2 HTML BASICS

Lesson overview

In this lesson, you'll familiarize yourself with HTML and learn:

- What HTML is and where it came from
- Frequently used HTML tags
- How to insert special characters
- What semantic web design is and why it's important
- New features and capabilities in HTML

This lesson will take about 25 minutes to complete. This lesson does not have support files.

[[DW2017-02-00.tif]] (To Come)

HTML is the backbone of the web, the skeleton of your webpage. Like the bones in your body, it is the structure and substance of the Internet, although it is usually unseen except by the web designer. Without it, the web would not exist. Dreamweaver has many features that help you access, create, and edit HTML code quickly and effectively.

What is HTML?

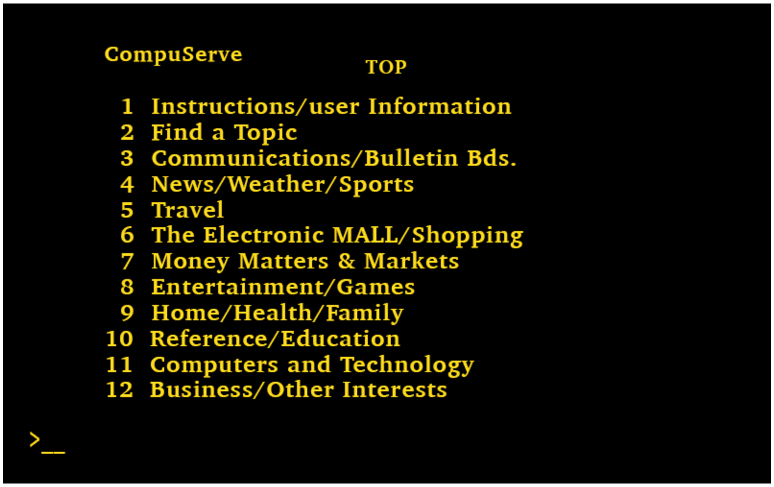
“What other programs can open a Dreamweaver file?” asked a student in a Dreamweaver class. Although it might seem obvious to an experienced developer, it illustrates a basic problem in teaching and learning web design. Most people confuse the program with the technology. Some may assume that the extension .htm or .html belongs to Dreamweaver or Adobe. This isn’t as unusual as it seems. Print designers are used to working with files ending with extensions such as .ai, .psd, .indd, and so on; it’s just a part of their jobs. They have learned over time that opening these file formats in a different program may produce unacceptable results or even damage the file.

On the other hand, the goal of the web designer is to create a webpage for display in a browser. The power or functionality of the originating program has little bearing on the resulting browser display, because the display is all contingent on the HTML code and how the browser interprets it. Although a program may write good or bad code, it’s the browser that does all the hard work.

The web is based primarily on the HyperText Markup Language (HTML). The language and the file format don’t belong to any individual program or company. In fact, it is a *non*-proprietary, plain-text language that can be edited in any text editor, in any operating system, on any computer. Dreamweaver is, in part, an HTML editor, although it is also much more than this. But to maximize the potential of Dreamweaver, it’s vital that you have a good understanding of what HTML is and what it can (and can’t) do. This lesson is intended as a concise primer on HTML and its capabilities. It will be a helpful foundation for understanding Dreamweaver.

Where did HTML begin?

HTML and the first browser were invented in 1989 by Tim Berners-Lee, a computer scientist working at the CERN (Conseil Européen pour la Recherche Nucléaire, which is French for European Council for Nuclear Research) particle physics laboratory in Geneva, Switzerland. He intended the technology to be a means for sharing technical papers and information via the fledgling Internet that existed at the time. He shared his HTML and browser inventions openly as an attempt to get the scientific community at large and others to adopt them and engage in the development themselves. The fact that he did not copyright or try to sell his work started a trend for openness and camaraderie on the web that continues to this day.



```
CompuServe                    TOP

  1  Instructions/user Information
  2  Find a Topic
  3  Communications/Bulletin Bds.
  4  News/Weather/Sports
  5  Travel
  6  The Electronic MALL/Shopping
  7  Money Matters & Markets
  8  Entertainment/Games
  9  Home/Health/Family
 10  Reference/Education
 11  Computers and Technology
 12  Business/Other Interests

>__
```

The Internet before HTML looked more like MS DOS or the macOS Terminal application. There was no formatting, no graphics, and no user-definable color.

The language that Berners-Lee created over 25 years ago was a much simpler construct of what we use now, but HTML is still surprisingly easy to learn and master. At the time of this writing, HTML is now at version 5, officially adopted as of October 2014. It consists of over 120 *tags*, such as html, head, body, h1, p, and so on.

The tag is inserted between less-than (<) and greater-than (>) angle brackets, as in `<p>`, `<h1>`, and `<table>`. These tags are used to identify, or *mark up*, text and graphics to signal the browser to display them in a particular way. HTML code is considered properly *balanced* when the markup features both an opening (`<...>`) and a closing (`</...>`) tag, such as `<h1>...</h1>`.

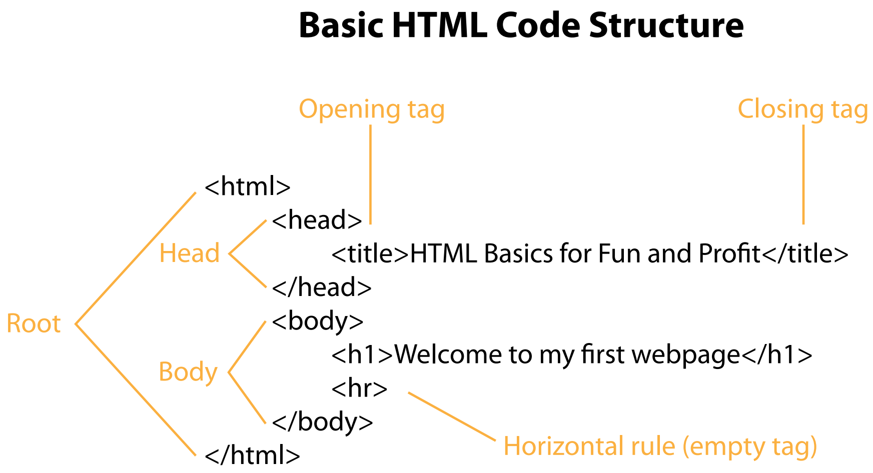
When two matching tags appear this way, they are referred to as an *element*; an element encompasses any contents contained within the two tags. Empty, or void, elements, like the horizontal rule, can be written in an abbreviated fashion using only one tag, such as `<hr/>`, essentially opening and closing the tag at the same time. In HTML5, empty elements can also be validly expressed without the closing slash, such as `<hr>`. Some web applications require the closing slash, so it's a good idea to check before using one form over the other.

● **Note:** If you are dead set against learning how to read and write good HTML, you should check out Adobe Muse. This program allows you to create professional-looking webpages and complete websites using point-and-click techniques in a graphical user interface similar to Adobe InDesign while never exposing you to the code running behind the scenes.

Some elements are used to create page structures, others to structure and format text, and yet others to enable interactivity and programmability. Even though Dreamweaver obviates the need for writing most of the code manually, the ability to read and interpret HTML code is still a recommended skill for any burgeoning web designer. Sometimes it's the only way to find an error in your webpage. The ability to understand and read code may also become an essential skill in other fields as more information and content is created and disseminated via mobile devices and Internet-based resources.

Basic HTML code structure

Here you see the basic structure of a webpage:



Basic HTML Code Structure

```
<html>
    <head>
        <title>HTML Basics for Fun and Profit</title>
    </head>
    <body>
        <h1>Welcome to my first webpage</h1>
        <hr>
    </body>
</html>
```

You may be surprised to learn that the only text from this code that displays in the web browser is "Welcome to my first webpage." The rest of the code creates the page structure and text formatting. Like an iceberg, most of the content of the actual webpage remains out of sight.

● **Note:** Go to the book's online resources at Peachpit.com for bonus hands-on exercises to gain some vital skills and experience writing and editing HTML code. See the "Getting Started" section at the beginning of the book for more details.

Frequently used HTML elements

HTML code elements serve specific purposes. Tags can create distinct objects, apply formatting, identify logical content, or generate interactivity. Tags that make their own space on the screen and stand alone are known as *block* elements; the ones that perform their duties within the flow of another tag are known as *inline* elements. Some elements can also be used to create *structural* relationships within a page, like stacking content in vertical columns or collecting several elements together in logical groupings. Structural elements can behave like block or inline elements or do their work entirely invisible to the user.

HTML tags

Table 2.1 shows some of the most frequently used HTML tags. To get the most out of Dreamweaver and your webpages, it helps to understand the nature of these elements and how they are used. Remember that some tags can serve multiple purposes.

Table 2.1 Frequently used HTML tags

TAG	DESCRIPTION
<!--...-->	Comment. Designates an HTML comment. Allows you to add notes within the HTML code (represented by … in the tag) that are not displayed within the browser.
<a>	Anchor. The basic building block for a hyperlink.
<blockquote>	Quotation. Creates a standalone, indented paragraph.
<body>	Body. Designates the document body. Contains the visible portions of the webpage content.
<br>	Break. Inserts a line break without creating a new paragraph.
<div>	Division. Used to divide webpage content into discernible sections.
<em>	Emphasis. Adds semantic emphasis. Displays as italics by default in most browsers and readers.
<form>	Form. Designates an HTML form. Used for collecting data from users.
<h1> to <h6>	Headings. Creates headings. Default formatting is bold.
<head>	Head. Designates the document head. Contains code that performs background functions, such as meta tags, scripts, styling, links, and other information not overtly visible to site visitors.
<hr>	Horizontal rule. Empty element that generates a horizontal line.
<html>	Root element of most webpages. Contains the entire webpage, except in certain instances where server-based code must load before the opening <html> tag.
<iframe>	Inline frame. A structural element that can contain another document or load content from another website.
<img>	Image. Provides the source reference to display an image.
<input>	Input. An input element for a form such as a text field.
<li>	List Item. The content of an HTML list.
<link>	Link. Designates the relationship between a document and an external resource.
<meta>	Metadata. Additional information provided for search engines or other applications.

TAG	DESCRIPTION
<ol>	Ordered list. Defines a numbered list. List items display in an alphanumeric or roman numeral sequence.
<p>	Paragraph. Designates a standalone paragraph.
<script>	Script. Contains scripting elements or points to an internal or external script.
<span>	Span. Designates a section within an element. Provides a means to apply special formatting or emphasis to a portion of an element.
<strong>	Strong. Adds semantic emphasis. Displays as bold by default in most browsers and readers.
<style>	Style. Embedded or inline container for CSS styling.
<table>	Table. Designates an HTML table.
<td>	Table data. Designates a table cell.
<textarea>	Text area. Designates a multi-line text input element for a form.
<th>	Table header. Identifies a cell as containing a header.
<title>	Title. Contains the metadata title reference for the current page.
<tr>	Table row. Structural element that delineates one row of a table from another.
<ul>	Unordered list. Defines a bulleted list. List items display with bullets by default.

HTML character entities

Text content is normally entered via a computer keyboard. But many characters don't appear on a typical 101-key input device. If a symbol can't be entered directly from the keyboard, it can be inserted within the HTML code by typing the name or a numeric value, referred to as an entity. Entities exist for every letter and character that can be displayed. Some popular entities are listed in Table 2.2:

● **Note:** Some entities can be created using either a name or a number, as in the copyright symbol, but named entities may not work in all browsers or applications. So either stick to numbered entities or test the specific named entities before you use them.

Table 2.2 HTML character entities

CHARACTER	DESCRIPTION	NAME	NUMBER
©	Copyright	&copy;	&#169;
®	Registered trademark	&reg;	&#174;
™	Trademark		&#153;
•	Bullet		&#149;
–	En dash		&#8211;
—	Em dash		&#8212;
	Nonbreaking space	&nbsp;	&#160;

Go to www.w3schools.com/html/html_entities.asp to see a complete description and list of entities.

What's new in HTML5

Every new version of HTML has made changes to both the number and purpose of the tags that make up the language. HTML 4.01 consisted of approximately 90 tags. HTML5 has removed some of those HTML 4 tags from its specification altogether, and some new ones have been adopted or proposed.

Changes to the list usually revolve around supporting new technologies or different types of content models, as well as removing features that were bad ideas or infrequently used. Some changes simply reflect customs or techniques that have been popularized within the developer community over time. Other changes have been made to simplify the way code is created, to make it easier to write and faster to disseminate.

HTML5 tags

Table 2.3 shows some of the important new tags in HTML5. The specification features nearly 50 new tags in total, while at least 30 old tags were deprecated. As we move through the exercises of this book, you will learn how to use many of these new HTML5 tags, as appropriate, to help you understand their intended role on the web. Take a few moments to familiarize yourself with these tags and their descriptions.

Go to www.w3schools.com/tags/default.asp to see the complete list of HTML5 elements.

Table 2.3 Important new HTML5 tags

TAG	DESCRIPTION
`<article>`	Article. Designates independent, self-contained content, which can be distributed independently from the rest of the site.
`<aside>`	Aside. Designates sidebar content that is related to the surrounding content.
`<audio>`	Audio. Designates multimedia content, sounds, music, or other audio streams.
`<canvas>`	Canvas. Designates graphics content created using a script.
`<figure>`	Figure. Designates a section of standalone content containing an image or video.
`<figcaption>`	Figure caption. Designates a caption for a `<figure>` element.
`<footer>`	Footer. Designates a footer of a document or section.
`<header>`	Header. Designates the introduction of a document or section.
`<hgroup>`	Heading group. Designates a set of <h1> to <h6> elements when a heading has multiple levels.
`<nav>`	Navigation. Designates a section of navigation.
`<picture>`	Picture. Designates one or more resources for a webpage image to support the various resolutions available on smartphones and other mobile devices. This is a new tag that may not be supported in older browsers or devices.
`<section>`	Section. Designates a section in a document.
`<source>`	Source. Designates media resources for video or audio elements. Multiple sources can be defined for browsers that do not support the default file type.
`<video>`	Video. Designates video content, such as a movie clip or other video streams.

Semantic web design

Many of the changes to HTML were made to support the concept of *semantic web design*. This movement has important ramifications for the future of HTML, its usability, and the interoperability of websites on the Internet. At the moment, each webpage stands alone on the web. The content may link to other pages and sites, but there's really no way to combine or collect the information available on multiple pages or multiple sites in a coherent manner. Search engines do their best to index the content that appears on every site, but much of it is lost due to the nature and structure of old HTML code.

HTML was initially designed as a presentation language. In other words, it was intended to display technical documents in a browser in a readable and predictable manner. If you look carefully at the original specifications of HTML, it looks like a list of items you would put in a college research paper: headings, paragraphs, quoted material, tables, numbered and bulleted lists, and so on.

The element list in the first version of HTML basically identified how the content would be displayed. These tags did not convey any intrinsic meaning or significance. For example, using a heading tag displayed a particular line of text in bold, but it didn't tell you what relationship the heading had to the following text or to the story as a whole. Is it a title or merely a subheading?

HTML5 has added a significant number of new tags to help us add semantic meaning to our markup. Tags such as `<header>`, `<footer>`, `<article>`, and `<section>` allow you for the first time to identify specific content without having to resort to additional attributes. The result is simpler code and less of it. But most of all, the addition of semantic meaning to your code allows you and other developers to connect the content from one page to another in new and exciting ways—many of which haven't even been invented yet. It's truly a work in progress.

New techniques and technology

HTML5 has also revisited the basic nature of the language to take back some of the functions that over the years have been increasingly handled by third-party plug-in applications and programming.

If you are new to web design, this transition will be painless because you have nothing to relearn and no bad habits to break. If you already have experience building webpages and applications, this book will guide you safely through some of these waters and introduce the new technologies and techniques in a logical and straightforward way. But either way, you don't have to trash all your old sites and rebuild everything from scratch.

Valid HTML 4 code will remain valid for the foreseeable future. HTML5 was intended to make web design easier by allowing you to do more with less work.
So let's get started!

See www.w3.org/TR/2014/WD-html5-20140617 to learn more about HTML5.

See www.w3.org to learn more about W3C.

Review questions

1 What programs can open HTML files?

2 What does a markup language do?

3 HTML is composed of how many code elements?

4 What are the three main parts of most webpages?

5 What is the difference between block and inline elements?

6 What is the current version of HTML?

Review answers

1 HTML is a plain-text language that can be opened and edited in any text editor and viewed in any web browser.

2 A markup language places tags contained within brackets (< >) around plain-text content to pass information concerning structure and formatting from one application to another.

3 HTML5 contains over 100 tags.

4 Most webpages are composed of three main sections: root, head, and body.

5 A block element creates a standalone element. An inline element can exist within another element.

6 HTML5 was formally adopted at the end of 2014. However, full support may take several more years. And, as with HTML 4, some browsers and devices may support the specification in differing ways.