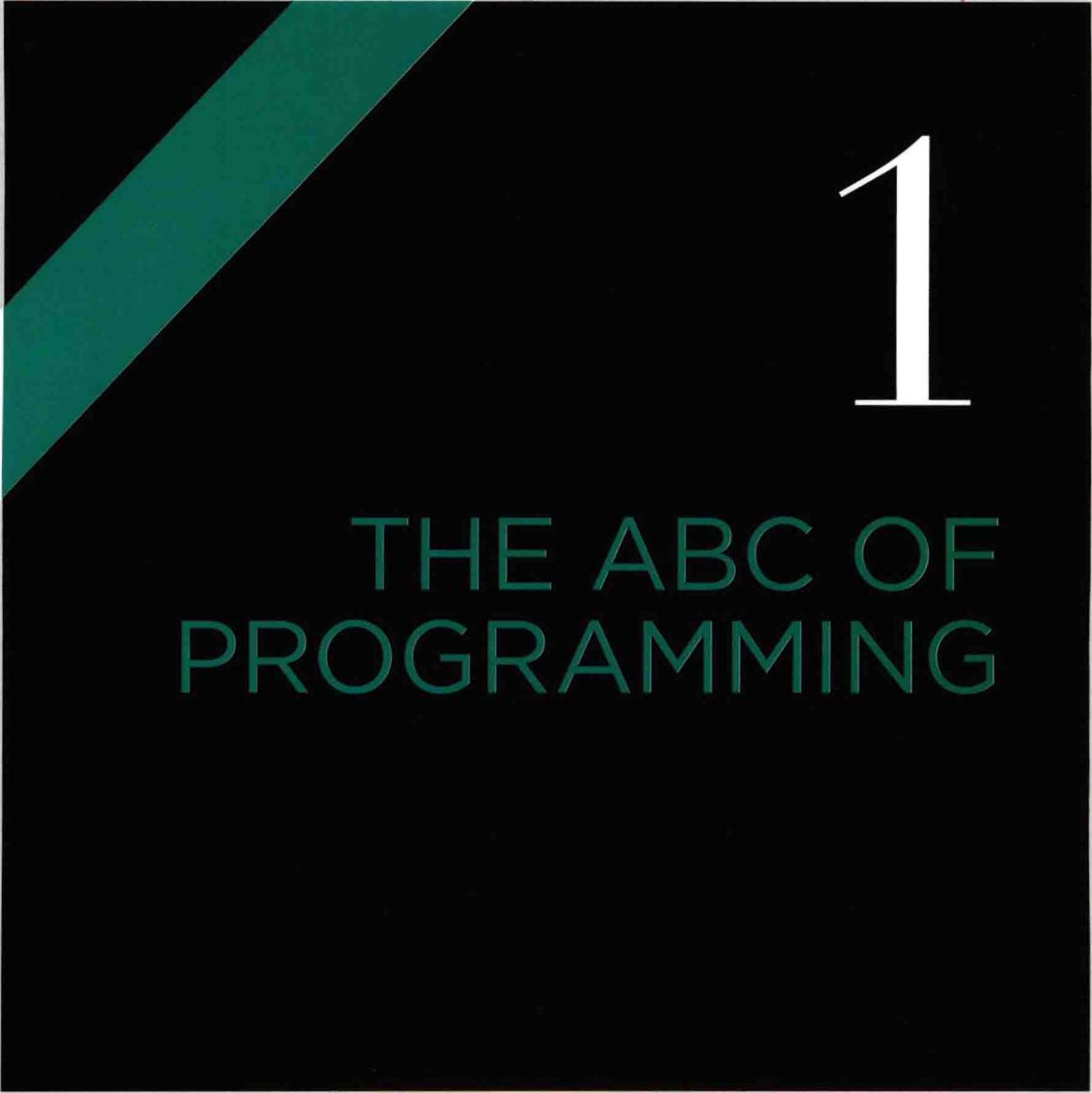
Each version of a web browser adds new features. Often these new features make tasks easier, or are considered better, than using older techniques.

But, website visitors do not always keep up with the latest browser releases, so website developers cannot always rely upon the latest technologies.

As you will see, there are many inconsistencies between browsers that affect JavaScript developers. jQuery will help you deal with cross-browser inconsistencies (it is one of the major reasons why jQuery rapidly gained popularity amongst web developers). But, before you learn jQuery, it helps to know what it is helping you to achieve.

To make JavaScript easier to learn, the first few chapters use some features of JavaScript that are not supported in IE8. But:

- You **will** learn how to deal with IE8 and older browsers in later chapters (because we know that many clients expect sites to work in IE8). It just requires knowledge of some extra code or requires you to be aware of some additional issues.
- Online, you will find alternatives available for each example that does not work in IE8. But please check the comments in those code samples to make sure you know about the about issues involved in using them.



1

THE ABC OF
PROGRAMMING

Before you learn how to read and write the JavaScript language itself, you need to become familiar with some key concepts in computer programming. They will be covered in three sections:

A

What is a script and how do I create one?

B

How do computers fit in with the world around them?

C

How do I write a script for a web page?

Once you have learned the basics, the following chapters will show how the JavaScript language can be used to tell browsers what you want them to do.