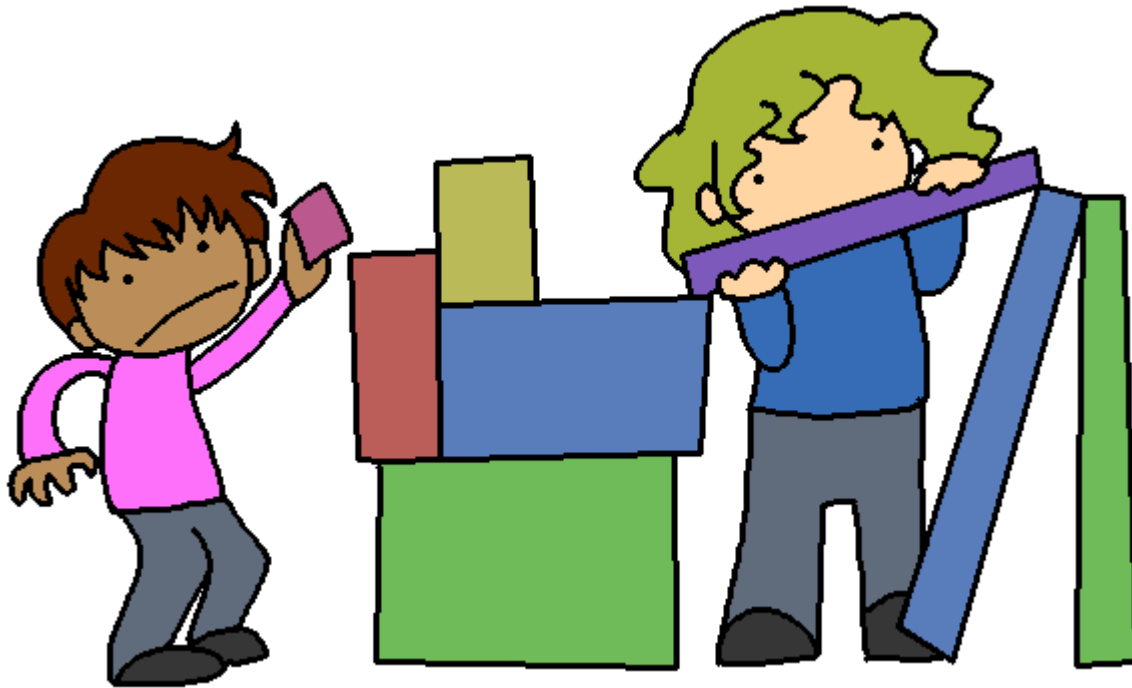


Building HTML



CS/IT 490 WD, Fall 2013

Last update 2013-08-28

Written by Rachel J. Morris
Creative Commons Attribution-NonCommercial-ShareAlike
3.0 Unported License

Breakdown

- Tools!
- Structuring your webpage

Tools!

- Make sure you have a good **text editor** with **syntax highlighting** when writing your HTML!
- Lighter weight:
 - Notepad++
 - Geany
- Heavier weight:
 - Eclipse
 - Aptana Studio
 - NetBeans

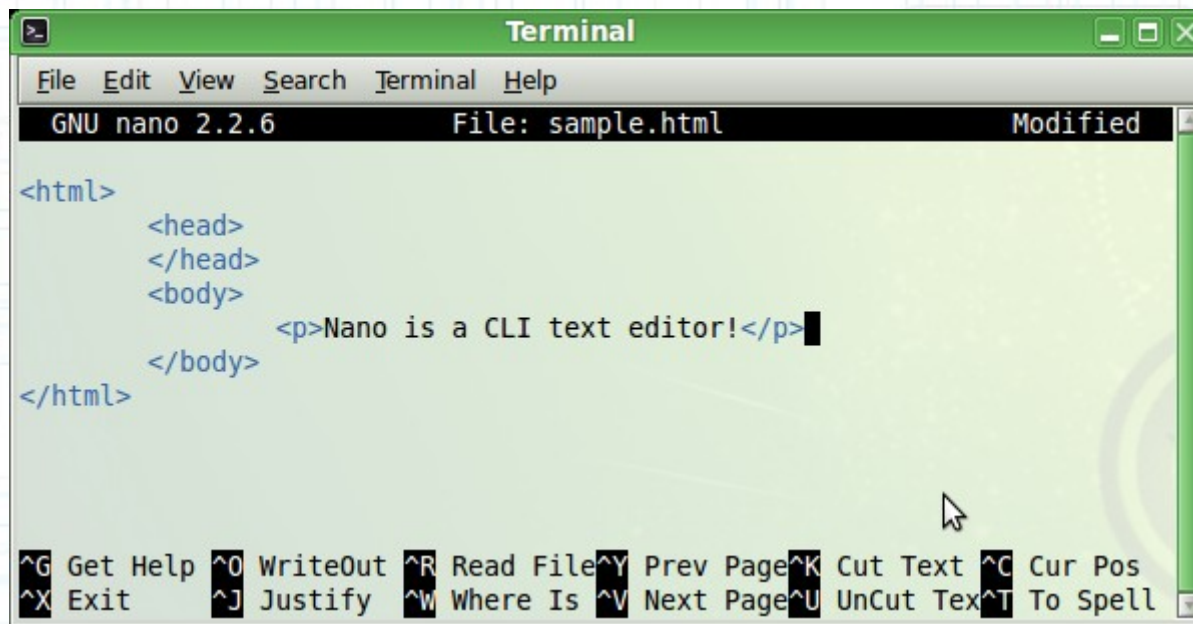


These are completely fine!

Tools!

- Command Line Interface tools:
 - Nano
 - Vim
 - Emacs

- *nix Text Editors:
 - Gedit
 - Kate
 - Kwrite
 - Many things



```
Terminal
File Edit View Search Terminal Help
GNU nano 2.2.6 File: sample.html Modified
<html>
  <head>
  </head>
  <body>
    <p>Nano is a CLI text editor!</p>
  </body>
</html>
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Tex ^T To Spell
```

Tools!

Don't use the version of Notepad that comes with Windows.

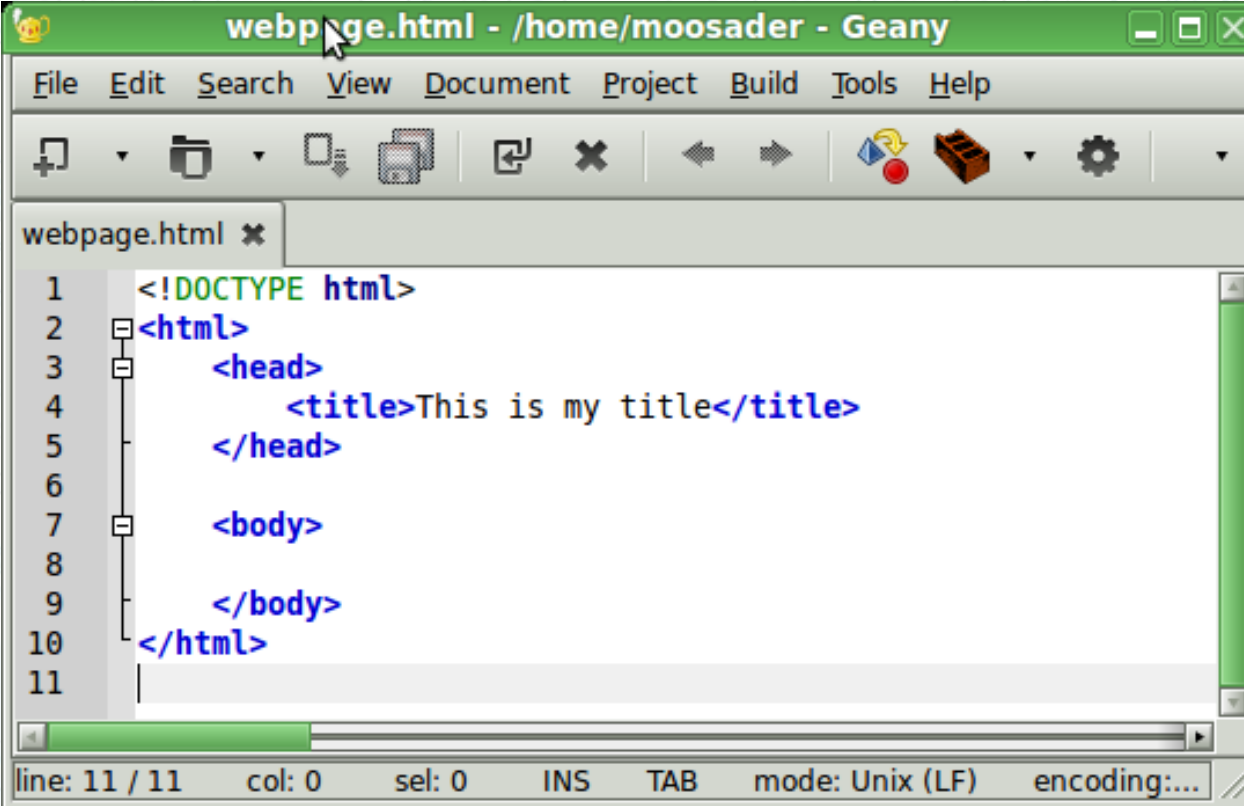
- It has no syntax highlighting
- It doesn't auto-tab
- It only stores an undo history of 1 command!

Structuring your webpage

- Even when you know CSS, you should begin working on a webpage by just working on the HTML first.
- Think about the markup's structure and the layout you want before messing with any styling!
- Keep the markup clean! Indent! Whitespace is OK!

Structuring your webpage

- Start with your core markup:



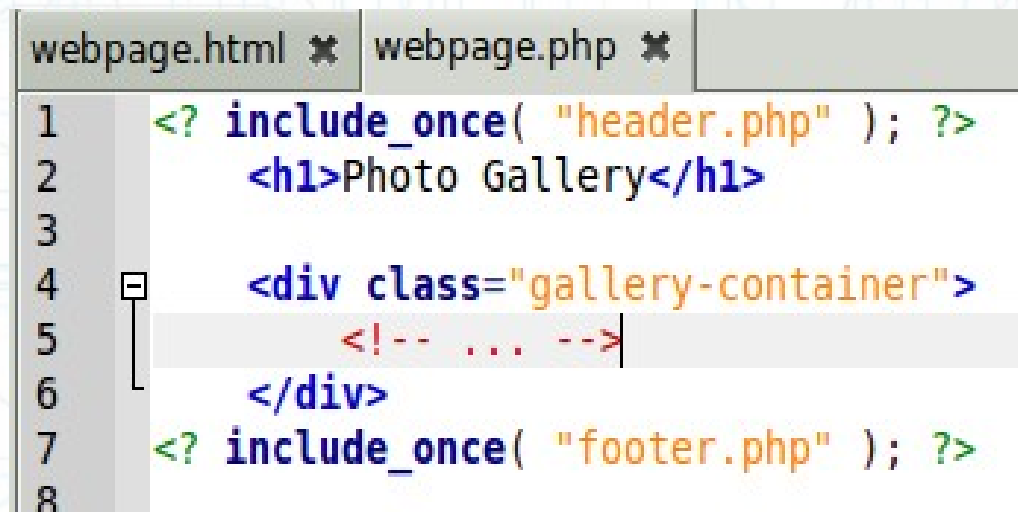
The screenshot shows the Geany text editor window titled "webpage.html - /home/moosader - Geany". The menu bar includes File, Edit, Search, View, Document, Project, Build, Tools, and Help. The toolbar contains icons for file operations and editing. The editor window shows a single file named "webpage.html" with the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>This is my title</title>
5   </head>
6
7   <body>
8
9   </body>
10 </html>
11
```

The status bar at the bottom indicates "line: 11 / 11", "col: 0", "sel: 0", "INS", "TAB", "mode: Unix (LF)", and "encoding:..."

Structuring your webpage

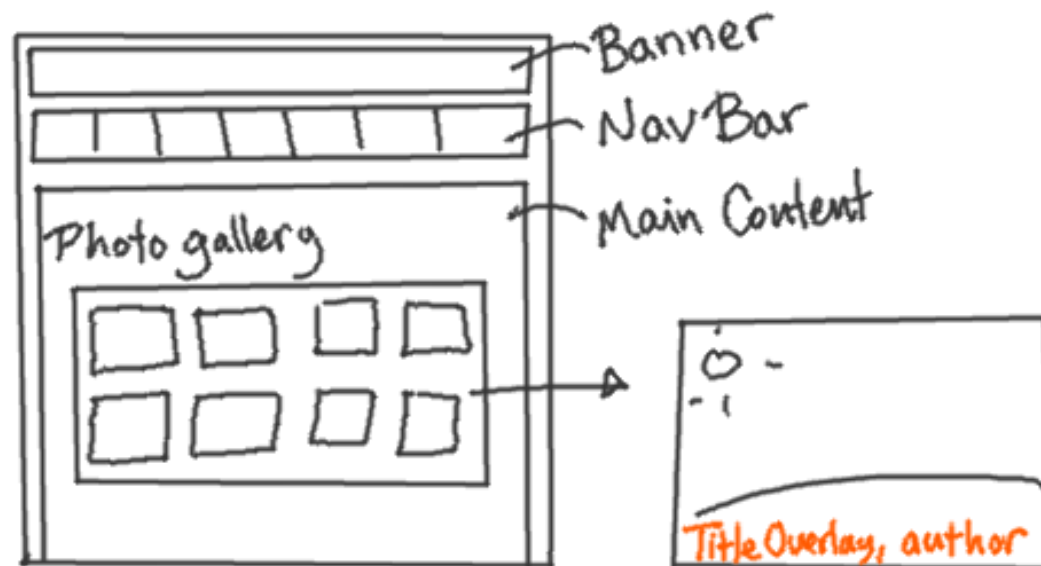
- When we start using PHP, we can extract all of the duplicate code (<head> contents, nav bar, etc.) and just have unique code in each .php page



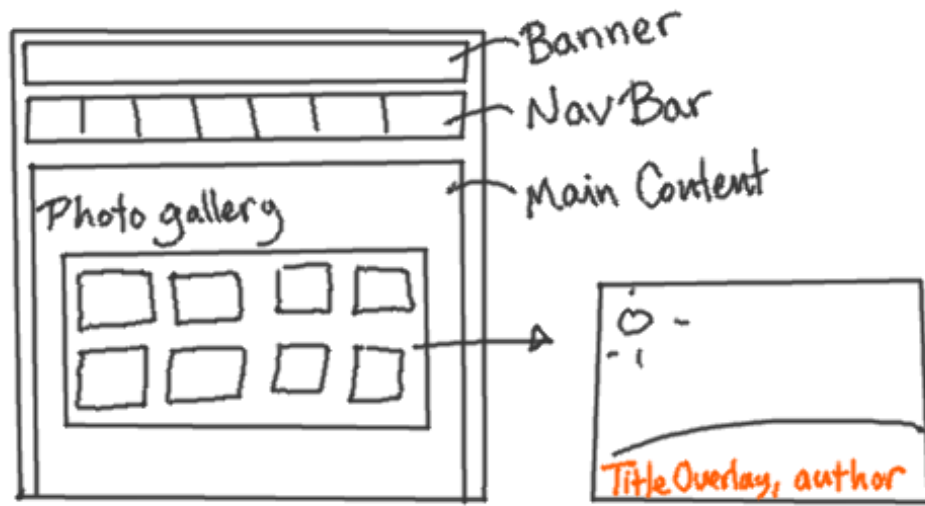
```
webpage.html * webpage.php *
1  <? include_once( "header.php" ); ?>
2      <h1>Photo Gallery</h1>
3
4      <div class="gallery-container">
5          <!-- ... -->
6      </div>
7  <? include_once( "footer.php" ); ?>
8
```


Structuring your webpage

- Make sure you have an idea for a page layout first. Know how things should be split up on the webpage.

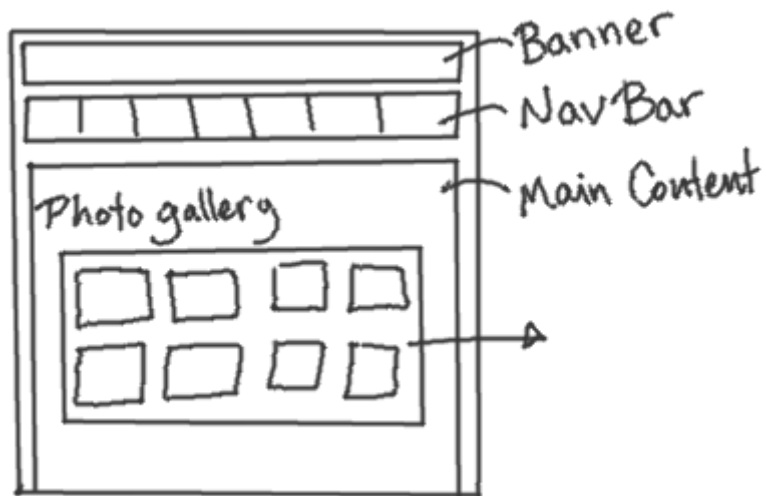


Structuring your webpage



What is in the body?

- Banner bar
- Navigation bar
- Navigation buttons
- Main content region
- Photo gallery container
- Photo thumbnails with title/author overlay

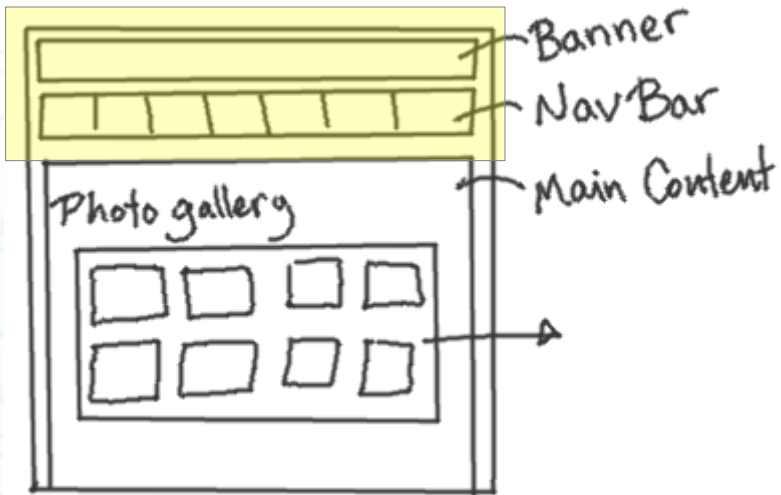


Sample markup -->

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12      <nav>
13        <!-- There will be multiple of these -->
14        <a href="">Home</a>
15      </nav>
16    </header>
17
18    <div class="main-content">
19      <h1>Photo Gallery</h1>
20
21      <div class="gallery-container">
22
23        <!-- There will be multiple of these -->
24        <div class="photo">
25          
26          <span class="title">Sunset</span>
27          <span class="author">RJM</span>
28        </div>
29
30      </div>
31    </div>
32  </body>
33 </html>

```

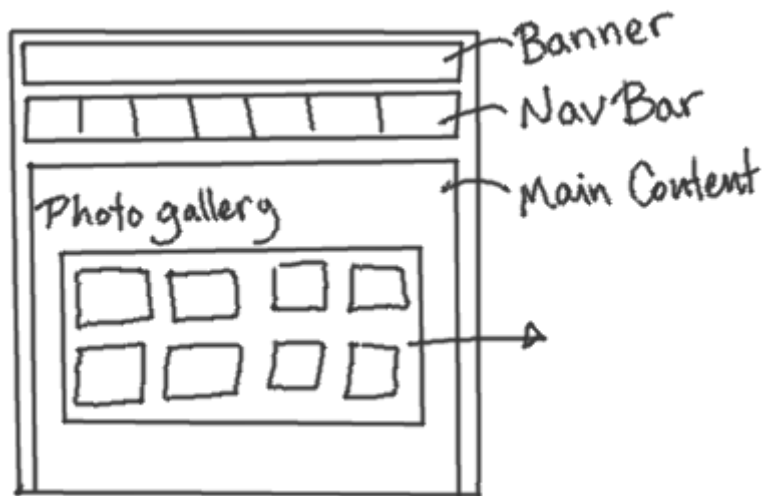
Notice that the header has been contained within the HTML5 `<header>` tags.

The navbar & links go within the `<nav>` tag (also HTML5)

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12       <nav>
13         <!-- There will be multiple of these -->
14         <a href="">Home</a>
15       </nav>
16     </header>
17
18     <div class="main-content">
19       <h1>Photo Gallery</h1>
20
21       <div class="gallery-container">
22
23         <!-- There will be multiple of these -->
24         <div class="photo">
25           
26           <span class="title">Sunset</span>
27           <span class="author">RJM</span>
28         </div>
29
30       </div>
31     </div>
32   </body>
33 </html>

```



As you build the markup, make sure to add **classes** to your **divs** and **spans**.

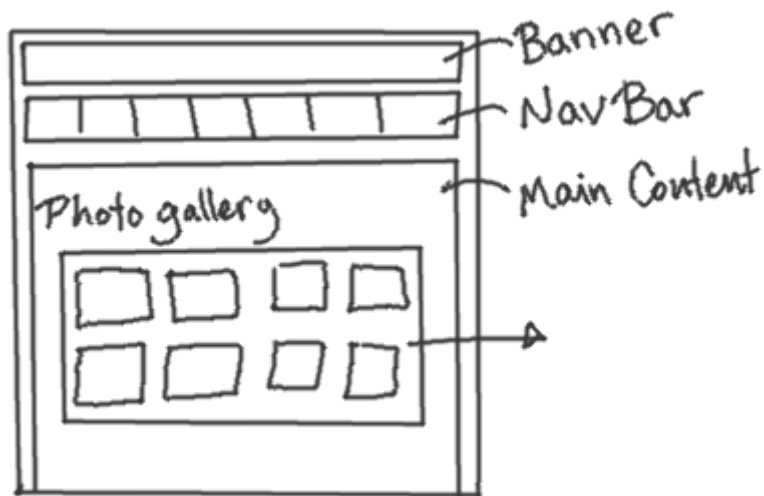
For now, think of them as labels for what they contain.

In CSS, we can target these classes and style them in a specific way.

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12      <nav>
13        <!-- There will be multiple of these -->
14        <a href="">Home</a>
15      </nav>
16    </header>
17
18    <div class="main-content">
19      <h1>Photo Gallery</h1>
20
21      <div class="gallery-container">
22
23        <!-- There will be multiple of these -->
24        <div class="photo">
25          
26          <span class="title">Sunset</span>
27          <span class="author">RJM</span>
28        </div>
29
30      </div>
31    </div>
32  </body>
33 </html>

```



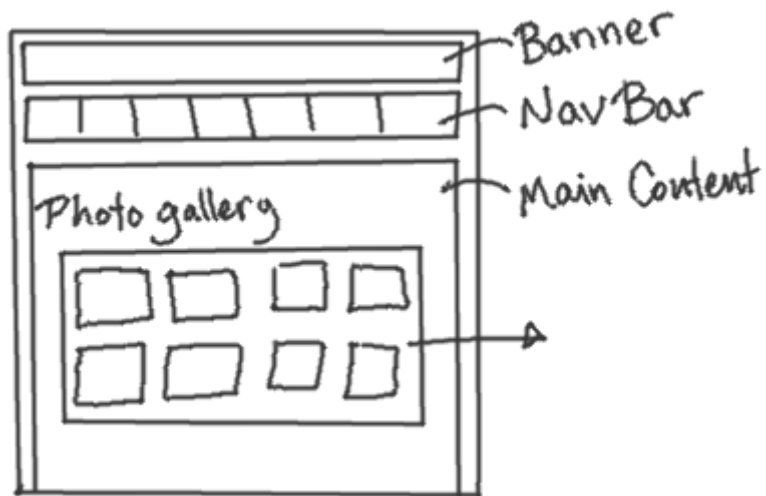
You'll usually want to create some sort of “main content” div to store all your webpage's content in.

You will also want to create “container” divs for anything relatively complex, such as this photo gallery – There will be many photos displayed, so we want to wrap it in a container so we can keep the style sane.

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12      <nav>
13        <!-- There will be multiple of these -->
14        <a href="">Home</a>
15      </nav>
16    </header>
17
18    <div class="main-content">
19      <h1>Photo Gallery</h1>
20
21      <div class="gallery-container">
22
23        <!-- There will be multiple of these -->
24        <div class="photo">
25          
26          <span class="title">Sunset</span>
27          <span class="author">RJM</span>
28        </div>
29
30      </div>
31    </div>
32  </body>
33 </html>

```

Note that, for now, this will display an image and some text without any overlap.

This will be changed in CSS so that the text overlays the image nicely.

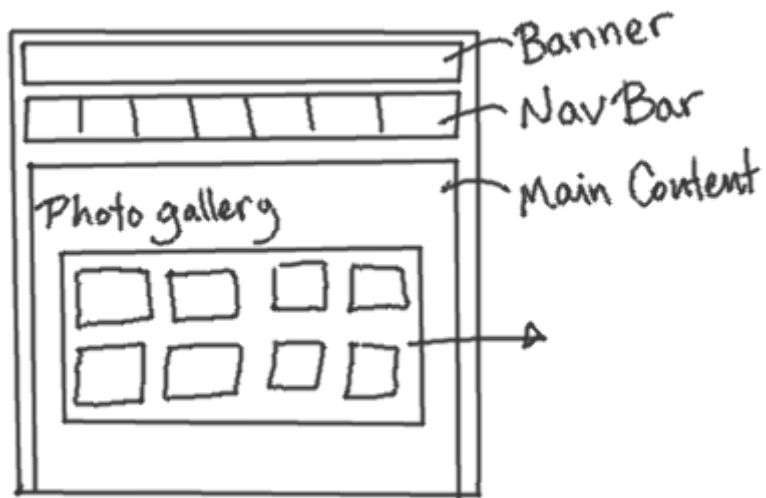
All this is within a “photo” class div to help simply positioning once we get into the CSS.

Don't be afraid to make containers!

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12      <nav>
13        <!-- There will be multiple of these -->
14        <a href="">Home</a>
15      </nav>
16    </header>
17
18    <div class="main-content">
19      <h1>Photo Gallery</h1>
20
21      <div class="gallery-container">
22
23        <!-- There will be multiple of these -->
24        <div class="photo">
25          
26          <span class="title">Sunset</span>
27          <span class="author">RJM</span>
28        </div>
29
30      </div>
31    </div>
32  </body>
33 </html>

```



It takes practice, but you'll gain foresight as you work on webpages – thinking about how the CSS and PHP interconnect.

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Aw.S.M. Photoz</title>
5    </head>
6
7    <body>
8      <header>
9        <div class="banner">
10       </div>
11
12      <nav>
13        <!-- There will be multiple of these -->
14        <a href="">Home</a>
15      </nav>
16    </header>
17
18    <div class="main-content">
19      <h1>Photo Gallery</h1>
20
21      <div class="gallery-container">
22
23        <!-- There will be multiple of these -->
24        <div class="photo">
25          
26          <span class="title">Sunset</span>
27          <span class="author">RJM</span>
28        </div>
29
30      </div>
31    </div>
32  </body>
33 </html>

```

References

-