

oving forward into deeper realms of JavaScript, this month we will be discussing how to further extend the capabilities of the active menu for a website. The last time the topic of an active menu arose, we looked at ways to make the menu items stand out when the mouse moves over them. The next step in our progressive battle toward the perfect website navigation bar is learning some tricks to make our navigation bar as consistent as possible.

## eb Designer: A Job in Duality

There is one issue in designing content for the Internet which every single HTML programmer must face: where do I get the graphics? This is why it is important for each and every person who wishes to do web page development to also be able to design graphics. Now, I'm not talking visually-stunning three-dimensional images like a professional graphics artist would render. But every web developer should be able to create some simple buttons with text and/or icons. Every web developer should be able to handle the creation of the graphical elements of his site's navigation interface. So as a web designer, your job description implicitly contains the titles of HTML Programmer and Graphic Designer.

## etting Oneself Framed

No, we're not talking about some sort of Scooby-Doo mystery with a malaccused butler. We're talking about that wonderful HTML paradigm called frames. Frames are used to divide the visible area of a browser into

separate panes. Each pane can have its own HTML source file embedded within it. Up until now, every page we've talked about in Toolbox has been a single file of HTML code. With frames, we have the ability to go beyond the one-file-per-page layout. Consider, for example, if we divide the visible area of our browser into two frames, a left and a right area. We could then use the right-hand area for the site content — the actual pages of information we wish to display. That leaves the left-hand area for use as a site navigation bar and various information which pertains to the entire site.

Take, for instance, the navigation bar shown at the right. This bar remains constant across all of the pages on the site. In fact, users can easily find which page they are currently viewing just by checking which of the buttons in the navigation bar is depressed. It is simple visual clues like this that will make your website easier to navigate.

Not only does the left-hand frame contain navigation buttons, but it also contains the company logo at the top. The frame is used not only as a site-specific navigation aid, but also as a reminder of whose page is being viewed. In addition to the logo, there are two links to the websites for the companies who produce the tools used in designing the page. Directly below the content creators can be found links to other sites which have some pertinence to the site.

The sky's the limit when you're adding items to the left-hand frame as a navigation bar. You may want to add links to the personal web pages of the individuals who created the site. Another shoo-in is a hit-counter — what better place to put a count of how many people have accessed your page than in an area that the user will see on each page of your site? Other possibilities include a "last modified on" line of text and/or copyright information for the site.

## here to Start

The starting point for creating such a site is specifying how the frames for your page should be arranged visually. In the past, all of our code for our pages have gone into one file — be prepared to now make the step to using sets of files to represent the individual segments of a single page. The first file which must be created is a file which tells the browser what files need to be loaded into the browser to form the complete page:

```
<FRAMESET COLS="200,*" FRAMEBORDER=0>
<NOFRAMES>
<BODY BGCOLOR="#FFFFFF">
<P>This page is designed to be viewed by a browser which supports Netscape's Frames extension.
Your browser does not support the Frames extension.</P>
You should go to a <A HREF="main.html">page without frames</A>.<P>
<IMG SRC="buttons/line.gif" WIDTH="100%" HEIGHT=4>
</BODY>
</NOFRAMES>
  <FRAME SRC="menu.html" name="menu" SCROLLING=NO>
  <FRAME SRC="main.html" name="content">
</FRAMESET>
```

Let's dissect the code. First we see the `<FRAMESET>` tag. This tag is used to let the browser know that it should create some frames. Within the tag we see two parameters. The first parameter, `COLS`, specifies how many columns should be created and how wide they should be. The browser's display is divided into as many frames as there are width specifications in the `COLS` parameter. In the code listing above, for example, there are two column-type frames created, one which is 200 pixels wide and a second which uses the remainder of the browser's width (the asterisk acts as a sort of wildcard). Using the wildcard character basically tells the browser to divide up the available space proportionally. Setting the `COLS` parameter to `"200,*"` would effectively divide the browser into three column-type frames, the left-most being 200 pixels wide and the remaining two each being 50% of the space not used by the first frame. Setting the `COLS` parameter to `"*,*,*"` would create three column-type frames, each of which is 33% of the horizontal size of the browser. The `FRAMEBORDER` tag specifies whether or not the browser should visually indicate the divisions between frames.

The next set of tags specify what a browser should do if it does not support frames. You really should put these tags into your page, since some people may still be using text-based browsers or merely much older versions of Netscape. In a browser which supports frames, the `<NOFRAMES>` tag is skipped, thus the information between `<NOFRAMES>` and `</NOFRAMES>` is not displayed. Typically, the code between the tags simply tells the user that this page is meant to be viewed as frames — including a link to a page

designed to be viewed without frames can also be provided. Generally it is a good idea to design pages which can be viewed with or without the aid of frames — frames should be looked upon as an extension to the regular properties of your web pages.

Finally, a set of `<FRAME>` tags appear which specify what files are to be used as the source for the frames you've created. You should have the same number of `<FRAME>` tags as you had column width specifications in the `COLS` parameter. There are several parameters for the `<FRAME>` tag with which you must be concerned. First there is the `src` parameter, which is simply the URL for the file of HTML code which should be placed in the frame. Additionally, each frame can be given a name — the frame name will come in handy later in our programming efforts. Since our page has two column-type frames with the left-hand frame being our navigation area and the right-hand frame our content area, we will name them "menu" and "content," respectively. Finally, we can specify whether or not the frame should have scroll bars using the `SCROLLING` parameter. The entire segment is terminated by the `</FRAMESET>` tag.

## or Next Time...

Now you know the basics to defining frames. While that is the first step in creating a global navigation bar for your site, you can also use your new knowledge to create regular pages which use frames. For instance, imagine a website which presents two versions of the same document side-by-side, one in German and the other in English. Students who are learning the German language can compare and contrast the two documents. Remember, web pages are only limited by the programmer's imagination and skills. And hopefully, as you're going through your career as a web designer you're picking up plenty of helpful, new skills.

## Feedback...Please?

I haven't been hearing much from those of you out there in cyberspace lately. If you have any ideas for future columns, critiques of current or past columns, or a really funny Clinton joke that I may not have heard yet, email me!

Jeff Frey

[jeff@applewizards.net](mailto:jeff@applewizards.net)

<http://applewizards.net/>