javascript-step-by-step-3rd-edition

- Introduction
- Part I: Javawhat? The where, why, and how of JavaScript
  - Chapter 1: JavaScript is more than you might think
    - A brief history of JavaScript
      - Enter Internet Explorer 3.0
      - And then came ECMAScript
      - So many standards...
      - The DOM
    - What's in a JavaScript program?
    - JavaScript placement on your webpage
    - What JavaScript can do
    - What JavaScript can't do
      - JavaScript can't be forced on a client
      - JavaScript can't guarantee data security
      - JavaScript can't cross domains
      - JavaScript doesn't do servers
    - Tips for using JavaScript
      - Where JavaScript fits
      - Which browsers should the site support?
    - And then came Windows 8
    - Exercises
  - Chapter 2: Developing in JavaScript
    - JavaScript development options
    - Configuring your environment
    - Writing JavaScript with Visual Studio 2012
      - Your first web (and JavaScript) project with Visual Studio 2012
      - Using external JavaScript files with Visual Studio 2012
    - Debugging JavaScript
    - Exercises
  - Chapter 3: JavaScript syntax and statements
    - A bit of housekeeping
      - Case sensitivity
      - White space
      - Comments
      - Semicolons
      - Line breaks
      - Placing JavaScript correctly
    - JavaScript statements
      - What's in a statement?
      - The two types of JavaScript statements
    - Reserved words in JavaScript
    - A quick look at functions
    - JavaScript's strict mode
    - Exercises
  - Chapter 4: Working with variables and data types
    - Data types in JavaScript
      - Working with numbers
      - Working with strings
      - Booleans
      - Null
      - Undefined
      - Objects
      - Arrays
    - Defining and using variables
      - Declaring variables
      - Variable types
      - Variable scope
      - The Date object

        Counting down to a certain date in the future

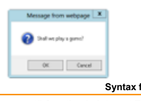
        1. Using Visual Studio, Eclipse, or another editor, edit the file countdown.html in the Chapter04 sample files folder, which you can find in the companion content.
        2. Add the following code shown in boldface type to the page:

        ```
        <!doctype html>
        <html>
        <head>
            <title>the dates</title>
        </head>
        <body>
            <p id="dateField">debug</p>
            <script type = "text/javascript">
            var today = new Date();
            var then = new Date();
            // January 1, 2014
            then.setFullYear(2014,0,1);
            var diff = then.getTime() - today.getTime();
            diff = Math.Floor(diff / (1000 * 60 * 60 * 24));
            var dateLoc = document.getElementById("dateField");
            dateLoc.innerHTML = "There are " + diff + " days until 1/1/2014";
            </script>

        </body>
        </html>
        ```
    - Using the RegExp object
      - The syntax of regular expressions
      - References and garbage collection
    - Learning about type conversions
      - Number conversions
      - String conversions
      - Boolean conversions
    - Exercises
  - Chapter 5: Using operators and expressions
    - Meet the operators
    - Additive operators
    - Multiplicative operators
    - Bitwise operators
    - Equality operators
    - Relational operators
      - The in operator
      - The instanceof operator
    - Unary operators
      - Incrementing and decrementing
      - Converting to a number with the plus sign
      - Creating a negative number with the minus sign
      - Negating with bitwise not and logical not
      - Using the delete operator
      - Returning variable types with the typeof operator
    - Assignment operators
    - The comma operator
    - Exercises
  - Chapter 6: Controlling flow with conditionals and loops
    - If (and how)
      - Syntax for if statements

        ```
        function processConfirm(answer) {
            var result = "";
            if (answer) {
                result = "Excellent.  We'll play a nice game of chess.";
            } else {
                result = "Maybe later then.";
            }
            return result;
        }
        </script>
        </head>
        <body>
        <script type="text/javascript">
        var confirmAnswer = confirm("Shall we play a game?");
        var theAnswer = processConfirm(confirmAnswer);
        alert(theAnswer);
        </script>
        </body>
        </html>
        ```

        3. Save the page, and view it in a web browser. You are presented with a dialog box that looks like this:

      - The prompt() function in Internet Explorer
      - Compound conditions
    - Using else if and else statements
    - Working with ternary conditionals
    - Testing with switch
    - Looping with while
      - The while statement
      - The do...while statement
    - Using for loops
      - The for loop
      - The for...in loop
      - The for each...in loop
    - Validating forms with conditionals
    - Exercises
  - Chapter 7: Working with functions
    - What's in a function?
      - Function parameters
      - Variable scoping revisited
      - Return values
      - More on calling functions
      - Anonymous/unnamed functions (function literals)
      - Closures
    - Methods
    - A look at dialog functions
    - Exercises
  - Chapter 8: Objects in JavaScript
    - Object-oriented development
      - Objects
      - Properties
      - Methods
      - Classes
    - Creating objects
      - Adding properties to objects
      - Adding methods to objects
    - Finding out more about arrays
      - The length property
      - Array methods
    - Taking advantage of built-in objects
      - Making URIs safe

        Making URIs safe

        The encodeURI() method takes a Uniform Resource Identifier (URI) that contains characters that are not allowed in a given URI scheme, and it encodes them so that they can be used according to the standard. For example, RFC (Request For Comments) 2396 defines a generic syntax for URIs. The encodeURI() method can be used to correct the following URI:

        ```
        http://www.braingia.org/a url with spaces.htm
        ```

        The preceding URI contains spaces—which are not allowed in an HTTP URI—and therefore the URI needs to be encoded (found as encode.html in the companion content):

        ```
        alert(encodeURI("http://www.braingia.org/a url with spaces.htm"));
        ```

        148 PART I Javawhat? The Where, Why, and How of JavaScript

        www.it-ebooks.info

        Figure 8-5 shows the result.

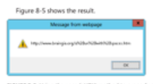
        

        FIGURE 8-5 Using the encodeURI() method to properly encode a URI in JavaScript.
      - Using the eval() method
    - Exercises
  - Chapter 9: The Browser Object Model
    - Introducing the browser
      - The browser hierarchy
      - Events
    - A sense of self

      Determining a visitor's screen height and width

      1. Using Microsoft Visual Studio, Eclipse, or another editor, edit the screen.html file in the Chapter09 sample files folder in the companion content.
      2. In the page, add the boldface code shown here (you can find this in the screen.txt file in the companion content):

      ```
      <!doctype html>
      <html>
      <head>
          <title>Screen</title>

      </head>
      <body>

        <script type="text/javascript">
              alert("Available Height: " + screen.availHeight);
              alert("Total Height: " + screen.height);
              alert("Available Width: " + screen.availWidth);
              alert("Total Width: " + screen.width);
        </script>

      </body>
      </html>
      ```

      3. Save and view the page in a web browser. You receive four *alert()* dialog boxes, one for each of the properties called. The following sample screen shots reflect a 1,024 × 768 pixel display.

      - Getting information about the screen
    - Using the navigator object
    - The location object
    - The history object
    - Exercises
  - Chapter 10: An introduction to JavaScript libraries and frameworks
    - Understanding programming libraries
    - Defining your own JavaScript library
    - Looking at popular JavaScript libraries and frameworks
      - jQuery
      - Modernizr
      - Yahoo! User Interface
      - MooTools
      - Other libraries
    - Exercises
  - Chapter 11: An introduction to jQuery
    - jQuery primer
    - Using jQuery
      - The two jQuery downloads
      - Including jQuery
      - Basic jQuery syntax
      - Connecting jQuery to the load event
    - Using selectors
      - Selecting elements by ID
      - Selecting elements by class
      - Selecting elements by type
      - Selecting elements by hierarchy
      - Selecting elements by position
      - Selecting elements by attribute
      - Selecting form elements
      - More selectors
    - Functions
      - Traversing the DOM
      - Working with attributes
      - Changing text and HTML
      - Inserting elements
      - Callback functions
    - More jQuery
    - Exercises
- Part II: Integrating JavaScript into Design
  - Chapter 12: The Document Object Model
    - The Document Object Model defined
      - DOM Level 0: The legacy DOM
      - DOM Levels 1 through 3
      - The DOM as a tree
      - Working with nodes
    - Retrieving elements
      - Retrieving elements by ID
      - Retrieving by tag name
      - HTML collections
      - Working with siblings
    - Working with attributes

      ```
      </head>
      <body>
      <p id="http://www.braingia.org/" id="BraingaFile"><bisa Sudbring's Web Site</a>
      <p id="dateField">debug</p>
      <script type = "text/javascript">
          var p = document.getElementById("BraingaFile");
          alert(p.getAttribute("id"));
          var d = document.getElementById("dateField");
          alert(d.getAttribute("id"));
      </script>
      </body>
      </html>
      ```

      When you view this page in a web browser, you see an alert() dialog box that displays the current value of the *id* attribute, as shown in Figure 12-5.

      FIGURE 12-5 The initial value of the *id* attribute.

      When the dialog box closes, the setAttribute() method executes and the *id* attribute changes, as shown in Figure 12-6.

      FIGURE 12-6 The new value of the *id* attribute.
      - Viewing attributes
      - Setting attributes
    - Creating elements
      - Adding text
      - Adding an element and setting an ID
    - Deleting elements
    - Exercises
  - Chapter 13: JavaScript events and the browser
    - Understanding window events
      - The event models
      - A generic event handler
      - jQuery event handling
      - Binding and unbinding
    - Detecting visitor information
      - A brief look at the userAgent property
      - Feature testing
      - Keeping JavaScript away from older browsers
      - Other navigator properties and methods
    - Opening, closing, and resizing windows
      - Window opening best practices
      - Opening tabs: no JavaScript necessary
      - Resizing and moving windows
    - Timers
    - Events
      - Mouse events and hover
      - Many more event handlers
    - Exercises
  - Chapter 14: Working with images in JavaScript
    - Working with image hovers
      - A simple hover

        3. View the page in a browser. You should see a page similar to the following screen shot. If you run into problems, make sure that each of the images and jQuery are located in the current directory, because that's where the <MM> tag is looking for them.

        4. Move the mouse over the buttons one at a time. Each image should change to its corresponding hover image. Here's what the screen looks like when the mouse is over the Blog graphic:

      - Modern hovers with jQuery
      - A closer look at the exercise
    - Preloading images
    - Working with slide shows
      - Creating a slide show
      - Moving backward
      - A jQuery slide show
    - Working with image maps
    - Exercises
  - Chapter 15: Using JavaScript with web forms
    - JavaScript and web forms
      - Validation with JavaScript
      - Validation with jQuery
    - Working with form information
      - Working with select boxes
      - Working with check boxes
      - Working with radio buttons

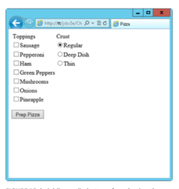
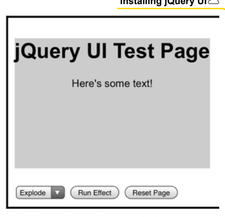
        

        FIGURE 15-8 Adding radio buttons for selecting the crust type.
      - Hacking JavaScript validation
      - Validating a text field

        FIGURE 15-13 Providing inline feedback on a webpage rather than an alert() dialog box.
      - Pre-validating form data
    - Exercises
  - Chapter 16: JavaScript and CSS
    - What is CSS?
      - Using properties and selectors
      - Applying CSS
    - The relationship between JavaScript and CSS
      - Setting element styles by ID
      - Setting element styles by type
      - Setting CSS classes with JavaScript
      - Retrieving element styles with JavaScript
    - Exercises
  - Chapter 17: jQuery effects and plug-ins
    - Installing jQuery UI
      - Obtaining jQuery UI
      - Installing jQuery UI

      - Building a jQuery UI demonstration page
    - Creating a jQuery UI calendar

      The result is a calendar that looks like Figure 17-3.

      FIGURE 17-3 A datepicker with a button panel.
      - Customizing the calendar
    - Adding a dialog box
      - Creating a modal dialog
      - Adding buttons
    - More JQuery UI
    - Exercises
  - Chapter 18: Mobile development with jQuery Mobile
    - A walkthrough of jQuery Mobile
    - Getting jQuery Mobile
      - Downloading jQuery Mobile
      - Testing jQuery Mobile
    - Linking with jQuery Mobile
      - Creating a link
      - Changing the page transition
      - Linking without AJAX
    - Enhancing the page with toolbars
      - Adding a navigation bar
      - Adding a footer navigation bar

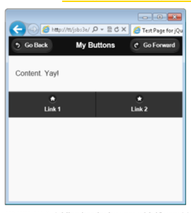
        

        FIGURE 18-8 Adding header buttons with jQuery Mobile.
      - Adding buttons to toolbars
      - Other toolbar enhancements
    - Even more jQuery Mobile
    - Exercises
  - Chapter 19: Getting data into JavaScript
    - JavaScript application architecture
      - The big three: display, behavior, data
    - JavaScript and web interfaces
    - Introduction to AJAX
    - AJAX with jQuery
    - AJAX without the X
    - What's Next?
- Part III: AJAX and Server-Side Integration
  - Chapter 20: Using AJAX
    - The XMLHttpRequest object
      - Instantiating the XMLHttpRequest object
      - Sending an AJAX request
      - Processing an AJAX response
      - Processing XML responses
      - Working with JSON
      - Processing headers
      - Using the POST method
    - AJAX and jQuery
      - AJAX errors and time-outs
      - Sending data to the server
      - Other important options
    - Exercise
  - Chapter 21: Developing for Windows 8
    - Windows 8 apps
    - Developing Windows 8 apps
      - Development guidelines
      - The development process
    - Distributing Windows apps
      - Distributing in the Windows Store
      - Distributing in an enterprise
    - Summary
- Part IV: JavaScript and Windows 8
  - Chapter 22: Using Visual Studio for Windows 8 development
    - Installing Visual Studio 2012
    - Installing Visual Studio 2012 Express for Windows 8
    - Windows 8 app templates
      - Blank App template
      - Grid App template
      - Split App template
    - Setting app details in the App Manifest
    - Packaging apps for the Windows Store
      - Certification requirements
      - The technical process
    - Exercises
  - Chapter 23: Creating a Windows app
    - The app development process
    - Starting the app design and programming
    - Customizing the app
      - Customize the JavaScript
      - Customize the main HTML
      - Customize the detail HTML
      - Customize the CSS
      - Finalizing the app
    - Customizing the Package Manifest
      - Adding a splash screen, logo, and tile image
      - Defining capabilities
    - Testing the app
    - Summary
- Appendix A: Answer key to exercises
- Appendix B: Writing JavaScript with other tools
- Index