

# HTML Workshop

## An Introduction

**HTML** (*Hypertext Markup Language*) is a standardized computer language for certain files on the Internet. A *standardized language* means that all programs can read and process files written in this language. Not all languages are standardized. For example, there is no standard for word processing documents.

Word processing programs, e.g. *Microsoft Word* vs. *Wordperfect* cannot work with each other's files. As a result, each program needs to convert files to its own version. Sometimes the results of the conversions are better; sometimes, they are worse.

**HTML** is the standard that all designers follow for transferring documents on the Internet. It works just like **USMARC** for catalog records: the information and coding is the same, but it may work or look differently depending on each system, such as the differences in RLIN displays, OCLC displays, and NOTIS displays.

**HTML** has changed significantly since it was first introduced, and it is still changing.

This workshop will attempt to give a familiarity with basic HTML coding and tasks. There are a few practical exercises included. To complete these exercises, please open: *WordPad* and *Netscape*.

Everything in this workshop will refer to the HTML section in *Princeton's Cataloging Documentation* located at:

**<http://infoshare1.princeton.edu/katmandu/html/htmltoc.html>**

- You may also need to refer to the handout: *Basic Skills for HTML*.

## URL

Everything in HTML documents is based on transferring files. When you click on something, the computer retrieves files and reformats them in various ways. The files you retrieve are on another computer (called a *server*).

- A **server** is just like any other computer, except that it is connected to the Internet and has special software.

When you call a URL (*Uniform Resource Locator*), you are actually asking to retrieve a file, along with any other files associated with it (e.g. pictures). Your computer then takes these files, and the browser on your computer reformats them for you to see.

- Your computer is called the **client**.
- A **browser** is the program you use to search the Internet, e.g., **Netscape** or **Explorer**.

## What Makes a URL?

**<http://infoshare1.princeton.edu/katmandu/html/htmltoc.html>**

This is the URL for the **HTML Documentation Table of contents** from **Princeton's Cataloging Documentation**.

**http://** infoshare1.princeton.edu/katmandu/html/htmltoc.html

The first part of the URL tells the type of transfer. In this case, it is **http://** (*hypertext transfer protocol*), but it can also be **ftp://** (*file transfer protocol*) or **telnet://**, or other types.

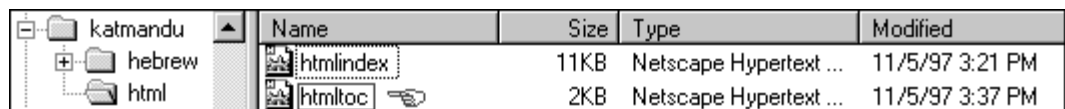
http:// **infoshare1.princeton.edu** /katmandu/html/htmltoc.html

Next comes the name of the server. Each server has a unique name assigned to it. In this case, the Princeton library server is called **infoshare1.princeton.edu**. This is not the only library server, for instance, there is also **libnt1.princeton.edu**.

We know that **infoshare1.princeton** is located at an educational institution because of the **.edu** attached to it. This part of the URL is called the *subdomain*. There are many other subdomains: **.com** (*commercial*), **.mil** (*military*), **.org** (*organization*). Some are based on countries: **.jp**, **.ca**, **.it** (*Japan, Canada, Italy*)

Next comes the file structure of the URL.

http://infoshare1.princeton.edu **katmandu/html/htmltoc.html**



Name	Size	Type	Modified
htmlindex	11KB	Netscape Hypertext ...	11/5/97 3:21 PM
htmltoc	2KB	Netscape Hypertext ...	11/5/97 3:37 PM

The slashes show changes to different **directories** (also called **folders**).

The final part of the file name **.html** does not display in *Windows 95* or *Windows NT*, but it **needs to be added** to the URL.

### **Special note: index.html**

There is a special file name which should be kept in mind: **index.html**. This filename is designated as a default file. This means that if there is no specific file indicated in the URL, the browser will automatically go to the one named **index.html**.

**http://infoshare1.princeton.edu/katmandu/html/**

This URL only gives the directory name (*html*). As a result, the browser will now automatically try to find the file named **index.html**.

As we can see above, there is no file named **index.html** and the browser would say that it could not find what you requested.

Web designers use this a lot, e.g. **http://www.princeton.edu/~reserve/**, but it is always **clearer** to add the entire name of the URL:

**http://www.princeton.edu/~reserve/index.html**

# Tags

## HTML coding

Within the HTML file, you can tell the computer coding by all the information in **angle brackets** <>. This is called an **element**. ( <a> is an **element**) Some of this coding is important for the computer and must be input. Other codes are important for the way the text appears on the screen, or for other actions when you enter text or click on something.

Tags can have **attributes** which can make further refinements. ( <a href *href* is an **attribute**) All of these attributes can be found in the documentation.

- In this workshop, all text that represent tags is in boxes.

## Codes which must be made

```
<html>
<head>
<title></title>
</head>
<body></body>
</html>
```

**These codes must appear** in the file for the browser to understand that it is an HTML file.

Most HTML coding is achieved by marking the **beginning** and **ending** of text. The beginning is shown by the code in plain angle brackets <???, while the end is shown by a slash </???, e.g. <table> </table>

Discussion of the above coding.

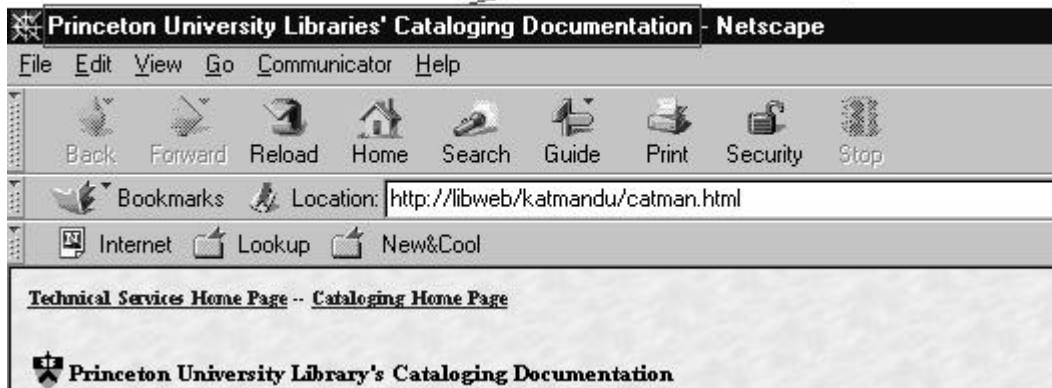
<html> - Tells the computer that this is the beginning of an HTML file.

<head> - Header. Important later for more complex coding.

<title></title> - The information that displays at the very top of the browser window. This is the information stored when you save a bookmark, and is also important for "search engines" (Webcrawler, Yahoo, etc.)

```
<title>Princeton University Libraries' Cataloging Documentation</title>
```

From the <title></title> area



</head> - End of the Header

**<body></body>** - Where the text of the document will be. This tag has several attributes:

BACKGROUND=URL, BGCOLOR=#RRGGBB, TEXT=#RRGGBB,  
LINK=#RRGGBB, VLINK=#RRGGBB, ALINK=#RRGGBB

This tells us that the tag **<body bgcolor="#FF0000">**

will give us a red background. Each color has a number, and there is a page giving the colors at:

**<http://infoshare1.princeton.edu/katmandu/html/color2.html>**

**</html>** - Tells the computer that this is the end of the HTML file.

## Tags That Affect What You See

In the **HTML Documentation Table of Contents**, the HTML tags are arranged in various ways.

### Tags which affect the text

#### Appearance

Most of these tags mark the text, which means that there is a beginning **<???.>** and end **</???.>**. Most make sense as you try them. For example, the **<h#></h#>** tags give varying sizes of fonts, depending on what number (1-7) you substitute for #.

**It is important to realize** that many tags do more than you would think. For example, the **<h#> </h#>** tags give a blank line before and after the text.

An alternative to **<h#>** is **<font size=+#>** and **<font size=-#>** which give the same sizes, but do not put in the blank lines before and after the text.

#### Example of **<h1>**

Please **<h1>help</h1>** me

Displays as:

Please  
**help**  
me

#### Example of **<font size>**

Please **<font size=+7>help</font>** me

Displays as:

Please **help** me

Another couple of important points to keep in mind are:

- **All empty space** is ignored by the browser.
- **The only way** to affect how the text looks is by putting in the tags.

For example, these will display as the first example above:

```
Please <h1>help</h1> me
```

```
Please <h1>help</h1> me
```

```
Please      <h1>          help      </h1>          me
```

```
Please  
                <h1>  
help  
                </h1>  
me
```

## ↳ Practice

- Open **WordPad** and **Netscape**..
- Go to the **HTML Documentation** copy the **Standard Template** and paste it into your Word Pad document.
- Type your name a few times and format it in different ways, based on the codes in the **Appearance** section of the **HTML Documentation**. Save it under the file name: **test.html** according to the instructions in **Basic Skills for HTML**.
- Open the file in **Netscape** and see what it looks like.
- Don't forget to **Reload** the page if you work on it.
- Give the page you just made a yellow background.

## Dividers

The Divider tags provide different methods of keeping the text from running together. Without these tags, all the text in a file will display as a single block of text. For example:

The Technical Services Department -- along with the departments of Administrative Services, Reference and Collection Development, Rare Books and Special Collections, and Special Libraries --is one of five major units which constitute the Princeton University Library. The Technical Services Department provides technical processing (acquisition, cataloging, binding and record management) for the vast majority of materials acquired for the Princeton collections, with the chief exceptions of federal and New Jersey State government documents, technical reports and ephemera. Specialized processing operations, complementing those provided by the Technical Services Department, are those of the Technical Services Section of the Gest Oriental Library and East Asian Collections, the Rare Books and Manuscripts Cataloging Team of the Rare Books and Special Collections Department, and the Documents Division of the Reference and Collection Development Department.

To break up this text to make it more readable, we must add **<br>** (*line break*) and **<p>** (*line break with a blank line*)

The Technical Services Department -- along with the departments of Administrative Services, Reference and Collection Development, Rare Books and Special Collections, and Special Libraries --is one of five major units which constitute the Princeton University Library.

**<br>**

The Technical Services Department provides technical processing (acquisition, cataloging, binding and record management) for the vast majority of materials acquired for the Princeton collections, with the chief exceptions of federal and New Jersey State government documents, technical reports and ephemera. **<p>**Specialized processing operations, complementing those provided by the Technical Services Department, are those of the Technical Services Section of the Gest Oriental Library and East Asian Collections, the Rare Books and Manuscripts Cataloging Team of the Rare Books and Special Collections Department, and the Documents Division of the Reference and Collection Development Department.

This displays as:

The Technical Services Department -- along with the departments of Administrative Services, Reference and Collection Development, Rare Books and Special Collections, and Special Libraries --is one of five major units which constitute the Princeton University Library.  
The Technical Services Department provides technical processing (acquisition, cataloging, binding and record management) for the vast majority of materials acquired for the Princeton collections, with the chief exceptions of federal and New Jersey State government documents, technical reports and ephemera.

Specialized processing operations, complementing those provided by the Technical Services Department, are those of the Technical Services Section of the Gest Oriental Library and East Asian Collections, the Rare Books and Manuscripts Cataloging Team of the Rare Books and Special Collections Department, and the Documents Division of the Reference and Collection Development Department.

## ↳ Practice

Copy the text marked **Example 1** in the documentation, and add it to the file you've already made. Break it in various places using the codes you find in the **Dividers** Section. You can change the format of different parts of the text by using the **Appearance** codes.

## Lists

Special codes for **lists** are also available. If you want to put in a bullet before a line or word, you must use **<li>**. To get the text to line up, you must precede this with **<ul>** (*Unnumbered list*) or **<ol>** (*Ordered list*).

**Example:**

```
<li>Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me.
```

Displays as:

- Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me.

Another way of displaying keeps the lines arranged. This is with **<ul></ul>**

```
<ul><li>Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me.</ul>
```

Displays as:

- Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me. Please help me.

Another way is to make a complete list with several `<li>`.

```
<ul><li>Please help me. <li>Please help me. <li>Please help me. <li>Please help me. <li>Please help me. <li>Please help me. <li>Please help me.</ul>
```

Displays as:

- Please help me.
- Please help me.
- Please help me.
- Please help me.
- Please help me.
- Please help me.
- Please help me.

By changing the attributes (e.g. `<li type=square>`) you can replace the dots with all manner of things: circles, squares, even images. With `<br>` instead of `<li>`, there is none at all.

If you want numbers instead instead of bullets, you can change the `<ul></ul>` to `<ol></ol>`, while the attributes can change to numbers, letters, roman numerals, etc.

## Practice

Copy the text marked **Example 2** in the documentation, and add it to the file you've already made. Use the section marked **Lists**.

- Make an unnumbered list, and change the dots to squares.
- Change it to numbered list, and change the numbers to small case roman numerals.

## Special characters

Certain characters will not display correctly in HTML, and a number of special codes have been made. These codes begin with an **& [Ampersand]** and end with a **;** [Semicolon]. Inside, there can be text or numbers, e.g.

`&#224;` displays à      `&#225;` displays á  
`&agrave;` displays à      `&aacute;` displays á

## Practice

Look at the codes and make the following sentence work in HTML.

How would you put in:

Les Françaises pendant la guerre de Cent Ans :‡b début du XIVe siècle-milieu du XVe siècle /‡c Jean Verdon.

You can copy this sentence from **Example 3**.

# Lay-out

## Tables

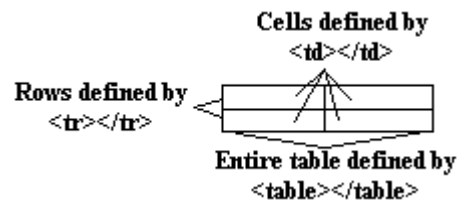
With the advent of the newer browsers, the `<table>` tag was implemented. The tags that make tables are more complicated than the other tags, but allow for a more precise placement of items on the computer screen. Before tables, it was impossible to make columns, or even place text or an image on the right side of the screen, but it can now be done through the correct use of tables.

There are several parts to a table.

- `<table></table>` defines the **table**, itself.
- `<tr></tr>` defines the **row** of a table.
- `<td></td>` or `<th></th>` defines an **individual cell** of a row.

The coding to make a table (2 x 2) is:

```
<table>
<tr><td></td><td></td></tr>
<tr><td></td><td></td></tr>
</table>
```



Each code has several attributes for size, color, and other properties. As you can see, text, or images can be placed in any one, or all, of the cells, and therefore, anywhere on the screen.

Individual cells can be defined. If you want a single cell to be bigger than the others, you can use *colspan* or *rowspan* attributes.

```
<table>
<tr><td></td><td></td></tr>
<tr><td colspan =2></td></tr>
</table>
```


```
<table>
<tr><td></td><td rowspan=2></td></tr>
<tr><td></td></tr>
</table>
```


Tables can also be included in other tables.

```
<table>
<tr><td>
<table><tr><td></td><td></td></tr></table>
</td><td></td></tr>
<tr><td></td><td></td></tr>
</table>
```

<table border="1"> <tr> <td></td> <td></td> </tr> </table>			

The information in each cell can be aligned, e.g. `<td align=right>`

## Practice

Make the following words look this way in Netscape:

Beginning	End
-----------	-----

## Frames

Frames are a way of allowing the browser to show more than one HTML page at a time.

Pages based on frames are made up of several files.

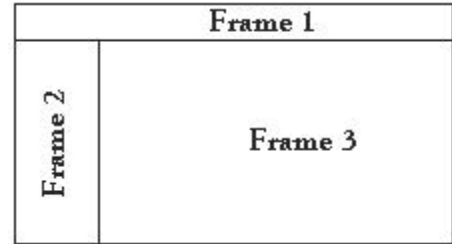
The first file defines the frame itself, but carries no formatted information. It then directs the browser to load others files into each frame. In this illustration, four HTML files make up this frame.

File #1 defines how the frames are arranged and display on the screen.

File #2 displays in the area marked Frame 1.

File #3 displays in the area marked Frame 2.

File #4 displays in the area marked Frame 3.



The designer can specify that if the user clicks on a link in Frame 1, something will happen in Frame 2 or 3, and vice versa.

Creating and maintaining frames can become exceedingly complex, and the current move in web designing is to stay away from frames.

### `<pre>`/`</pre>` – Preformatted text

A very simple way of creating an HTML page is with `<pre>` `</pre>`. This is a way of allowing the tabs and line breaks from a word processor to appear in an HTML page.

The problem with `<pre>` is that it's not very flexible and text in `<pre>` looks like a typewriter.

**The Technical Services Department -- along with the departments of Administrative Services, Reference and Collection Development, Rare Books and Special Collections, and Special Libraries -- is one of five major units which constitute the Princeton University Library.**

`<pre>` was used extensively before the `<table>` tag, since it was the only way to get text in columns. On the downside, text in `<pre>` tends to scroll off the right side of the screen on smaller monitors.

- It is therefore best to avoid the seemingly easy solutions of `<pre>`.

## Computer Commands

Computer commands can be grouped into those tags which control the look of the text, and those which control the tasks of the computer.

### Appearance

#### Graphics, Backgrounds and Colors

Although a picture may appear to be part of a HTML page, it is not. All pictures are separate, special files. The codes direct the browser to look for the image files. These files are downloaded and then included into the correct position on the computer screen.

The tag for loading an image is:

```

```

Again, there are several attributes to this tag. It can be aligned higher or lower than a line of text. It can let text flow around it as it is aligned to the left, right or center of a paragraph.

Browsers can currently display two types of images: GIF (*Graphic Interchange Format*) and JPEG (*Joint Photographic Experts Group*)

#### ↳ Practice

Somewhere in your page, put in the following image

```
http://imagecat1.princeton.edu/images/card2.gif
```

#### Links

Links are the main purpose of hypertext. It allows the user to go directly to another part of the document, or to a completely different document with only a mouse click. To make a link, text or an image must be "anchored" (which means putting it in the `<a></a>` tags) and use various attributes.

For example, to make a link from the word *Now* to the HTML Documentation Table of Contents:

```
<a href="http://infoshare1.princeton.edu/katmandu/html/htmltoc.html">Now</a>
```

This link will go to the top of the HTML Table of Contents.

Sometimes, you may prefer to go to particular information inside a document. The same link will work, but it needs something more. For example, perhaps you want to link directly to the **HTML Copy&Paste** section, instead of the top of the file.

Two things are needed:

1. There needs to be a way of letting the computer know exactly where to stop in the page (making an **anchor**).
2. The link itself, needs to be modified.

In this case, in the **Table of Contents** page, another anchor needs to be entered to let the computer know where to stop.

```
<a name="HTML">HTML Copy & Paste</a>
```

The original link is modified to:

```
<a href="http://infoshare1.princeton.edu/katmandu/html/htmltoc.html#HTML">Now</a>
```

The # tells the computer to go to the page, and then look for the anchor "HTML".  
Some other different types of links:

### Absolute links

```
<a href="http://infoshare1.princeton.edu/katmandu/html/htmltoc.html">HTML Table of Contents</a>
```

The URL above is an **absolute link**. This means that no matter where you are in the world, the URL will always work.

### Relative links

Another way to make the link is with a *relative link*. This works by putting in "./" or "../" the computer looks up or down within the file structure, itself. For example, if you were looking at a file in "katmandu" above, and wanted to make a



Name	Size	Type	Modified
htmlindex	11KB	Netscape Hypertext ...	11/5/97 3:21 PM
htmltoc	2KB	Netscape Hypertext ...	11/5/97 3:37 PM

link to the file *htmltoc.html* in the folder *html*, you could either make the absolute link above, or shorten it with

```
<a href="./html/htmltoc.html">HTML Table of Contents</a>
```

Relative links are more efficient than absolute links, but they can also be more difficult to interpret.

### Practice

1. Make an *absolute link* from your name to the Main page of the **Cataloging Documentation**.
2. Save an extra copy of the file you have made under the name: **test2.html**
3. Make an anchor to first item of the list you created earlier.
4. Make a *relative link* from your name in the file **test.html** to the anchor you made in **test2.html**.

## Actions

The last HTML tags are those which mostly cannot be seen. These are used when you enter information into the computer, such as mail, or doing a web search.

### Forms

The `<form></form>` with various attributes tag creates **interactive pages**. With forms, you can decide if someone can input information (by typing in text,

```
<input type="text"> [allows the user to type information]
```

clicking on boxes

**<input type="checkbox">** *[creates a box to click on]*

and what it looks like

**<input maxlength=?>** *[how much text can be input]*

- Information in forms must work with another program somewhere on the web.

## Miscellaneous

The last tags are mainly for informational purposes. Comments that will appear only in the coding, certain indexes, and so on. Much of the information used in these codes is used for advertising purposes (which program created the file), or gives various comments concerning copyright or credits. This information is normally put into a comment and is formatted in the following manner:

```
<!--***-->
```

There is other information which relates to the file. Some codes can give a referring URL

```
<base href="url">
```

which directs the computer to refer all relative links to this URL.

There is also **<isindex>** which can create an index to the information in the file.

## Dynamic HTML

The latest version of HTML is called Dynamic HTML. In this version of HTML, the designer can determine how text is displayed. For example,

```
<HTML>
<HEAD>
<TITLE>Dynamic HTML Example</TITLE>
<STYLE>
H1 { font-size: 26pt; color: red }
H2 { font-size: 18pt; color: blue }
</STYLE>
</HEAD>
```

The new **<style>** **</style>** tags go inside the header and can be used to change the display of almost any tag. In this case, any text coded **<h1>****</h1>** will display as extra-large in red, while **<h2>****</h2>** will display smaller in blue.

The new HTML coding works only in Netscape's Communicator 4, and Microsoft's Explorer 4. All earlier coding still works in these browsers, while the earlier browsers will ignore the new codes.

## Final Notes

A final consideration is one that drives many designers crazy: the same HTML file will **display differently** on different browsers. This is similar to the same USMARC records displaying differently in RLIN, OCLC, NOTIS, HORIZON, WorldCat, and other catalogs. Some of these differences are decisions that are made by the browser developers, but other differences lie in the lesser or greater power of the browsers, themselves.

For example, Netscape 2 cannot deal with the same tags as Netscape 3, which also cannot deal with the same tags as Netscape Communicator 4. None of the versions of Microsoft's Explorer work the same way as Netscape products. Additionally, the way each product works on a separate type of computer can vary tremendously. Netscape on the Macintosh will display and work differently than on a PC, and still differently from WindowsNT.

The only way to know how your files will display on any browser is to try them on different machines and and different browsers.