

HTML / PERL INSTRUCTIONS

Here are various bits of HTML code and PERL inserts you can use to build web pages. The BLUE comments are background information (remove them from your web page). The RED notes are specific additions you need to make these work.

Download and print this as a handy reference while working on your web site.

SETTING UP YOUR WEB SITE:

Before you start building pages, you need to set up some internal structure on the server. You will need the following folders:

1IMAGES - this is where all your images go.

CGI-BIN - All your CGI, SSI and related material goes in here. Your SERVER will install this automatically when you subscribe.

RANDTEXT - Within the CGI-BIN you may want a RANDTEXT folder to hold various rotating 'SSI includes' in order to keep them handy. See below for further details.

1DOWNLOADS - all downloadable files (If you use any) go here.

Create these folders using your word processing software, and load them onto the server. You should preface each folder name with a '1' to get them to group together at the top of your server's file index in order to keep track of them. Unfortunately, the CGI-BIN title can't be altered.

Now, on to making pages:

START WITH THIS:

At the very top of your INDEX page:

```
<HTML>
<HEAD>
  <TITLE>Your Site Name Here</TITLE>
  <META NAME="description" CONTENT="My Site Does This Neat Stuff">
  <META NAME="keywords" CONTENT="science fiction, fantasy, horror, writing, sci fi,
speculative fiction, and anything else you can think of.">
  <META NAME="revisit-after" CONTENT="30 days">
  <META NAME="Robots" CONTENT="ALL"> (These four META lines are search functions)
</HEAD>
<BODY BGCOLOR="#ffffff"(white background) BACKGROUND="/1images/background.gif">
(overlays a pattern to act as backdrop for your page content.)

<P><CENTER><IMG SRC="1images/title.gif" ALIGN="BOTTOM" BORDER="0"
WIDTH="720" HEIGHT="90"></CENTER> (This is your header logo.)
```

This is for the first - INDEX - page. The 'META NAME' information controls the internet search function, calling upon the net to 'REVISIT' your site for an update on a regular basis. This search is done with a 'ROBOT' software, also known as a 'spider'. The 'KEYWORDS' are

specific terms the 'SEARCH' function will look for when someone does a search for sites like yours. **Your Site Name** is your official web name, thus: My Cool Web Site. This is what will show on web search results.

The 'TITLE.GIF' will call up your logo (prepared as a GIF) and load it at the top of your Index page. Image inclusions must be done as GIFs and stored in your 'IMAGES' folder. See below for further info.

INTERIOR PAGES

The interior pages of your site will each start with this:

```
<HTML>
<HEAD>
  <TITLE>YourPageTitle here</TITLE>
</HEAD>
<BODY BGCOLOR="#ffffff" (white background) BACKGROUND="/1images/background.gif">
```

In each case, the first text (<HTML>) should go right in the upper left corner with no spaces in front or above it. Otherwise your browser will not be able to find it.

Finally, at the bottom of each page you should have </BODY></HTML> to close out that page. Your best bet is to work up a blank page, then copy it as often as needed.

ADDING LINKS AND A LINK TREE:

Links, when clicked on, will send the reader to that page, one of the cool things about HTML. In order to keep your guests from getting lost in the maze of links, you should include a LINK TREE at the bottom of each page as shown here:

```
<P><CENTER><B><I><FONT SIZE="+1"><A HREF="index.html">Index</A> -
<A HREF="books.html">My Books</A> - <A HREF="contact.html">Contact Me</A> -
  <A HREF="complaints.html">My Whining</A> - <A HREF="stuff.html">Stuff</A> -
  <A HREF="map.html">Site Map</A></FONT></I></B></CENTER>
```

Note that there are no <P> callouts above to break the link tree into several lines - this is one long line of text. It should appear as follows on your web page:

Index - My Books - Contact Me - My Whining - Stuff - Site Map

If you have an extensive link tree, you might want to break it up with <P> or
 callouts, forming it into several rows. Each item above will have a specific link to its corresponding web page. If you have a lot of topics, you should link (above) to TOPIC INDEX pages, which will direct the reader in turn to specific topics (Good for focussing the reader's attention). Finally, you should include a SITE MAP page with links to every topic, in case they get lost.

When establishing a link, always be sure the URL is correct. HTML links are case sensitive.

CREATING A LINK

LINKS are structured as follows:

`Page Name`. This callout will turn any text between the `<A HREF>` and the `` into a hypertext link based on material in the " ". The Page Name is what will appear on the web page. See the Link Tree data above for details.

In order to link to another web site, use the following:

`Page Name`

HTML CALLOUTS:

In the samples given earlier you will notice various `<CALLOUTS>` which tell the computer how to lay out each line. These are the key to structuring your site, so use them carefully. The callouts below go at the start of each line, and do not need a closer (see below.)

`<P>` This will start a new line with a space between it and the previous line.

`
` This creates a new line without the intervening space, useful for lists.

`` This creates a bullet list indented from the main text.

`<HR>` This creates a line across the page, useful as a divider.

`<BR CLEAR="ALL">` This will move everything below it down far enough to clear images, creating a blank space.

`<TARGET="NEW">` This will open the linked-to page as a separate window, rather than going to it. This is useful when you need to go off site, such as to your Amazon page, without losing track of your place on your site.

The callouts below affect specific areas of text, images, etc within a given line. You must include both halves of the callout to isolate the material you seek to influence. Failing to do so will change everything below the first callout.

`<i></i>` This will cause the indicated text to display in *ITALICS*.

`` This will cause the indicated text to display in **BOLD**.

`<u></u>` This will UNDERLINE the indicated text.

`` (black - not needed for body text unless you want another color.)

`` (Changes the size of indicated text.)

`` This closes the special font zone. You must include this or the rest of your page will be altered as well.

`<H2></H2>` (comes as '1' through '6') Increases font size, same as `` callout.

`<BLOCKQUOTE></BLOCKQUOTE>` This will indent the text 1/2" for each callout used, and can be stacked for deeper indents, but must be unstacked at the end of the indented zone.

`<CENTER></CENTER>` This will center any text, SSI or image between the callouts.

`<LEFT></LEFT>` Justifies enclosed text, images, etc to the left margin. This is the default position for HTML so you will seldom have to use it.

`<RIGHT></RIGHT>` Justifies text, etc to the right margin.

INSERTING IMAGES:

Another cool gimmick HTML offers is the ability to include images on a page. These images can be done in JPEG, PNG, or GIF, but you should stick to GIF (Graphic Interface Format) since it offers a special feature.

In order to create or modify a GIF, you will need a graphics software such as Photoshop. Yes, Photoshop is expensive, but the newer upgrades can do just about anything imaginable. For your purposes, an older version, or a Student Version will do nicely, and is a lot cheaper. Be sure you get a version suitable for your computer: Photoshop comes in many 'builds' for different operating systems, and while it is supposed to be 'backwards compatible' the changes in operating systems can defeat that purpose.

There are several key features you need to focus on:

RGB - Under the IMAGE pulldown menu, select MODE, then set the image to RGB COLOR. This is essential for material appearing on line.

SIZE - Under IMAGE again, select CANVAS SIZE which will adjust the overall image. This done, set the RESOLUTION to PIXELS / INCH. The best setting is around 300 DPI (Dots Per Inch). After that, use the submenu pulldowns to select PIXELS, then type in the size you want.

From there, go ahead and create your image using the various PAINT, PENCIL, and TEXT settings. When done, SAVE the image, and pull a COPY for further work.

Now comes the cool part. After creating your image, select MAGIC ERASER (the button may be hidden under the ERASER) and click on the background you don't want to keep. This includes enclosed areas in letters like A and O. These will vanish, leaving the background transparent, which allows the image to float over the desktop cloth.

Finally, trim the image to size using the CROP feature, and save.

Now that we have an image, its time to add it to the web page. Select the spot you want to put the image in, and add the following:

```
<IMG SRC="1images/title.gif" ALIGN="CENTER" BORDER="0" WIDTH="720" HEIGHT="90">
```

You may have to adjust the line spacings for image-crowded pages, although this can also be influenced by the INSPECTOR, using the WRAP feature to set how the image relates to the text on the page.

Use the BORDER function to include a line border around your image (or not, if desired), and set the WIDTH and HEIGHT to match the dimensions of the image. You can also use these to enlarge or reduce an image on the web page, although it is better to create a separate, appropriately sized, image. Label this something like 'LogoLarge'.

Add all your images into the '1IMAGES' folder.

SERVER SIDE INCLUDES:

A really handy trick HTML can do is the SERVER SIDE INCLUDE. This is a brief line of code which tells the server to access a file and load it where indicated. This can be a photo, a sound, a RANDOM ROTATING IMAGE (see below), or a block of TEXT / HTML. In its simplest form, it works much like downloading an image:

```
<P><CENTER><!--#include virtual="cgi-bin/copyright.txt"--></CENTER>
```

The <CALLOUTS> can be added either in the TXT file, giving you the ability to post widely variable materials, or on the web page surrounding the SSI, as above. This latter is especially good for rotating quotes and comments.

PREPACKAGED SSI APPLICATIONS:

In addition to home brewed CGI applications, there are a number of apps supplied by the Server. The server will send you prepackaged SSI to copy and post to your page.

HIT COUNTERS:

One useful feature for your INDEX page is a hit counter, which records each time someone clicks to your web site. This is a helpful MARKETING tool because it shows you how many visits you are getting. Add these to your MARKET TRENDS computations to provide a running account of your traffic.

E-MAIL FORM:

Another useful prepackaged SSI is an e-mail form your readers can use to connect with you. Giving your readers a venue to communicate can be both a helpful FEEDBACK tool and helps to build your STAGE PRESENCE / PUBLIC IMAGE.

This can also be a part of a POLL form and a GUESTBOOK form, although I recommend you avoid these. I had both on my site for some time, and the amount of SPAM and GARBAGE posted was dismaying. I finally had to give both up due to one pernicious and incoherent SPAMMER who kept up a running barrage of randomness. English was *not* his first language. Sigh.

ROTATING MATERIAL:

Now we venture off the Purer Path into the dark recesses of Programming. PERL is a rugged, reliable software which can handle the few basic functions you'll need, but unlike HTML, it is not for the faint of heart. No, we have not reached PERL yet, although we are drawing perilously close. Fortunately, from here on, we can rely on the canned examples below. Let's start with a basic callout to place a block of text:

```
#!/usr/bin/perl
$RandomTextFile = "/home/YourSiteAccess/www/cgi-bin/randtext/quote.txt";
$Delimiter = "\n--NEXT--\n*";
$UseLog = 0;
$LogFile = "/Absolute/path/to/Random-Text-Log.txt";
    srand(time);
    open(LINKS,$RandomTextFile) || &Error("Cannot Open Links File :
$RandomTextFile, Error $!\n");
    @TextFile = <LINKS>;
    close(LINKS);

    $NbLines = @Text = split(/$Delimiter/, join(",@TextFile));
    $Phrase = $Text[int rand $NbLines];

    if ($UseLog) {
        @date = localtime(time); $date[4]++; $date[5] += 1900;
        $Time = "$date[4]/$date[3]/$date[5]";
        open(LOG,">>$LogFile") || &Error("Cannot Write Log File : $LogFile,
Error $!\n");
        print LOG "[ $Time] - $ENV{'REMOTE_HOST'} -> $Url\n";
        close(LOG);
    }

    print "Content-type: text/html\n\n";
    print $Phrase.\n";

sub Error {
    my($ErrorText) = @_ ;
    print "Content-type: text/html\n\n";
    print "Error: ".$ErrorText;
    exit;
}
```

Calm. Breathe deeply. Think Happy Thoughts. All you have to do is copy and paste the above, and change one line of text. Yes, this is PERL. Welcome to the jungle.

For the first time we are leaving the safe confines of your web site and wandering off in search of adventure. This document must be composed as a separate TEXT file using ASCII text. This is the most common text form, found almost everywhere, but check to be sure: there are some old TEXT applications around which still use obsolete formats. The title of your text file should be the same used in the PERL CGI, above.

This will give you a RANDOM ROTATING INCLUDE, useful for Quote-Of-The-Day material, randomly displayed comments or tidbits, and variations in major text blocks. Once you have the PERL, above, set up, you can generate a TEXT file with the material you want randomly displayed each time that page is accessed. You can set the format <CALLOUTS> in this file. Below is a typical example of how these are structured:

```
<B><CENTER>"<I>He</I> is about as pathetic as an inflatable dominatrix."</B>
<P>(Admiral MacKenna - 'Diplomacy's Stepchild')</CENTER>
--NEXT--
```

The Include will look like this:

```
"He is about as pathetic as an inflatable dominatrix."
(Admiral MacKenna - 'Diplomacy's Stepchild')
```

Some specific features of these PERL applications which will help you understand this somewhat better include:

#!/usr/bin/perl This tells the server you are switching to PERL, which it keeps safely locked in a special file. Once open, PERL will do the rest.

\$Delimiter = "\n--NEXT--\n" This defines the point where the PERL drops one quote and starts reading the next. Otherwise it will display the entire text file. This delimiter should also go after the last entry to start it all over again from the beginning.

\$Time = "\$date[4]/\$date[3]/\$date[5]" This is a random counting function using TIME from the server's internal clock, essential to various random selection processes.

\$RandomTextFile = "/home/YourSiteAccess/www/cgi-bin/copyright.txt"; This should be changed to include your site access and the address of the TEXT file you want to upload.

print "Content-type: text/html\n\n"; This tells the server to print only TEXT or HTML from your CGI-BIN, which prevents it from slinging no end of random static and material from who knows where.

exit;

} This tells the server to lock PERL in its dungeon again, preventing endless confusion when it tries to usurp control of your site from HTML.

Fortunately you don't have to (don't *WANT* to) edit or change anything except for a few file addresses. All CGI and TEXT files should go into the CGI BIN where they can be conveniently accessed.

NOTE: PERL files can be written in several minor variations. These examples come from my site and my server. If they don't work properly, have your server's TECH SUPPORT check them out.

RANDOM ROTATING IMAGES

The above is ideal for placing text, including images (like book covers) embedded in HTML, but to embed rotating or random images independently we need to journey deeper into the Heart Of Darkness in search of JAVA.

JAVA is a thoroughly modern software, which means it was written by and for SuperGeeks, and no semi-literate Spec Fi authors need apply. The one advantage of using this APPLET comes for rotating TITLE LOGOS, illustrations, or your 'About The Author' photos. So this is the only JAVA you might need - or want:

```
<P><CENTER><script language="JavaScript">
```