

Brad Dayley
Brendan Dayley

Sams **Teach Yourself**

AngularJS, JavaScript, and jQuery

All
in **One**

SAMS



About This eBook

ePUB is an open, industry-standard format for eBooks. However, support of ePUB and its many features varies across reading devices and applications. Use your device or app settings to customize the presentation to your liking. Settings that you can customize often include font, font size, single or double column, landscape or portrait mode, and figures that you can click or tap to enlarge. For additional information about the settings and features on your reading device or app, visit the device manufacturer's Web site.

Many titles include programming code or configuration examples. To optimize the presentation of these elements, view the eBook in single-column, landscape mode and adjust the font size to the smallest setting. In addition to presenting code and configurations in the reflowable text format, we have included images of the code that mimic the presentation found in the print book; therefore, when the reflowable format may compromise the presentation of the code listing, you will see a “Click here to view code image” link. Click the link to view the print-fidelity code image. To return to the previous page viewed, click the Back button on your device or app.

Sams Teach Yourself

AngularJS, JavaScript, and

jQuery All in One in 24 Hours

Brad Dayley
Brendan Dayley

SAMS

800 East 96th Street Indianapolis, IN 46240 USA

Sams Teach Yourself AngularJS, JavaScript, and jQuery All in One

Copyright © 2016 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-672-33742-0

ISBN-10: 0-672-33742-8

Library of Congress Control Number: 2015907445

Printed in the United States of America

First Printing August 2015

Acquisitions Editor

Mark Taber

Managing Editor

Kristy Hart

Project Editor

Andy Beaster

Copy Editor

Barbara Hacha

Indexer

Brad Herriman

Proofreader

Sarah Kearns

Technical Editor

Jesse Smith

Publishing Coordinator

Vanessa Evans

Interior Designer

Gary Adair

Cover Designer

Mark Shirar

Compositor

Nonie Ratcliff

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The authors and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact international@pearsoned.com.

Contents at a Glance

[Introduction](#)

[Part I: Introduction to AngularJS, jQuery, and JavaScript Development](#)

[LESSON 1 Introduction to Dynamic Web Programming](#)

[LESSON 2 Debugging JavaScript in Web Pages](#)

[LESSON 3 Understanding Dynamic Web Page Anatomy](#)

[LESSON 4 Adding CSS/CSS3 Styles to Allow Dynamic Design and Layout](#)

[LESSON 5 Jumping into jQuery and JavaScript Syntax](#)

[LESSON 6 Understanding and Using JavaScript Objects](#)

[Part II: Implementing jQuery and JavaScript in Web Pages](#)

[LESSON 7 Accessing DOM Elements Using JavaScript and jQuery Objects](#)

[LESSON 8 Navigating and Manipulating jQuery Objects and DOM Elements with jQuery](#)

[LESSON 9 Applying JavaScript and jQuery Events for Richly Interactive Web Pages](#)

[LESSON 10 Dynamically Accessing and Manipulating Web Pages with JavaScript and jQuery](#)

[LESSON 11 Working with Window, Browser, and Other Non-Web Page Elements](#)

[Part III: Building Richly Interactive Web Pages with jQuery](#)

[LESSON 12 Enhancing User Interaction Through jQuery Animation and Other Special Effects](#)

[LESSON 13 Interacting with Web Forms in jQuery and JavaScript](#)

[LESSON 14 Creating Advanced Web Page Elements in jQuery](#)

[LESSON 15 Accessing Server-Side Data via JavaScript and jQuery AJAX Requests](#)

[Part IV: Utilizing jQuery UI](#)

[LESSON 16 Introducing jQuery UI](#)

[LESSON 17 Using jQuery UI Effects](#)

[LESSON 18 Advanced Interactions Using jQuery UI Interaction Widgets](#)

[LESSON 19 Using jQuery UI Widgets to Add Rich Interactions to Web Pages](#)

[Part V: Building Web Applications with AngularJS](#)

[LESSON 20 Getting Started with AngularJS](#)

[LESSON 21 Understanding AngularJS Application Dynamics](#)

[LESSON 22 Implementing the Scope as a Data Model](#)

[LESSON 23 Using AngularJS Templates to Create Views](#)

[LESSON 24 Implementing Directives in AngularJS Views](#)

[LESSON 25 Creating Your Own Custom Directives to Extend HTML](#)

[LESSON 26 Using Events to Interact with Data in the Model](#)

[LESSON 27 Implementing AngularJS Services in Web Applications](#)

[LESSON 28 Creating Your Own Custom AngularJS Services](#)

[LESSON 29 Creating Rich Web Application Components the AngularJS Way](#)

[Index](#)

Table of Contents

[Introduction](#)

[Lesson 1: Introduction to Dynamic Web Programming](#)

[Understanding the Web Server/Browser Paradigm](#)

[Setting Up a Web Development Environment](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 2: Debugging JavaScript in Web Pages](#)

[Viewing the Developer Tools Console](#)

[Debugging HTML Elements](#)

[Debugging CSS](#)

[Debugging JavaScript](#)

[Analyzing the Network Traffic](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 3: Understanding Dynamic Web Page Anatomy](#)

[Using HTML/HTML5 Elements to Build a Dynamic Web Page](#)

[Understanding HTML Structure](#)

[Implementing HTML Head Elements](#)

[Adding HTML Body Elements](#)

[Adding Some Advanced HTML5 Elements](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 4: Adding CSS/CSS3 Styles to Allow Dynamic Design and Layout](#)

[Adding CSS Styles to the Web Page](#)

[Adding CSS Styles to HTML Elements](#)

[Preparing CSS Styles for Dynamic Design](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 5: Jumping into jQuery and JavaScript Syntax](#)

[Adding jQuery and JavaScript to a Web Page](#)

[Accessing the DOM](#)

[Understanding JavaScript Syntax](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 6: Understanding and Using JavaScript Objects](#)

[Using Object Syntax](#)

[Understanding Built-in Objects](#)

[Creating Custom-Defined Objects](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 7: Accessing DOM Elements Using JavaScript and jQuery Objects](#)

[Understanding DOM Objects Versus jQuery Objects](#)

[Accessing DOM Objects from JavaScript](#)

[Using jQuery Selectors](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 8: Navigating and Manipulating jQuery Objects and DOM Elements with jQuery](#)

[Chaining jQuery Object Operations](#)

[Filtering the jQuery Object Results](#)

[Traversing the DOM Using jQuery Objects](#)

[Looking at Some Additional jQuery Object Methods](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 9: Applying JavaScript and jQuery Events for Richly Interactive Web Pages](#)

[Understanding Events](#)

[Using the Page Load Events for Initialization](#)

[Adding and Removing Event Handlers to DOM Elements](#)

[Triggering Events Manually](#)

[Creating Custom Events](#)

[Implementing Callbacks](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 10: Dynamically Accessing and Manipulating Web Pages with JavaScript and jQuery

[Accessing Browser and Page Element Values](#)

[Dynamically Manipulating Page Elements](#)

[Dynamically Rearranging Elements on the Web Page](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 11: Working with Window, Browser, and Other Non-Web Page Elements

[Understanding the Screen Object](#)

[Using the Window Object](#)

[Using the Browser Location Object](#)

[Using the Browser History Object](#)

[Controlling External Links](#)

[Adding Pop-up Boxes](#)

[Setting Timers](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 12: Enhancing User Interaction Through jQuery Animation and Other Special Effects

[Understanding jQuery Animation](#)

[Animating Show and Hide](#)

[Animating Visibility](#)

[Sliding Elements](#)

[Creating Resize Animations](#)

[Implementing Moving Elements](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 13: Interacting with Web Forms in jQuery and JavaScript

[Accessing Form Elements](#)

[Intelligent Form Flow Control](#)

[Dynamically Controlling Form Element Appearance and Behavior](#)

[Validating a Form](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 14: Creating Advanced Web Page Elements in jQuery

[Adding an Image Gallery](#)

[Implementing Tables with Sorting and Filters](#)

[Creating a Tree View](#)

[Using Overlay Dialogs](#)

[Implementing a Graphical Equalizer Display](#)

[Adding Sparkline Graphics](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 15: Accessing Server-Side Data via JavaScript and jQuery AJAX Requests

[Making AJAX Easy](#)

[Implementing AJAX](#)

[Using Advanced jQuery AJAX](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 16: Introducing jQuery UI

[Getting Started with jQuery UI](#)

[Applying jQuery UI in Your Scripts](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 17: Using jQuery UI Effects

[Applying jQuery UI Effects](#)

[Adding Effects to Class Transitions](#)

[Adding Effects to Element Visibility Transitions](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 18: Advanced Interactions Using jQuery UI Interaction Widgets

[Introducing jQuery UI Interactions](#)

[Using the Drag-and-Drop Widgets](#)

[Resizing Elements Using the Resizable Widget](#)

[Applying the Selectable Widget](#)

[Sorting Elements with the Sortable Widget](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 19: Using jQuery UI Widgets to Add Rich Interactions to Web Pages](#)

[Reviewing Widgets](#)

[Adding an Expandable Accordion Element](#)

[Implementing Autocomplete in Form Elements](#)

[Applying jQuery UI Buttons to Form Controls](#)

[Creating a Calendar Input](#)

[Generating Stylized Dialogs with jQuery UI](#)

[Implementing Stylized Menus](#)

[Creating Progress Bars](#)

[Implementing Slider Bars](#)

[Adding a Value Spinner Element](#)

[Creating Tabbed Panels](#)

[Adding Tooltips to Page Elements](#)

[Creating Custom Widgets](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 20: Getting Started with AngularJS](#)

[Why AngularJS?](#)

[Understanding AngularJS](#)

[An Overview of the AngularJS Life Cycle](#)

[Separation of Responsibilities](#)

[Integrating AngularJS with Existing JavaScript and jQuery](#)

[Adding AngularJS to Your Environment](#)

[Bootstrapping AngularJS in an HTML Document](#)

[Using the Global APIs](#)

[Using jQuery or jQuery Lite in AngularJS Applications](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 21: Understanding AngularJS Application Dynamics](#)

[Looking at Modules and Dependency Injection](#)

[Defining an AngularJS Module Object](#)

[Creating Providers in AngularJS Modules](#)

[Implementing Providers and Dependency Injection](#)

[Applying Configuration and Run Blocks to Modules](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 22: Implementing the Scope as a Data Model

[Understanding Scopes](#)

[Implementing Scope Hierarchy](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 23: Using AngularJS Templates to Create Views

[Understanding Templates](#)

[Using Expressions](#)

[Using Filters](#)

[Creating Custom Filters](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 24: Implementing Directives in AngularJS Views

[Understanding Directives](#)

[Using Built-In Directives](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 25: Creating Your Own Custom Directives to Extend HTML

[Understanding Custom Directive Definitions](#)

[Implementing Custom Directives](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

Lesson 26: Using Events to Interact with Data in the Model

[Browser Events](#)

[User Interaction Events](#)

[Adding \\$watches to Track Scope Change Events](#)

[Emitting and Broadcasting Custom Events](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 27: Implementing AngularJS Services in Web Applications](#)

[Understanding AngularJS Services](#)

[Using the Built-In Services](#)

[Using the \\$q Service to Provide Deferred Responses](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 28: Creating Your Own Custom AngularJS Services](#)

[Understanding Custom AngularJS Services](#)

[Integrating Custom Services into Your AngularJS Applications](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Lesson 29: Creating Rich Web Application Components the AngularJS Way](#)

[Summary](#)

[Q&A](#)

[Workshop](#)

[Index](#)

About the Authors

Brad Dayley is a senior software engineer with more than 20 years of experience developing enterprise applications and web interfaces. He has used JavaScript, jQuery, and AngularJS to develop a wide array of feature-rich web applications. He has a passion for new technologies, especially ones that really make a difference in the software industry. He is the author of *Node.js*, *MongoDB*, and *AngularJS Web Development*, *Learning AngularJS*, *jQuery*, and *JavaScript Phrasebook*, and *Sams Teach Yourself jQuery and JavaScript in 24 Hours*.

Brendan Dayley is a university student majoring in computer science. He is an avid web application developer who loves learning and implementing the latest and greatest technologies. He recently attended Dev-Mountain's Immersive Web Development program, specializing in web application development and AngularJS in particular. He has written a number of web applications using JavaScript, jQuery, and AngularJS and is excited about the future of these technologies.

Dedication

For D!

A & F

Jessie

My one and only

Acknowledgments

I'd like to take this opportunity to thank all those who made this title possible. First, thanks to my wonderful wife and boys for giving me the inspiration and support I need. I'd never make it far without you.

Thanks to Mark Taber for getting this title rolling in the right direction, Jesse Smith for keeping me honest with his technical review, Barbara Hacha for turning the technical ramblings of my brain into fine text, and Andy Beaster for managing everything on the production end and making sure the book is the finest quality.

We Want to Hear from You!

As the reader of this book, *you* are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

We welcome your comments. You can email or write to let us know what you did or didn't like about this book—as well as what we can do to make our books better.

Please note that we cannot help you with technical problems related to the topic of this book.

When you write, please be sure to include this book's title and author as well as your name and email address. We will carefully review your comments and share them with the author and editors who worked on the book.

Email: feedback@samspublishing.com

Mail: Sams Publishing
ATTN: Reader Feedback
800 East 96th Street
Indianapolis, IN 46240 USA

Reader Services

Visit our website and register this book at informit.com/register for convenient access to any updates, downloads, or errata that might be available for this book.

Introduction

Welcome to *AngularJS, JavaScript, and jQuery All in One*. This book is designed to jumpstart you into the world of dynamic web application development using JavaScript, jQuery, and AngularJS. This book covers the basics of the JavaScript language, jQuery library, and AngularJS framework and how to use them to build well-designed, reusable components for web applications.

With billions of people using the Internet today, there is a rapidly growing trend to replace traditional websites, where one page links to another page and so on, with single page applications that have richly interactive elements.

The main reason is that users have become less patient with clicking, waiting, and then having to navigate back and forth between web pages. Instead, they want websites to behave more like the applications they are used to on their computers and mobile devices.

In fact, in the next 24 hours, millions of new web pages will be added to the Internet. The majority of these pages will be written in HTML, with CSS to style elements and with JavaScript to provide interaction between the user interface and back-end services.

As you complete the lessons in this book, you will gain a practical understanding of how to incorporate JavaScript with the powerful jQuery library as well as the exciting AngularJS framework to provide rich user interactions in your web pages. You will gain the valuable skills of adding dynamic code that allows web pages to instantly react to mouse clicks and finger swipes, interact with back-end services to store and retrieve data from the web server, and create robust Internet applications.

Each lesson in the book provides fundamentals that are necessary to create professional web applications. The book includes some basics on using HTML and CSS to get you started, even if you've never used them before. You are provided with code examples that you can implement and expand as your understanding increases. In fact, in just the first lesson in the book, you create a dynamic web page using jQuery and JavaScript.

So pull up a chair, sit back, and enjoy the ride of programming rich Internet applications with AngularJS, jQuery, and JavaScript.

Who Should Read This Book

This book is aimed at readers who already have an understanding of the basics of HTML and have done some programming in a modern programming language. Having an understanding of JavaScript will make this book easier to digest, but it is not required because the basics of JavaScript are covered.

Why You Should Read This Book

This book will teach you how to create powerful, interactive web applications that have a well-structured, easy-to-reuse code base that will be easy to maintain. The typical readers of this book want to learn JavaScript, jQuery, and AngularJS for the purpose of building highly interactive web applications. The typical reader will also want to leverage the innovative Model View Controller (MVC) approach of AngularJS to extend HTML and implement well-designed and structured web pages and web applications.

What You Will Learn from This Book

Reading this book will enable you to build rich, dynamic interactions into your web pages and applications. Websites are no longer simple static content that consist of HTML pages with integrated images and formatted text. Instead, websites have become much more dynamic, with a single page providing a wide array of functionality and interactions.

Using AngularJS, jQuery, and JavaScript enables you to build logic directly into your web applications that allows you to interact with the user from your client-side application. These technologies also allow you to interact with back-end web services on the web server to create a comprehensive client-side web application. The following are a few of the things you will learn while reading this book:

- ▶ The basics of the JavaScript language
- ▶ How to implement JavaScript, jQuery, and AngularJS in your web pages
- ▶ How to dynamically modify page elements in the browser
- ▶ How to use browser events to interact with the user directly
- ▶ How to implement client-side services that can interact with the web server
- ▶ How to implement rich user interface (UI) components, such as zoomable images and expandable lists
- ▶ How to quickly build AngularJS templates with built-in directives that enhance the user experience
- ▶ How to bind UI elements to the data model so that when the model changes, the UI changes, and vice versa
- ▶ How to bind mouse and keyboard events directly to the data model and back-end functionality to provide robust user interactions
- ▶ How to define your own custom AngularJS directives that extend the HTML language
- ▶ How to build dynamic browser views that provide rich user interaction
- ▶ How to create custom services that can be easily reused in other AngularJS applications

Why AngularJS, jQuery, and JavaScript in the Same Book?

The reason we decided to put AngularJS, jQuery, and JavaScript in the same book is that they all relate to each other. We've been asked questions like "Should I use AngularJS or jQuery?" or "Should I use JavaScript or jQuery?" many times. We see them as a stack that works together very well.

JavaScript is the base language that is supported by the browser. jQuery extends JavaScript with a syntax that is much more powerful and user friendly. AngularJS is an extension of jQuery (or at least a stripped-down version of jQuery) that provides an extremely powerful MVC framework with robust data binding functionality.

Understanding all three of these technologies and how they work together will make you a better web developer, even if you use another JavaScript framework or library to develop, because they provide the fundamental functionality that all good web applications need. You may decide that simple JavaScript fits the needs in one area, or jQuery/jQueryUI provides the perfect functionality for some web forms, or that you need the robust functionality of AngularJS for your web application. Either way, you will have the skills and understanding to be able to choose and implement the right technology.

What Is JavaScript?

JavaScript is a programming language much like any other. What separates JavaScript the most from other programming languages is that the browser has a built-in interpreter that can parse and execute the language. That means you can write complex applications that have direct access to the browser events and Document Object Model (DOM) objects.

Access to the DOM means that you can add, modify, or remove elements from a web page without reloading it. Access to the browser gives you access to events such as mouse movements and clicks. This is what gives JavaScript the capability to provide functionality such as dynamic lists and drag and drop.

What Is jQuery?

jQuery is a library that is built on JavaScript. The underlying code is JavaScript; however, jQuery simplifies a lot of the JavaScript code into simple-to-use functionality. The two main advantages to using jQuery are selectors and built-in functions.

Selectors provide quick access to specific elements on the web page, such as a list or table. They also provide access to groups of elements, such as all paragraphs or all paragraphs of a certain class. This allows you to quickly and easily access specific DOM elements.

jQuery also provides a rich set of built-in functionality that makes it easy to do a lot more with a lot less code. For example, tasks such as hiding an element on the screen or animating the resizing of an element take just one line of code.

What Is AngularJS?

AngularJS is a client-side framework developed by Google. It is written in JavaScript with a reduced jQuery library called jQuery lite. The entire ideology behind AngularJS is to provide a framework that makes it easy to implement well-designed and well-structured web pages and applications using an MVC framework.

AngularJS provides all that functionality to handle user input in the browser, manipulate data on the client side, and control how elements are displayed in the browser view. Here are some of the benefits AngularJS provides:

- ▶ **Data Binding**—AngularJS has a very clean method to bind data to HTML elements using its powerful scope mechanism.
- ▶ **Extensibility**—The AngularJS architecture enables you to easily extend almost every aspect of the language to provide your own custom implementations.
- ▶ **Clean**—AngularJS forces you to write clean, logical code.
- ▶ **Reusable Code**—The combination of extensibility and clean code makes it very easy to write reusable code in AngularJS. In fact, the language often forces you to do so when you're creating custom services.
- ▶ **Support**—Google is investing a lot in this project, which gives it an advantage where other similar initiatives have failed.
- ▶ **Compatibility**—AngularJS is based on JavaScript and has a close relationship with jQuery. This makes it easier to begin integrating AngularJS into your environment and reuse pieces of your existing code within the structure of the AngularJS framework.

Beyond AngularJS, jQuery, and JavaScript

This book covers more than jQuery and JavaScript because you need to know more than the language structure to create truly useful web applications. The goal of this book is to give you the fundamental skills needed to create fully functional and interactive web applications in just 29 short, easy lessons. This book covers the following key skills and technologies:

- ▶ HTML is the most current recommendation for web page creation. Every example in this book validated HTML5, the most recent recommended version.
- ▶ CSS is the standard method for formatting web elements. You not only learn how to write CSS and CSS3, but also how to dynamically modify it on-the-fly using jQuery and JavaScript.
- ▶ JavaScript is the best method to provide interactions in web pages without the need to load a new page from the server. This is the standard language on which most decent web applications are built.
- ▶ jQuery and jQueryUI are some of the most popular and robust libraries for JavaScript. jQuery provides very quick access to web page elements and a robust set of features for web application interaction. jQuery provides additional UI libraries that provide rich UI components for web applications.
- ▶ AJAX is the standard method that web applications use to interact with web servers and other services. The book includes several examples of using AJAX to interact with web servers, Google, Facebook, and other popular web services.

Code Examples

Many of the examples in the book provide the following elements:

- ▶ **HTML code**—Code necessary to provide the web page framework in the browser.
- ▶ **CSS code**—Code necessary to style the web page elements correctly.
- ▶ **JavaScript code**—This includes the AngularJS, jQuery, and JavaScript code that provide interactions between the user, web page elements, and web services.
- ▶ **Figures**—Most of the examples include one or more figures that illustrate the behavior of the code in the browser.

The titles for the listing blocks include a filename of the file that contains the source. These files can be obtained from the book's website (follow the directions on the back cover of this book).

The examples in the book are basic to make it easier for you to learn and implement. Many of them can be expanded and used in your own web pages. In fact, some of the exercises at the end of each lesson have you expand on the examples.

Development Web Server

I chose Node.js with Express as the web server for the development environment for this book. You will get a chance to set up Node.js as the web server in [Lesson 1](#). There are several reasons I chose Node.js over a more traditional web server like Apache or IIS, including the following:

- ▶ Node.js is extremely easy to install and set up.
- ▶ You can use Node.js to test your JavaScript snippets without having to use a web browser.
- ▶ There is a great Node.js plug-in for Eclipse that allows you to easily debug JavaScript.
- ▶ You do not need to understand a back-end scripting language such as PHP, Python, or Ruby because you can write your server-side script for Node.js in JavaScript.

Special Elements

As you complete each lesson, margin notes help you immediately apply what you just learned to your own web pages.

Whenever a new term is used, it is clearly explained. No flipping back and forth to a glossary!

Tip

Tips and tricks to save you precious time are set aside in Tips so that you can spot them quickly.

Note

Notes highlight interesting information you should be sure not to miss.

Caution

When there's something you need to watch out for, you'll be warned about it in a Caution.

Q&A, Quizzes, and Exercises

Every lesson ends with a short question-and-answer session that addresses the kind of “dumb questions” everyone wants to ask. A brief but complete quiz lets you test yourself to be sure you understand everything presented in the lesson. Finally, one or two optional exercises give you a chance to practice your new skills before you move on.

Finally

We hope you enjoy this book and enjoy learning about JavaScript, jQuery, and AngularJS as much as we did. These are great, innovative technologies that are really fun to use. Soon you'll be able to join the many other web developers who use them to build rich, dynamic, and interactive websites and web applications.

- [Turning Point \(Sholan Alliance, Book 1\) pdf, azw \(kindle\), epub, doc, mobi](#)
- [read Wire's Pink Flag \(33 1/3 Series\) for free](#)
- [read Mastering VMware vSphere 4 \(Computer/Tech\) online](#)
- [read «Che t'importa di ci che dice la gente?» Altre avventure di uno scienziato curioso pdf](#)

- <http://musor.ruspb.info/?library/Wine-and-Society.pdf>
- <http://rodrigocaporal.com/library/Marijuanamerica--One-Man---s-Quest-to-Understand-America---s-Dysfunctional-Love-Affair-with-Weed.pdf>
- <http://thermco.pl/library/Mastering-VMware-vSphere-4--Computer-Tech-.pdf>
- <http://econtact.webschaefer.com/?books/--Che-t-importa-di-ci---che-dice-la-gente---Altre-avventure-di-uno-scienziato-curioso.pdf>