Creating and Building Websites

Stanford University Continuing Studies CS 21

Mark Branom
branom@alumni.stanford.edu

Course Web Site: http://web.stanford.edu/group/csp/cs21/

Course Description

In this class, we will explore the fundamentals of web site creation. We'll look at the software, on-line resources, and reading materials which will help you build outstanding web sites. The class will consist of a mixture of hands-on exercises, lectures, and demonstrations. Topics covered will include:

- Web design principles
- HTML5 basics
- Adding images to web pages
- Linking web pages
- Using CSS to apply style to web content

By the end of the class, you will have built a website and be ready to strike out on your own. Students MUST have access to a computer, be familiar with computer basics, and have access to the Internet. Since the Web is cross-platform, you can apply the knowledge you gain from this class to any computer platform.

Administrivia

Grading

You have the option of taking this course for a letter grade, CR/NC, or as an Audit.

	Letter Grade	**CR/NC**	**NGR**
Attendance	Required	Required	Recommended
Homework	Required	Required	Recommended
Project	Required	Required	Not required
	By request	By Request	Default choice

More Administrivia

- Resources
  - Each week you will receive supplemental handouts and in-class exercises. Copies of the handouts and links to resources will be posted to the CS21 website:
    - http://web.stanford.edu/group/csp/cs21/

- How to reach me:
  - branom@alumni.stanford.edu
  - markbranom@gmail.com
  - mark.branom@sjcc.edu
  - 650-224-1728

More Administrivia

- You will need:
  - Access to a computer
  - Familiarity with computer basics
  - Access to the internet (many free ISPs exist)
  - A Byet.Host account (we'll do it in class)
  - Access to the following tools:
    - A text editor (*TextEdit* for Macs; *Notepad* for Windows)
      - Better choices: *TextWrangler* for Macs; *Notepad++* for Windows
      - Even better choice: *Dreamweaver* (both Mac & Windows)
    - A web browser (*Firefox* / *Internet Explorer* / *Chrome* / *Safari*)

Assignments & Final Project

- The weekly class assignments and homework will build toward the final project
- The final project will be to turn in a web site consisting of one home page and four linked subsequent pages with content, links, and graphics on all of them.
- The assignment will be "turned in" by being loaded onto your webspace.

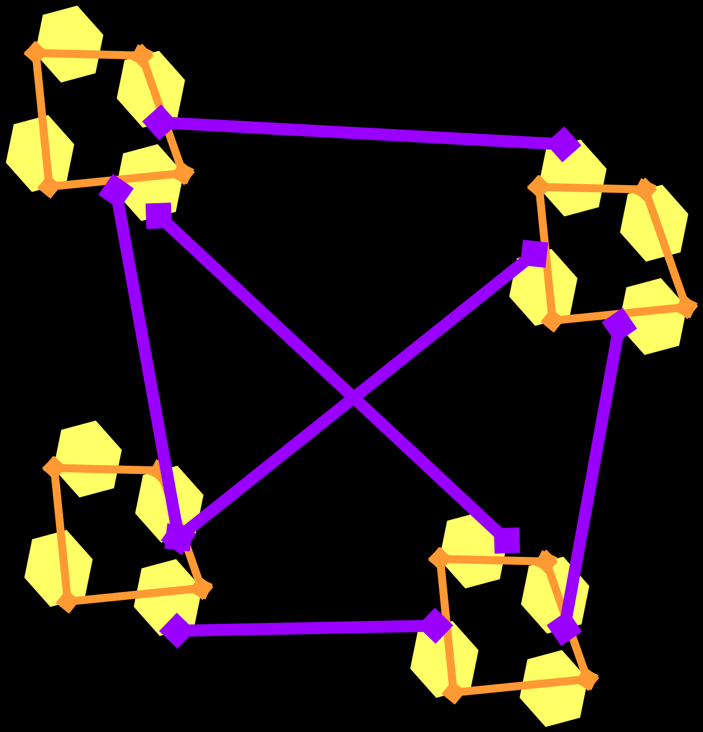
Course Syllabus

- Week 1: The Web & HTML basics
- Week 2: Adding Links, Fonts, and Colors
- Week 3: Cascading Style Sheets (CSS) – The Basics
- Week 4: Adding Graphics – The Basics
- Week 5: *Dreamweaver*, Imagemaps, Sound & Video (HTML5); Lists; Tables; E-mail Newsletters
- Week 6: Cascading Style Sheets (CSS) - more advanced usage
- Week 7: HTML5 Forms and Basic PHP Scripting
- Week 8: iFrames; Internal links; meta tags; site promotion; Basics of Content Management Systems

Week 1 Agenda

- Demystifying the internet ("How does it work?")
- Demystifying Web Pages
- HTML basics

How the internet works...



The Internet is a network of networks that allows for communication and sharing of information.

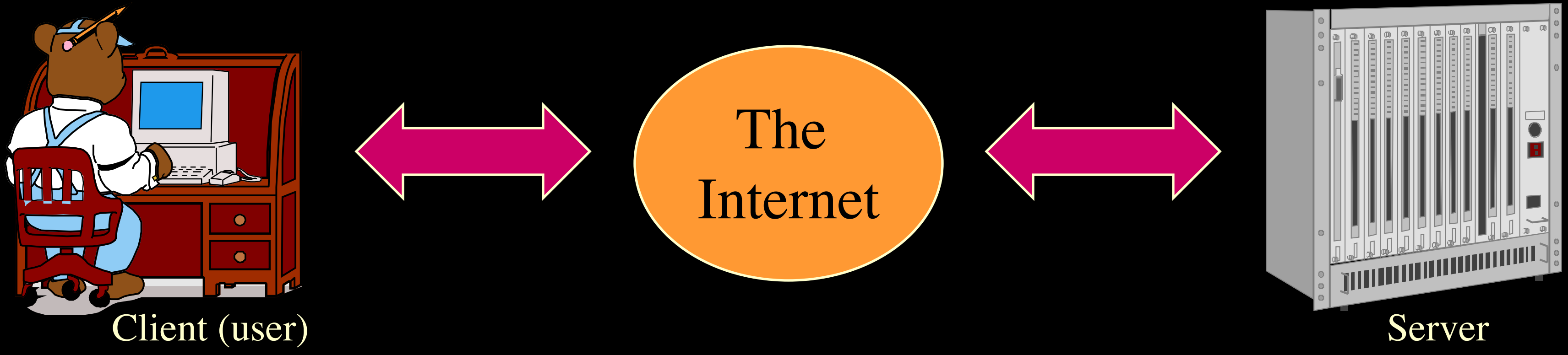
The World Wide Web is just one of several applications which can be used on the Internet.

Web Terminology

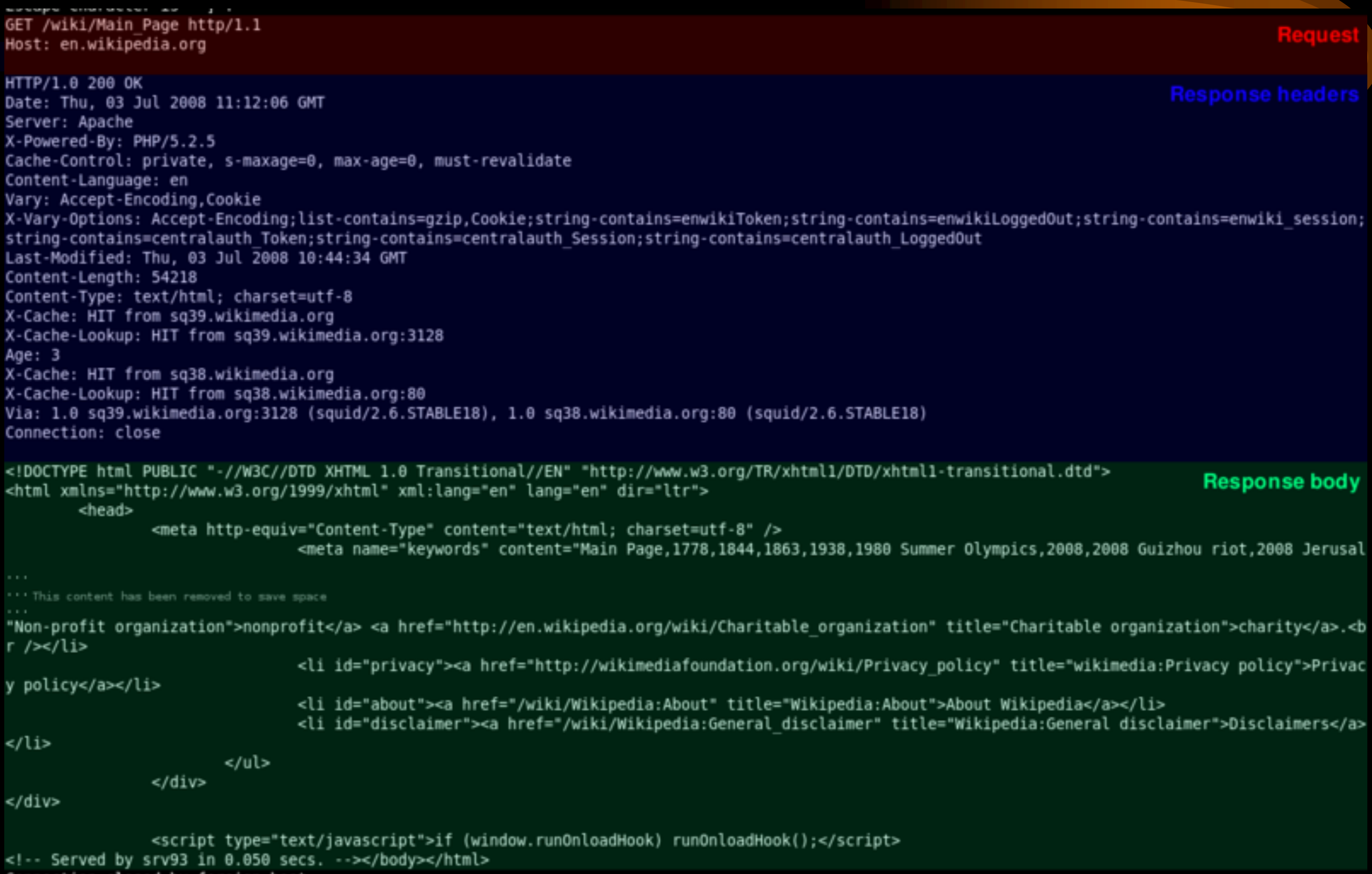
- ***Protocol*** - ground rules or "languages" that computers use to "talk" with each other
- ***HTML*** (HyperText Markup Language) - the "programming" language used to write web pages
- ***Source file*** - the set of tags and text which make up a web page. Browsers process the source file to make the web page look the way the designer wanted it to look.
- ***URL*** (Uniform Resource Locator) - a web address; indicates the location of a web resource as well as the protocol needed to access it
- ***HTTP*** (HyperText Transfer Protocol) - the internet protocol which allows web pages to work
- ***FTP*** (File Transfer Protocol) - allows computers to exchange files over a network
- ***Web page*** - a single page on the web (a "homepage" is the first web page on a web site)
- ***Web site*** - a collection of web pages, usually on a particular topic or business
- ***Web browser/navigator/client*** - the software application which displays web pages
- ***Web server*** - the computer or network of computers which stores web pages

How web pages work…

1) Client (user) tells the browser to request a web page using the *http* protocol
2) Request goes over the internet to the web server
3) Web page is found on the web server
4) Copy of the source code for the web page is sent back to the original computer
5) Browser processes ("translates") code into web page



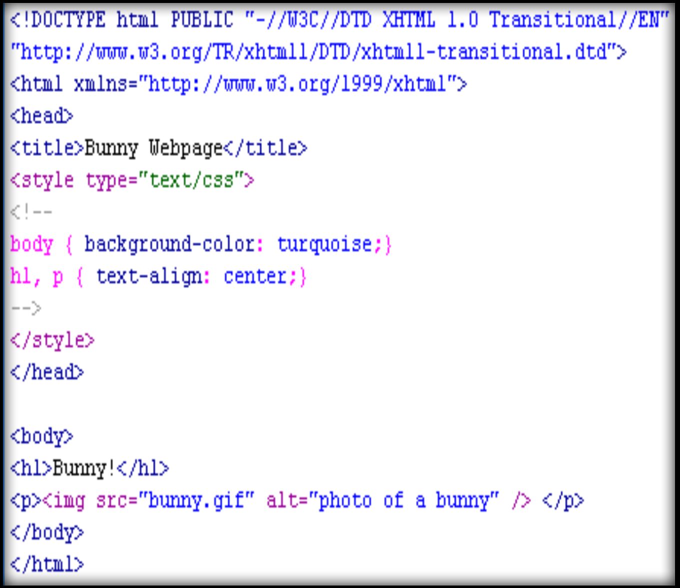
What browsers actually GET...



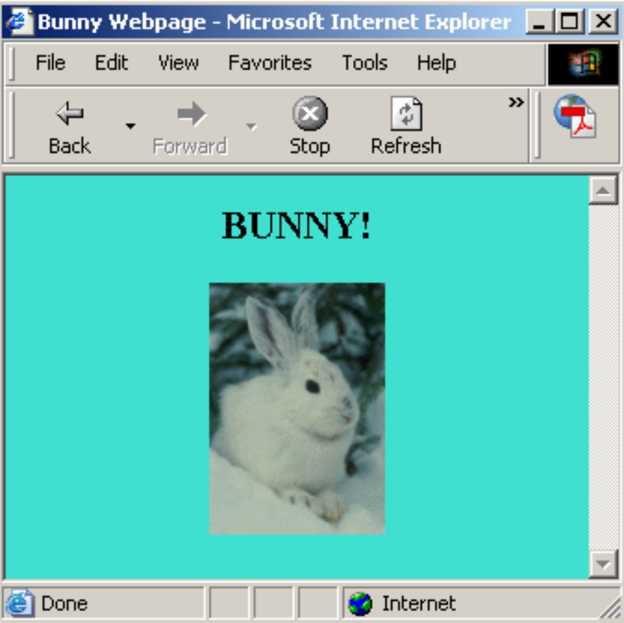
```
GET /wiki/Main_Page http/1.1
Host: en.wikipedia.org

HTTP/1.0 200 OK
Date: Thu, 03 Jul 2008 11:12:06 GMT
Server: Apache
X-Powered-By: PHP/5.2.5
Cache-Control: private, s-maxage=0, max-age=0, must-revalidate
Content-Language: en
Vary: Accept-Encoding,Cookie
X-Vary-Options: Accept-Encoding;list-contains=gzip,Cookie;string-contains=enwikiToken;string-contains=enwikiLoggedOut;string-contains=enwiki_session;
string-contains=centralauth_Token;string-contains=centralauth_Session;string-contains=centralauth_LoggedOut
Last-Modified: Thu, 03 Jul 2008 10:44:34 GMT
Content-Length: 54218
Content-Type: text/html; charset=utf-8
X-Cache: HIT from sq39.wikimedia.org
X-Cache-Lookup: HIT from sq39.wikimedia.org:3128
Age: 3
X-Cache: HIT from sq38.wikimedia.org
X-Cache-Lookup: HIT from sq38.wikimedia.org:80
Via: 1.0 sq39.wikimedia.org:3128 (squid/2.6.STABLE18), 1.0 sq38.wikimedia.org:80 (squid/2.6.STABLE18)
Connection: close

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en" dir="ltr">
        <head>
                <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
                                <meta name="keywords" content="Main Page,1778,1844,1863,1938,1980 Summer Olympics,2008,2008 Guizhou riot,2008 Jerusal
...
''' This content has been removed to save space
...
"Non-profit organization">nonprofit</a> <a href="http://en.wikipedia.org/wiki/Charitable_organization" title="Charitable organization">charity</a>.<b
r /></li>
                                <li id="privacy"><a href="http://wikimediafoundation.org/wiki/Privacy_policy" title="wikimedia:Privacy policy">Privac
y policy</a></li>
                                <li id="about"><a href="/wiki/Wikipedia:About" title="Wikipedia:About">About Wikipedia</a></li>
                                <li id="disclaimer"><a href="/wiki/Wikipedia:General_disclaimer" title="Wikipedia:General disclaimer">Disclaimers</a>
</li>
                        </ul>
                </div>
</div>

                <script type="text/javascript">if (window.runOnloadHook) runOnloadHook();</script>
<!-- Served by srv93 in 0.050 secs. --></body></html>
```

How does it work?

Web pages are simply made up of plain ASCII text. Web browsers process the codes used on web pages to display a fully formatted web page. But the pictures, colors, and other non-text items on a web page aren't actually on the web page itself -- the web page itself is simply text,with the formatting and graphics encoded. You can look at the code for any web page by going to the "View" menu and choosing "Page Source".



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Bunny Webpage</title>
<style type="text/css">
<!--
body { background-color: turquoise;}
h1, p { text-align: center;}
-->
</style>
</head>

<body>
<h1>Bunny!</h1>
<p><img src="bunny.gif" alt="photo of a bunny" /> </p>
</body>
</html>
```



Introduction to HTML

- HTML stands for "HyperText Markup Language". There are several “dialects” of HTML; the most recent are XHTML 1.1 and HTML 5.
- HTML is a collection of text surrounded by tags which modify the text of the document.
- All tags are encoded in angle brackets (`< >`).
- In general, tags work in pairs -- one to turn on the modification, one to turn it off. Stop tags look just like start tags except they have a slash (/) in front.

```
<tag>affected text</tag>
```

- Some tags work on their own. We refer to these tags as “empty” or “singlet” tags. Since they don’t have a closing tag, in xHTML, we place a `/` at the end of the tag. In HTML5, there is no need for the trailing slash. The Horizontal Rule tag `<hr />` is an example.
- Most tags have attributes which can modify how they function:

```
<tag attribute="value">affected text</tag>
```

Tag examples

```
<em>I want this in italics</em>
```

is processed by the browser to look like:

I want this in italics

```
<p style="text-align: center;">I want this new
paragraph to be centered</p>
```

is processed by the browser to look like:

I want this new paragraph
to be centered

Structural tags (elements)

- These are the basic tags which must be a part of every web page.
  - The *DocType Declaration* tells the browser which dialect of HTML the web page is using (`<!doctype html>` is the doctype for HTML5).
  - The `<html>` element contains the entire web page.
  - The `<head>` element provides information about the web page (and mostly contains the title of the web page, any JavaScripts, Meta tags, and Cascading Style Sheet references).
  - The `<body>` element contains the visible part of the web page.

```
<!doctype html>
<html>
  <head>
     <title>Title of Webpage</title>
  </head>
  <body>
     This is where the main part of the web page would go
  </body>
 </html>
```

HTML5 structural tags

- With the advent of HTML5, there are several new structural elements:
  - `<header>Used to indicate header information</header>`
  - `<footer>Used to indicate footer information</footer>`
  - `<main>Used to indicate the main content</main>`
  - `<nav>Used for site or in-page navigational items</nav>`
  - `<article>Used if content could be on its own page (e.g., individual comment, post, sidebar widget)</article>`
  - `<aside>Sidebars, comments section, ads, etc.</aside>`
  - `<section>Section within a page or chapter within an article, etc.</section>`
  - `<div>Other non-semantic division of text</div>`

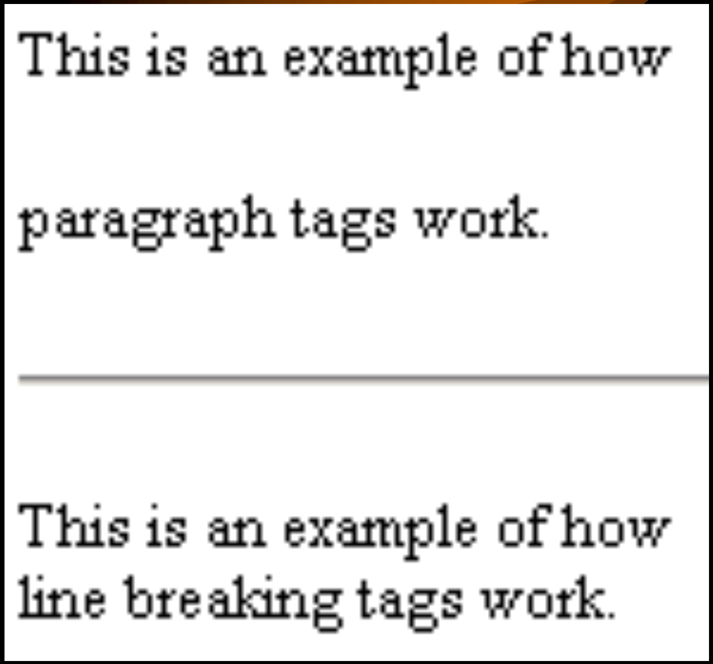
- For details, read *Dive into HTML5* or look at the *HTML5 Element Flowchart*: http://html5doctor.com/downloads/h5d-sectioning-flowchart.png

Breaking Lines and Paragraphs

- `<p>` text text text `</p>`
  - Paragraph tag
  - Most browsers render (process) this with blank lines between each paragraph
- `<br />`
  - Line break tag. Used when the webmaster wants a carriage return but doesn't want a blank line to follow
- `<hr />`
  - Horizontal Rule. Used to place a straight line across the page.

Examples of Breaking Lines

```
<p>This is an example of how</p>
<p>paragraph tags work.</p>
<hr />
<br />This is an example of how
<br />line breaking tags work.
```



This is an example of how

paragraph tags work.

This is an example of how
line breaking tags work.

Header Tags

Header Tags – Used for marking sections and subsections in a document. Using Cascading Style Sheets (CSS), you can change the default meaning for the tags (more on this next week).

`<h1>` Header 1 – Main section; Giant-sized and bold `</h1>`

`<h2>` Header 2 – Subsection; Large and bold `</h2>`

`<h3>` Header 3 – Sub-subsection; Normal-sized and bold `</h3>`

`<h4>` Header 4 – Sub-subsection; Small and bold `</h4>`

`<h5>` Header 5 – Sub-subsection; Very Small and bold `</h5>`

`<h6>` Header 6 – Sub-subsection; Tiny and bold `</h6>`

Formatting text

- Bolding and Italicizing:
  - `<strong>text you want bold (strongly emphasized) </strong>`
    - `<b> The bold tag should not be used. </b>`
  - `<em>text you want in italics (emphasized) </em>`
    - `<i> The italics tag should not be used. </i>`
    - `<cite> The cite tag is used for citations (books, journal articles, etc.) </cite>`
    - `<address> The address tag is used to indicate the main contact for the web site. </address>`
- Aligning text or graphics left/center/right:
  - The CSS style `text-align` has a value of `left`, `center`, `right`, or `justify`.
  - We'll cover CSS starting in week 3. Stay tuned!

Planning for your site

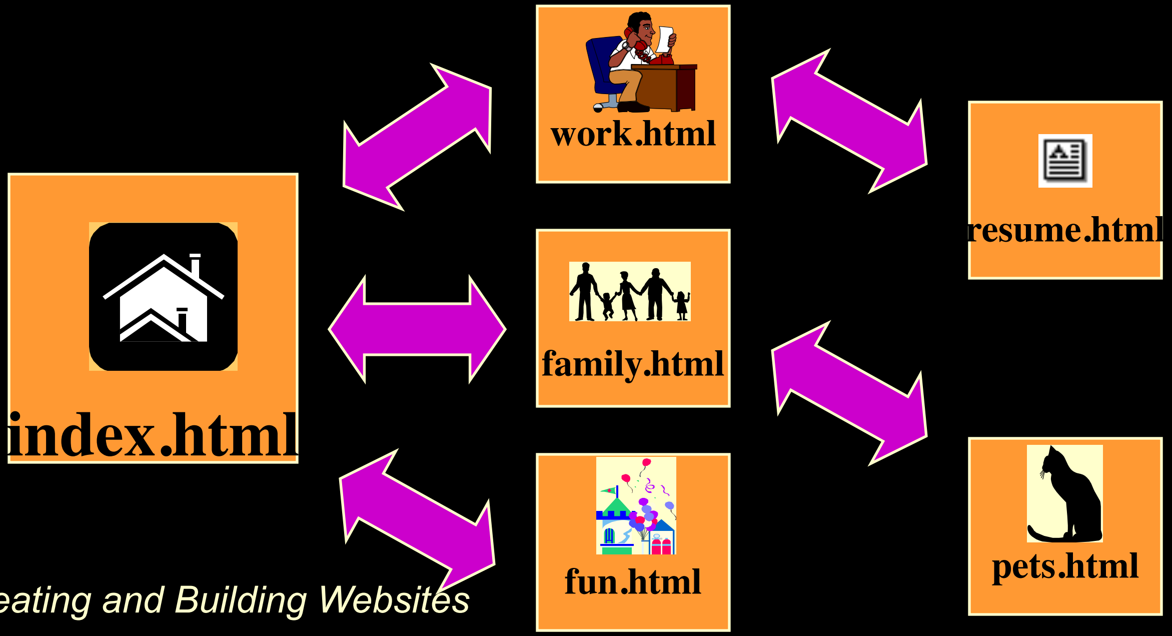
- BEFORE you start creating pages and coding tags -- before you begin to write content, even -- you need to do some planning
- AUDIENCE -- who do you think will visit your site?
- GOALS -- what do you want to accomplish with this site?
- FLOWCHART -- map out a logical layout of pages to guide visitors through your site

Planning -- types of webpages

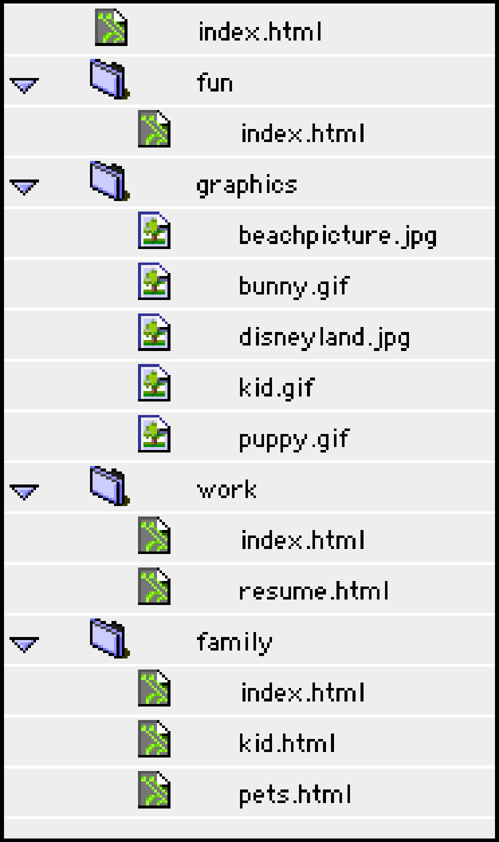
- Types of pages
  - ***Personal Home Pages*** -- http://web.stanford.edu/people/markb/
    - Audience: family, friends, future employers, people who want to get to know you better
    - Goals: To share personal information
  - ***Business pages*** -- http://www.fedex.com/
    - Audience: clients, associates, workers
    - Goals: Bring in new business, sell products, provide information about products
  - ***Informational pages*** -- http://dizpins.com/
    - Audience: general public, specifically those with an interest in the topic
    - Goals: to provide information

Basic Web Site Organization

Once you know your audience and the goals of your site, it's time to think about *how* you're going to present your information. Create a flowchart of the webpages you plan to design. Group web pages together by topic -- putting all your family web pages in one directory and web pages related to a hobby in another directory. However you decide to organize your site, it should make sense to YOU. Put some thought into the process before you start -- and map it out well -- so that you only have to do it once.



Organization chart



- index.html
- fun
  - index.html
- graphics
  - beachpicture.jpg
  - bunny.gif
  - disneyland.jpg
  - kid.gif
  - puppy.gif
- work
  - index.html
  - resume.html
- family
  - index.html
  - kid.html
  - pets.html

After you figure out what pages you want on your website, you need to think about creating logical directories (folders) to store them. Again, it's a good idea to group pages by themes in different directories.

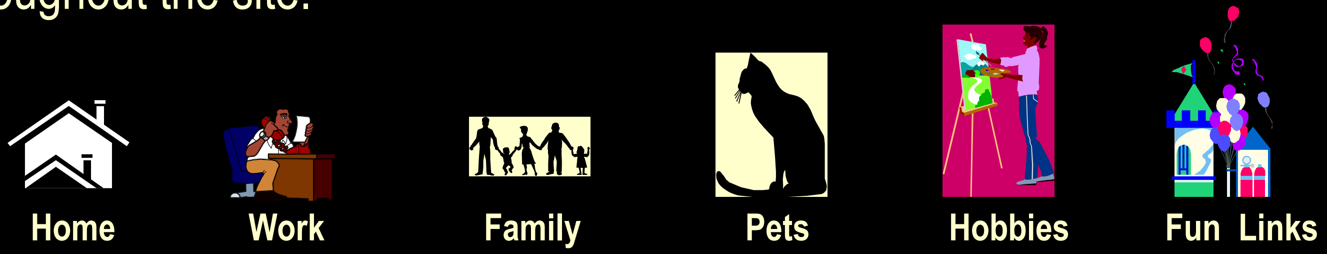
You should also think about creating a separate directory for all graphics/photos.

Organizing for your visitors

In addition to thinking about the organization of your web site from the creator's perspective, you must also put some thought into presenting your web pages so that visitors to your site will be able to easily move from page to page. One helpful way to do this is through the use of a navigation bar. These can take several forms, from simple text to more complicated graphical forms. Text bars can be something like this:

| Home | Job | Family | Pets | Hobbies | Fun Links |

Graphics navigation bars could be a graphical version of the same thing. It is important to not abandon your visitors deep within your site. Having a navigation bar also gives a sense of continuity throughout the site.



In-class assignments

- In-class HTML assignments
- Byet Host account creation (if time permits - otherwise, it’s homework for next class!):

  https://byet.host/free-hosting/news

Preview of next week…

- Adding links
- Adding style (color, font, size)