

CMPSCI 120 Fall 2011
Lab #4
Professor William T. Verts

Setting Up

Go to the class site and download to a folder on your computer the file called **Buttons_for_Assignment.zip**, and then unpack its contents into the same folder. There are eight tiny **.GIF** image files (four pairs, with and without beveled edges):

Button_CNN_Normal.gif



Button_CNN_Selected.gif



Button_Yahoo_Normal.gif



Button_Yahoo_Selected.gif



Button_XKCD_Normal.gif



Button_XKCD_Selected.gif



Button_Gray_Normal.gif



Button_Gray_Selected.gif



First Steps – Moving the Buttons

Using WinSCP or Fugu as you did in the previous assignment, copy these eight files (in binary mode) into the `public_html/cmpsci120` folder of your UNIX account on the `elsrv3` server.

Logging in and Renaming Files

Using PuTTY (PC) or `ssh` from terminal (Mac) as you did in the previous assignments, log in to the UNIX server. Change into the correct directory as follows:

1. `cd public_html` (Change into the Web nest)
2. `cd cmpsci120` (Change into the inner folder)
3. `ls -al` (See what files are there)

You should see `index.html` and `Button_Dr_Bill.png` from the previous assignment, and the eight new `.GIF` files. Do not proceed unless you see all 10 files.

Rename the `index.html` file to `Default.html` with the following command:

```
mv index.html Default.html
```

List the files again to verify that the renaming went correctly:

```
ls -al
```

You should now see `Default.html` (note the capitalization) and the names of the nine image files listed, but not `index.html`. Do not proceed unless this is correct.

Building a new `index.html`

We are going to build a new version of `index.html` to be the driver for a set of frames, one frame a table of contents of links, and the other frame the target for the links. The default page will be the `Default.html` page. Start up the emacs text editor on the new (empty) `index.html` file:

```
emacs index.html
```

Type in the following new code exactly as you see it here, but replace the blank in the title tag with your name:

```
<HTML>
  <HEAD>
    <TITLE>_____</TITLE>
  </HEAD>

  <FRAMESET COLS="220,*">
    <FRAME SRC="Contents.html" NAME="TOC">
    <FRAME SRC="Default.html" NAME="MAIN">
  </FRAMESET>

  <NOFRAMES>
    <H2>Frames are not supported</H2>
    <A HREF="Contents.html">
      Click for Table of Contents
    </A>
  </NOFRAMES>
</HTML>
```

Save your page and exit emacs with `Ctrl`x`Ctrl`c (and answer Y when it asks if you wish to save the changes).

Building Contents.html

Now we will build the links page to use in the left frame. Start up the emacs text editor on a brand new Contents.html file:

```
emacs Contents.html
```

Type in the following code exactly as you see it here, with the same spacing, except for the final comment (it is a placeholder for new text from the following page):

```
<HTML>
  <HEAD>
    <TITLE>Table of Contents</TITLE>

    <STYLE TYPE="text/css">
      BODY {background-color:#C0C0C0}
    </STYLE>

    <SCRIPT TYPE="text/javascript">
      <!--
        function ShowImage (item_name,image_name)
        {
          eval ('document.' + item_name +
                '.src = "' + image_name + '"') ;
          return true ;
        }
      //-->
    </SCRIPT>
  </HEAD>

  <BODY>
    <H1>Contents</H1>

    <!-- the stuff on the next page goes here -->

  </BODY>
</HTML>
```

As you type in this code, pay particular attention to commas, semicolons, and quotes. Missing one, or adding one that does not belong, can make the difference between a page that works and one that does not!

Replace the comment in the body with the code to drive four buttons, each with a normal and a selected variant (the eight button files). Here is the template to use for each button. Make four copies, one after the other. We'll talk about the ■ and □ symbols shortly.

```
<A HREF="http://_____"
  onMouseOver="ShowImage('■', '□_Selected.gif') ;"
  onMouseOut ="ShowImage('■', '□_Normal.gif') ;"
  TARGET="MAIN">

  <IMG SRC="□_Normal.gif" BORDER=0 NAME="■">
</A>

<BR>
```

The ■ symbol is to *identify each image with a unique JavaScript name* so that the appropriate `onMouseOver` and `onMouseOut` event handlers can grab it. Each of the four copies of the template will use different strings, but within each copy it must be the same string in the three places where the symbol appears. The string does not matter: it could be `frog` or `MyImage1` or whatever you want, but in order to make the code easy to debug please follow the guidelines below. Similarly, the □ symbol is to help uniquely identify one of the eight .GIF image files: `Button_CNN_Normal.gif`, `Button_CNN_Selected.gif`, etc.

The first button will be to CNN, so in the first copy of the template all the □ and ■ symbols must be replaced with the unique string `Button_CNN` (i.e., change all ■ to `Button_CNN` for the resource name, and change all copies of the file names `□_Selected.gif` and `□_Normal.gif` to `Button_CNN_Selected.gif` and `Button_CNN_Normal.gif`, respectively). Point the link in the HREF attribute to `http://www.cnn.com/`.

Edit the second copy of the template to use the `Button_Yahoo` buttons and point the link to `http://www.yahoo.com/`.

Edit the third copy of the template to use the `Button_XKCD` buttons and point the link to `http://www.xkcd.com/`.

Edit the fourth and final copy of the template to use the blank `Button_Gray` buttons, and point the link to your `Default.html` page. Since `Default.html` is in the same folder as `Contents.html`, you do not need a complete URL; the link need only be `HREF="Default.html"`, without any `http://` part.

Save your page and exit emacs with `Ctrl)xCtrl)c` (and answer Y when it asks if you wish to save the changes).

Make Sure Everything is Visible

Set the permissions to `rw-r--r--` on all files in the `public_html/cmpsci120` folder to make them visible on the Web (including those you've already set):

```
chmod 644 *.*
```

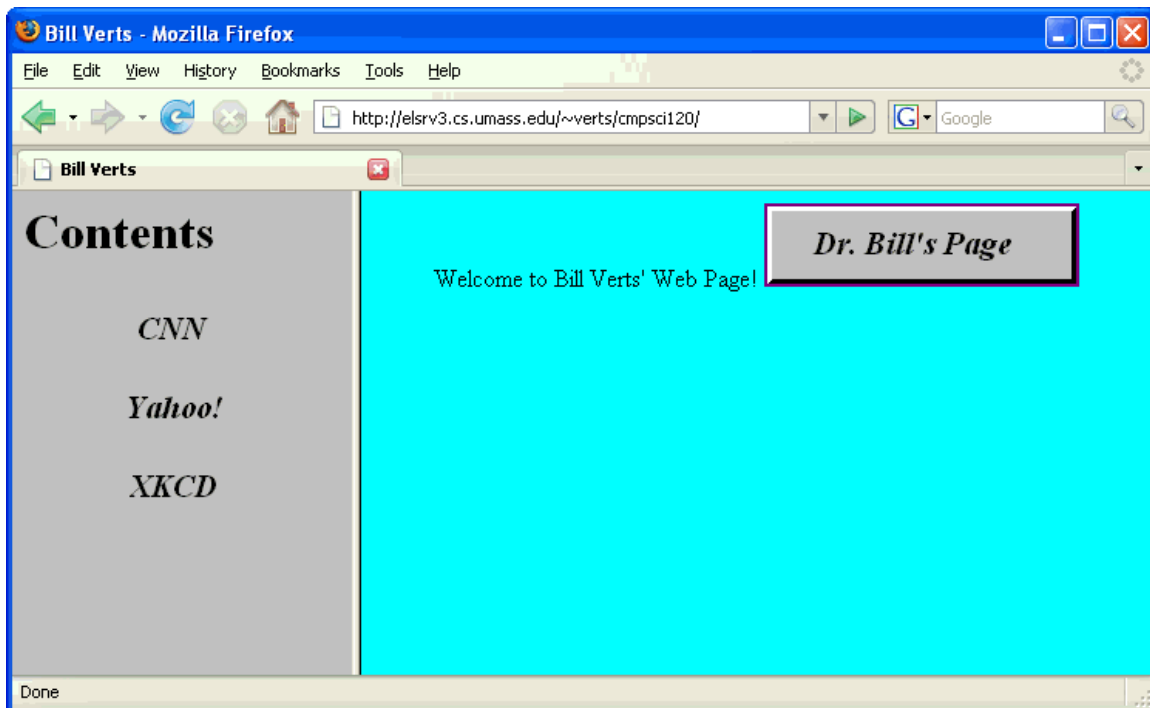
List all the files in the current directory:

```
ls -al
```

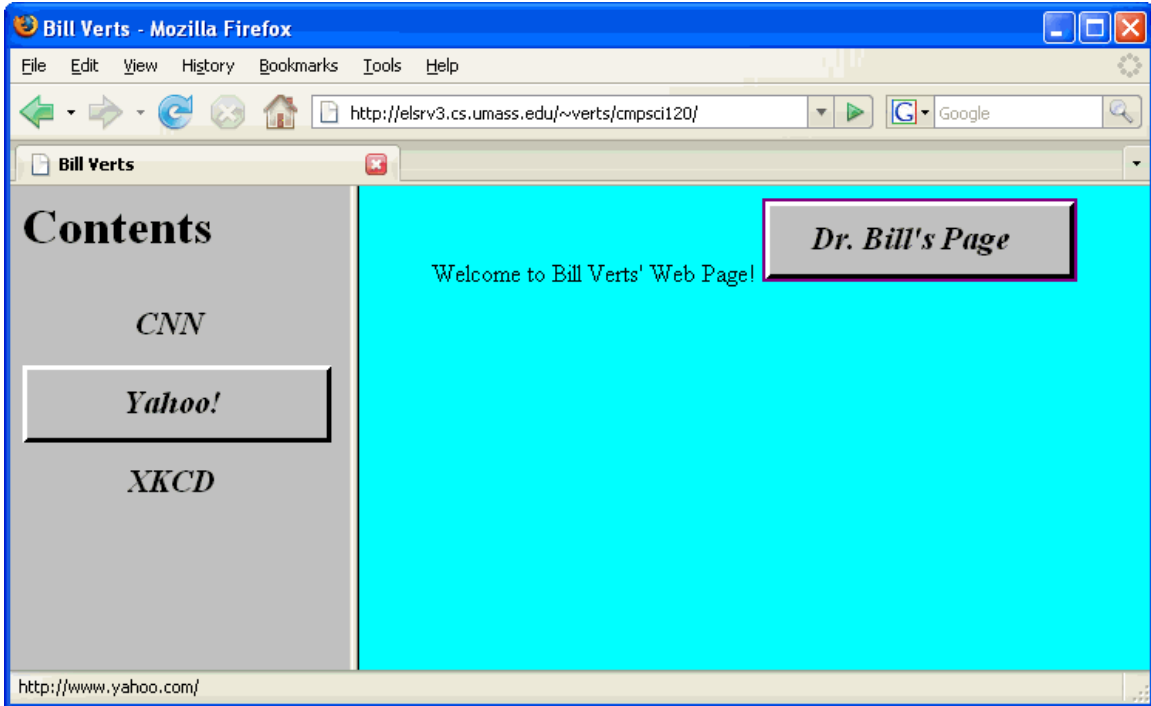
Do not proceed until all files have `rw-r--r--` permissions.

Testing

Along with all the image files, your `public_html/cmpsci120` folder should now contain three Web pages: **index.html** (the frame driver), **Contents.html** (the left frame containing the links and buttons), and **Default.html** (the default right frame which was your old `index.html` page). Load in and test your `index.html` page. Make sure to hit the refresh button. Your page should look as follows (with your name instead of mine in the appropriate places). Note that in this view (and all that follow) I accidentally omitted the `
` tag between the text and the button in `Default.html`. In *your* pages, the button will be underneath the text.



Floating the mouse over the Yahoo button should appear as follows:



Clicking that Yahoo button should replace the right-hand pane with Yahoo's home page.

Even though there are four defined buttons, the fourth button "blends in" with the frame background until you float the mouse over it. This is shown below:



Make sure that clicking each button jumps to the correct page. Use emacs to correct any mistakes.

If You Get Stuck

If you miss any of the instructions, you may need to go back and fix certain things. Possible problems include misspelling the name of the .GIF files in Contents.html, setting or omitting certain file permissions, or copying files into the wrong place. Please email me or the TA if you need help.

What To Turn In

When your pages are correct and visible on the Web, send an email message to **both** the TA **and** the **literacy@cs.umass.edu** account. The subject line must be set to the exact phrase **CMPSCI 120 ASSIGNMENT #4** and the body of the message must contain your name, your username, and your full Web address. The next assignment will also depend on you successfully completing this one.