

Basic HTML 1

October 2003

Copyright © 2003 University of Delaware. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage or profit and that the material duplicated is attributed to Information Technologies, University of Delaware. To copy otherwise or to republish requires specific permission and a fee.

Naming your home page

Most Web sites contain multiple files that are linked to a “home page” (your main or introductory page). The most efficient way to handle a home page is to give it the name

index.html

The advantage to naming your home page “index.html” is that most browsers will display the “index.html” file if no specific HTML filename is given. Thus, people searching for your personal Web page on the “udel.edu” Web server only have to know your username to be able to find your files, and people on the University’s central Web server only have to know the name of the directory in which your files reside. They do not have to remember an HTML filename to get to your information.

Headings

The organization of your document depends upon your effective use of headings to produce manageable sections. There are six heading levels in HTML, numbered 1 through 6, from most important to least important:

| | |
|--------------------------|-----------------|
| <h1>Heading level 1</h1> | most important |
| <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> | least important |

Headings are usually larger and bolder than regular HTML text. However, do not assume that the way each heading appears on your computer screen will be the

way it looks on every computer. Each browser can define how the headings look, and the user can often change settings too. For this reason, use headings to show how your information is organized logically. This means that you would use a level 3 heading after a level 2 heading and not before.

The first heading of a document—which often repeats the text used for the title—should use an `<h1>` tag. Most documents use only one `<h1>` tag. Headings are automatically followed by a blank line.

Paragraphs

To indicate paragraphs in HTML, use the `<p>` and `</p>` tags.

1. The paragraph tag (`<p>`) indicates breaks between paragraphs by inserting a blank line between them on the browser page.
2. Blank lines in your text file are ignored. Only a `<p>` tag produces a blank line on the browser page.
3. Tabs and multiple spaces are collapsed into one space.
4. Word wrapping can occur anywhere in your text document and doesn't affect the appearance of your browser page. In other words, you could type whole paragraphs or your entire HTML document on one line and still have the page nicely formatted in the browser. (This isn't recommended because it makes editing the text difficult.)
5. To include short lines in your Web document with no extra spaces between lines (for example, lines in a mailing address or a poem), use the line break `
` tag.

To see how the paragraph and line break tags work in HTML, complete the following exercise:

Exercise 3: Adding paragraphs

1. In your “index.html” file, between the `</h1>` and the `</body>` lines, type the following as it appears here, including the blank line. To make your text file easy to edit, press the ENTER key at the end of each line.

```
<p>
Creating HTML pages takes planning,
but the results are worthwhile.
```

```
Imagine the excitement of having your Web page seen by
people all over the world.
</p>
```

Continued on next page

Exercise, continued

```
<p>
Keep your Web pages to a few screens each.
You can link information together, so don't worry about
including everything at once.
</p>
<p>
Remember that readers do not always find
your home page first.
A link from somewhere else can lead them to
the middle of your information,
so try not to use words like
<br />
back<br />
forward<br />
previous<br />
next
<br />
to indicate navigation on your pages.
</p>
```

2. Save your text file.
 3. In the browser window, click “Reload current page” to see the text you have added.
-

You might prefer to leave blank lines between paragraphs in your HTML text file to make it more readable as you're editing, but that's optional. Remember that you will see blank lines on the browser screen only if you include a `<p>` tag.

HTML lists

After headings, lists are the next most important way of organizing information for your viewers. Lists are useful for creating tables of contents, step-by-step instructions, outlines, glossaries, etc. The most common HTML lists are

| Type of list | Appearance on page |
|--|---------------------------------|
| Ordered lists <code></code> | numbered |
| Unordered lists <code></code> | bulleted |
| Definition lists <code><dl></code> | a term followed by a definition |

Note: You can include a `
` tag within a list item to move text to the following line. The same indentation will be preserved, but no new bullet or number will be produced.

Ordered lists

Ordered lists () number their items sequentially. Tags for an ordered list are

```
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ol>
```

The list begins and ends with the ordered list tags and . Each item in the list is indicated by a “list item” tag. This pattern is the same for most lists with only the tag for the type of list changing.

The ordered list tags above would produce the following when you view your HTML document in a browser:

1. List item 1
2. List item 2
3. List item 3

Unordered lists

Unordered lists () use bullets to indicate items. Tags for an unordered list are

```
<ul>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ul>
```

The unordered list begins and ends with the tags and . Each item in the list is indicated by a list item tag.

The unordered list tags above would produce the following in your HTML document:

- List item 1
- List item 2
- List item 3

Definition lists

Definition lists (<dl>) are a little different from other lists because they use an additional tag. Tags for a definition list are

```
<dl>
  <dt>Definition term</dt>
  <dd>Definition</dd>
  <dt>Definition term</dt>
  <dd>Definition</dd>
</dl>
```

The list begins and ends with the definition list tags <dl> and </dl>. There are separate tags for the definition term <dt></dt> and for the definition itself <dd></dd>. The definition term consists of a single line, while the definition can contain multiple lines or paragraphs.

Definition lists are useful for glossaries and for any situation where you want to set off one element (the definition term) and expand upon it (the definition).

The definition list tags above would produce something like the following in your HTML document:

```
Definition term
Definition
Definition term
Definition
```

You can “nest” lists and types of lists (put one inside the other). Nesting is another way to show the organization of your material.

The next exercise gives you practice using HTML lists.

Exercise 4: Building a list

1. In your “index.html” file, type the following text just above the </body> tag:

```
<p>
Following are some HTML features to help you build a
great page:
</p>
<h2>HTML structural elements</h2>
<ol>
  <li>Formatting tags</li>
```

Continued on next page

Exercise, continued

```
        <ul>
          <li>Heading tags</li>
          <li>Paragraph tags</li>
          <li>List tags</li>
        </ul>
      </ol>
```

2. Save your file and then click “Reload current page” in the browser to view it.
-

Note: You’ll continue the ordered list in the next exercise.

Customizing the look of your lists

The challenge in using lists and nested lists is to begin and end each list in the proper place. Failure to do so results in errors that become immediately apparent when you view your Web document. One way to keep things straight for yourself is to indent successive lists so that like elements are under each other. Notice that Web browsers indent nested lists, making the structure clear.

How lists appear depends upon the browser being used and upon how the individual user has set up the browser. For example, the bullet that appears in a bulleted list might be a filled-in circle on one computer, an open circle on another, and a square on a third. Some browsers put definitions on separate lines, some do not. As the author, your concern is to indicate only what kind of list is being presented.

You can exercise *some* control over how lists are numbered or over what kinds of bullets are used by including modifiers with your HTML list tags. These modifiers are called “attributes,” and they indicate exactly how the tag is to be modified. For example, the “type” attribute, used with a list tag, allows you to specify different numbering or bullet schemes. Attributes, in turn, can have “values” that further define them. Thus, if you give the “type” attribute a value of “A,” your ordered list will appear with capital letters. Values should be enclosed in quotation marks. Similarly, other HTML tags have associated attributes and values that let you control how your page looks. Directions for using the “type” attribute with list tags follow:

Ordered lists

| | |
|--|------------------------------|
| Tag: <ol type=“ <i>type of numbering you prefer</i> ”> | |
| type=“1” | for numbers (default) |
| type=“A” | for capital letters |
| type=“a” | for lowercase letters |
| type=“I” | for capital Roman numerals |
| type=“i” | for lowercase Roman numerals |

Unordered lists

Tag: `<ul type="type of bullet you prefer">`
`type="disc"` for a solid round bullet
`type="circle"` for a hollow round bullet
`type="square"` for a hollow square bullet

Writing HTML with style

Another way to set off text in your Web documents is through the use of styles like bold, italics, and so on. To indicate that text is to have a certain style, enclose it in tags that define the style:

| | |
|---|--------------------|
| <code>bold text</code> | bold text |
| <code><i>italic text</i></code> | <i>italic text</i> |
| <code><tt>typewriter text</tt></code> | typewriter text |

Styles can be combined with each other. To combine styles, nest one style within another. For example,

| | |
|---|----------------------------|
| <code><i>bold italics</i></code> | <i>bold italics</i> |
|---|----------------------------|

The next exercise continues the ordered list and lets you see the effects of the style tags on the text they surround.

Exercise 5: Using style tags

1. In your “index.html” file, type the following text just above the `` tag:

```
<li>Style tags</li>
<ul>
  <li><b>Bold text</b></li>
  <li><i>Italic text</i></li>
  <li><tt>Typewriter text</tt></li>
</ul>
```

2. Save your file and then click “Reload current page” in the browser to view it.
-