

Beyond the Mouse – A Short Course on Programming

7. HTML/CSS or:
“I’m on the

INTERNET!!!11!!”

Ronni Grapenthin

Geophysical Institute, University of Alaska
Fairbanks

October 24, 2010

OUR PAGES ARE NOW
HTML, XHTML-STRICT, AND
HAIKU-COMPLIANT.)

HAIKU? <DIV CLASS="MAIN">

BLOG! </DIV>



“Not Enough Work” (panel 3),
<http://xkcd.com/554>

“We are here to help you learn how to learn.”

*Prof. Dr. Klaus Bothe,
Humboldt-Universität Berlin*

“We are here to make you appreciate free knowledge after presenting the concepts.”

me, here

`http://www.google.com/#q=html+tutorial`

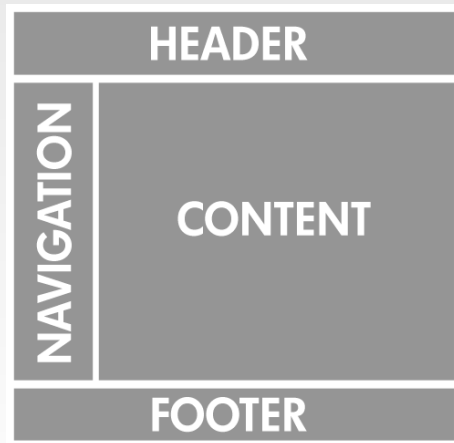
`http://www.google.com/#q=css+tutorial`

`http://www.w3schools.com`

Before anything ...

Think about:

- the content
- a structure
- a layout
- maintenance

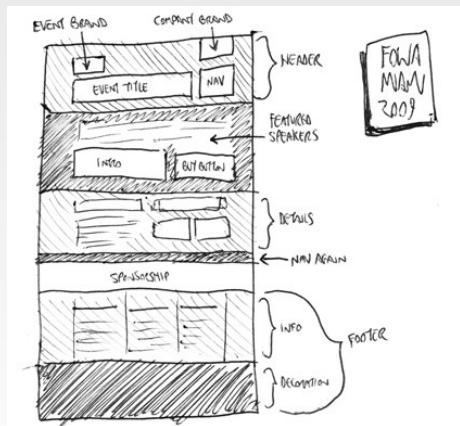


stolen off the Internet

Before anything ...

Think about:

- the content
- a structure
- a layout
- maintenance

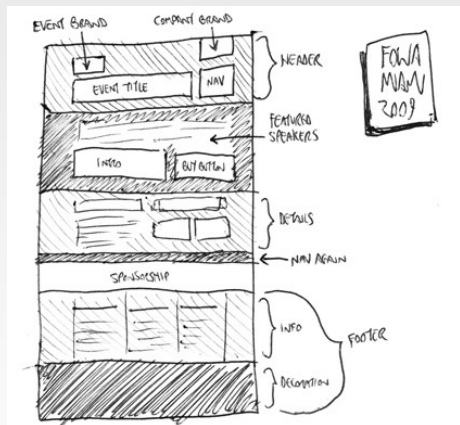


stolen off the Internet

Before anything ...

Think about:

- the content
- a structure
- a layout
- maintenance
- nothing's more lame than an outdated website!



stolen off the Internet

website = content + structure + layout
= content + **HTML** + **CSS**

website = content + structure + layout
= content + **HTML** + **CSS**

- **HTML:** HyperText Markup Language
- **CSS:** Cascading Style Sheets

- (e**X**tensible) HyperText **M**arkup **L**anguage: The innovate idea were links!

- (e**X**tensible) HyperText Markup Language: The innovate idea were links!
- *hyper- + text*; coined by Ted Nelson circa 1965 (source: wiktionary.org):

A hypertext system, then, is a memex-like device for creating and manipulating hypertexts, both for on-line browsing, and for reducing selected portions of such texts . . .

S. Carmody, W. Gross, T. Nelson, D. Rice, and A. van Dam

"A Hypertext Editing System for the /360" (1969)

- (e**X**tensible) HyperText Markup Language: The innovate idea were links!
- *hyper- + text*; coined by Ted Nelson circa 1965 (source: wiktionary.org):

A hypertext system, then, is a memex-like device for creating and manipulating hypertexts, both for on-line browsing, and for reducing selected portions of such texts . . .

S. Carmody, W. Gross, T. Nelson, D. Rice, and A. van Dam

"A Hypertext Editing System for the /360" (1969)

- different versions exist – use XHTML (call it HTML anyway)!
- it's a markup language, **not** a programming language
- **tags** to describe web sites:

```
<TAGNAME attribute="value"> ...</TAGNAME>
```

HTML tags

- Websites are interpreted/displayed by browsers (Firefox, Chrome, Opera, Safari, IE, lynx)
- Browser manufacturers have different ideas of standards / functionality – then, there's confusion.

simple (invalid) website

```
<html>
  <head>
    <title> Browser header</title>
  </head>

  <body>
    <h1>Dear Mama, I'm on the Internet!</h1>
    <a href="http://www.gps.alaska.edu/programming">Here's where I learned that stuff!</a>
  </body>
</html>
```

simple (invalid) website

```
<html>           <!-- HTML tag opens document (this is a comment!) -->
<head>          <!-- start the header part ( here goes meta information) -->
  <title>Browser header</title>
  <!-- e.g. the title of the website, nested in the header -->
</head>         <!-- close ALL tags! -->

<body>         <!-- This is where the good stuff is – the body of the page -->
  <h1>Dear Mama, I'm on the Internet!</h1> <!-- a header (includes linebreak) -->
  <a href="http://www.gps.alaska.edu/programming">Here's where I learned that stuff!</a>
  <!-- finally, a link! -->
</body>
</html>        <!-- end of the story -->
```

simple valid website

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<!-- A DOCTYPE Declaration is mandatory, without one validation of a document is impossible -->

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<!-- Document Types based on XML need a mandatory xmlns="" on the root element. That's life -->

<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <!-- Declaring a character encoding helps ... use this line -->
  <title> Browser header</title>
</head>

<body>
  <h1>Dear Mama, I'm on the Internet!</h1>
  <a href="http://www.gps.alaska.edu/programming">Here's where I learned that stuff!</a>
</body>
</html>
```

<http://validator.w3.org> is your friend!

- **Cascading Style Sheets**
- define how to display HTML elements
- allow to keep data and their representation separate
- can be external (in separate file) – saves a lot of work
- different CSS for different media (screen, print, ...)
- definition of CSS inside a HTML file contrary to original intention
- **Syntax:** `TAGNAME[#id] [TAGNAME] { attribute1: value; attribute2: value; }`

simple CSS example

```
body{
  color: black; background-color: white;
  font-size: 100.01%;
  font-family: Helvetica , Arial ,sans-serif;
  margin: 0; padding: 1em 0;
  text-align: center; /* centering in internet explorer */
}

a{
  color: black; background-color: white;
  font-size: 100.01%;
  font-family: Helvetica , Arial ,sans-serif;
  margin: 0; padding: 0em 0;
  text-align: center; /* centering in internet explorer */
}

h1 {
  font-size: 1.25em;
  color: #666666;
  margin: 0; padding: 0.3em;
  text-align: right;
  background: #fff url(./background.jpg) no-repeat -10% 20%;
  background-width:100%;
  border: 1px solid black;
}
```


Different CSS for different occasions:

```
...  
<link rel="stylesheet" href="/common/shared.css?254z2" type="text/css" media="screen" />  
<link rel="stylesheet" href="/common/commonPrint.css?254z2" type="text/css" media="print" />  
<link rel="stylesheet" href="/monobook/main.css?254z2" type="text/css" media="screen" />  
<link rel="stylesheet" href="/chick/main.css?254z2" type="text/css" media="handheld" />  
...
```

From <http://en.wikipedia.org>

It's a **website**, right?

Necessities for websites:

- the Internet
- a webserver (check with your advisor, or Paul Delys / Dave Covey) to get access to GI server
- a copy tool: scp, sftp, rsync (Windows: winscp, FileZilla)
- work on a local mirror, though!

It's a **website**, right?

Necessities for websites:

- the Internet
- a webserver (check with your advisor, or Paul Delys / Dave Covey) to get access to GI server
- a copy tool: scp, sftp, rsync (Windows: winscp, FileZilla)
- work on a local mirror, though!

my `webupdate` script

```
#!/bin/csh
# takes folder in ~/www that's to be updated on fairweather as
# argument
```

```
if ($#argv < 1) then
  echo "Usage: $0 <folder in ~/www>"
  exit
endif
```

```
rsync -avz --delete --exclude="*~" ~/www/$1 ronni@fairweather.gps.alaska.edu:/export/ftpweb/htdocs
```

It's a **website**, right?

Some useful things

- most webservers per default look for `index.html` as a start site
- use Gimp or other tools that have color palettes to get “webcolor”
- whitespaces in HTML document are ignored by browser – format neatly though!
- work on a local mirror, though!

Then there are Dynamic Websites . . .

Program reacting on input from website, is executed on server, creates HTML code and sends this back to client.