
CHAPTER 3

HTML: How to Make a Web Page from Scratch

Now that you have an operational server, you're ready to begin creating web pages. Almost all web pages are coded in HTML, so we'll discuss how to create basic HTML pages in this chapter. Specifically, I'll show you how to create a simple page, format text (i.e., use bold, italics, line breaks), insert images, and use hyperlinks.

We'll be coding our HTML "by hand." There are a number of programs available, such as Microsoft's FrontPage, Macromedia's Dreamweaver, or Netscape's Composer, that allow you to create a web page as if you were working with a word processing program. If you want to place some text in bold, for example, you simply highlight the text and click on the "bold" button. With these WYSIWYG (What You See Is What You Get) programs, the way the page looks on your screen is the way the final product is supposed to appear. The HTML code is automatically created in the background, so you don't need to know anything about HTML to create beautiful web pages.

I do not have anything against WYSIWYG programs, and I often use Dreamweaver to manage most of my personal and teaching-related web pages. However, for research purposes, you will need to code most of your HTML the old fashioned way for two reasons. First, for much of our CGI scripting, we'll need to create new web pages on-the-fly. In other words, the server will construct a web page that may be tailored to a specific user, and you'll need to program the generic HTML code into the CGI script in order for this to work. Second, programs such as FrontPage often insert a lot of useless, and sometimes obstructive, HTML code into your page. Moreover, as you revise pages over time in Dreamweaver, the HTML code is not efficiently updated. This will lead to formatting problems that simply cannot be corrected without explicit knowledge of HTML. You'll want to have *full* control over the pages you create for research purposes, and these programs will not give you the control you need.

It is easiest to write and edit HTML code in a simple text editor. Although you can use popular word processing programs, such as Microsoft Word, to edit text (*.txt) files, it is not unusual for MS Word to reformat your text against your will. Therefore, I discourage you from using MS Word for creating HTML files. I also discourage you from using the standard Windows text editor, Notepad, because it has a nasty habit of adding .txt as an extension to all of your files, even if you instruct the program to save the file with an alternative extension.

In this book, I will encourage you to use **1st Page**—a program I asked you to download in the previous chapter. This program has several desirable features: (a) you can keep

multiple files open at once, and move between them easily by simply clicking a button; (b) the program is “code smart” and will use different colors for various HTML (and Perl) codes, thereby making editing and trouble shooting much easier; (c) the program includes a preview pane, which allows you to see how your HTML code will appear in a browser; (d) like Dreamweaver, the program will do some of the HTML coding for you via drop-down menus, but in a way that is more easily controlled; and (e) the program includes a useful spell-checker.

What is required of all HTML pages

Open 1st Page. To open a new document, click on File, then select the following options: New → Blank Document. A blank page should appear in your 1st Page editor window.

All HTML pages require certain, unchanging default commands.

```
<HTML>
<BODY>

</BODY>
</HTML>
```

The things in brackets are called tags. **Tags** are pieces of code that are used to modify the contents of a web page. Many, but not all, tags require a closing tag, which contains a backslash (e.g., `</HTML>`). Throughout this book, I will capitalize the commands within tags to help separate them from the rest of the text. It is not necessary that you do this, but you may find it helpful for organizing your code.

The first tag, `<HTML>`, tells the browser that it is about to receive HTML information. The close tag, `</HTML>` tells the browser that the HTML code is complete. The `BODY` tag isn't necessary unless you want to alter certain page attributes, such as the background color. We'll probably want to change these attributes on occasion, so I've included the `BODY` tag above.

Go ahead and type these basic tags into your new document. We'll build on these basics in the examples that follow.

Note. Technically, the `<HTML>` tags are not necessary for most web applications because browsers often assume that they are receiving HTML code. Nonetheless, it is good form to include these tags in your pages.

Creating Text

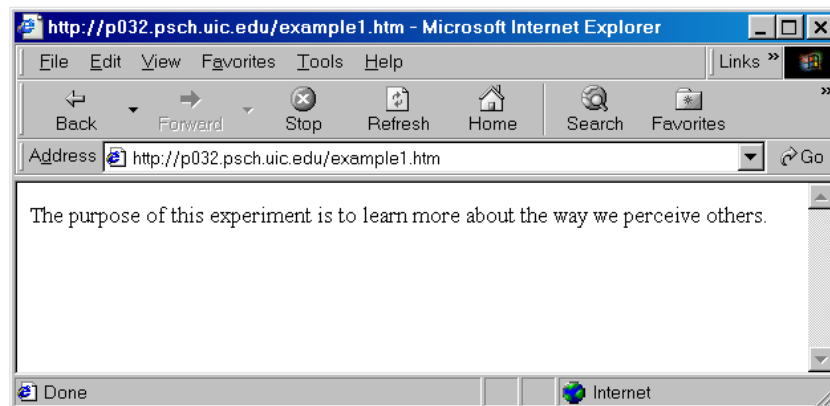
To insert some text into your web page, believe it or not, all you have to do is type the text that you would like to appear when someone views your web page.

For example, let's say I wanted to create a home page for my study that listed the name of the study and explained its purpose. The following simple page will do the trick.

```
<HTML>
<BODY>
The purpose of this experiment is to learn more about the way we
perceive others.
</BODY>
</HTML>
```

Try typing this into your document, and save the file as **example1.htm**. You'll want to save this file (and all HTML documents we create) to a specific directory, specifically, the one called **htdocs**. If you went with the defaults when you were installing Apache, this folder should be located at C:/Program Files/Apache/Apache Group/Apache2/htdocs.

Let's see how the page looks. Open your browser and type in your site's URL, followed by example1.htm. For example, my example1.htm page will be located at <http://P032.psch.uic.edu/example1.htm>. Here is what you should see:



Note. When writing HTML code, it is often useful to have your browser window open. That way you can see what your page looks like as you create it. To do this, however, you'll need to "refresh" or "reload" you page each time you edit it. As you may know, you browser window is configured to keep a copy of the web page stored in memory. Thus, once you've loaded a page, your computer will not continue to reload the contents continually. If the code for the page changes on the server, your browser won't know that unless you hit the "reload" button to see revised pages.

Note. One of the useful features of 1st Page is that it has a preview pane built into it, so you don't necessarily need to open your browser to see what the page will look like. Simply click the "preview" tab near the top of the window in order to switch between the HTML code and the browser-like interpretation of that code.

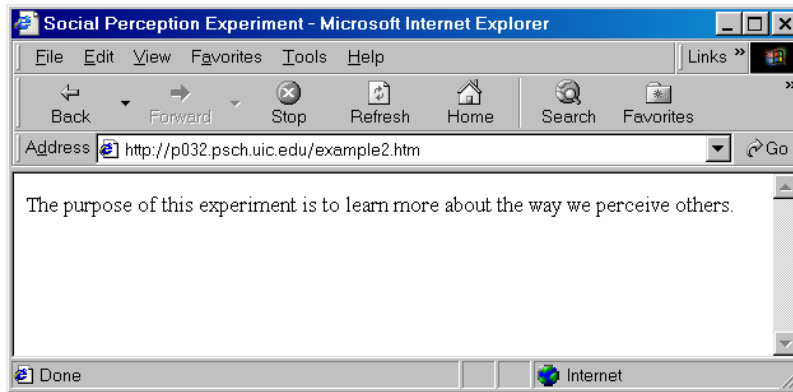
Page Properties

You can easily manipulate certain features of your web page, such as the background color and the title of the page (specifically, the title that appears on the browser's bar). The title of the page is critical because most search engines use this information to index web pages. You should include a title for your page that is as descriptive as possible. Think of the title as an advertisement for your page. If you want people doing web searches on terms such as "personality tests," you might want to title your page "Free Psychology Experiments and Personality Tests." If you don't want people to discover your page via a simple web search (e.g., the page represents a specific condition in an experimental design that you don't want people to see unless they are doing the experiment per se), then you should consider using no title or a meaningless title.

To create a title, simply use the <TITLE> tag. Here is an example:

```
<HTML>
<TITLE>Social Perception Experiment</TITLE>
<BODY>
The purpose of this experiment is to learn more about the way we
perceive others.
</BODY>
</HTML>
```

Notice below that the title of the experiment now appears in the top of the browser bar.



You may also want to modify the background color of your web page. By default, browser will often assume the color is white or gray. If you want to specify the background color explicitly, you can do so within the `<BODY>` tag, using the `BGCOLOR` command:

```
<BODY BGCOLOR = 'blue'>
```

Note. Although it is common to use double quotes around values that are assigned in tags, we will use single quotes. The reason for using single quotes is that when we begin to insert HTML code in our CGI programs, the CGI programs will often reserve the double quotes for specific purposes. If we adopt the convention of using single quotes from the outset, we will not run into problems later.

For many colors, you can simply insert common color names (e.g., 'white', 'blue', 'red'), as we did in this example. If you want more control over the colors you use, you can enter the **hexadecimal** code for various colors. 'ffffff', for example, is one way of representing white in hexadecimal code. For a thorough discussion of hexadecimal codes and color mixtures, I encourage you to visit the following web site:
http://www.w3schools.com/html/html_colors.asp.

You may also specify the color of your text in the `<BODY>` tag using the `TEXT` command. (The default text color is black on most browsers.)

```
<BODY BGCOLOR = 'blue' TEXT = 'white'>
```

Formatting Text

In many instances you will want to modify the appearance of some of your text. For example, you may want to place some words in bold, italicize others, or use different font sizes throughout a document. Here are some commands for some common formatting alternatives.

Bold

Use the `` tag to place your text in boldface.

```
I want the following words <B>to appear in boldface type</B>, but not these.
```

Italics

Use the `<I>` tag to italicize text.

```
I would like to <I>italicize</I> some words.
```

Font

To change the font face, use the `` tag with the `FACE` attribute.

```
Some words look nice in <FONT FACE = 'arial'>arial</FONT>, whereas others look better in <FONT FACE = 'courier'>courier.</FONT>
```

Text size

Use the `` tag with the `SIZE` attribute to alter the size of your text.

```
Some words are <FONT SIZE = '5'>bigger</FONT> than others.
```

Note. If we want to revert to the previous font settings (which were the browser defaults here), we have to use close the font tag: ``.

Text color

If we want to set some text in a different color than the rest of the text (specified in the BODY tag), we can use the `` tag with the COLOR attribute.

```
Some words are <FONT COLOR= 'red'>red</FONT> more than others.
```

Line spacing

Even if you type your text on different lines in your text editor, the browser will not display the text on different lines unless you specifically ask it to do so. The tag for a line break is `
`.

```
I want this to appear on one line. <BR>
And this to appear on the next. <BR>
<BR>
<BR>
This line will be separated from the rest by two blank lines.
```

Note. There are also paragraph tags (i.e., `<P>` and `</P>`) that allow you to separate lines of text into discrete paragraphs. We will not use that tag here because the same result can be achieved with the `
` tag.

Centering text

By default, the text will be “flush left”—aligned against the left-hand side of the page. If you would like to center certain portions of the page, you can use the `<CENTER>` tag.

```
<CENTER>
Results
</CENTER>
The age and sex breakdown for the sample is presented in Table 1.
```

Blank space

One interesting feature of HTML is that you cannot insert two blank spaces into your text unless you specifically request it. Thus, while you can insert a blank space after the end of a sentence by simply using the space bar, you cannot hit the spacebar twice to insert two blank spaces.

To insert more than one-character's worth of blank space, you'll need to use a special command: ** **. Here is how it is used:

```
Two blank spaces follow this period.&nbsp; &nbsp;Nice, isn't it?
```

Another way to accomplish the same feat, based on the idea that we can get away with one space by using the spacebar:

```
Two blank spaces follow this period. &nbsp; Nice, isn't it?
```

Let's combine some of this stuff into a single file and see what it looks like. Copy the following HTML code to a blank document and save it as **example3.htm** in your htdocs directory.

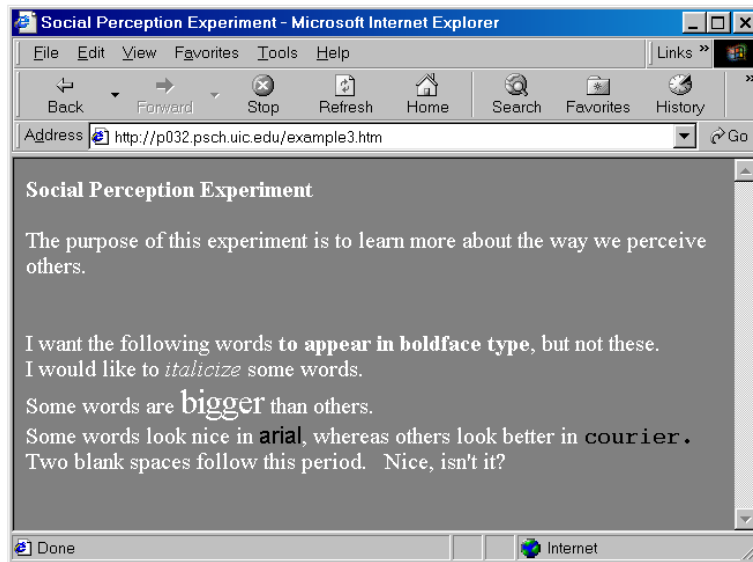
```
<HTML>
<TITLE>Social Perception Experiment</TITLE>
<BODY BGCOLOR='gray' TEXT='white'>

<B>Social Perception Experiment </B><BR><BR>

The purpose of this experiment is to learn more about the way we
perceive others. <BR><BR><BR>
I want the following words <B> to appear in boldface type</B>, but not
these.<BR>

I would like to <I> italicize</I> some words.<BR>
Some words are <FONT SIZE = '5'> bigger</FONT> than others.<BR>
Some words look nice in <FONT FACE = 'arial' COLOR = 'black'>
arial</FONT>, whereas others look better in <FONT FACE = 'courier'
COLOR = 'black'> courier.</FONT>

Two blank spaces follow this period.&nbsp; &nbsp;Nice, isn't it?
</BODY>
</HTML>
```



Inserting an image

To place an image in your document, you'll need to do one of two things. First, if you want to use an image that already exists somewhere on the WWW, you'll need to know the full URL for the image. Alternatively, if you want to place a new image on the Internet (e.g., an image you've scanned or created yourself), you'll need to place that image in your **htdocs** folder. The URL for the image then becomes the root URL for your site plus the image name. For example, if I wanted to use an image called `figure1.jpg`, I would save it to my **htdocs** folder, and the URL for the image would become `http://P034.psch.uic.edu/figure1.jpg`.

Once you know the URL for the image, all you need to do is insert an image tag into your document. Here is an example:

```
Please imagine what this person's personality is like.<BR> <BR>
<IMG SRC = 'http://P032.psch.uic.edu/figure1.jpg' >
```

Please imagine what this person's personality is like.



If you specify the tag in this way, the image will appear at its *natural* size or dimensions (i.e., the size of the image itself). If you want to adjust the size of the image, you can do so using the size attributes (HEIGHT and WIDTH) of the IMG tag. The metric is pixels, the itty-bitty pieces of light on your screen. Here is an example:

```
Which of the following faces seems more friendly?<BR>
<IMG SRC = 'http://P032.psch.uic.edu/figure1.jpg' >
<IMG SRC = 'http://P032.psch.uic.edu/figure1.jpg' WIDTH = '280' HEIGHT
= '171'>
```

Which of the following faces seems more friendly?



It might be useful to use the width and height attributes when you have visual stimuli that you would like to present in the same dimensions, but the graphics files are not *exactly* same dimensions. As long as the original files are not too different in their dimensions, this method will allow you to present the stimuli in a standard size without distortion.

As a general rule, it is better to save the image using the desired dimensions than to adjust the dimensions for which the image is presented. Images can take a while for people to download, so you don't want your users to spend a lot of time downloading large image files that you plan on downsizing in the presentation mode.

Creating hyperlinks

One of the useful features of web pages is the ability to click on a hyperlink in order to instantly jump to another web page. To create a hyperlink, you can use the <A> tag. Here is an illustration:

```
For more information on UIC, please click <A HREF =  
'http://www.uic.edu'> here </A>.
```

The URL that you would like to refer your user to when he or she clicks on the link is specified by the HREF attribute.

Here is what this page would look like in the browser.

For more information on UIC, please click [here](http://www.uic.edu) .

Notice that the text that we wanted to turn into a hyperlink falls between the opening tag and the closing tag. Notice also that the HREF attribute is used to specify the full web address for the link. If the link goes to another file within the same directory as the current page, the full extension will not be necessary. Mistakes can be made, however, so, until you're feeling more HTML savvy, play it safe and always specify the full URL following the HREF command.

Now, when a user clicks on the word "here", he or she will be transported to the UIC home page.

Using images as links

There may be times when you find it useful to use images instead of text as links. To do this, you can simply insert an image, instead of text, in between the hyperlink tags. Here is an example:

```
Please click on the image below for more information on UIC. <BR>  
<A HREF = 'http://www.uic.edu'>  
<IMG SRC = 'http://www.uic.edu/depts/psch/social/Uiclogo.jpg' >  
</A>.
```

Please click on the image below for more information on UIC.



Using Tables Effectively

Oftentimes, you'll find it useful to place images, text, or response scales within a table. Using tables to format the placement of your text or images is a highly efficient technique (i.e., it eats up less code, memory, and time than other approaches). There three tags involved in creating a simple table: (a) a `<TABLE>` tag, used to begin and end the table contexts; (b) a row tag, `<TR>`, used to begin and end a row; and (c) column tag, `<TD>`, used to begin and end the contents of a column within a row. Here is an example:

```
<TABLE>
<TR>
<TD>
first row, first column
</TD>
<TD>
first row, second column
</TD>
</TR>
<TR>
<TD>
second row, first column
</TD>
<TD>
second row, second column
</TD>
</TR>
</TABLE>
```

Notice that the flow of commands is row by column. In other words, we first begin a row using `<TR>`, then create the columns *within that row* using `<TD>` and `</TD>`, then we end the row with `</TR>` and begin again with the next row. It is noteworthy that we didn't need to tell the browser how many rows or columns we were going to use; we simply used as many as we needed. This feature will turn out to be very convenient later when we discuss ways to customize experimental feedback for subjects and we don't know in advance how many rows will be necessary.

Here is what the code will look like in the browser window:

first row, first column first row, second column
second row, first column second row, second column

Notice that, by default, the columns are as wide as the contents of the largest cell.

It is not necessary that each of command be placed on a separate line. (In fact, an entire HTML file could be written on a single line, if desired!) Here is the way I would typically format the previous set of commands in my text editor:

```
<TABLE>
<TR>
<TD>first row, first column</TD>
<TD>first row, second column</TD>
</TR>
<TR>
<TD>second row, first column</TD>
<TD>second row, second column</TD>
</TR>
</TABLE>
```

The `<TABLE>` tag has a number of attributes that allow you to alter the appearance of the table in ways that may be useful. For example, you can create a border for the table by adding the `BORDER` attribute to the `<TABLE>` tag. Also, you can alter the color of the border using the `BORDERCOLOR` attribute. You can adjust the spacing between cells and the padding between cells by adding the `CELLSPACING` and `CELLPADDING` attributes to the `<TABLE>` tag. Here is an example of how some of these attributes affect the appearance of a table. I encourage you to experiment with the various options.

```
<TABLE BORDER CELLSPACING = 2>
<TR>
<TD>first row, first column</TD>
<TD>first row, second column</TD>
</TR>
<TR>
<TD>second row, first column</TD>
<TD>second row, second column</TD>
</TR>
</TABLE>
```

first row, first column	first row, second column
second row, first column	second row, second column

There will be situations in which you want a particular row of the table to span one or more columns. You can span both columns and rows by using the COLSPAN and ROWSPAN attributes within a <TD> or <TR> tag, respectively. Here is an example.

```
<TABLE BORDER CELLSPACING = 2>
<TR>
<TD COLSPAN='2'>Factor A</TD>
</TR>
<TR>
<TD>Level 1</TD>
<TD>Level 2</TD>
</TR>
<TR>
<TD>4.50</TD>
<TD>2.46</TD>
</TR>
<TR>
<TD>8.7</TD>
<TD>5.6</TD>
</TR>
</TABLE>
```

Factor A	
Level 1	Level 2
4.50	2.46
8.7	5.6

For more information on the various ways you can create and format tables, please visit the following link: http://www.w3schools.com/html/html_tables.asp.

Note. Unfortunately, the various text attributes that you define in an HTML document (e.g., font face, font color) do not apply to the cells within a table. If you don't want to use the browser defaults for elements within a table, you'll need to respectively these attributes within each cell of a table. I will illustrate how to do this in the next example.

Bringing It All Together

Okay, now let's use these basic tools to create a home page for your web site. The purpose of this page will be to introduce visitors to the kinds of research you're conducting. Although we haven't created any on-line research studies yet, I'll create links to some of my on-line studies, just for the purpose of illustration. You may want to copy this code and modify the contents to convey information about the kinds of research you plan to conduct on-line.

In this example, we'll illustrate the use of the TABLE tag to format text. We'll also use links and images, as well as different kinds of fonts and text sizes. You may or may not want all these tweaks on your own site, but I want to show you how these commands work.

```
<HTML>
<TITLE> Online Attachment Style Survey </TITLE>
<BODY>

<TABLE>
<TR>
<TD BGCOLOR = '333366'>
<IMG SRC = 'http://p032.psch.uic.edu/relationalstructures.jpg' >
</TD>

<TD><FONT FACE = 'Times' SIZE = '5'>Online Attachment Style Surveys
</FONT><BR><BR></TD>
</TR>

<TR>
<TD BGCOLOR = '333366'></TD>

<TD>
<FONT FACE = 'arial' SIZE = '2'>
The objective of this site is to provide up-to-date links to
interactive web surveys on adult attachment styles. Each of these
interactive questionnaires take less than 10 minutes to complete, and
will provide you with immediate feedback on some aspect of your
interpersonal style.
<BR><BR><BR>

<A HREF = 'http://P034.psch.uic.edu/crq.htm'><B>Adult Attachment
Style</B></A><BR><BR>
People differ considerably in how secure or insecure they feel in their
close relationships. This 7-minute interactive questionnaire is
designed to assess how secure or insecure you feel in your close
relationships. When you are finished answering the questions, the
program will tell you about your attachment style.
<BR><BR><BR>

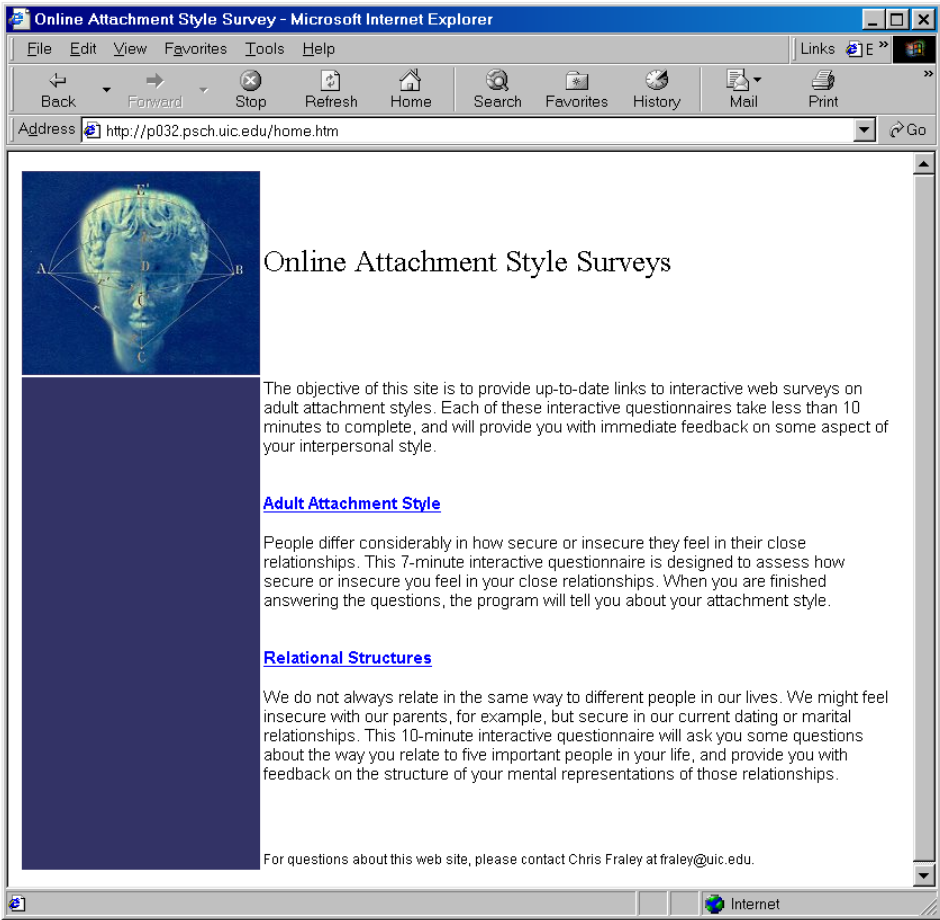
<A HREF =
'http://p034.psch.uic.edu/Claudia/relstructures.htm'><B>Relational
Structures </B></A><BR><BR>
```

```
We do not always relate in the same way to different people in our
lives. We might feel insecure with our parents, for example, but
secure in our current dating or marital relationships. This 10-minute
interactive questionnaire will ask you some questions about the way you
relate to five important people in your life, and provide you with
feedback on the structure of your mental representations of those
relationships.
</FONT>
<BR><BR><BR><BR>

<FONT FACE = 'arial' SIZE = '1'>
For questions about this web site, please contact Chris Fraley at
fraley@uic.edu.</FONT>
</TD>
</TR>

</BODY>
</HTML>
```

Copy this code to a blank document in your text editor, modify it accordingly, and save it as **home.htm** in your htdocs directory. Next, enter the URL for the page in the address bar of your web browser. On my server, the appropriate URL is <http://P032.psch.uic.edu/home.htm>. Here is what the page should look like:



Note. There are some files that are handled in special ways by the server software. For example, if you were to type the base URL for your web site into your address bar, you will see the Apache page that we noted in the previous chapter. Although the name of the file was not listed in the URL, the file does have a name: **index.html**. Files with the name `index.htm` are automatically displayed by Apache when a user specifies only a directory and not a specific file. If you would like your `home.htm` page to be the default page that is displayed when a user enters in the root of your URL, you'll need to do two things. First, you'll need to delete all the files in your `htdocs` directory that have "index" in their name. Second, you'll need to rename your `home.htm` file to `index.html`. After doing this, you should see your new page by default when you enter the root URL (e.g., <http://P032.psych.uic.edu/>) in the browser address bar.

In the next chapter, we'll explore HTML forms—ways of obtaining input or responses from the person using the page.