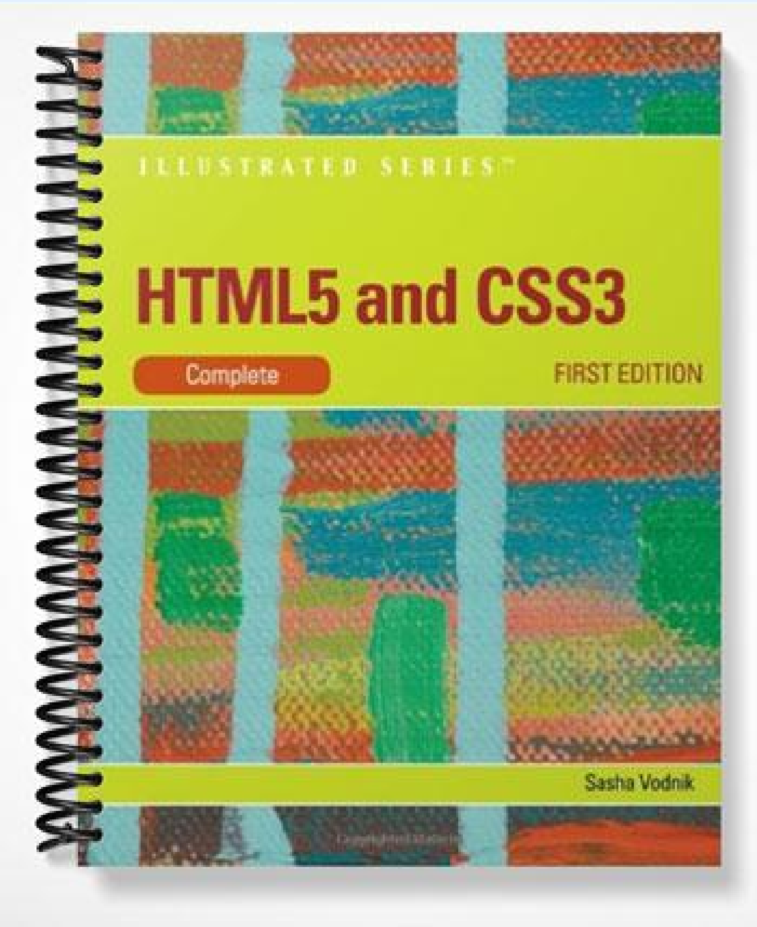
SOLUTIONS MANUAL



ILLUSTRATED SERIES™

HTML5 and CSS3

Complete

FIRST EDITION

Sasha Vodnik

HTML 5 and CSS 3 - Illustrated Complete

Unit B: Getting Started with HTML

A Guide to this Instructor's Manual:

We have designed this Instructor's Manual to supplement and enhance your teaching experience through classroom activities and a cohesive chapter summary.

This document is organized chronologically, using the same heading in **dark blue** that you see in the textbook. Under each heading you will find (in order): Lecture Notes that summarize the section, Figures and Boxes found in the section, if any, Teacher Tips, Classroom Activities, and Lab Activities. Pay special attention to teaching tips, and activities geared towards quizzing your students, enhancing their critical thinking skills, and encouraging experimentation within the software.

In addition to this Instructor's Manual, our Instructor's Resources CD also contains PowerPoint Presentations, Test Banks, and other supplements to aid in your teaching experience.

For your students:

Our latest online feature, CourseCasts, is a library of weekly podcasts designed to keep your students up to date with the latest in technology news. Direct your students to http://coursecasts.course.com, where they can download the most recent CourseCast onto their mp3 player. Ken Baldauf, host of CourseCasts, is a faculty member of the Florida State University Computer Science Department where he is responsible for teaching technology classes to thousands of FSU students each year. Ken is an expert in the latest technology and sorts through and aggregates the most pertinent news and information for CourseCasts so your students can spend their time enjoying technology, rather than trying to figure it out. Open or close your lecture with a discussion based on the latest CourseCast.

Table of Contents

Unit Objectives

Students will have mastered the material in Unit B when they can:

- Assess the history of HTML
- Compare HTML and XHTML
- Create an HTML document
- Set up the document head and body
- Add Web page text
- Preview your Web page
- Implement one-sided tags
- Validate your HTML code

26: Assessing the History of HTML

LECTURE NOTES

- Explain that the World Wide Web Consortium (W3C) is the organization responsible for maintaining HTML standards, incorporating new features into the language while keeping it consistent, and maintaining a vision of the next steps in HTML development.
- Stress that although HTML has several defined versions, it is constantly transforming.
- Point out that HTML 4 codified HTML as a semantic language, which is intended to indicate the meanings of elements in a Web page, but not to define how the elements should appear.
- Note that Cascading Style Sheets (CSS) is a companion language to HTML, which is a presentational language designed for describing the appearance of items.
- Explain deprecated HTML features, which are presentational HTML features that do not fit the current model of HTML, and are therefore not recommended for use. Point out that alternatives for their use are available.

TABLE: B-1

BOXES:

1. **Clues to Use: HTML and Web browser versions**
In the early years of the Web, browser creators Microsoft and Netscape added to their browsers' proprietary features that weren't supported by their competitors. This provided opportunities for Web developers to add new elements to their Web pages. However, it also meant that any Web page incorporating these features would be displayed differently depending on which browser was used to open it. This situation created difficulties for developers trying to reach the widest possible audience with their Web pages. Eventually, the major browser companies recognized that creating their own features would not be a major key to gaining or maintaining market share, and they have joined with developers to support a largely standards based Web.

On the flip side, recent updates to Web standards have not always been quickly or fully adopted by makers of browsers and other user agents. As a result, when considering whether to use a particular HTML feature in your Web pages, it's important to research which user agents support it and how widely the supporting user agents are in use. In some cases, implementing new features will need to wait until your target audience has caught up with available Web technologies.

CLASSROOM ACTIVITIES

1. **Critical Thinking**: Ask students to think of reasons why it is important for HTML to be standardized. What are the pros and cons of including user-agent specific features?

2. **Classroom Discussion:** Ask students to raise issues or problems that may have come up during prior Web-browsing experiences. Then, have them discuss possible changes that could be made to HTML standards or to Web-browsers and user-agents that would improve their Web-browsing experience.

28: Comparing HTML and XHTML

LECTURE NOTES:

- Explain that XML (Extensible Markup Language) is another markup language, which co-exists with HTML and has spawned many additional specialized "child" languages. XML allows a user to describe the structure of a document.
- Point out that XML is more generic than HTML and allows a user to define many different types of documents, not only Web pages.
- Note that XHTML is a markup language intended to make HTML comply with the rules of XML.
- Show how adapting existing HTML standards to adhere to XML syntax requires Web page authors to change some of the way they write code. For example, in order to adhere to XML syntax, Web page authors must write a closing tag for every opening tag, or, put in other words, Web page authors must use only two-sided tags. You can also discuss other more examples covered in Table B-2.

> TEACHER TIP
> The definition of XHTML as complying with the rules of XML means that all Web pages written in XHTML are valid according to the requirements of HTML, but not all pages written in HTML are valid according to the rules of XHTML.

- Explain that HTML is a flexible language, and stress the importance of this flexibility. For example, if a Web-page author made a minor mistake in writing code, user agents would still be able to display the Web page correctly.
- Stress that XML does not tolerate errors, and therefore XHTML, which adheres to XML rules, does not tolerate errors. Specify that this means that if a user agent encounters a coding error in XHTML code, it must display an error message.
- Point out that initially the idea was to use XHTML as a successor to HTML, but some of the Web standards community did not like this idea, because it would mean that Web pages containing coding errors would not be displayed correctly (or at all), and new features developed in later XHTML versions would not be available to existing HTML Web pages unless they were edited to comply with XHTML rules. This then led to the development of the HTML 4.01 standard.
- Note that the W3C is improving and expanding both XHTML and HTML, as they are now seen as parallel languages.

- Explain to students that they will learn to create both XHTML and HTML documents; and in the future will be able to decide which language is more suitable for specific projects.

TABLE: B-2

BOXES:

1. **Clues to Use**: **The W3C and the WHATWG**
After XHTML 1.2 was finalized, the W3C moved forward with adding new features to XHTML, and drafted a proposal for XHTML 2.0. However, a number of community members, including several major technology companies, felt that HTML, rather than XHTML, would better serve the future of the Web, and proposed that the W3C change course. When a W3C committee voted against this proposal, these companies formed the Web Hypertext Application Technology Working Group (WHATWG) to begin a process of creating a new HTML specification. Over time, the two organizations bridged their differences and agreed to collaborate in creating HTML5, with the W3C coordinating work on XHTML5.

Before finalizing the latest version of HTML, the two organizations are following a process of soliciting input and incorporating feedback from members of the Web community, including makers of user agents and other software and Web developers. As a result, the development process will likely continue for several years before all parties involved finalize a single standard. In the interim, however, different features of the developing specification may become available at different times.

CLASSROOM ACTIVITIES

1. **Quick Quiz**:

   a. T/F XML is a more generic language than HTML. (T)

   b. T/F Every HTML document is a valid XHTML document. (F)

   c. T/F XHTML replaced HTML as the only language for writing Web pages. (F)

   d. T/F XHTML was developed in order to facilitate the interoperation of Web pages with documents coded using XML-based languages. (T)

2. **Classroom Discussion:** Look at Table B-2, and discuss possible considerations for using XHTML over HTML. Try to think of specific types of projects that would benefit from being written in XHTML rather than HTML. Repeat the exercise for HTML.

LAB ACTIVITY

1. Table B-2 lists specific differences between HTML and XHTML. Ask students to look at the source code of an HTML Web page of their choice, and to use the information in Table B-2 to determine whether or not the Web page complies with XHTML rules.

30: Creating an HTML Document

LECTURE NOTES

- Explain that an HTML document consists solely of text, and that HTML code does not include any images. As a result, an HTML document or Web page can be written in any text editor.
- Illustrate that to create a Web page you enter text that you want to display on the page, along with tags which specify how a user agent should treat each item in the document.
- Explain that a tag pair assigns meaning to a Web page element, which is a specific component of the page. The opening tag is placed at the start of the element and the closing tag at the end. HTML tags always start with an opening angle bracket (<) and end with a closing angle bracket (>). A closing tag is the same as its corresponding opening tag except that the opening angle bracket is followed by a slash (/). The text between the angle brackets specifies the HTML element type being applied to the selection.
- Show that most tags occur in pairs and include an opening tag, such as <tag>, and a closing tag, such as </tag>, but some tags, which are called one-sided tags, are used by themselves. An example of such a tag is the line break tag: <br/>.
- Explain that to create a Web page, one must define the document as being written in HTML by using the DOCTYPE declaration <!DOCTYPE html>.

  TEACHER TIP
  In order for a Web page to be rendered correctly in some Web browsers, the DOCTYPE declaration must be the first thing to appear on the Web page.

- Point out that in order to write an html page, the contents of the page must be surrounded by <html> tags. Therefore, advise students to always write these tags as a first step to creating their document.
- Recommend to your students that for the clarity of their code, they should leave two lines of space between the opening and closing html tags.

FIGURES: B-1, B-2

BOXES

1. **Quick Tip**: To read more about any HTML tag used in these steps, see Appendix A.

2. **Trouble**: If you are using TextEdit on a Mac, uncheck the Hide extension check box if necessary.

3. **Clues to Use: Other Web page creation software**
Many other programs are available that allow you to create Web pages visually by clicking buttons and using drag-and-drop to place items on a page. However, creating your first Web pages by entering HTML directly—sometimes referred to as hand-coding—is one of the best ways to get familiar with HTML and the underlying structure of a Web page.

CLASSROOM ACTIVITIES

1. **Critical Thinking**: Ask students to consider what would happen if someone wrote a Web page including an opening <html> tag, but forgot to add the corresponding closing tag. How would a user agent treat the student's Web page?

2. **Critical Thinking**: Based on students' knowledge of the format for creating an HTML document, ask how they expect to create an XHTML document?

32: Setting Up the Document Head and Body

LECTURE NOTES

- Explain that each Web page is divided into a head section, which contains elements that are not part of the main Web page, and a body section, the contents of which are visible in the main window of the Web browser.
- Explain the concept of nesting elements – some elements in an HTML page are included within other elements. For example, both head and body tag pairs are located within the html tag pair.
- Explain to students that adding spaces before a nested tag makes it appear indented, making complex code easier to read and elements easier to identify.
- Introduce students to the meta element, which enables a Web page author to pass information about a Web page to the user agent that opens it.
- Note that some HTML elements may include attributes, which specify details about a given element's properties. For example, the charset attribute specifies the character encoding, which is the system the user agent should employ to translate the electronic information into human recognizable symbols, such as letters and numbers.
- Show that in Figure B-3, the <head> and <body> tag pairs are nested within the <html> tag pair, and are indented with respect to the <html> tag pair but not with respect to each other.

> TEACHER TIP
> In order to make sure that all opening tags have corresponding closing tags, recommend that whenever students write an opening tag they immediately write the closing tag, and only then add the contents of the tag pair between the opening and closing tags.

FIGURES: B-3, B-4

BOXES

1. **Quick Tip**: To save your work without using the menus, you can press **[Ctrl]+[S]** (Windows) or **[command]+[S]** (Mac).

2. **Clues to Use: Describing nested elements**
An element nested within another element is called a child element of the enclosing element, and the enclosing element is known as the parent element. Two elements that are both children of the same element are known as sibling elements.
In the code:

```
<html>
```

```
<head>
   <meta charset="utf-8" />
```

the head element is both a child of the html element and the parent of the meta element. In addition, here the html element is the grandparent element of the meta element, which can be referred to as a grandchild element of the html element.

3. **Clues to Use: HTML attributes**
Many, but not all, HTML elements allow you to set attributes to specify details about a given element's properties. To use an attribute, you provide two pieces of information: an attribute name and the value you are assigning to the attribute. Together, these two pieces are known as a name/value pair. In the element <meta charset="utf-8" />, *charset* is the attribute name, and *utf-8* is the attribute value. In a tag pair, you specify any attributes in the opening tag, never in the closing one.

CLASSROOM ACTIVITIES

1. **Classroom Discussion**: As mentioned in the Unit, the meta element enables a Web page author to pass information about a Web page to the user agent that opens it. Discuss with your class what kinds of information should be included in attributes of the meta tag (possible answers: description of the Web page, keywords, when a page expires, or should be refreshed). What kind of information should you include in attributes of an image tag? (possible answers: image dimensions, alignment, text to show if the image doesn't load, what to do if you mouse-over the image).

2. **Critical Thinking**: Based on their personal experience with other languages/systems, ask students to think about the advantages of splitting a Web page into head and body sections

34: Adding Web Page Text

LECTURE NOTES

- Explain that because an HTML document is a simple text document that includes HTML codes, adding text to a Web page is as simple as typing it and adding the appropriate HTML tags to specify the relevant element type for each text item.

  TEACHER TIP
  Sketching the layout of your Web site, using a piece of paper and a pen or a word processing application, often helps determine how the various elements should be included in a Web page, and find errors in the resulting Web page.

- Show students that when adding text to a Web page, they can also add surrounding elements, such as the Web page title that will appear in the user-agent title bar. Stress that such elements, such as the title element, are part of the document's head section because this element does not appear in the main browser window.
- Point out the meaning of various basic HTML tags, such as <h1> for a top level heading and <p> for a paragraph of text.

FIGURES: B-5, B-6

BOXES

1. **Quick Tip**: This line is indented four spaces because it is nested within two elements: html and head.

2. **Clues to Use**: **Adding comments to your HTML code**
In addition to marking text that appears on your Web pages, you can create text elements in your Web page code that user agents ignore. These elements, known as comments, are not rendered by user agents and are viewable only by people who examine the HTML code of your Web pages. Comments can be especially helpful when you are creating or adding on to a large, complex Web document or Web site, or when other Web developers will be working with your code—now or in the future. Common uses for comments include explaining what a particular section of HTML does or pointing out the beginning and end of parts of a Web page containing numerous HTML elements. The comment tag pair begins with <!-- and ends with -->.

CLASSROOM ACTIVITIES

1. **Critical Thinking**: Assume that you are looking at a Web page that includes a heading, then a paragraph of text, and then another heading. You want to change this Web page to include only plain paragraphs of text, with no headings. How many tags will you need to change in order to achieve this?

2. **Quick Quiz:**

a. T/F In order to write a Web page you need to first write all the text and then add all the tags. (F)

b. T/F Nested elements must be indented with respect to parent elements in order for the HTML code to be properly displayed in a browser. (F)

36: Previewing Your Web Page

LECTURE NOTES:

- Explain the importance of previewing a Web page as an aid in finding errors in the code before publishing the page.
- In order to understand how your code would be rendered in different Web browsers, stress that it is advisable to preview a Web page in multiple user agents, since the interpretation of the code may vary.
- Show students that in order to preview a Web page in their default Web browser, they have to use their file manager to find the Web page they wish to preview, and double click the name of the relevant .html file.
- Show students that in order to preview a Web page in a non-primary Web browser they need to return to the file manager, and right-click (Windows) or control-click (Mac) the name of the relevant .html file, point to "Open with" and then click the selected browser name.
- Explain that if the Web page being previewed is not rendered correctly, it can be edited, saved, and previewed again, until it looks correct.

TEACHER TIP
Stress to students that in order for changes made to the source HTML to take effect, the Web page must be refreshed or re-opened in the Web browser.

FIGURES: B-7, B-8
BOXES:

1. **Trouble**: If your Web page does not match Figure B-6, return to Notepad, compare your code to Figure B-7, edit as necessary, and save the file, then repeat Step 2 to preview your edited Web page.

2. **Clues to Use: Why browsers display Web pages differently**
The display of Web pages in HTML5 starts with the standards created by the W3C and the WHATWG. The standards list and describe all the available elements, along with parameters for how user agents should handle them. User agents are built around software known as rendering engines that translate Web page elements into visual, auditory, or tactile representations based on these standards. Because the standards require some interpretation, no two engines render the same HTML code in exactly the same way. In addition, the creators of rendering engines do not always implement all of the current standards in their software. Because the audience for your Web pages will almost always be using a number of different user agents, it's important to test your code in a variety of popular browsers and on multiple operating systems (such as Windows 7, Windows XP, and Mac OS X).

CLASSROOM ACTIVITIES

1. **Quick Quiz**:

   a. T/F All Web browsers render a given Web page in the same way. (F)

   b. T/F To preview a Web page, you should open it in one or more Web browsers. (T)

2. **Class Discussion**: Look at Figure B-8, and discuss why certain elements of the Illustrated Web page are rendered identically in the different browsers, while other elements are portrayed differently in the different browsers.

LAB ACTIVITY

1. Select a Web page of your choice, and open it in two different Web browsers. Identify the elements of the Web page that are different between the different Web browsers.

38: Implementing One-Sided Tags

LECTURE NOTES:

- Explain that some HTML elements do not require tag pairs, but only a single tag. These tags usually do not affect text that needs to be delineated by them, but rather generally represent a specific occurrence of an item or behavior.
- Show that a one-sided tag is implemented by typing a triangular opening bracket, the name of the tag, a space followed by a slash and a triangular closing bracket.
- Explain that Web browsers ignore the [ENTER] characters in the HTML code, and that in order to insert a line break in the Web page one must use the <br /> one-sided tag.

TEACHER TIP
The fact that browsers ignore [ENTER] characters does not mean that these should be omitted from HTML source code. The [ENTER] characters in the source code are required in order to make the code legible and understandable.

- Emphasize that in order to observe changes made to their HTML code, they must press the refresh or reload button of their browser after saving the changes to their code.

FIGURES: B-9, B-10

TABLE: B-3

BOXES:
1. **Quick Tip**: Be sure to type a space between the r and the /.

CLASSROOM ACTIVITY

1. **Quick Quiz**:

   a. T/F All HTML tags can be implemented as one-sided tags. (F)

   b. T/F A one-sided HTML tag is identical to the closing tag of an HTML tag pair. (F)

2. **Quick Quiz**: In order to view changes made to an updated Web page, the page must be ____ or ____. (refreshed, reloaded)

40: Validating Your HTML Code

LECTURE NOTES:

- Explain to students the importance of spotting problems with HTML code and making sure that users can view your Web pages as expected.
- Present the idea of validation, an automated process of comparing HTML code against the HTML5 coding standards, as a way of finding errors in your code. This is helpful because it not only shows you that there is an error, but may identify the specific source of the problem.
- Illustrate to your students how to validate a Web page by opening http://validator.w3.org/ in a browser, uploading a simple HTML file using the "validate by file upload" tag and have the validator check your HTML file. Show students how the validation results look and where they can be found in the validator Web page.

FIGURES: B-11, B-12

BOXES:
1. **Trouble**: If your document does not successfully validate as HTML5, return to your text editor, compare your document to Figure B-9 and make any necessary changes, then save your work and revalidate your file.

CLASSROOM ACTIVITIES

1. **Quick Quiz:**

   a. T/F It is important to correct errors in HTML code because it can ensure that a Web page will continue to work with future versions of HTML standards. (T)

   b. T/F If the HTML code of a specific Web page is not validated you will not be able to open the Web page in a Web browser. (F)

2. **Critical Thinking:** Is it possible to use a single validation tool to validate documents written in different mark-up languages, such as HTML, XML, XHTML, CSS etc.? Why or why not?

LAB ACTIVITY

1. Select a Web page of your choice, and validate it using the "validate by URI" tab of http://validator.w3.org/. If the validation tool shows no comments as to the validity of the Web page, copy the source code of the Web page into a new text document and save it. Make a few changes to the tags included in the copied source code, and then validate the edited Web page. How did your changes affect the validity of the Web page?

End of Unit Material

- **Concepts Review** – Includes screen identification, multiple choice, and matching questions.

- **Skills Review** – Provides additional hands-on exercises that mirror the progressive style of the lesson material.

- **Independent Challenges 1, 2 and 3** – Case projects that require critical thinking and application of the unit skills. The Independent Challenges increase in difficulty, with the first being the easiest (with the most step-by-step detailed instructions). Independent Challenges 2 and 3 become increasingly more open-ended.

- **Real Life Independent Challenge** – Using a real world focus to apply the unit skills, students perform tasks or create documents that will benefit their everyday lives.

- **Advanced Challenge Exercises** – Optional steps that are set within 2 of the 4 Independent Challenges for each unit. They are designed to challenge more advanced students. Two solution files are provided for easy grading.

- **Visual Workshop** – a practical, self-graded capstone project that requires independent problem solving.

Glossary of Key Terms

- attribute (32)
- body section (32)
- Cascading Style Sheets (CSS) (26)
- character encoding (32)
- child element (32)
- closing tag (30)
- comments (34)
- deprecated (26)
- DOCTYPE declaration (30)
- element (30)
- Extensible Hypertext Markup Language (XHTML) (28)
- Extensible Markup Language (XML) (28)
- grandchild element (32)
- grandparent element (32)
- hand-coding (30)
- Hypertext Markup Language (HTML) (25)
- Internet Engineering Task Force (IETF) (26)
- name/value pair (33)
- nesting (32)
- one-sided tags (30)
- opening tag (30)
- parent element (32)
- preview (36)
- rendering engines (36)
- renders (36)
- semantic (26)
- sibling elements (32)
- tags (30)
- Web Hypertext Application Technology Working Group (WHATWG) (29)
- World Wide Web Consortium (W3C) (26)

Top of Document