

HTML AND CSS

A webpage is created using a language called **Hypertext Markup Language (HTML)**. The term *hypertext* refers to the fact that the information is not organized linearly like a book. Instead, you can embed links to other information and jump from one page to another as needed. A more accurate term would be *hypermedia*, because HTML handles many types of information in addition to text, including images, audio, and video.

The term **markup language** refers to the fact that the primary elements of the language take the form of **tags** that you insert into a document to annotate the information stored there. It's as if you took a printed document and marked it up with extra notation to specify other details, as shown in **FIGURE 1**.

HTML documents are regular text and can be created in any general-purpose editor or word processor. Special software tools are also available to help designers create web pages, but these tools ultimately generate HTML documents. When a web page is requested, HTML documents are transported over the Web.

The current version of the HTML standard, **HTML5**, was released in 2011. The discussion in this section applies to HTML5. All of the major browsers support the HTML5 standard. If you're doing web development, be sure you're focusing on HTML5.

HTML works in conjunction with another technology: Cascading Style Sheets (CSS). In general, HTML tags indicate what the information is (such as a paragraph, an image, or a list), and the style information defined by CSS indicates how you want that information to be displayed (such as centered text, a border around an image, or a background color). Since it's of little use to

HTML and CSS

Hypertext Markup Language (HTML)

The language used to create or build a web page



FIGURE 1 A marked-up document

Markup language A language that uses tags to annotate the information in a document

Tag The syntactic element in a markup language that indicates how information should be displayed

HTML5 The newest HTML standard, with a streamlined tag system and support for dynamic content

discuss HTML without CSS, or vice versa, we'll explore both technologies at the same time.

Let's look at an example web page displayed by a browser and then examine the underlying HTML document that defines it. **FIGURE 2** shows a web page displayed in the Firefox browser. The page contains information about a fictional student organization called *Student Dynamics*.

This web page contains an image at the top showing the name of the group. Below the image, offset by a pair of horizontal lines, is a single phrase. Below that is some information about the organization, including a bulleted list of upcoming events followed by some short paragraphs. The small image at the end of one bulleted item indicates that this information has been recently updated. The text in blue represents a link that, when clicked, opens a new web page.

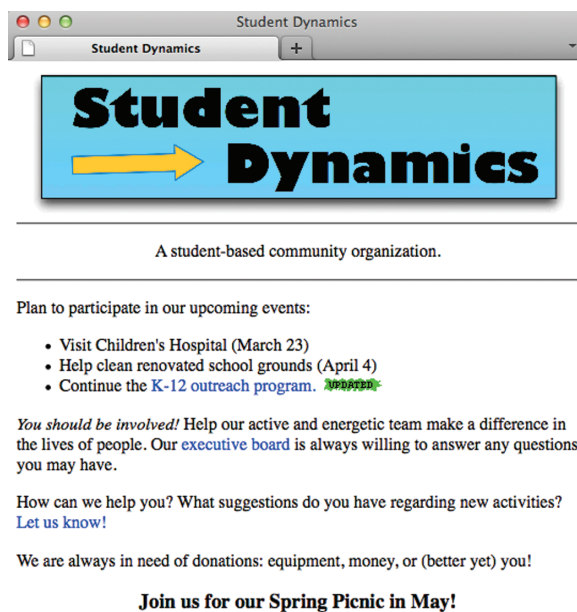


FIGURE 2 The Student Dynamics web page as displayed in Firefox

Courtesy of John Lewis

FIGURE 3 shows the underlying HTML document for this web page. The tags embedded among the document contents are highlighted in blue.

Tags are enclosed in angle brackets (< . . . >). Words such as **head**, **title**, and **body** are called *elements* and specify the type of tag. Elements often consist of a *start tag* such as <body> and a corresponding *end tag* with a / before the element name, such as </body>.

Every HTML file contains two main sections: the head of the document and the body of the document. The head contains information about the document itself, such as its title. The body of the document contains the information to be displayed.

The entire HTML document is enclosed between <html> and </html> tags. The head and body sections of the document are similarly indicated. The text between the <title> and </title> tags appears in the title bar of the web browser when the page is displayed.

HTML and CSS

```

<!DOCTYPE HTML>
<html>
<head>
  <title>Student Dynamics</title>
  <meta charset="utf-8">
  <style type="text/css">
    img.banner {display:block; margin:auto;}
    a:link {color:#0000FF; text-decoration:none;}
    a:visited {color:#00FF00; text-decoration:none;}
    a:hover {color:#FF00FF; text-decoration:underline;}
  </style>
</head>
<body>
  
  <hr>
  <p style="text-align:center;">A student-based community organization.</p>
  <hr>
  <p>Plan to participate in our upcoming events:</p>
  <ul>
    <li>Visit Children's Hospital (March 23)</li>
    <li>Help clean renovated school grounds (April 4)</li>
    <li>Continue the <a href="outreach.html">K-12 outreach program.</a>
      </li>
  </ul>
  <p><em>You should be involved!</em> Help our active and energetic team make
  a difference in the lives of people. Our <a href="execBoard.html">
  executive board</a> is always willing to answer any questions you may
  have.</p>
  <p>How can we help you? What suggestions do you have regarding new
  activities? <a href="suggestions.html">Let us know!</a></p>
  <p>We are always in need of donations: equipment, money, or (better yet)
  you!</p>
  <h3 style="text-align:center;">Join us for our Spring Picnic in May!</h3>
</body>
</html>

```

FIGURE 3 The HTML document defining the Student Dynamics web page

The browser uses the tags, in conjunction with the styles specified by CSS, to determine how the page should be displayed. It ignores the way we format the HTML document using carriage returns, extra spaces, and blank lines. Indenting some lines of the document makes it easier

for a human to read, but such formatting is irrelevant to the way it is finally displayed. A browser takes into account the width and height of the browser window. When you resize the browser window, the contents of the web page are reformatted to fit the new size.

A browser does its best to make sense of the way a document is marked up with tags and displays the page accordingly. If HTML tags conflict, or if they are not properly ordered or nested, the results may be surprising—and unattractive.

Basic HTML Elements

Let's explore some of the core elements of HTML. This discussion will merely scratch the surface of HTML's capabilities, but even so it will give you the ability to create fairly versatile and useful web pages.

Paragraph tags (`<p>` . . . `</p>`) enclose text that should be treated as a separate paragraph. A browser usually begins each paragraph on a new line, with some space separating it from preceding and following paragraphs.

The `<hr>` tag inserts a horizontal rule (that is, a line) across the page. Horizontal rules are helpful in breaking a webpage into sections. The horizontal rule element doesn't enclose content, so there is no closing tag.

It is often useful to display a list of items. The `ul` element defines an unordered list, while the `li` element represents a single list item. In the Student Dynamics example, three list items are enclosed in the `` . . . `` tags. By default, the major browsers display an unordered list using bullets. If the ordered list element (`ol`) is used, the list items are numbered sequentially. Both unordered lists and ordered lists can be nested, creating a hierarchy of lists. Unordered nested lists use different bullet types for each level, and the numbering for each ordered list begins over again at each level.

Several elements are used to define headings in a document. HTML includes six predefined heading elements: `h1`, `h2`, `h3`, `h4`, `h5`, and `h6`. Text enclosed in `<h3>` . . . `</h3>` tags, for instance, is treated as a level 3 heading, which by default is displayed in a larger font than level 4 and a smaller font than level 2. Headings are also displayed in bold by default. Heading tags don't have to specify text that introduces a section; they can be used anywhere you want to change the look.

The `em` element specifies text that should be emphasized. By default, browsers display emphasized text in italics. The `strong` element is similar, and by default is displayed in bold type.

HTML and CSS

Note that the manner in which browsers display a particular element by default may vary slightly from browser to browser. Therefore, the same web page may look different depending on which browser you use to view it. But the default styling of any element can be modified using CSS styles.

Tag Attributes

Many tags can contain **attributes** that indicate additional details about the information or describe how the enclosed information should be displayed. Attributes take the following form:

```
attribute-name = value
```

For example, the paragraph element just below the title image on the Student Dynamics page contains the attribute

```
style="text-align:center"
```

which is a CSS style specification indicating that the text of the paragraph should be centered horizontally on the page.

An image can be incorporated into a web page using the `img` element, which takes an attribute that identifies the image file to display. The attribute name is `src`, meaning the source of the image. There is no separate closing tag for the `img` element. For example,

```

```

inserts the image stored in the file `myPicture.gif` into the HTML document.

In the Student Dynamics example, an image is used as a banner at the top of the page. In another location, a small image indicates which information on the website has been recently updated.

Finally, links are specified using the element `a`, which stands for *anchor*. An anchor tag contains an attribute called `href` that specifies the destination URL. For example,

```
<a href="http://google.com/">Google It!</a>
```

shows the text "Google It!" on the screen as a link. When the user clicks the link, the Google Home page is retrieved and displayed in the browser, replacing the current page.

By default, most browsers display a link in blue, underlined text. As with other elements, this default styling can be overridden with CSS style specifications. Let's explore CSS in more detail.

Attribute Part of a tag that provides additional information about the element

More About CSS

We've seen how CSS styles can be specified as an attribute of an HTML element:

```
<p style="text-align:center">This text is centered!</p>
```

In this case, the style attribute indicates that the text of this paragraph should be centered instead of the default left alignment. But this style applies only to this single paragraph. What if we want to establish a style for all paragraphs on the page?

CSS styles can also be expressed in the head section of the HTML document. For example, we could put the following `style` tag (not an attribute this time, but a tag) between the `<head>` and `</head>` tags:

```
<style type="text/css">
  p {color:#00FF00;}
</style>
```

If we put this tag in the head of the document, it will apply to all paragraph tags in the document (unless it is overridden by a style attribute on a particular paragraph). In this case, the tag will cause all paragraph text to be colored green.

In the Student Dynamics example, the following style tag is used in the head section of the document:

```
<style type="text/css">
  img.banner {display:block; margin:auto;}
  a:link {color:#0000FF; text-decoration:none;}
  a:visited {color:#00FF00; text-decoration:none;}
  a:hover {color:#FF00FF; text-decoration:underline;}
</style>
```

The first style rule in this example applies to all `img` tags that have been designated with the class `banner` (using the `class` attribute). Element classes are used to specify a subset of elements. In the Student Dynamics example, the `banner` class is applied only to the image at the top of the page. By setting the margins to `auto`, we center the image horizontally on the page.

The other three style rules affect the various states of an anchor link. Recall that by default, a link is shown in blue underlined text. If we set `text-decoration` to `none`, the underline is removed. So, these rules establish that an unvisited link is shown in blue text with no underline, a visited link is shown in green with no underline, and when the mouse hovers over a link, the link text turns purple with an underline.

Union: © matka_Wariatka/Shutterstock, Inc.

Tim Berners-Lee

Tim Berners-Lee is the first holder of the 3COM (Computer Communication Compatibility) Chair at the Computer Science and Artificial Intelligence Laboratory (CSAIL) at Massachusetts Institute of Technology. He is the 3COM Founders Professor of Engineering in the School of Engineering with a joint appointment in the Department of Electrical Engineering and CSAIL at MIT. Berners-Lee is a researcher, evangelist, and visionary rather than an academician. He is Director of the World Wide Web Consortium, which coordinates web development worldwide. The Consortium, with teams at MIT, ERCIM in France, and Keio University in Japan, aims to help the Web achieve its full potential, ensuring its stability through rapid evolution and revolutionary transformations of its usage.

How did Tim Berners-Lee arrive at this very important position? He built his first computer while a student at Queen's College, Oxford, in the United Kingdom. After graduation, he worked for two years with Plessey Telecommunications Ltd., a major telecom equipment manufacturer in the United Kingdom; he then worked as an independent consultant for a year and a half, followed by three years at Image Computer Systems Ltd. His various projects during this time included real-time control firmware, graphics and communications software, and a generic macro language.

In 1984, Berners-Lee took up a fellowship at CERN, the European Organization for Nuclear Research in Geneva, Switzerland, where he worked on a heterogeneous remote procedure call system and a distributed real-time system for scientific data acquisition and system control. In 1989, he proposed a global hypertext project to be known as the *World Wide Web*. It was designed to allow people to work together by combining their knowledge in a web of hypertext documents. Berners-Lee wrote the first World Wide Web server, "httpd," and the first client, "World Wide Web," a what-you-see-is-what-you-get hypertext browser/editor. The work began in October 1990, and the program "World Wide Web" was made available within CERN in December 1990 and on the Internet at large in the summer of 1991.

Between 1991 and 1993, Berners-Lee continued working on the design of the Web, coordinating feedback from users across the Internet. His initial specifications of URLs, HTTP, and HTML were refined and discussed in larger circles as the web technology spread. Eventually, it became apparent that the physics lab in Geneva was not the appropriate place for the task of developing and monitoring the Web. In October 1994, the World Wide Web



© Hank Morgan/Science Source

Unicon: © matka_Wanatka/Shutterstock, Inc.

Tim Berners-Lee (continued)

Consortium was founded by Berners-Lee at the MIT Laboratory for Computer Science.

In a *New York Times* article in 1995, Berners-Lee was asked about private corporations trying to dominate web standards for profit. He responded, “There’s always the threat that a particular company would dominate the market and control the standards of the Web.” But he feels strongly that this should not happen. “The essence of the Web is that it’s a universe of information,” he said. “And it wouldn’t be universal if it was tied, in any way, to one company.”

Michael Dertouzos, the director of the Computer Science Laboratory at MIT, has said that Berners-Lee seems to embody the “libertarian idealism” of the Internet culture. “He has a real commitment to keeping the Web open as a public good, in economic terms,” Dertouzos said. “That’s his mission.” Berners-Lee concludes: “Reasonable competition speeds the pace of innovation. Companies will promote the proprietary aspects of their browsers and applications, and they should. But the navigation of the Web has to be open. If the day comes when you need six browsers on your machine, the World Wide Web will no longer be the World Wide Web.”

Berners-Lee was one of *Time* magazine’s 100 most important people of the twentieth century. In recognition of his work on the

World Wide Web, Queen Elizabeth II made Berners-Lee a Knight Commander, Order of the British Empire (KBE).

In June 2007, he received the Order of Merit from Queen Elizabeth II, which entitles him to use “OM” after his name and gives him membership in this dynastic order recognizing distinguished service. In September 2008, he was awarded the IEEE/RSE Wolfson James Clerk Maxwell Award for conceiving and further developing the World Wide Web, and in 2009 he was elected as a foreign associate of the National Academy of Sciences.

At the opening ceremony of the 2012 Summer Olympics in London, Berners-Lee was honored as the “Inventor of the World Wide Web.” He tweeted the words “This is for everyone,” which appeared in LCD lights attached to the chairs of the 80,000 people in the stadium. His philosophy: Everyone everywhere should have affordable access to the Internet. He is president of the Alliance for Affordable Internet, an organization launched in October 2013 that seeks to make the Internet more accessible and affordable worldwide.

HTML and CSS

In addition to being able to specify styles at the element level and the document level, you can also put CSS style rules in a separate external file. That way, multiple pages, or even an entire website, can easily share a consistent set of style rules. And that's where the word *cascading* comes from in Cascading Style Sheets: Styles can be specified at multiple levels and overridden at lower levels as needed.

More HTML5 Elements

The HTML5 standard streamlines many of the issues that were problematic in previous versions. It features many new tags to represent content, including:

- `<section>` - to define sections of pages
- `<header>` - to define the header of a page
- `<footer>` - to define the footer of a page
- `<nav>` - to define the navigation elements on a page
- `<article>` - to define an article or primary content of a page
- `<aside>` - to define secondary content that might appear in a sidebar
- `<figure>` - to define images that annotate an article

Another exciting aspect of HTML5 is its support of dynamic content—content that changes even if the user is not interacting with the page. Some dynamic techniques are discussed later in this chapter, though HTML5 may make some of them unnecessary over time.

One example of the support of dynamic content in HTML5 is the idea of a context menu for presenting drop-down menus as appropriate. There is also an additional attribute (`async`) that can be included in a tag to indicate content that should be loaded asynchronously, which helps increase page loading speed.

HTML5 also includes several new tags that support accepting input from a form, such as tags for time and dates, as well as fields for ranges, email addresses, and URLs.

Of course, this book can only provide a glimpse into the world of web page development using HTML and CSS. You are encouraged to use this as a starting point to explore this area of computing in more detail.