

Beginning HTML/CSS

11 November 2015 / 4:00 - 6:00 PM / Olin 155

WICC Instructors

Haewon Hwang (hh474)

Nikita Gupta (ng354)

Jerica Huang (jh2263)

Inyoung Hong (ih235)

Outline

Before we begin...

Sublime

File Directory Structure

HTML: Basics

blah

blah

CSS: Basics

blah

blah

Next time...

Before we begin...

Download Sublime

File Directory Structure

HTML Basics

1. The Skeleton

```
<!DOCTYPE html>

<head>
    ...information about your website, not visible on the page
</head>

<html>
    <body>
        ...the meat of your page that is visible
    </body>
</html>
```

2. Tags

- Open <body>; close </body>
 - Everything inside <body> goes between these open/close tags
 - Think of them as (opening and closing) parentheses
- Nesting and indentation

```
...
<body>
    <div>
        <h1> ... </h1>
        <p> ... </p>
        <img />
    </div>
</body>
```

3. Elements

- In <head>...
 - <title>Your Website Title</title>
 - Link to CSS file... see CSS section below
 - Links to JS files
 - Other information such as favicon, meta tags, etc.
- In <body>...
 - Text

```
<p>Paragraph</p>
```

```
<h1>Header 1</h1>
```

- There are 6 of these! h1 ... h6

○ Links

- Attribute: href

- Note: absolute vs. relative paths (see below #4)

```
<a href = "http://www.google.com">Google</a>
```

○ Images

- Attributes: src, alt

- Note: absolute vs. relative paths (see below #4)

- Note: self-closing tags (see below #6)

```
<img src = "img.jpg" alt = "image" />
```

○ Lists

- Ordered (numbered)

```
<ol>
  <li> List item </li>
  <li> List item </li>
  <li> List item </li>
</ol>
```

Output:

```
1. List item
2. List item
3. List item
```

- Unordered (not numbered)

```
<ul>
  <li> List item </li>
  <li> List item </li>
  <li> List item </li>
</ul>
```

Output:

```
List item
List item
List item
```

- You can have lists inside lists (nesting)!

```

<h1>Info. Sci. Requirements</h1>
<ul>
  <li> Core
    <ol>
      <li>INF01300</li>
      <li>MATH1110</li>
      <li>CS2800</li>
    </ol>
  </li>
  <li> Track
    <ol>
      <li>Classes...</li>
    </ol>
  </li>
</ul>

```

Output:

```

Info. Sci. Requirements
Core
  1. INF01300
  2. MATH1110
  3. CS2800
Track
  1. Classes...

```

○ Elements that will help with CSS styling

■ Divisions: <div> things </div>

- Use these to “section” off your website content into reasonable chunks. This is mainly for ease in CSS styling.

■ Span: text

- Use these to style small, specific portions inside textual (paragraph, header, etc.) tags

4. Absolute vs. Relative paths (<a> and)

- Absolute: using URLs to link things

```
<a href = “google.com”> Google </a>
```

```
<img src = “google.com/img.jpg” alt = “google image” />
```

- Relative: using your file path to link things

```
<a href = "page2.html"> Page 2 </a>
```

```
<img src = "img/image.jpg" alt = "image" />
```

5. Wrapping

- Make your images clickable/hyperlinked!

```
<a href = "google.com"><img src = "img/image.jpg" alt =  
"image" /></a>
```

Clicking on that image will take you to Google.com

6. Self closing tags

- and <link> elements don't need closing tags because nothing goes "inside" or "between" them. Instead, just close with /> of the main tag

```
<img ... /> instead of <img ... > </img>
```

```
<link ... /> instead of <link ... > </link>
```

7. Comments

- You can write notes, skeleton code, anything inside comments that won't get processed and displayed on the page

```
<!-- This is an HTML comment -->
```

8. Formatting

- Escape characters

- Certain characters are special and need escaping inside HTML text

■ "&" → &

- Text formatting

- **bold letters**

- *italicized letters* <u> underlined letters </u>

CSS Basics

1. Types

- Inline
- Internal
- External

2. External stylesheets

```
<link type = "text/css" rel = "stylesheet" href = "stylesheet.css" />
```

3. Syntax

```
selector {
    property: value ;
}
//
p {
    color: red;
}
```

Output: all <p> element text will be red

- Comments

```
/* This is a CSS comment */
```

- Universal selector: asterisk, will format ALL html elements on the page

```
* { styling...; }
```

- Nested selectors

HTML:

```
<div>
    <div>
        <h1> Header </h1>
    </div>
</div>
```

CSS:

```
div div h1 { styling... ; }
```

4. Styling

- Color: `color`, `background-color`
 - Can be defined with standard names: `red`, `blue`, `green`, etc.
 - Can be defined with hex codes: `#0000ff`
 - Can be defined with `rgb` and `rgba` coordinates: `rgb(255, 255, 255)` or `rgba (0, 0, 0, .5)` ← `a` (alpha coordinate) = transparency
- Font
 - Size: `font-size`
 - `px`
 - `em`
 - Family: `font-family`
 - backup font families in case one isn't currently available
 -
- Align: `align`
 - `Left`, `center`, `right`
- Height / Width


```
height: 500px;
width: 200px;
```
- Text decoration


```
text-decoration: none;
```

<a> links are by default blue and underlined. This will get rid of that.
- Display
 - Block: makes the element a block box, won't let anything sit next to it, takes up the full width


```
display: block;
```
 - Inline-block: makes the element a block box, will allow other elements to sit next to it on the same line


```
display: inline-block;
```
 - Inline: not a block box, makes element sit on same line as another element, does not take up the full width -- only as much as it needs to take up


```
display: inline;
```
 - None: renders that element invisible


```
display: none;
```
- Float / Clear
 - `Left`, `right`
 - `Clear: both;`

- Positioning

- Default = static
- Absolute

```
position: absolute;
```

- Relative

```
position: relative;
```

- Fixed: anchors that element to the browser window... if you scroll up/down, the fixed element stays put even as other elements scroll past it

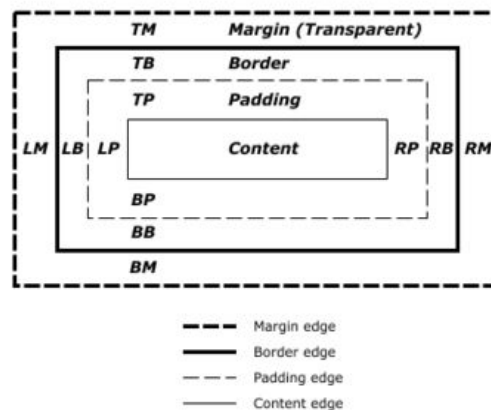
```
position: fixed;
```

5. ID vs. Class

- ID: only once
- Class: style multiple items at the same time
 - can have multiple, whichever class is further down the CSS page will override the rest
- Single element can have both an ID and a class(es)

6. Box Model

- Margin → Border → Padding → Content
- Top Right Bottom Left



7. Chrome Inspector

Resources

CodePen

Codecademy

W3Schools

Next time...

Responsive design

Javascript / JQuery (scrolling, buttons change colors, size, event handling)

PHP (forms, basic name email text send)

User testing / design

Make it live!