CMPS 179

UX for Designing 3D, Animation, and Interaction for the Web

Professors: Reid Swanson & Matt Maclaurin
TA: Peter Mawhorter
• Bring laptops if you have them
• Please vote for office hours
• If you are going to form groups let me know who is in your group by next Wednesday (April 10).
• Tufte giving one day lecture in San Jose on May 2 (Link to event on course website)
CMPS 179: Some Notes & Announcements

• Skills
  – Statistics & Data analysis
  – Computation & programming
  – Web programming
  – Design
  – Art

• Can’t cover all in one course
• New course
• Be creative & playful
• Let’s get started with some basic HTML
• Simple markup language like XML
• The fist line of an HTML 5 file should be `<!DOCTYPE html>`
• Opening tags are specified by `<tagname>` and closing tags with `</tagname>`
  – Not case sensitive
• Usually come in pairs but not always
  – E.g., `<br>`
• All documents start with `<html>` and end with `</html>`
• Comments `<!-- A comment -->`
Two broad categories of tags
- *Block*: `<p>`, `<h1>`, `<ul>`, `<table>`, etc.
- *Inline*: `<b>`, `<a>`, `<img>`, `<td>`

Tags generally have semantic meaning
- `<p>` is for paragraphs
- `<code>` is for code
- `<ul>` is for lists
- `<em>` for adding emphasis
- In HTML 5 even `<B>`, `<I>`, `<U>` have meaning

Use styles to change the display
- `<div>` and `<span>` especially useful
• Full list of HTML tags are available at:

• Some interesting new HTML 5 tags
  – `<audio>` for embedding sound content
  – `<video>` for embedding video content
  – `<canvas>` for drawing graphics

• Some tags were removed
  – `<applet>, <font>, <frame>, <center>`
  – See the diff at:
    • [http://www.w3.org/TR/html5-diff/#obsolete-elements](http://www.w3.org/TR/html5-diff/#obsolete-elements)
• Tags can have attributes
• Some apply to all html tags
  – *id*: give the element a unique identifier
  – *class*: apply a named style rule to the element
  – *style*: allows direct input of style code
• Some only apply to specific tags
  – *href*: `<a>`
  – *border*: `<table>`
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
</head>
<body>
  <h1>Header</h1>
  <a href='index.html'>home</a>
</body>
</html>
Code

```html
<h1>HTML List</h1>
<ul>
  <li>First Item</li>
  <li>Second Item</li>
  <li>Third Item</li>
</ul>
```

Rendered

**HTML List**

- First Item
- Second Item
- Third Item
Code

```html
<h1>HTML List</h1>
<ul>
  <li>First Item</li>
  <li>Second Item</li>
  <li>Third Item</li>
</ul>
```

Rendered?

HTML List

First Item    Second Item    Third Item
• CSS allows you to define the display attributes of HTML nodes
• Simple syntax: `<selector> { <declaration>;* }`

```html
li
{
  display: inline;
  list-style-type: none;
}
```
• *: applies to every element on the page
  – * { width: 100%; }

• tag: applies to every element whose tag name is tag.
  – p { text-align: right; }

• #identifier: applies to the single element on the page whose id attribute is identifier.
  – myid { color: red; }

• .name: applies to any element on the page whose class attribute is name.
  – .center { margin: 0 auto; }
**CMPS 179: DOM Selectors**

- **X Y**: applies to selectors Y who are descendants of X.
  \[
  \text{div p \{...\}}
  \]
- **X > Y**: Like X Y but only applies to direct children.
  \[
  \text{div p \{...\}}
  \]
- **X + Y**: Only select Y if it is the element immediately after X.
  \[
  \text{p div \{...\}}
  \]
• Many more...
  – http://www.w3.org/TR/selectors/

• Can be combined in interesting ways
  – p.horizontal_list
  – p, h1 {...}

• Class attributes can have multiple styles
  – <p class="center dialog">
• Formatting text
  – E.g., p { font-size: 12pt; }
• Sizing elements
  – E.g., div { width: 100%; }
• Setting the spacing around elements
  – E.g., h1 { margin: 10px; }
• Positioning elements
  – E.g., #footer { position: absolute; }
• Adding graphics
  – E.g., body { background-image: url('myimage.png'); }
• Coloring elements
  – E.g., h1 { color: green; }
• % - percentage value
• \texttt{in, cm, mm} – value in real measurement units
• \texttt{em} – 1em equals the current font size. \textit{nem} is a multiplier to the current value.
• \texttt{pt} – Common measurement for fonts (1/72in)
• \texttt{px} – Value in pixels.
• CSS3 adds lots of cool stuff
  – Better box model (e.g., rounded corners)
  – 2D / 3D transformations
  – Animations
• **External Style Sheet** (in the `<head>` section)
  – `<link rel="stylesheet" type="text/css" href="doc.css">`
  – (note no closing link tag)
• **Internal Style Sheet** (in the `<head>` section)
  – `<style> (valid css) </style>`
• **Inline**
  – In the *style* attribute of an HTML element
• **Multiple style sheets can be included (cascaded)**
CMPS 179: Dive In

- **CSS Reference**
  - [http://www.w3schools.com/cssref/#gsc.tab=0](http://www.w3schools.com/cssref/#gsc.tab=0)

- **Skeleton**
  - [http://codepen.io/reidswanson/pen/zynwf](http://codepen.io/reidswanson/pen/zynwf)

- **Target**
  - [http://codepen.io/reidswanson/pen/CKIxh](http://codepen.io/reidswanson/pen/CKIxh)

---

**CSS Layout**

**Min-height**

The `.colmask` element of this page has a min-height of 100%. That way, if the content requires more height than the viewport provides, the height of `.singlecolumn` forces `.colmask` to become longer as well. Possible columns in `.singlecolumn` can then be visualized with a background image on `.colmask`; divs are not table cells, and you don’t need (or want) the physical elements to create such a visual effect. If you’re not yet convinced, think wobbly lines and gradients instead of straight lines and simple color schemes.

**Relative positioning**

Because `.colmask` has a relative position, #footer will always remain at its bottom: since the min-height mentioned above does not prevent `.colmask` from scaling, this will work even if (or rather especially when) `.singlecolumn` forces `.colmask` to become longer.

**Padding-bottom**

Since it is no longer in the normal flow, padding-bottom of `.singlecolumn` now provides the space for the absolute #footer. This padding is included in the scrolled height by default, so that the footer will never overlap the above content.

Scale the text size a bit or resize your browser window to test this layout. The [CSS file is over here](http://games.soe.ucsc.edu).