Terry Ann Felke-Morris

Basics of Web Design

HTML5 & CSS

SIXTH EDITION

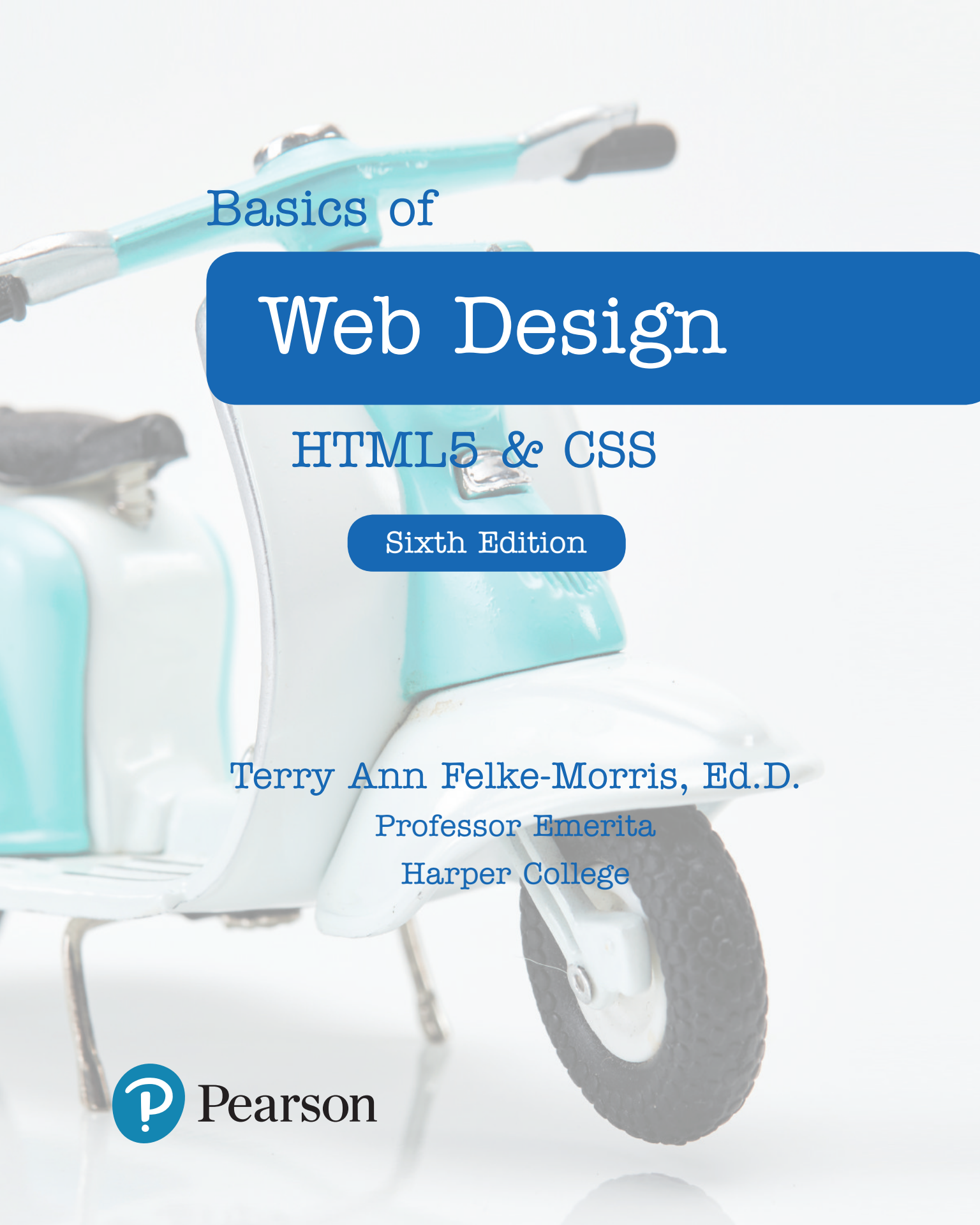
Basics of

Web Design

HTML5 & CSS

Sixth Edition

This page intentionally left blank



Basics of Web Design

HTML5 & CSS

Sixth Edition

Terry Ann Felke-Morris, Ed.D.
Professor Emerita
Harper College

Pearson

Content Development: Tracy Johnson
Content Management: Dawn Murrin, Tracy Johnson
Content Production: Carole Snyder
Product Management: Holly Stark
Product Marketing: Wayne Stevens
Rights and Permissions: Anjali Singh

Please contact https://support.pearson.com/getsupport/s/ with any queries on this content

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability, whether express, implied or statutory, fitness for a particular purpose, title and non-infringement. In no event shall Microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time. Partial screen shots may be viewed in full within the software version specified.

Microsoft® and Windows® are registered trademarks of the Microsoft Corporation in the U.S.A. and other countries. This book is not sponsored or endorsed by or affiliated with the Microsoft Corporation.

Copyright © 2022 by Pearson Education, Inc. or its affiliates, 221 River Street, Hoboken, NJ 07030. All Rights Reserved. Manufactured in the United States of America. This publication is protected by copyright, and permission should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise. For information regarding permissions, request forms, and the appropriate contacts within the Pearson Education Global Rights and Permissions department, please visit www.pearsoned.com/permissions/.

Acknowledgments of third-party content appear on page 484, which constitutes an extension of this copyright page.

PEARSON, ALWAYS LEARNING, and ETEXT are exclusive trademarks owned by Pearson Education, Inc. or its affiliates in the U.S. and/or other countries.

Unless otherwise indicated herein, any third-party trademarks, logos, or icons that may appear in this work are the property of their respective owners, and any references to third-party trademarks, logos, icons, or other trade dress are for demonstrative or descriptive purposes only. Such references are not intended to imply any sponsorship, endorsement, authorization, or promotion of Pearson's products by the owners of such marks, or any relationship between the owner and Pearson Education, Inc., or its affiliates, authors, licensees, or distributors.

Library of Congress Cataloging-in-Publication Data

Names: Felke-Morris, Terry, author.
Title: Basics of web design : HTML5 & CSS / Terry Ann Felke-Morris, Ed.D., Professor Emerita.
Description: Sixth edition. | Hoboken : Pearson, 2022.
Identifiers: LCCN 2020030388 | ISBN 9780137313266 (Print Offer) | ISBN 9780137313211 (Rental)
Subjects: LCSH: HTML (Document markup language) | Cascading style sheets. | Web site development—Computer programs. | Web sites—Design.
Classification: LCC QA76.76.H94 F455 2020 | DDC 006.7/4—dc23
LC record available at https://lccn.loc.gov/2020030388

ScoutAutomatedPrintCode

Rental
ISBN 10: 0-13-731321-7
ISBN 13: 978-0-13-731321-1

Print Offer
ISBN-10: 0-13-731326-8
ISBN-13: 978-0-13-731326-6

Preface

Basics of Web Design: HTML5 & CSS is intended for use in a beginning web design or web development course. Topics are introduced in two-page sections that focus on key points and often include a hands-on practice exercise. The text covers the basics that web designers need to develop their skills:

- Introductory Internet and World Wide Web concepts
- Creating web pages with HTML5
- Configuring text, color, and page layout with Cascading Style Sheets
- Configuring images and multimedia on web pages
- Exploring CSS Flexbox and CSS Grid layout systems
- Web design best practices
- Creating responsive web pages that display well on both desktop and mobile devices
- Accessibility, usability, and search engine optimization considerations
- Obtaining a domain name and a web host
- Publishing to the Web

Student files include solutions to the Hands-On Practice exercises, starter files for the Hands-On Practice exercises, and the starter files for the Case Study. The eText offers student file downloads by chapter (where used) wiithin each chapter introduction. Student files are also available for download from the companion website for this book at www.pearson.com/felke-morris.

Building on this textbook's successful fifth edition, the sixth edition features:

- Additional Hands-On Practice exercises
- Updated code samples, case studies, and web resources
- Updates for HTML5 elements and attributes
- Expanded treatment of page layout design and responsive web design techniques
- Expanded treatment of CSS Flexible Layout Module (Flexbox) and CSS Grid Layout systems
- Expanded coverage of responsive image techniques including lazy loading
- Updated reference sections for HTML5 and CSS

Features of the Text

Design for Today and Tomorrow. The textbook prepares students to design web pages that work today in addition to being ready to take advantage of new HTML5 and CSS coding techniques of the future.

Well-Rounded Selection of Topics. This text includes both "hard" skills such as HTML5 and Cascading Style Sheets (Chapters 1–2 and 4–11) and "soft" skills such as web design

(Chapter 3) and publishing to the Web (Chapter 12). This well-rounded foundation will help students as they pursue careers as web professionals. Students and instructors will find classes more interesting because they can discuss, integrate, and apply both hard and soft skills as students create web pages and websites. The topics in each chapter are typically introduced on concise two-page sections that are intended to provide quick overviews and timely practice with the topic.

Two-Page Topic Sections. Most topics are introduced in a concise, two-page section. Many sections also include immediate hands-on practice of the new skill or concept. This approach is intended to appeal to your busy students—especially the millennial multitaskers—who need to drill down to the important concepts right away.

Hands-On Practice. Web design is a skill, and skills are best learned by hands-on practice. This text emphasizes hands-on practice through practice exercises within the chapters, end-of-chapter exercises, and the development of a website through ongoing real-world case studies. This variety provides instructors with a choice of assignments for a particular course or semester.

Website Case Study. There are case studies that continue throughout most of the text (beginning at Chapter 2). The case studies serve to reinforce skills discussed in each chapter. Sample solutions to the case study exercises are available on the Instructor Resource Center available through https://pearsonhighered.com/felke.

Focus on Web Design. Every chapter offers an additional activity that explores web design topics related to the chapter. These activities can be used to reinforce, extend, and enhance the course topics.

FAQs. In her web design courses, the author is frequently asked similar questions by students. They are included in the book and are marked with the identifying FAQ icon.

Focus on Accessibility. Developing accessible websites is more important than ever, and this text is infused with accessibility techniques throughout. The special icon shown here makes accessibility information easy to find.

Focus on Ethics. Ethical issues related to web development are highlighted throughout the text with the special ethics icon shown here.

Quick Tips. Quick tips, which provide useful background information, or help with productivity, are indicated with this Quick Tip icon.

Explore Further. The special icon identifies enrichment topics along with web resources useful for delving deeper into a concept introduced in book.

Reference Materials. The appendices offer reference material, including an HTML5 reference, a Cascading Style Sheets reference, a WCAG 2.1 Quick Reference, an overview of ARIA Landmark Roles and a Web Safe Color Palette.

VideoNotes. VideoNotes are Pearson's visual tool designed for teaching students key programming concepts and techniques. These short step-by-step videos demonstrate how to solve problems from design through coding. VideoNotes allow for self-placed instruction with easy navigation including the ability to select, play, rewind, fast-forward,

and stop within each VideoNote exercise. Margin icons in your textbook let you know when a VideoNote video is available for a particular concept or hands-on practice.

Supplemental Materials

Student Resources. Student resources provide both reinforcement and practice of new concepts and skills include:

- VideoNotes
- Student Files containing the following:
  - Hands-On Practice starter files
  - Hands-On Practice solutions
  - Case Study starter files

Author's Website. In addition to the publisher's companion website for this book, the author maintains a website at https://www.webdevbasics.net. This website contains a page for each chapter with additional resources and updates. This website is not supported by the publisher.

Acknowledgments

Very special thanks go to the people at Pearson, including Tracy Johnson, Carole Snyder, Scott Disanno, and Erin Sullivan.

Most of all, I would like to thank my family for their patience and encouragement. My wonderful husband, Greg Morris, has been a constant source of love, understanding, support, and encouragement. Thank you, Greg! A big shout-out to my children, James and Karen, who grew up thinking that everyone's Mom had their own website. Thank you both for your understanding, patience, and timely suggestions. Finally, a very special dedication to the memory of my father who will be greatly missed.

About the Author

Dr. Terry Ann Felke-Morris is a Professor Emerita at Harper College in Palatine, Illinois. She holds a Doctor of Education degree, a Master of Science degree in information systems, and numerous certifications, including Adobe Certified Dreamweaver 8 Developer, WOW Certified Associate Webmaster, Microsoft Certified Professional, Master CIW Designer, and CIW Certified Instructor.

Dr. Felke-Morris received the Blackboard Greenhouse Exemplary Online Course Award in 2006 for use of Internet technology in the academic environment. She is the recipient of two international awards: the Instructional Technology Council's Outstanding e-Learning Faculty Award for Excellence and the MERLOT Award for Exemplary Online Learning Resources—MERLOT Business Classics.

With more than 25 years of information technology experience in business and industry, Dr. Felke-Morris published her first website in 1996 and has been working with the Web ever since. A long-time promoter of web standards, she was a member of the Web Standards Project Education Task Force. Dr. Felke-Morris is the author of the popular textbook *Web Development and Design Foundations with HTML5*, currently in its tenth edition. She was instrumental in developing the Web Development degree and certificate programs at Harper College. For more information about Dr. Terry Ann Felke-Morris, visit https://terrymorris.net.

CONTENTS

VideoNotes

VideoNote

Locations of **VideoNotes**
www.pearson.com/felke-morris

CHAPTER 1

Internet and Web Basics

The Internet and the Web are parts of our daily lives. How did they begin? What networking protocols and programming languages work behind the scenes to display a web page? This chapter provides an introduction to some of these topics and is a foundation for the information that web developers need to know. This chapter also gets you started with your very first web page. You'll be introduced to Hypertext Markup Language (HTML), the language used to create web pages.

You'll learn how to...

- Describe the evolution of the Internet and the Web
- Explain the need for web standards
- Describe universal design
- Identify benefits of accessible web design
- Identify reliable resources of information on the Web
- Identify ethical uses of the Web
- Describe the purpose of web browsers and web servers
- Identify Internet protocols
- Define URIs and domain names
- Describe HTML, XHTML, and HTML5
- Create your first web page
- Use the body, head, title, and meta elements
- Name, save, and test a web page

The Internet and the Web

The Internet

The **Internet**, the interconnected network of computer networks, seems to be everywhere today. You can't watch television or listen to the radio without being urged to visit a website. Even newspapers and magazines have their place on the Internet. It is possible that you may be reading an electronic copy of this book that you downloaded over the Internet. With the increased use of mobile devices, such as tablets and smartphones, being connected to the Internet has become part of our daily lives.

The Birth of the Internet

The Internet began as a network to connect computers at research facilities and universities. Messages in this network would travel to their destinations by multiple routes or paths, allowing the network to function even if parts of it were broken or destroyed. The message would be rerouted through a functioning portion of the network while traveling to its destination. This network was developed by the Advanced Research Projects Agency (ARPA)—and the ARPAnet was born. Four computers (located at University of California, Los Angeles; Stanford Research Institute; University of California, Santa Barbara; and the University of Utah) were connected by the end of 1969.

Growth of the Internet

As time went on, other networks, such as the National Science Foundation's NSFnet, were created and connected with the ARPAnet. Use of this interconnected network, or Internet, was originally limited to government, research, and educational purposes. The ban on commercial use of the Internet was lifted in 1991.

The growth of the Internet continues—Internet World Stats[1] reported that over 4.8 billion users, about 62% of the world's population, were using the Internet by 2020.

When the restriction on commercial use of the Internet was lifted, the stage was set for future electronic commerce. However, while businesses were no longer banned, the Internet was still text based and not easy to use. The further developments addressed this issue.

The Birth of the Web

VideoNote
Evolution of the Web

While working at CERN, a research facility in Switzerland, **Tim Berners-Lee** envisioned a means of communication for scientists by which they could easily "hyperlink" to another research paper or article and immediately view it. For this purpose, Berners-Lee created the World Wide Web. In 1991, Berners-Lee posted the code in a newsgroup and made it freely available. This version of the World Wide Web used **Hypertext Transfer Protocol** (**HTTP**) to communicate between the client computer and the web server, and it was text based, employing **Hypertext Markup Language** (**HTML**) to format the documents.

The First Graphical Browser

In 1993, Mosaic, the first graphical web browser, became available. Marc Andreessen and graduate students working at the National Center for Supercomputing Applications (NCSA) at the University of Illinois Urbana–Champaign developed Mosaic. Some individuals in this group later created another well-known web browser, Netscape Navigator, which is an ancestor of today's Mozilla Firefox.

Convergence of Technologies

By the early 1990s, personal computers with easy-to-use graphical operating systems (such as Microsoft's Windows, IBM's OS/2, and Apple's Macintosh OS) were increasingly available and affordable. Online service providers such as CompuServe, AOL, and Prodigy offered low-cost connections to the Internet. Figure 1.1 depicts this convergence of available computer hardware, easy-to-use operating systems, low-cost Internet connectivity, the HTTP protocol and HTML language, and a graphical browser that made information on the Internet much easier to access. The **World Wide Web**—the graphical user interface providing access to information stored on web servers connected to the Internet—had arrived!

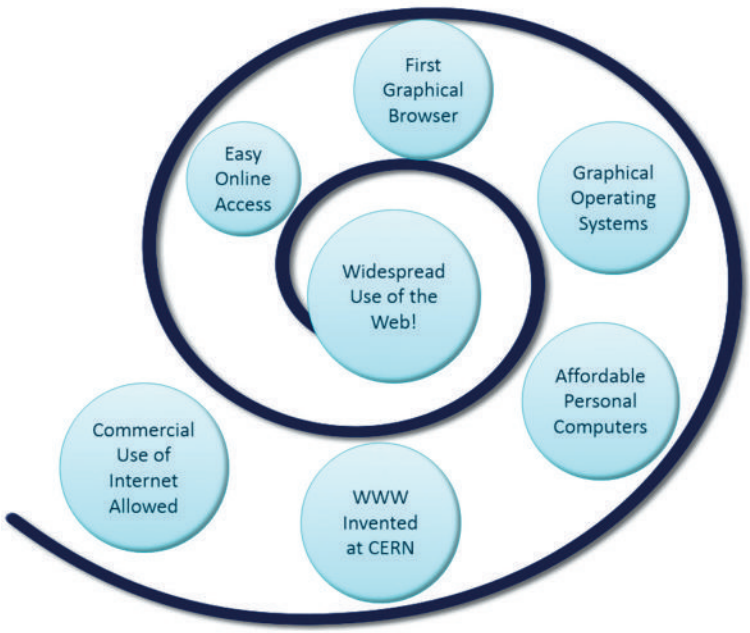


FIGURE 1.1 *Convergence of technologies.*

Web Standards and Accessibility

You are probably aware that no single person or group runs the World Wide Web. However, the **World Wide Web Consortium**, commonly referred to as the **W3C**, takes a proactive role in developing recommendations and prototype technologies related to the Web.[2] Topics that the W3C addresses include web architecture, standards for web design, and accessibility. In an effort to standardize web technologies, the W3C produces guidelines called recommendations.

W3C Recommendations

The W3C Recommendations are created in working groups with input from many major corporations involved in building web technologies. These recommendations are not rules; they are guidelines. Major software companies that build web browsers do not always follow the W3C Recommendations. This makes life challenging for web developers because not all web browsers will display a web page in exactly the same way. The good news is that there is a trend toward conforming to the W3C Recommendations in new versions of major web browsers. You'll follow W3C Recommendations as you code web pages in this book. Following the W3C Recommendations is the first step toward creating a website that is accessible.

Web Standards and Accessibility

The Web Accessibility Initiative, referred to as the WAI, is a major area of work by the W3C.[3] The Web can present barriers to individuals with visual, auditory, physical, and neurological disabilities. An **accessible website** provides accommodations that help individuals overcome these barriers. The WAI has developed the **Web Content Accessibility Guidelines** (**WCAG**) for web content developers, web authoring tool developers, and web browser developers to facilitate use of the Web by those with special needs.[4] The most recent version of WCAG is WCAG 2.1, which extends WCAG 2.0 and introduces additional success criteria including requirements for increased support of mobile device accessibility, low vision accessibility, and cognitive and learning disability accessibility.

Accessibility and the Law

The **Americans with Disabilities Act** (**ADA**) of 1990 is a federal civil rights law that prohibits discrimination against people with disabilities. The ADA requires that business, federal, and state services are accessible to individuals with disabilities.

Section 508 of the Federal Rehabilitation Act was amended in 1998 to require that U.S. government agencies give individuals with disabilities access to information technology that is comparable to the access available to others. This law requires developers creating information technology (including web pages) for use by the federal government to provide for accessibility. The GSA Government-wide IT Accessibility Initiative provides accessibility requirement resources for information technology developers.[5] As the Web and Internet technologies developed, it became necessary to revise the original Section 508 requirements. In 2017, an update to Section 508 Standards became official which requires meeting the requirements of WCAG 2.0 Level A & AA Success Criteria. This textbook focuses on WCAG 2.0 and 2.1 guidelines to provide for accessibility.

In recent years, state governments have also begun to encourage and promote web accessibility. The Illinois Information Technology Accessibility Act (IITAA) guidelines are an example of this trend.[6]

Putting It All Together: Universal Design for the Web

Universal design is a "strategy for making products, environments, operational systems, and services welcoming and usable to the most diverse range of people possible".[7] Examples of universal design are all around us. The cutouts in sidewalk curbs providing for wheelchair accessibility also benefit a person pushing a stroller or riding a Segway Personal Transporter (Figure 1.2). Doors that open automatically improve accessibility and also benefit people carrying packages. A ramp is useful for a person dragging a rolling backpack or carry-on bag.

FIGURE 1.2 *A smooth ride is a benefit of universal design.*

Awareness of universal design by web developers has been steadily increasing. Forward-thinking web developers design with accessibility in mind because it is the right thing to do. Providing access for visitors with visual, auditory, and other challenges should be an integral part of web design rather than an afterthought.

A person with visual difficulties may not be able to use graphical navigation buttons and may use a screen reader device to provide an audible description of the web page. By making a few simple changes, such as providing text descriptions for the images and perhaps providing a text navigation area at the bottom of the page, web developers can make the page accessible. Often, providing for accessibility increases the usability of the website for all visitors.

Accessible websites with alternate text for images, headings used in an organized manner, and captions or transcriptions for multimedia are more easily used not only by visitors with disabilities but also by visitors using a mobile browser. Finally, accessible websites may be more thoroughly indexed by search engines, which can be helpful in bringing new visitors to a site. As this book introduces web development and design techniques, corresponding web accessibility and usability issues are discussed.

Web Browsers and Web Servers

Network Overview

A **network** consists of two or more computers connected for the purpose of communicating and sharing resources. A diagram of a small network is shown in Figure 1.3. Common components of a network include the following:

- Server computer(s)
- Client computer(s)
- Shared devices such as printers
- Networking devices (routers, hubs, and switches) and the media that connect them

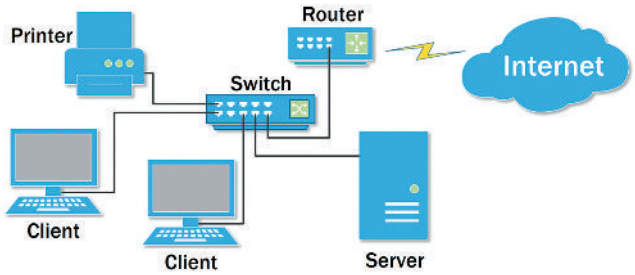


FIGURE 1.3 *Common components of a network.*

The **clients** are the computer workstations used by individuals, such as a PC on a desk. The **server** receives requests from clients for resources, such as a file. Computers used as servers are usually kept in a protected, secure area and are only accessed by network administrators. Networking devices such as hubs and switches provide network connections for computers, and routers direct information from one network to another. The **media** connecting the clients, servers, peripherals, and networking devices may consist of copper cables, fiber optic cables, or wireless technologies.

The Client/Server Model

The term **client/server** dates from the 1980s and refers to computers joined by a network. Client/server can also describe a relationship between two computer programs—the client and the server. The client requests some type of service (such as a file or database access) from the server. The server fulfills the request and transmits the results to the client over a network. While both the client and the server programs can reside on the same computer, typically they run on different computers (Figure 1.4). It is common for a server to handle requests from multiple clients.

The Internet is a great example of client/server architecture at work. Consider the following scenario: A person is at a computer using a web browser client to access the Internet. The person uses the web browser to visit a website, say http://www.google.com. The server is the web server program running on the computer with an IP address that corresponds to google.com. The web server is contacted, it locates the web page and related resources that

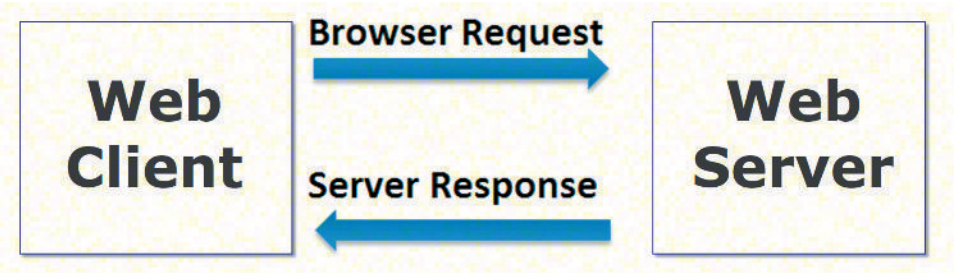


FIGURE 1.4 *Web client and web server.*

were requested, and it responds by sending them. Here's how to distinguish between web clients and web servers:

Web Client

- Connected to the Internet when needed
- Usually runs web browser (client) software such as Edge or Firefox
- Uses HTTP
- Requests web page from a web server
- Receives web page and associated files from a web server

Web Server

- Continually connected to the Internet
- Runs web server software (such as Apache or Microsoft Internet Information Server)
- Uses HTTP
- Receives a request for the web page
- Responds to the request and transmits the status code, web page, and associated files

When clients and servers exchange files, they often need to indicate the type of file that is being transferred; this is done through the use of a **Multi-Purpose Internet Mail Extensions (MIME) type**, which is a standard for the exchange of multimedia documents among different computer systems. MIME was initially intended to extend the original Internet e-mail protocol, but it is also used by HTTP. MIME provides for the exchange of seven different media types on the Internet: audio, video, image, application, message, multipart, and text. MIME also uses subtypes to further describe the data. The MIME type of a web page is text/html. MIME types of gif and jpeg images are image/gif and image/jpeg, respectively.

A web server determines the MIME type of a file before it is transmitted to the web browser. The MIME type is sent along with the document. The web browser uses the MIME type to determine how to display the document.

How does information get transferred from the web server to the web browser? Clients (such as web browsers) and servers (such as web servers) exchange information through the use of communication protocols such as HTTP, TCP, and IP, which are introduced in the next section.

Internet Protocols

Protocols are rules that describe how clients and servers communicate with each other over a network. There is no single protocol that makes the Internet and the Web work—a number of protocols with specific functions are needed.

E-Mail Protocols

Most of us take e-mail for granted, but there are two servers involved in its smooth functioning—an incoming mail server and an outgoing mail server. When you send e-mail to others, **Simple Mail Transfer Protocol** (**SMTP**) is used. When you receive e-mail, **Post Office Protocol** (POP; currently **POP3**) and **Internet Message Access Protocol** (**IMAP**) can be used.

Hypertext Transfer Protocol

Hypertext Transfer Protocol (**HTTP**) is a set of rules for exchanging files such as text, graphic images, sound, video, and other multimedia files on the Web. Web browsers and web servers usually use this protocol. When the user of a web browser requests a file by typing a website address or clicking a hyperlink, the browser builds an HTTP request and sends it to the server. The web server in the destination machine receives the request, does any necessary processing, and responds with the requested file and any associated media files.

Hypertext Transfer Protocol Secure (HTTPS)

Hypertext Transfer Protocol Secure (**HTTPS**) combines HTTP with a security and encryption protocol. Using HTTPS provides a more secure transaction because the information passed between the browser and the web server is encrypted. See Chapter 12 for more information on HTTPS.

File Transfer Protocol

File Transfer Protocol (**FTP**) is a set of rules that allows files to be exchanged between computers on the Internet. Unlike HTTP, which is used by web browsers to request web pages and their associated files in order to display a web page, FTP is used simply to move files from one computer to another. Web developers commonly use FTP to transfer web page files from their computers to web servers.

Transmission Control Protocol/Internet Protocol

Transmission Control Protocol/Internet Protocol (**TCP/IP**) has been adopted as the official communication protocol of the Internet. TCP and IP have different functions that work together to ensure reliable communication over the Internet.

TCP. The purpose of TCP is to ensure the integrity of network communication. TCP starts by breaking files and messages into individual units called **packets**. These packets (see Figure 1.5) contain information such as the destination, source, sequence number, and checksum values used to verify the integrity of the data.

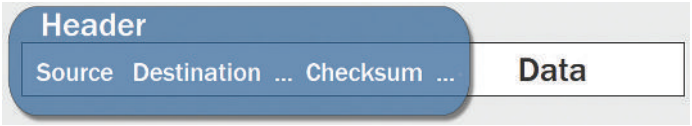


FIGURE 1.5 *TCP packet.*

TCP is used together with IP to transmit files efficiently over the Internet. IP takes over after TCP creates the packets, using IP addressing to send each packet over the Internet using the best path at the particular time. When the destination address is reached, TCP verifies the integrity of each packet using the checksum, requests resend if a packet is damaged, and reassembles the file or message from the multiple packets.

IP. Working in harmony with TCP, IP is a set of rules that controls how data are sent between computers on the Internet. IP routes a packet to the correct destination address. Once sent, the packet gets successively forwarded to the next closest router (a hardware device designed to move network traffic) until it reaches its destination.

IP Addresses

Each device connected to the Internet has a unique numeric **IP address**. These addresses consist of a set of four groups of numbers called octets. The current widely used version of IP, **IPv4**, uses 32-bit (binary digit) addressing. This results in a decimal number in the format of xxx.xxx.xxx.xxx, where each xxx is a value from 0 to 255. Theoretically, this system allows for at most 4 billion possible IP addresses (although many potential addresses are reserved for special uses). However, even this many addresses will not be enough to meet the needs of all of the devices expected to be connected to the Internet in upcoming years.

IPv6, Internet Protocol Version 6, intended to replace IPv4, was designed as an evolutionary set of improvements and is backwardly compatible with IPv4. Service providers and Internet users can update to IPv6 independently without having to coordinate with each other. IPv6 provides for more Internet addresses because the IP address is lengthened from 32 bits to 128 bits. This means that there are potentially 2^{128} unique IP addresses possible, or 340, 282, 366, 920, 938, 463, 463, 347, 607, 431, 768, 211, 456.

The IP address of a device may correspond to a domain name. The **Domain Name System (DNS)** associates these IP addresses with the text-based URLs and domain names you type into a web browser address box (more on this later). For example, at the time this was written an IP address for Google was 216.58.194.46.

You can enter this number in the address text box in a web browser (as shown in Figure 1.6), press Enter, and the Google home page will be displayed. Of course, it's much easier to type "google.com," which is why domain names such as google.com were created in the first place! Since long strings of numbers are difficult for humans to remember, the DNS was introduced as a way to associate text-based names with numeric IP addresses.

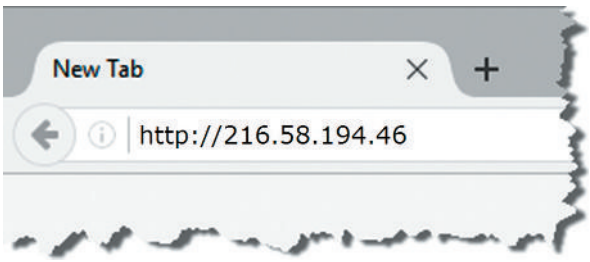


FIGURE 1.6 *Entering an IP address in a web browser.*

Uniform Resource Identifiers and Domain Names

URIs and URLs

A **Uniform Resource Identifier (URI)** identifies a resource on the Internet. A **Uniform Resource Locator (URL)** is a type of URI that represents the network location of a resource such as a web page, a graphic file, or an MP3 file. The URL consists of the protocol, the domain name, and the hierarchical location of the file on the web server.

The URL http://www.webdevbasics.net, as shown in Figure 1.7, denotes the use of HTTP protocol and the web server named www at the domain name of webdevbasics.net. In this case, the root file (which is usually index.html or index.htm) of the 6e directory will be displayed.

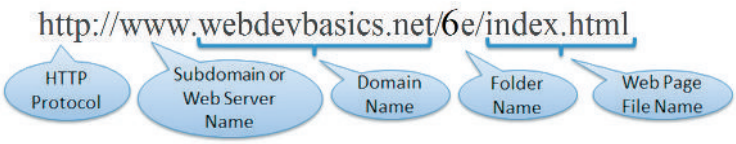


FIGURE 1.7
URL describing a file within a folder.

Domain Names

A **domain name** locates an organization or other entity on the Internet. A domain name is associated with a unique numeric IP address assigned to a device. This association is stored in the DNS database.

Let's consider the domain name www.google.com. The .com is the top-level domain name. The portion google.com is the domain name that is registered to Google and is considered a second-level domain name. The www is the name of the web server (sometimes called a **host**) at the google.com domain.

A **subdomain** can be configured to house a separate website located at the same domain. For example, Google's Gmail can be accessed by using the subdomain "gmail" in the domain name (gmail.google.com). Google Maps can be accessed at maps.google.com and Google News Search is available at news.google.com. The combination of host or subdomain, second-level domain, and top-level domain name (such as www.google.com or gmail.google.com) is called a **Fully Qualified Domain Name (FQDN)**.

Top-Level Domain Names

A **top-level domain** (**TLD**) identifies the rightmost part of the domain name. A TLD is either a **generic top-level domain** (**gTLD**), such as com for commercial, or a country-code top-level domain, such as fr for France. The Internet Assigned Numbers Authority (IANA) website has a complete list of TLDs.[8]

Generic Top-Level Domain (gTLD) Names

The Internet Corporation for Assigned Names and Numbers (ICANN) administers gTLDs.[9] The .com, .org, and .net gTLD designations are currently used on the honor system, which means that an individual who owns a shoe store (not related to networking) can register shoes.net. Table 1.1 shows a collection of gTLDs and their intended use.

TABLE 1.1 *Generic Top-Level Domains*

gTLD	Intended for Use By
.aero	Air-transport industry
.bank	Banks and other financial institutions
.biz	Businesses
.club	Clubs and groups with common interests
.com	Commercial entities
.coop	Cooperative
.edu	Restricted to accredited degree-granting institutions of higher education
.gov	Restricted to government use
.info	Unrestricted use
.int	International organization (rarely used)
.jobs	Human resource management community
.mil	Restricted to military use
.museum	Museums
.name	Individuals
.net	Entities associated with network support of the Internet, usually Internet service providers or telecommunication companies
.org	Nonprofit entities
.pro	Accountants, physicians, and lawyers
.tel	Contact information for individuals and businesses
.travel	Travel industry

Expect the number and variety of gTLD names to increase. As of 2020, there were over 1,900 gTLD names applications submitted with over 1200 new gTLD names approved.[10] The new gTLDs include place names (.quebec, .vegas, .moscow, and .amsterdam), retail terms (.blackfriday, .sale, .shop, and .shopping), financial terms (.cash, .trade, .loans, and .mortgage), technology terms (.systems, .technology, .digital, and .app), company names (.apple, .guardian, .toshiba, and .volkswagen), professions (.doctor, .accountant, .dentist, and .attorney), sports (.baseball, .football, and .hocky), and even whimsical fun terms (.ninja, .buzz, and .cool). ICANN has set a schedule to periodically launch new gTLDs.

Country-Code Top-Level Domain Names

Two-character country codes have also been assigned as TLD names. The country-code TLD (ccTLD) names were originally intended to designate the geographical location of the individual or organization that registered the name. The IANA website has a complete list of country-code TLDs.[11]

Domain names with country codes are often used for municipalities, schools, and community colleges in the United States. For example, the domain name www.harper.cc.il.us denotes, from right to left, the United States, Illinois, community college, Harper, and the web server named "www" as the website for Harper College in Illinois.

Although the original intent of country-code TLD names were to designate websites in a particular geographical location, some country codes have become available for commercial use. Examples of nongeographical use of ccTLDs include domain names such as mediaqueri.es, livestre.am, webteacher.ws, youtu.be, who.is, amc.tv, and bit.ly. Table 1.2 lists some popular country codes used on the Web.

TABLE 1.2 *Country-Code TLDs*

ccTLD	Country
.am	Armenia
.au	Australia
.be	Belgium
.dj	Dijbouti
.ch	Switzerland
.es	Spain
.in	India
.is	Iceland
.it	Italy
.lv	Latvia
.ly	Libya
.me	Montenegro
.tm	Turkmenistan
.tv	Tuvalu
.us	United States
.ws	Samoa

Domain Name System (DNS)

The DNS associates domain names with IP addresses. As shown in Figure 1.8, the following happens each time a new URL is typed into a web browser:

1. The DNS is accessed.
2. The corresponding IP address is obtained and returned to the web browser.
3. The web browser sends an HTTP request to the destination computer with the corresponding IP address.
4. The HTTP request is received by the web server.
5. The necessary files are located and sent by HTTP responses to the web browser.
6. The web browser renders and displays the web page and associated files.

We all get impatient sometimes when we need to view a web page. The next time you wonder why it's taking so long to display a web page, consider all of the processing that goes on behind the scenes before the web browser receives the files needed to display the web page.

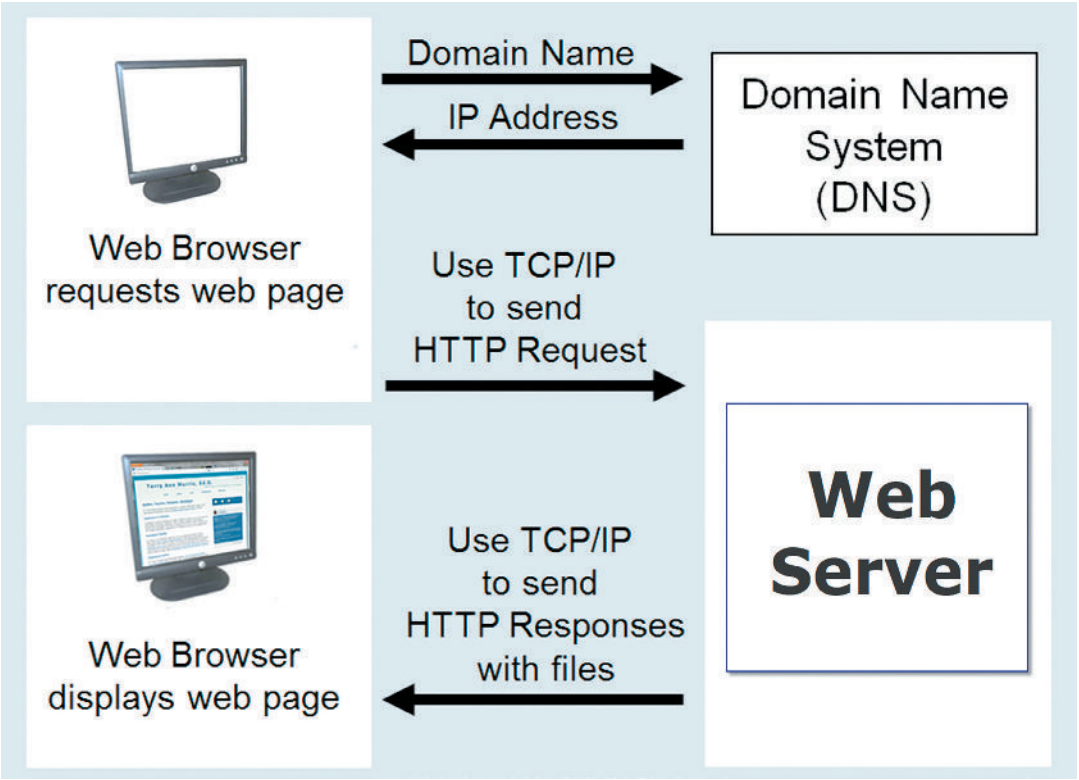


FIGURE 1.8 *Accessing a web page.*

Information on the Web

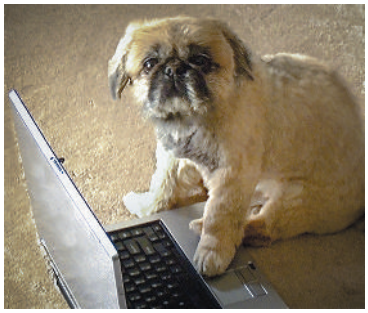


FIGURE 1.9 *Who really updated that web page you are viewing?*

These days anyone can publish just about anything on the Web. In this section, we'll explore how you can tell if the information you've found is reliable and also how you can use that information. There are many websites—but which ones are reliable sources of information? When visiting websites to find information, it is important not to take everything at face value (Figure 1.9).

- **Is the organization credible?**

Anyone can post anything on the Web! Choose your information sources wisely.

First, evaluate the credibility of the website itself. Does it have its own domain name, such as http://mywebsite.com, or is it a free website consisting of just a folder of files hosted on a free web hosting site (such as weebly.com, awardspace.com, or 000webhost.com)?

The URL of a site hosted on a free web server usually includes part of the free web host's domain name. Information obtained from a website that has its own domain name will usually (but not always) be more reliable than information obtained from a free website.

Evaluate the type of domain name—is it a nonprofit organization (.org), a business (.com or .biz), or an educational institution (.edu)? Businesses may provide information in a way that gives them an advantage, so be careful. Nonprofit organizations or schools will sometimes treat a subject more objectively.

- **How recent is the information?**

Another item to look at is the date the web page was created or last updated. Although some information is timeless, very often a web page that has not been updated for several years is outdated and may not be the best source of information.

- **Are there links to additional resources?**

Hyperlinks indicate websites with supporting or additional information that can be helpful to you in your research as you explore a topic. Look for these types of hyperlinks to aid you in your studies.

- **Is it Wikipedia?**

Wikipedia[12] is a good place to begin research, but don't accept what you read there for fact, and avoid using Wikipedia as a resource for academic assignments. Why? Well, except for a few protected topics, anyone can update Wikipedia with anything!

Usually it all gets sorted out eventually—but be aware that the information you read may not be valid.

Feel free to use Wikipedia to begin exploring a topic, but scroll down to the bottom of the Wikipedia web page and look for "References"—and explore those websites and others that you may find. As you gather information on these sites, also consider the other criteria: credibility, domain name, timeliness, and links to additional resources.

Ethical Use of Information on the Web

The wonderful technology called the World Wide Web provides us with information, graphics, music, and video—all virtually free (after you pay your Internet service provider, of course). Let's consider the following issues relating to the ethical use of this information:

- Is it acceptable to copy someone's graphic to use on your website?
- Is it acceptable to copy someone's music or video to use on your website?
- Is it acceptable to copy someone's website design to use on your site or on a client's site?
- Is it acceptable to copy an essay that appears on a web page and use it or parts of it as your writing?
- Is it acceptable to insult someone on your website or link to another website in a derogatory manner?

The answer to all these questions is no. Using a person's graphic, music, or video without permission is the same as stealing it. In fact, if you link to it, you are actually using up some of his or her bandwidth and may be costing him or her money. Copying the website design of another person or company is also a form of stealing. Any text or graphic on a website is automatically copyrighted in the United States whether or not a copyright symbol appears on the site. Insulting a person or company on your website or linking to another website in a derogatory manner could be considered a form of defamation.

Issues related to intellectual property, copyright, and freedom of speech are regularly discussed and decided in courts of law. Good web etiquette requires that you ask permission before using others' work, give credit for what you use as a student ("fair use" in the U.S. copyright law), and exercise your freedom of speech in a manner that is not harmful to others. The **World Intellectual Property Organization (WIPO)**[13] is dedicated to protecting intellectual property rights internationally.

What if you'd like to retain ownership but make it easy for others to use or adapt your work? **Creative Commons** is a nonprofit organization that provides free services that allow authors and artists to register a type of a copyright license called a Creative Commons license.[14] There are several licenses to choose from, depending on the rights you wish to grant. The Creative Commons license informs others exactly what they can and cannot do with your creative work.

HTML Overview

Markup languages consist of sets of directions that tell web browser software (and other user agent software that retrieves and renders web content) how to display and manage a web document. These directions are usually called tags and perform functions such as displaying graphics, formatting text, and referencing hyperlinks.

The World Wide Web is composed of files containing Hypertext Markup Language (HTML) and other markup languages that describe web pages. Tim Berners-Lee developed HTML using **Standard Generalized Markup Language (SGML)**. SGML prescribes a standard format for embedding descriptive markup within a document and for describing the structure of a document. SGML is not in itself a document language, but rather a description of how to specify one and create a document type definition. The W3C sets the standards for HTML and its related languages. HTML (like the Web itself) is in a constant state of change.

What Is HTML?

HTML (Hypertext Markup Language) is the set of markup symbols or codes placed in a file that is intended for display on a web page. These markup symbols and codes identify structural elements such as paragraphs, headings, and lists. HTML can also be used to place media (such as graphics, video, and audio) on a web page and describe fill-in forms. The web browser interprets the markup code and renders the page. HTML permits the platform-independent display of information across a network. No matter what type of computer a web page was created on, any web browser running on any operating system can display the page.

Each individual markup code is referred to as an **element** or a **tag**. Each tag has a purpose. Tags are enclosed in angle brackets, the < and > symbols. Most tags come in pairs: an opening tag and a closing tag. These tags act as containers and are sometimes referred to as container tags. For example, when an HTML document is displayed by a web browser, the text that appears between the `<title>` and `</title>` tags would be displayed in the title bar on the browser window.

Some tags are used alone and are not part of a pair. For example, a `<hr>` tag that displays a horizontal line on a web page is a stand-alone or self-contained tag and does not have a closing tag. You will become familiar with these as you use them. Most tags can be modified with **attributes** that further describe their purpose.

What Is XML?

XML (**eXtensible Markup Language**) was developed by the W3C to create common information formats and share the format and the information on the Web. It is a text-based syntax designed to describe, deliver, and exchange structured information, such as RSS (Rich Site Summary) web feeds. XML is not intended to replace HTML, but to extend the power of HTML by separating data from presentation. Using XML, developers can create any tags they need to describe their information.

What Is XHTML?

eXtensible Hypertext Markup Language (**XHTML**) uses the tags and attributes of HTML4 along with the more rigorous syntax of XML. XHTML was used on the Web for over a decade, and you'll find many web pages coded with this markup language. At one point, the W3C was working on a new version of XHTML, called XHTML 2.0. However, the W3C stopped development of XHTML 2.0 because it was not backward compatible with HTML4. Instead, the W3C decided to move forward with HTML5.

HTML5—The Newest Version of HTML

HTML5 is the successor to HTML and replaces XHTML. HTML5 incorporates features of both HTML and XHTML, adds new elements, provides new functionality such as form edits and native video, and is designed to be backward compatible.

The W3C approved HTML5 for final Recommendation status in late 2014 and continued to update the language. HTML 5.3 reached in working draft status in 2018.[15] A change in the procedure to update HTML5 occurred in 2019, when the W3C and the WHATWG (Web Hypertext Application Technology Working Group) agreed to collaborate on the development of HTML specifications.[16] The **WHATWG** is an open group founded by individuals working at many leading technology organizations including Apple, the Mozilla Foundation, and Opera Software. The WHATWG plans to prepare a Review Draft[17] of the HTML Living Standard every six months. Recent versions of popular browsers offer good support for HTML5, which you'll learn to use as you work through this textbook. HTML5 documentation is available at https://html.spec.whatwg.org.

Under the Hood of a Web Page

FIGURE 1.10 *It's what is under the hood that matters.*

You already know that the HTML markup language tells web browsers how to display information on a web page. Let's take a closer look at what's "under the hood" (Figure 1.10) of every web page you create.

Document Type Definition

Because multiple versions and types of HTML and XHTML exist, the W3C recommends identifying the markup language used in a web page document with a **Document Type Definition (DTD)**. The DTD identifies the version of HTML in the document. Web browsers and HTML code validators use the information in the DTD when processing the web page. The DTD statement, commonly called a **DOCTYPE** statement, is the first line of a web page document. The DTD for HTML5 is:

```
<!DOCTYPE html>
```

Web Page Template

Every single web page you create will include the html, head, title, meta, and body elements. We will follow the coding style to use lowercase letters and place quotes around attribute values. A basic HTML5 web page template (found in the student files at chapter1/template.html) is:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Page Title Goes Here</title>
<meta charset="utf-8">
</head>
<body>
. . . body text and more HTML tags go here . . .
</body>
</html>
```

With the exception of the specific page title, the first seven lines will usually be the same on every web page that you create. Review the code above and notice the DTD statement has its own formatting, but that the HTML tags all use lowercase letters. Next, let's explore the purpose of the html, head, title, meta, and body elements.

HTML Element

The purpose of the **html element** is to indicate that the document is HTML formatted. The html element tells the web browser how to interpret the document. The opening `<html>` tag

is placed on a line below the DTD. The closing </html> tag indicates the end of the web page and is placed after all other HTML elements in the document.

The html element also needs to indicate the spoken language, such as English, of the text in the document. This additional information is added to the <html> tag in the form of an **attribute**, which modifies or further describes the function of an element. The **lang attribute** specifies the spoken language of the document. For example, lang="en" indicates the English language. Search engines and screen readers may access this attribute.

The html element contains the two sections of a web page: the head and the body. The **head section** contains information that describes the web page document. The **body section** contains the actual tags, text, images, and other objects that are displayed by the web browser as a web page.

Head Section

Elements that are located in the **head section** include the title of the web page, meta tags that describe the document (such as the character encoding used and information that may be accessed by search engines), and references to scripts and styles. Many of these do not show directly on the web page.

Head Element. The **head element** contains the head section, which begins with the <head> tag and ends with the </head> tag. You'll always code at least two other elements in the head section: a title element and a meta element.

Title Element. The first element in the head section, the **title element**, configures the text that will appear in the title bar of the browser window. The text between the <title> and </title> tags is called the title of the web page. This title text is accessed when web pages are bookmarked and printed. Popular search engines, such as Google, use the title text to help determine keyword relevance and even display the title text on the results page of a search. A descriptive title that includes the website or organization name is a crucial component for establishing a brand or presence on the Web.

Meta Element. The **meta element** describes a characteristic of a web page, such as the character encoding. **Character encoding** is the internal representation of letters, numbers, and symbols in a file, such as a web page or other file, that is stored on a computer and may be transmitted over the Internet. There are many different character-encoding sets. A form of Unicode called utf-8 character encoding is typically used for web pages.[18] The meta tag is not used as a pair of opening and closing tags. It is a stand-alone self-contained tag (referred to as a **void** element in HTML5). The meta tag uses the **charset attribute** to indicate the character encoding. An example meta tag is:

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

Body Section

The **body section** contains text and elements that display directly on the web page within the browser window, also referred to as the browser viewport. The purpose of the body section is to configure the contents of the web page.

Body Element. The **body element** contains the body section, which begins with the <body> tag and ends with the </body> tag. You will spend most of your time writing code in the body of a web page. Text and elements typed between the opening and closing body tags will be displayed on the web page in the browser viewport.

Your First Web Page

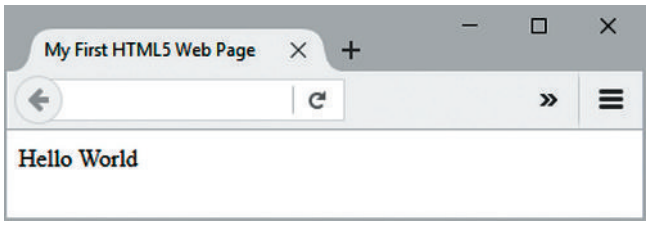
VideoNote
Your First Web Page

No special software is needed to create a web page document—all you need is a text editor. The Notepad text editor is included with Microsoft Windows. TextEdit is distributed with the Mac OS operating system. An alternative to using a simple text editor or word processor is to use a commercial web authoring tool, such as Adobe Dreamweaver. There are also many free or shareware editors available, including Notepad++, Brackets, Visual Studio Code, and BBEdit. Regardless of the tool you use, having a solid foundation in HTML will be useful. The examples in this book use Notepad.

Hands-On Practice 1.1

Now that you're familiar with the basic elements used on every web page, it's your turn to create your first web page, shown in Figure 1.11.

FIGURE 1.11
Your first web page.



Create a Folder

You'll find it helpful to create folders to organize your files as you develop web pages and create your own websites. Use your operating system to create a new folder named chapter1 on your hard drive or on a portable flash drive.

To create a new folder on a Mac:

1. In the Finder, go to the location where you'd like to create the new folder.
2. Choose File > New Folder. An untitled folder is created.
3. To rename the folder with a new name: select the folder and click on the current name. Type a name for the folder and press the Return key.

To create a new folder with Windows:

1. Right-click on the Start Button and select File Explorer. Then, navigate to the location where you'd like to create the new folder, such as Documents, your C: drive, or an external USB drive.
2. Select the Home tab. Select New folder.
3. To rename the new folder: right-click on it, select Rename from the context-sensitive menu, type the new name, and press the Enter key.

Now, you are ready to create your first web page. Launch Notepad or another text editor. Type in the following code.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>My First HTML5 Web Page</title>
<meta charset="utf-8">
</head>
<body>
Hello World
</body>
</html>
```

Notice that the first line in the file contains the DTD. The HTML code begins with an opening `<html>` tag and ends with a closing `</html>` tag. The purpose of these tags is to indicate that the content between the tags makes up a web page.

The head section is delimited by `<head>` and `</head>` tags and contains a pair of title tags with the words "My First HTML5 Web Page" in between along with a `<meta>` tag to indicate the character encoding.

The body section is delimited by `<body>` and `</body>` tags. The words "Hello World" are typed on a line between the body tags. See Figure 1.12 for a screenshot of the code as it would appear in Notepad. You have just created the source code for a web page document.

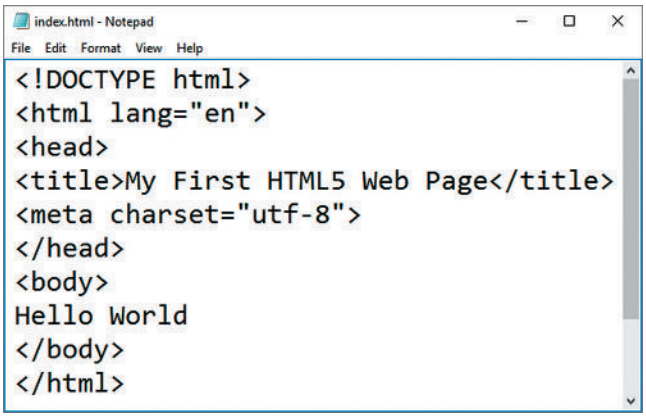


FIGURE 1.12 *Your web page source code displayed in Notepad.*

Do I have to start each tag on its own line?

No, you are not required to start each tag on a separate line. A web browser can display a page even if all the tags follow each other on one line with no spaces. Humans, however, find it easier to write and read web page code if line breaks and indentation are used.

Save Your File

Web pages use either an .htm or .html file extension. A common file name for the home page of a website is index.html or index.htm. The web pages in this book use the .html file extension.

You will save your file with the name of index.html.

1. Display your file in Notepad or another text editor.
2. Select File from the menu bar, and then select Save As.
3. The Save As dialog box appears. Using Figure 1.13 as an example, type the file name.
4. Click the Save button.

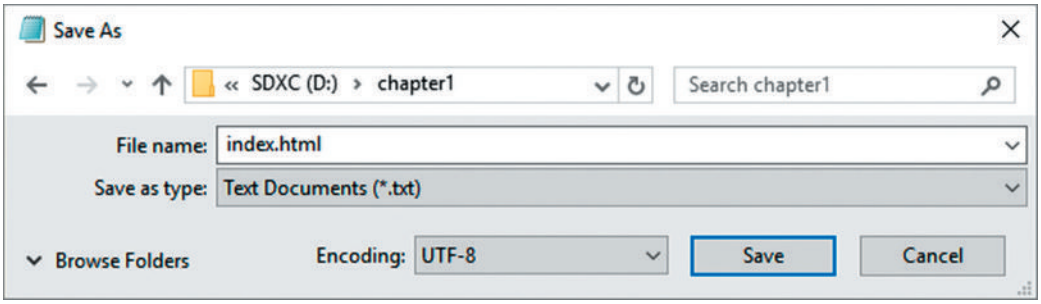


FIGURE 1.13 *Save and name your file.*

Sample solutions for Hands-On Practice exercises are available in the student files. If you would like, compare your work with the solution (chapter1/index.html) before you test your page.

FAQ

Why does my file have a .txt file extension?

In some older versions of Windows, Notepad will automatically append a .txt file extension. If this happens, rename your file index.html.

FAQ

Why should I create a folder, why not just use the desktop?

Folders will help you to organize your work. If you just used the desktop, it would quickly become cluttered and disorganized. It's also important to know that websites are organized on web servers within folders. By starting to use folders right away to organize related web pages, you are on your way to becoming a successful web designer.

Test Your Page

There are two ways to test your page:

1. In Windows Explorer (Windows) or the Finder (Mac), navigate to your index.html file. Double-click index.html. The default web browser will launch and will display your index.html page.
2. Launch a web browser. Select File > Open, and navigate to your index.html file. Double-click index.html and click OK. The browser will display your index.html page.

If you are using Microsoft Edge, your page should look similar to the one shown in Figure 1.14. A display of the page using Firefox is shown in Figure 1.11. Notice how the title text, "My First HTML5 Web Page" displays in the tab and the title bar of the browser window. Some search engines use the text surrounded by the `<title>` and `</title>` tags to help determine relevance of keyword searches, so make certain that your pages contain descriptive titles. The `<title>` tag is also used when viewers bookmark your page or add it to their Favorites. An engaging and descriptive page title may entice a visitor to revisit your page. If your web page is for a company or an organization, it's a good idea to include the name of the company or organization in the title.

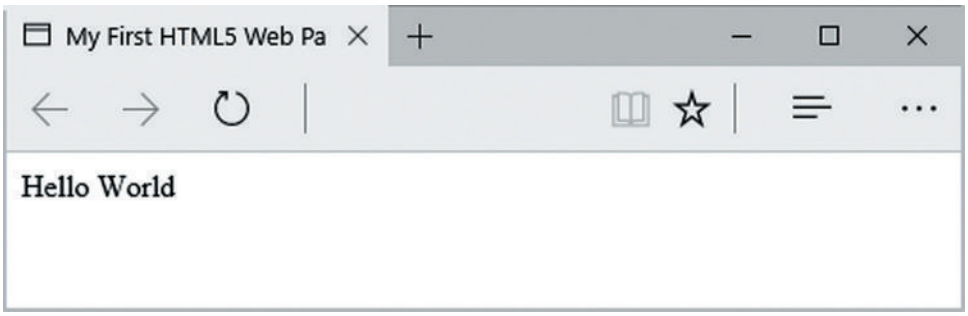


FIGURE 1.14 *Web page displayed by Microsoft Edge. Courtesy of Microsoft Corporation.*

FAQ **When I viewed my page in a web browser, the file name was index.html.html—why did this happen?**

This usually happens when your operating system is configured to hide file extension names. You will correct the file name, using one of the following two methods:

- Use the operating system to rename the file from "index.html.html" to "index.html".

  OR

- Open the index.html.html file in your text editor and save it with the name "index.html".

It's a good idea to change the settings in your operating system to show file extension names. Follow the steps at the resources below to show file extension names:

- *Windows:* http://www.file-extensions.org/article/show-and-hide-file-extensions-in-windows-10
- *Mac:* http://www.fileinfo.com/help/mac_show_extensions

CHAPTER 1

Review and Apply

Review Questions

Multiple Choice. Choose the best answer for each question.

1. Select the item below that indicates the top-level domain name for the URL http://www.mozilla.com.
   a. http
   b. com
   c. mozilla
   d. www
2. What is a unique text-based Internet address corresponding to a computer's unique numeric IP address called?
   a. IP address
   b. domain name
   c. URL
   d. user name
3. The purpose of _____ is to ensure the integrity of the communication.
   a. IP
   b. TCP
   c. HTTP
   d. FTP
4. Choose the true statement:
   a. The title of the web page is displayed by the meta element.
   b. Information about the web page is contained in the body section.
   c. The content that displays in the browser viewport is contained in the head section.
   d. The content that displays in the browser viewport is contained in the body section.

True or False. Choose the best answer, true or false, for each question.

5. _____ Markup languages contain sets of directions that tell web browser software how to display and manage a web document.
6. _____ A domain name that ends in .net indicates that the website is for a networking company.
7. _____ The World Wide Web was developed to allow companies to conduct e-commerce over the Internet.

Fill in the Blanks.

8. _____ is the set of markup symbols or codes placed in a file intended for display on a web browser.
9. Web page documents typically use the _____ or _____ file extension.
10. The home page of a website is typically named _____ or _____.

Review Answers

1. b 2. b 3. b 4. d 5. True 6. False 7. False 8. HTML
9. .htm, .html 10. index.htm, index.html

Hands-On Exercises

1. A blog, or web log, is a journal that is available on the Web—it's a frequently updated page with a chronological list of ideas and links. Blog topics range from political journals to technical information to personal diaries. It's up to the person, called a blogger, who creates and maintains the blog.

   Create a blog to document your learning experiences as you study web design. Visit one of the many sites that offers free blogs, such as https://blogger.com, https://tumblr.com, or https://www.wordpress.com. Follow their instructions to establish your blog. Your blog could be a place to note websites that you find useful or interesting. You might report on websites that contain useful web design resources. You might describe sites that have interesting features, such as compelling graphics or easy-to-use navigation. Write a few sentences about the site that you find intriguing. After you begin to develop your sites, you could include the URLs and reasons for your design decisions. Share this blog with your fellow students and friends.

2. Twitter (https://www.twitter.com) is a social networking website for microblogging, or frequently communicating with a brief message (280 characters or less) called a tweet. Twitter users (referred to as twitterers) tweet to update a network of friends and followers about their daily activities, observations, and information related to topics of interest. A hashtag (the # symbol) can be placed in front of a word or term within a tweet to categorize the topic, such as typing the hashtag #SXSWi in all tweets about the SXSW Interactive Conference for the web design industry. The use of a hashtag makes it easy to search for tweets about a category or an event in Twitter.

   If you don't already use Twitter, sign up for a free account at https://www.twitter.com. Use your Twitter account to share information about websites that you find useful or interesting. Post at least three tweets. You might tweet about websites that contain useful web design resources. You might describe sites that have interesting features, such as compelling graphics or easy-to-use navigation. After you begin to develop your own websites, you can tweet about them, too!

   Your instructor may direct you to include a distinctive hashtag (for example, something like #CIS110) in your tweets that are related to your web design studies. Searching Twitter for the specified hashtag will make it easy to collect all the tweets posted by the students in your class.

Web Research

The World Wide Web Consortium creates standards for the Web. Visit its site at https://www.w3c.org and then answer the following questions:

a. How did the W3C get started?
b. Who can join the W3C? What does it cost to join?
c. The W3C develops standards for a variety of technologies. Explore the W3C's website, locate a technology that interests you, click its link, and read several of the associated pages. List three facts or issues you discover.

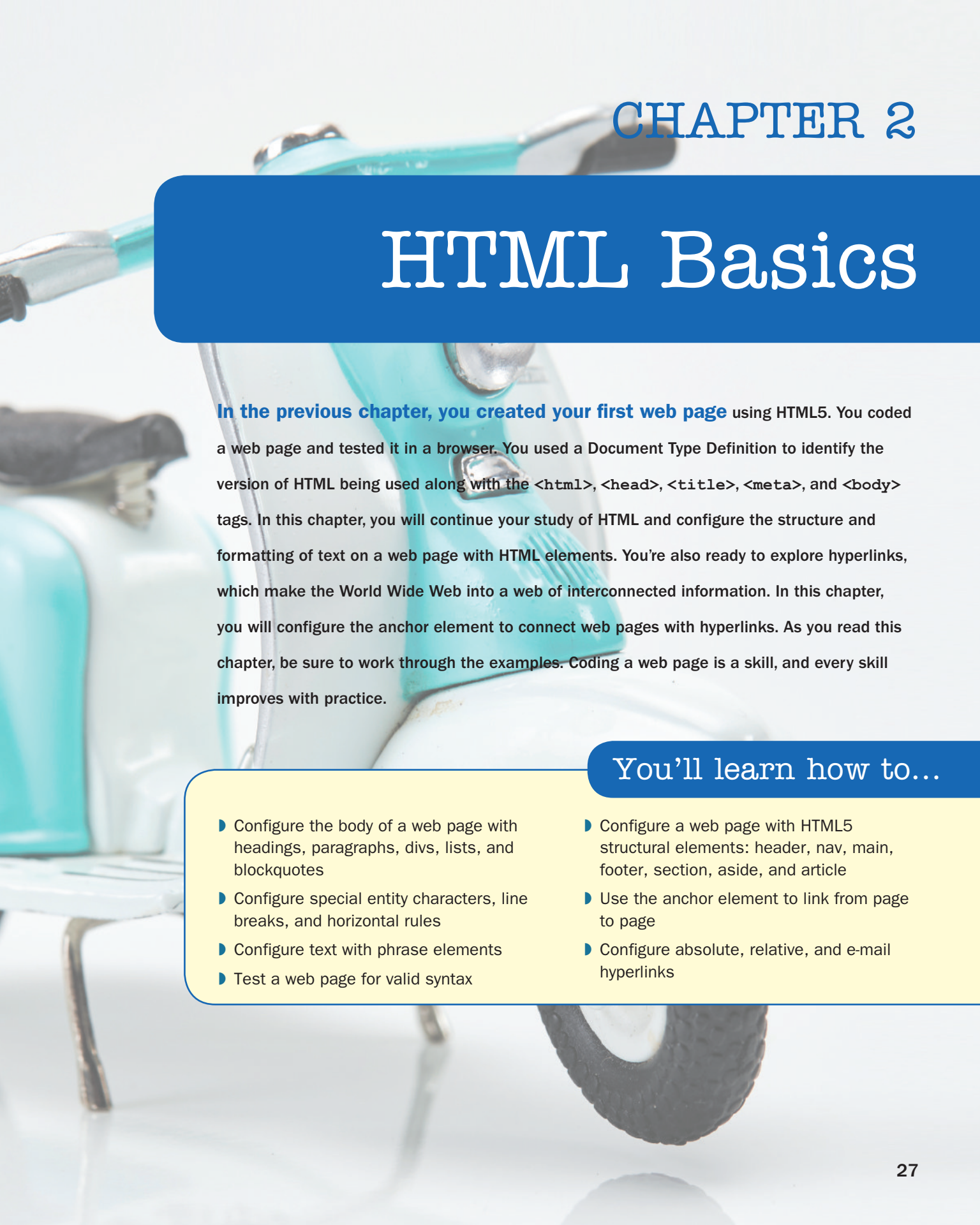
Focus on Web Design

Visit a website referenced in this chapter that interests you. Print the home page or one other pertinent page from the site. Write a one-page summary and your reaction to the site. Address the following topics:

a. What is the purpose of the site?
b. Who is the intended audience?
c. Do you think that the site reaches its intended audience? Why or why not?
d. Is the site useful to you? Why or why not?
e. List one interesting fact or issue that this site addresses.
f. Would you encourage others to visit this site?
g. How could this site be improved?

Endnotes

1. "World Internet Users Statistics and 2020 World Population Stats." *Internet World Stats*, 3 Mar. 2020, www.internetworldstats.com/stats.htm.
2. *W3C*, www.w3.org/.
3. w3c_wai. "Home Web Accessibility Initiative (WAI) W3C." *Web Accessibility Initiative (WAI)*, www.w3.org/WAI/.
4. w3c_wai. "WCAG 2.1 at a Glance." *Web Accessibility Initiative (WAI)*, 2019, www.w3.org/WAI/standards-guidelines/wcag/glance/.
5. "GSA Government-Wide IT Accessibility Program." *Section508.Gov | GSA Government-Wide IT Accessibility Program*, www.section508.gov/.
6. "IIITAA 2.0 Standards." *Illinois Department of Human Services*, www.dhs.state.il.us/page.aspx?item=96985.
7. "Universal Design." *Universal Design–Office of Disability Employment Policy–United States Department of Labor*, www.dol.gov/odep/topics/UniversalDesign.htm.
8. "Root Zone Database." *IANA*, www.iana.org/domains/root/db.
9. "By the Numbers & Survey Report." *ICANN,* www.icann.org/.
10. "Program Statistics." *Program Statistics | ICANN New GTLDs*, newgtlds.icann.org/en/program-status/statistics.
11. "Root Zone Database." *IANA*, www.iana.org/domains/root/db.
12. "Main Page." *Wikipedia*, Wikimedia Foundation, 5 Feb. 2020, www.wikipedia.org/.
13. *Inside WIPO*, www.wipo.int/about-wipo/en/.
14. "When We Share, Everyone Wins." *Creative Commons*, creativecommons.org/.
15. "HTML 5.3." Edited by Patricia Aas, et al., *W3C*, 18 Oct. 2018, www.w3.org/TR/html53/.
16. Jaffe, Jeff, et al. "W3C And WHATWG to Work Together to Advance the Open Web Platform." *W3C*, 28 May 2019, www.w3.org/blog/2019/05/w3c-and-whatwg-to-work- together-to-advance-the-open-web-platform/.
17. "HTML." *HTML Standard*, WHATWG, 1 Apr. 2020, html.spec.whatwg.org/.
18. "UTF-8, UTF-16, UTF-32 & BOM." *[Unicode]*, The Unicode Consortium, 24 Jan. 2020, www.unicode.org/faq/utf_bom.html#UTF8.



CHAPTER 2

HTML Basics

In the previous chapter, you created your first web page using HTML5. You coded a web page and tested it in a browser. You used a Document Type Definition to identify the version of HTML being used along with the `<html>`, `<head>`, `<title>`, `<meta>`, and `<body>` tags. In this chapter, you will continue your study of HTML and configure the structure and formatting of text on a web page with HTML elements. You're also ready to explore hyperlinks, which make the World Wide Web into a web of interconnected information. In this chapter, you will configure the anchor element to connect web pages with hyperlinks. As you read this chapter, be sure to work through the examples. Coding a web page is a skill, and every skill improves with practice.

You'll learn how to...

- Configure the body of a web page with headings, paragraphs, divs, lists, and blockquotes
- Configure special entity characters, line breaks, and horizontal rules
- Configure text with phrase elements
- Test a web page for valid syntax
- Configure a web page with HTML5 structural elements: header, nav, main, footer, section, aside, and article
- Use the anchor element to link from page to page
- Configure absolute, relative, and e-mail hyperlinks

Heading Element

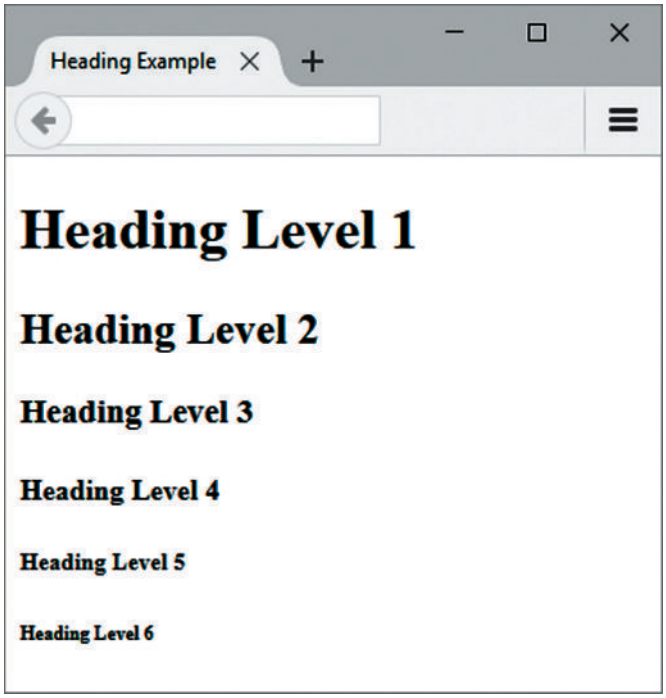


FIGURE 2.1 *Sample heading.html.*

Heading elements are organized into six levels: h1 through h6. The text within a heading element is rendered as a "block" of text by the browser (referred to as **block display**) and appears with empty space (sometimes called "white space" or "negative space") above and below. The size of the text is largest for **<h1>** (called a heading 1 tag) and smallest for **<h6>** (called a heading 6 tag). Depending on the font being used, the text within `<h4>`, `<h5>`, and `<h6>` tags may be displayed smaller than the default text size. All text within heading tags is displayed with bold font weight.

Figure 2.1 shows a web page document with six levels of headings.

Hands-On Practice 2.1

To create the web page shown in Figure 2.1, launch a text editor and open the template.html file from the chapter2 folder in the student files. Modify the title element and add heading tags to the body section as indicated by the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Heading Example</title>
<meta charset="utf-8">
</head>
<body>
<h1>Heading Level 1</h1>
<h2>Heading Level 2</h2>
<h3>Heading Level 3</h3>
<h4>Heading Level 4</h4>
<h5>Heading Level 5</h5>
<h6>Heading Level 6</h6>
</body>
</html>
```

Save the document as heading.html on your hard drive or flash drive. Launch a browser such as Edge or Firefox to test your page. It should look similar to the page shown in Figure 2.1. You can compare your work with the solution found in the student files (chapter2/heading.html).

FAQ

Why doesn't the heading tag go in the head section?

It's common for students to try to code the heading tags in the head section of the document, but someone doing this won't be happy with the way the browser displays the web page. Even though "heading tag" and "head section" sound similar, always code heading tags in the body section of the web page document.

Accessibility and Headings

Focus on ACCESSIBILITY

Heading tags can help to make your pages more accessible and usable. It is good coding practice to use heading tags to outline the structure of your web page content. To indicate areas within a page hierarchically, code heading tags numerically as appropriate (h1, h2, h3, and so on), and include page content in block display elements such as paragraphs and lists. In Figure 2.2, the `<h1>` tag contains the name of the website in the logo header area at the top of the web page, the `<h2>` tag contains the major topic or name of the page in the content area, and other heading elements are coded in the content area as needed to identify subtopics.

Visually challenged visitors who are using a screen reader can configure the software to display a list of the headings used on a page in order to focus on the topics that interest them. Your well-organized page will be more usable for every visitor to your site, including those who are visually challenged.

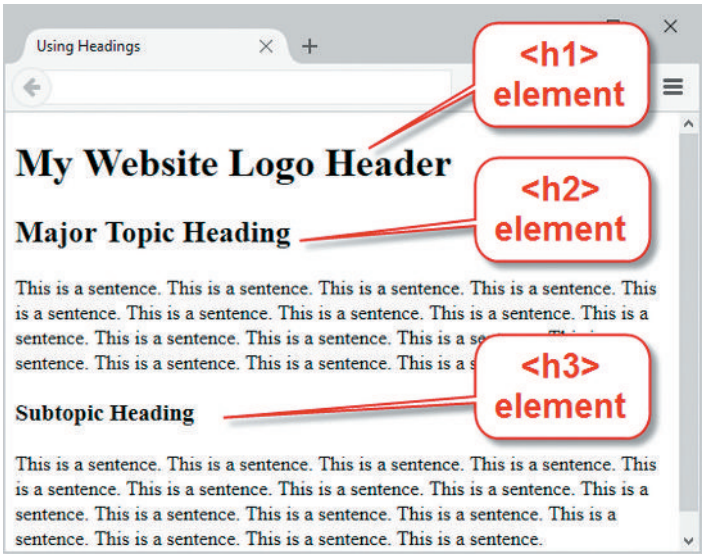


FIGURE 2.2 *Heading tags outline the page.*

More Heading Options in HTML5

You may have heard about the HTML5 header element. The header element offers additional options for configuring headings and typically contains an h1 element. We'll introduce the header element later in this chapter.

Paragraph Element

A **paragraph element** groups sentences and sections of text together. Text within ** ** and ** ** tags is rendered as block display with empty space above and below.

Figure 2.3 shows a web page document containing a paragraph after the first heading.

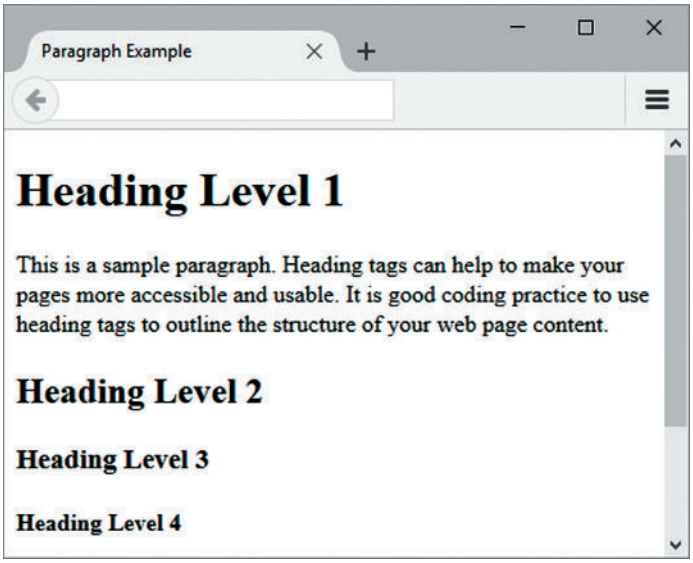


FIGURE 2.3 *Web page using headings and a paragraph.*

Hands-On Practice 2.2

To create the web page shown in Figure 2.3, launch a text editor and open the heading.html file from the chapter2 folder in the student files. Modify the page title and add a paragraph of text to your page below the line with the `<h1>` tags and above the line with the `<h2>` tags:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Paragraph Example</title>
<meta charset="utf-8">
</head>
<body>
<h1>Heading Level 1</h1>
<p>This is a sample paragraph. Heading tags can help to make your
pages more accessible and usable. It is good coding practice to use
heading tags to outline the structure of your web page content.
</p>
<h2>Heading Level 2</h2>
<h3>Heading Level 3</h3>
<h4>Heading Level 4</h4>
<h5>Heading Level 5</h5>
<h6>Heading Level 6</h6>
</body>
</html>
```

Save the document as paragraph.html on your hard drive or flash drive. Launch a browser to test your page. It should look similar to the page shown in Figure 2.3. You can compare your work with the solution found in the student files (chapter2/paragraph.html). Notice how the text in the paragraph wraps automatically as you resize your browser window.

Alignment

As you test your web pages, you may notice that the headings and text begin near the left margin. This is called **left alignment**, and it is the default alignment for web pages. There are times when you want a paragraph or heading to be centered or right aligned (justified). In previous versions of HTML, the align attribute can be used for this. However, the align attribute is **obsolete** in HTML5, which means that the attribute has been removed from the W3C HTML5 specification. You'll learn techniques to configure alignment with Cascading Style Sheets (CSS) in Chapters 6 through 8.

When writing for the Web, avoid long paragraphs. People tend to skim web pages rather than reading them word for word. Use heading tags to outline the page content along with short paragraphs (about three to five sentences each) and lists (which you'll learn about later in this chapter).

Line Break and Horizontal Rule

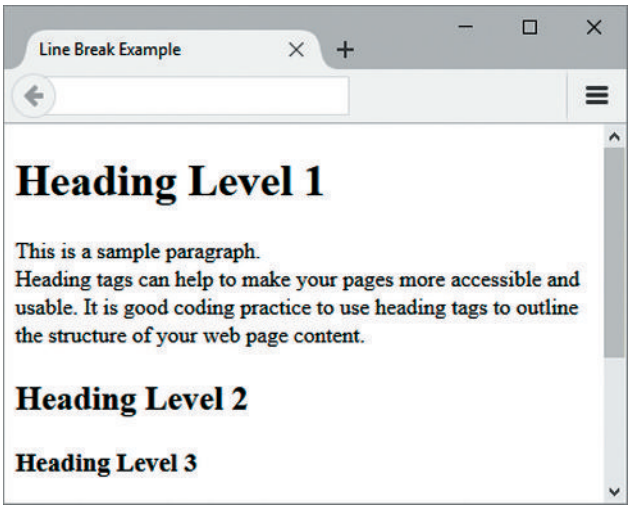


FIGURE 2.4 *Notice the line break after the first sentence.*

The Line Break Element

The **line break element** causes the browser to advance to the next line before displaying the next element or text on a web page. The line break tag is not coded as a pair of opening and closing tags. It is a void element and is coded as **
**. Figure 2.4 shows a web page document with a line break after the first sentence in the paragraph.

Hands-On Practice 2.3

To create the web page shown in Figure 2.4, launch a text editor and open the paragraph.html file from the chapter2 folder in the student files. Modify the text between the title tags to be "Line Break Example." Place your cursor after the first sentence in the paragraph (after "This is a sample paragraph."). Press the Enter key. Save your file. Test your page in a browser and notice that even though your source code displayed the "This is a sample paragraph." sentence on its own line, the browser did not render it that way. A line break tag is needed to configure the browser to display the second sentence on a new line. Edit the file in a text editor and add a <br> tag after the first sentence in the paragraph as shown in the following code snippet:

```
<body>
<h1>Heading Level 1</h1>
<p>This is a sample paragraph. <br> Heading tags can help to make your
pages more accessible and usable. It is good coding practice to use
heading tags to outline the structure of your web page content.
</p>
<h2>Heading Level 2</h2>
<h3>Heading Level 3</h3>
<h4>Heading Level 4</h4>
<h5>Heading Level 5</h5>
<h6>Heading Level 6</h6>
</body>
```

Save your file as linebreak.html. Launch a browser to test your page. It should look similar to the page shown in Figure 2.4. You can compare your work with the solution found in the student files (chapter2/linebreak.html).

The Horizontal Rule Element

Web designers often use visual elements such as lines and borders to separate or define areas on web pages. The **horizontal rule** element, **<hr>**, configures a horizontal line across a web page. Since the horizontal rule element does not contain any text, it is coded as a void element and not in a pair of opening and closing tags. The horizontal rule element has an additional purpose in HTML5, it can be used to indicate a thematic break or change in the content. Figure 2.5 shows a web page document (also found in the student files at chapter2/hr.html) with a horizontal rule after the paragraph. In Chapter 6, you'll learn how to configure lines and borders on web page elements with Cascading Style Sheets (CSS).

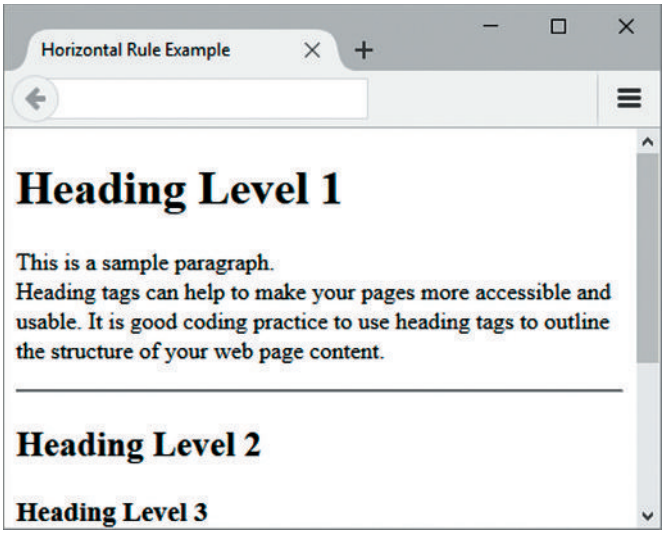


FIGURE 2.5 *The horizontal line is below the paragraph.*

Hands-On Practice 2.4

To create the web page shown in Figure 2.5, launch a text editor and open the linebreak.html file from the chapter2 folder in the student files. Modify the text between the title tags to be: Horizontal Rule Example. Place your cursor on a new line after the `</p>` tag. Code the `<hr>` tag on the new line as shown in the following code snippet:

```
<body>
<h1>Heading Level 1</h1>
<p>This is a sample paragraph. <br> Heading tags can help to make your
pages more accessible and usable. It is good coding practice to use
heading tags to outline the structure of your web page content.
</p>
<hr>
<h2>Heading Level 2</h2>
<h3>Heading Level 3</h3>
<h4>Heading Level 4</h4>
<h5>Heading Level 5</h5>
<h6>Heading Level 6</h6>
</body>
```

Save your file as hr.html. Launch a browser to test your page. It should look similar to the page shown in Figure 2.5. You can compare your work with the solution found in the student files (chapter2/hr.html).

When you are tempted to use a horizontal rule on a web page, consider whether it is really needed. Usually, just leaving extra blank space on the page will serve to separate the content.

Blockquote Element

Besides organizing text in paragraphs and headings, sometimes you need to add a quotation to a web page. The **blockquote element** is used to display a block of quoted text in a special way—indented from both the left and right margins. A block of indented text begins with a **<blockquote>** tag and ends with a **</blockquote>** tag.

Figure 2.6 shows a web page document with a heading, a paragraph, and a blockquote.

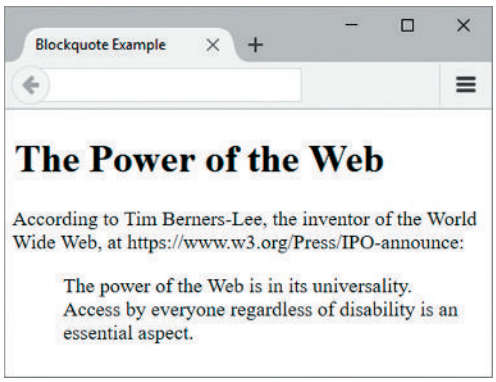


FIGURE 2.6 *The text within the blockquote element is indented.*

Quick TIP

You've probably noticed how convenient the `<blockquote>` tag could be if you needed to indent an area of text on a web page. You may have wondered whether it would be OK to use `<blockquote>` anytime you'd like to indent text or whether the blockquote element is reserved only for long quotations. The semantically correct usage of the `<blockquote>` tag is to use it only when displaying large blocks of quoted text within a web page. Why should you be concerned about semantics? Consider the future of the Semantic Web, described in *Scientific American* as "A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities." Using HTML in a semantic, structural manner is one step toward the Semantic Web. So, avoid using a `<blockquote>` just to indent text. You'll learn modern techniques to configure margins and padding on elements later in this book.

Hands-On Practice 2.5

To create the web page shown in Figure 2.6, launch a text editor and open the template.html file from the chapter2 folder in the student files. Modify the text in the title element. Add a heading tag, a paragraph tag, and a blockquote tag to the body section as indicated by the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Blockquote Example</title>
<meta charset="utf-8">
</head>
<body>
<h1>The Power of the Web</h1>
<p>According to Tim Berners-Lee, the inventor of the World Wide Web,
at https://www.w3.org/Press/IPO-announce:
</p>
<blockquote>
The power of the Web is in its universality. Access by everyone
regardless of disability is an essential aspect.
</blockquote>
</body>
</html>
```

Save the document as blockquote.html on your hard drive or flash drive. Launch a browser such as Edge or Firefox to test your page. It should look similar to the page shown in Figure 2.6. You can compare your work with the solution found in the student files (chapter2/blockquote.html).

FAQ

Why does my web page still look the same?

Often, students make changes to a web page but get frustrated because their browser shows an older version of the page. The following troubleshooting tips are helpful when you know you have modified your web page, but the changes do not show up in the browser:

1. Be sure to save your web page file after you make changes.
2. Verify the location where you save your file—the hard drive, a particular folder.
3. Verify the location from where your browser is requesting the file—the hard drive, a particular folder.
4. Be sure to click the web browser Refresh or Reload button.

Phrase Element

A **phrase element** indicates the context and meaning of the text between the container tags. It is up to each browser to interpret that style. Phrase elements are displayed directly in line with the text (referred to as inline display) and can apply to either a section of text or even a single character of text. For example, the **** element indicates that the text associated with it has strong importance and should be displayed in a "strong" manner in relation to normal text on the page.

Table 2.1 lists common phrase elements and examples of their use. Notice that some tags, such as <cite> and <dfn>, result in the same type of display (italics) as the tag in today's browsers. These tags semantically describe the text as a citation or definition, but the physical display is usually italics in both cases.

TABLE 2.1 *Phrase Elements*

Element	Example	Usage
<abbr>	WIPO	Identifies text as an abbreviation
<b>	**bold** text	Has no extra importance but is styled in bold font
<cite>	*cite* text	Identifies a citation or reference; usually displayed in italics
<code>	code text	Identifies program code samples; usually a fixed-space font
<dfn>	*dfn* text	Identifies a definition of a word or term; usually displayed in italics
	emphasized text	Causes text to be emphasized; usually displayed in italics
<i>	*italicized* text	Has no extra importance but is styled in italics
<kbd>	kbd text	Identifies keyboard input; usually has a fixed-space font
<mark>	mark text	Highlights text in order to be easily referenced
<q>	"quoted" text	Indicates a short quote; usually displays quotation marks
<samp>	samp text	Shows program sample output; usually a fixed-space font
<small>	small text	Defines a smaller text
	strong text	Strong importance; usually displayed in bold
<sub>	$_{sub}$text	Displays a subscript as small text below the baseline
<sup>	^{sup}text	Displays a superscript as small text above the baseline
<var>	*var* text	Identifies and displays a variable output; usually displayed in italics

Note that all phrase elements are container tags—both an opening and a closing tag is used. As shown in Table 2.1, the element indicates that the text associated with it has "strong" importance. Usually the browser (or other user agent) will display text in bold type. A screen reader, such as JAWS or Window-Eyes, might interpret text to indicate that the text should be more strongly spoken. In the following line, the phone number is displayed with strong importance:

Call for a free quote for your web development needs: **888.555.5555**

The code follows:

```
<p>Call for a free quote for your web development needs:
<strong>888.555.5555</strong></p>
```

Notice that the opening and closing tags are within the paragraph tags (and). This code is properly nested and is considered to be **well-formed**. When improperly nested, the and tag pairs overlap each other instead of being nested within each other. Improperly nested code will not pass validation testing (see the HTML Syntax Validation section later in this chapter) and may cause display issues.

Figure 2.7 shows a web page document (also found in the student files at chapter2/em.html) that uses the tag to display the emphasized phrase, "Access by everyone," in italics.

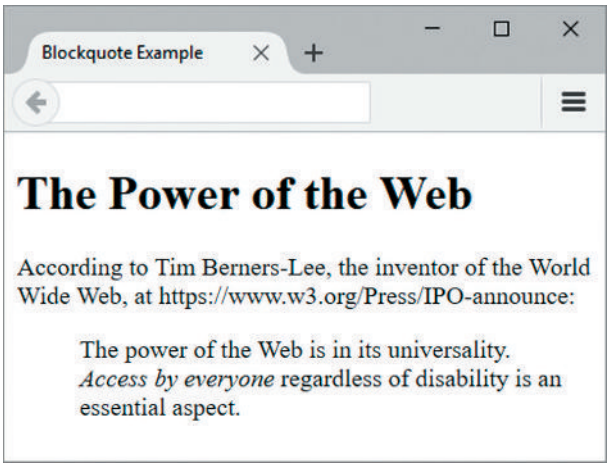


FIGURE 2.7 *The tag in action.*

The code snippet follows:

```
<blockquote>
The power of the Web is in its universality.
<em>Access by everyone</em> regardless of disability is an essential
aspect.
</blockquote>
```

Ordered List

Lists are used on web pages to organize information. When writing for the Web, headings, short paragraphs, and lists can make your page more clear and easy to read. HTML can be used to create three types of lists—description lists, ordered lists, and unordered lists. All lists are rendered as block display with empty space above and below. This section focuses on the **ordered list**, which displays a numbering or lettering system to sequence the information in the list. An ordered list can be organized using numerals (the default), uppercase letters, lowercase letters, uppercase Roman numerals, and lowercase Roman numerals. See Figure 2.8 for a sample ordered list.

FIGURE 2.8 *Sample ordered list.*

Ordered lists begin with an **
** tag and end with an **
** tag. Each list item begins with an **
** tag and ends with an **
** tag. The code to configure the heading and ordered list shown in Figure 2.8 follows:

```
<h1>My Favorite Colors</h1>
<ol>
  <li>Blue</li>
  <li>Teal</li>
  <li>Red</li>
</ol>
```

The `type`, `start`, and `reversed` Attributes

The **`type` attribute** configures the symbol used for ordering the list. For example, to create an ordered list organized by uppercase letters, use `<ol type="A">`. Table 2.2 documents the type attribute and its values for ordered lists.

TABLE 2.2 *The type Attribute for Ordered Lists*

Value	Symbol
1	Numerals (the default)
A	Uppercase letters
a	Lowercase letters
I	Roman numerals
i	Lowercase Roman numerals

Another handy attribute that can be used with ordered lists is the **`start` attribute**, with which you can specify the start value for the list (for example, `start="10"`). Use the **`reversed` attribute** `(reversed="reversed")` to indicate that a list is in descending order.

Hands-On Practice 2.6

In this Hands-On Practice, you will use a heading and an ordered list on the same page. To create the web page shown in Figure 2.9, launch a text editor and open the template.html file from the chapter2 folder in the student files. Modify the title element and add h1, h2, ol, and li tags to the body section, as indicated by the following code:

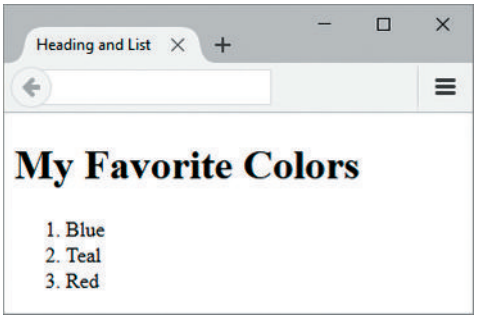


FIGURE 2.9 *An ordered list.*

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Heading and List</title>
<meta charset="utf-8">
</head>
<body>
<h1>My Favorite Colors</h1>
<ol>
  <li>Blue</li>
  <li>Teal</li>
  <li>Red</li>
</ol>
</body>
</html>
```

Save your file as ol.html. Launch a browser and test your page. It should look similar to the page shown in Figure 2.9. You can compare your work with the solution in the student files (chapter2/ol.html).

Take a few minutes to experiment with the `type` attribute. Configure the ordered list to use uppercase letters instead of numerals. Save your file as ola.html. Test your page in a browser. You can compare your work with the solution in the student files (chapter2/ola.html).

FAQ Why is the web page code in the examples indented?

Actually, it doesn't matter to the browser if web page code is indented, but humans find it easier to read and maintain code when it is logically indented. This makes it easier for you or another web developer to understand the source code in the future. For example, it's common practice to indent `<li>` tags a few spaces in from the left margin because it makes it easier to "see" the list with a quick glance at the source code. There is no "rule" as to how many spaces to indent. However, your instructor or the organization you work for may have a standard for you to follow. Web pages with consistent indentation are easier to maintain.

Unordered List

An **unordered list** displays a bullet, or list marker, before each list entry. The default list marker is determined by the browser but is typically a disc, which is a filled-in circle. See Figure 2.10 for a sample unordered list.

FIGURE 2.10 *Sample unordered list.*

Unordered lists begin with a **
** tag and end with a **
** tag. The ul element is a block display element and is rendered with empty space above and below. Each list item begins with an `<li>` tag and ends with an `</li>` tag. The code to configure the heading and unordered list shown in Figure 2.10 is as follows:

```
<h1>My Favorite Colors</h1>
<ul>
  <li>Blue</li>
  <li>Teal</li>
  <li>Red</li>
</ul>
```

Can I change the "bullet" in an unordered list?

Back in the day before HTML5, the type attribute could be included with a `<ul>` tag to change the default list marker to a square `(type="square")` or open circle `(type="circle")`. However, be aware that using the type attribute on an unordered list is considered obsolete in HTML5 because it is decorative and does not convey meaning. No worries, though—there are CSS techniques to configure list markers (bullets) to display images and shapes.

Hands-On Practice 2.7

In this Hands-On Practice, you will use a heading and an unordered list on the same page. To create the web page shown in Figure 2.11, launch a text editor and open the template.html file from the chapter2 folder in the student files. Modify the title element and add h1, ul, and li tags to the body section as indicated by the following code:

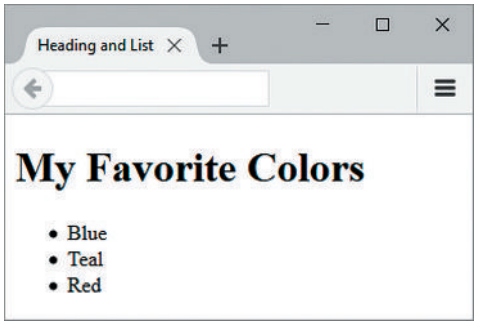


FIGURE 2.11 *An unordered list.*

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Heading and List</title>
<meta charset="utf-8">
</head>
<body>
<h1>My Favorite Colors</h1>
<ul>
  <li>Blue</li>
  <li>Teal</li>
  <li>Red</li>
</ul>
</body>
</html>
```

Save your file as ul.html. Launch a browser and test your page. It should look similar to the page shown in Figure 2.11. You can compare your work with the solution in the student files (chapter2/ul.html).

Description List

A **description list** can be used to organize terms and their descriptions. The terms stand out and their descriptions can be as long as needed to convey your message. Each term begins on its own line at the margin. Each description begins on its own line and is indented. Description lists are also handy for organizing Frequently Asked Questions (FAQs) and their answers. The questions and answers are offset with indentation. Any type of information that consists of a number of corresponding terms and associated descriptions is well suited to being organized in a description list. See Figure 2.12 for an example of a web page that uses a description list.

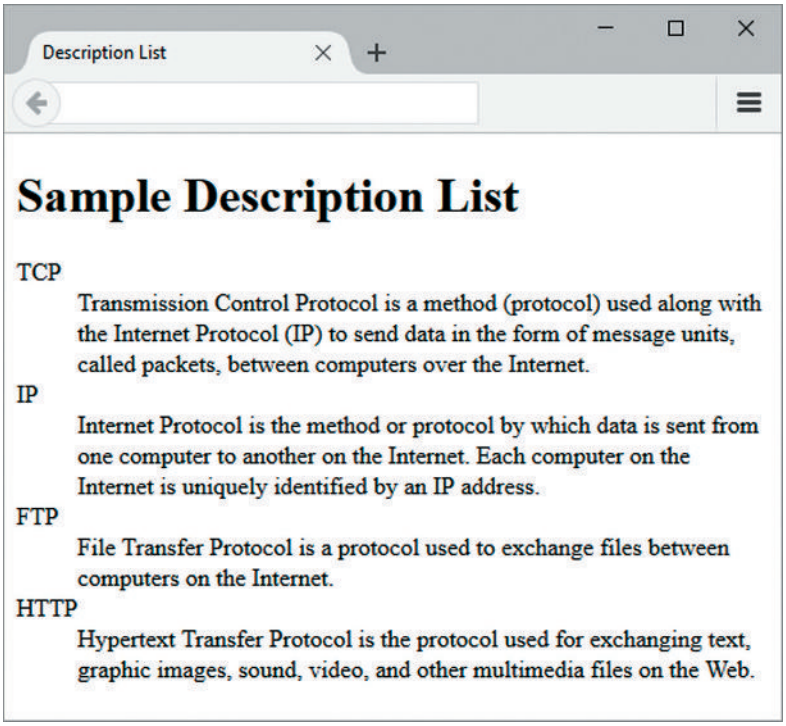


FIGURE 2.12 *A description list.*

Description lists begin with the **<dl>** tag and end with the **</dl>** tag. Each term or name in the list begins with the **<dt>** tag and ends with the **</dt>** tag. Each term description begins with the **<dd>** tag and ends with the **</dd>** tag.

Hands-On Practice 2.8

In this Hands-On Practice, you will use a heading and a description list on the same page. To create the web page shown in Figure 2.12, launch a text editor and open the template.html file from the chapter2 folder in the student files. Modify the title element and add h1, dl, dd, and dt tags to the body section as indicated by the following code: