

CSS

INFO 253A: Frontend Web Architecture
Kay Ashaolu

CSS

Cascading Style Sheets

Style

- color, size, visibility, positioning
- Specified separately from HTML
- Why separate?

CSS

```
1 <strong>hello</strong>
2
3 strong {
4     font-weight: bold;
5     color: magenta;
6     font-size: 45px;
7 }
```

How else could we do it?

Just like the funky ASCII symbols, we could set the style of each element

```
1 <!-- Warning: Invalid HTML -->
2 <strong font-color="magenta" font-size="45px">hello</strong>
```

CSS File

```
1 h1 {  
2     color: #B3D4FC; /* light blue */  
3     text-align: center;  
4 }
```

- Now all main headers are centered and have a light blue look
- What's the trade-off we made?

ID & Class attributes

- id: Identifier. Unique per page
- class: Grouping. Multiple per page and per element

```
1 <strong id="logo" class="big red">Pinterest</strong>
```

What's wrong with the above?

Compromises

- Truth is, most sophisticated sites make a compromise
- Sharing styles between semantic elements
- Frameworks that are used by different sites

Leaky Abstraction

- When details of the composition or layers merge
- Often must write your HTML with knowledge of how you will style it
- But avoid too much coupling!
- **Bootstrap**

Element Selectors

```
1 strong {  
2     font-weight: bold;  
3     color: red;  
4     font-size: 45px;  
5 }
```

- Selectors specify the element to apply a style
- This is selecting all strong elements

ID & Class Selectors

```
1 #logo {
2     font-size: 64px;
3     color: red;
4 }
5
6 .symbol {
7     font-size: 32px;
8     color: darkblue;
9 }
10 <span id="logo">Pinterest</span>
11 <span class="symbol">PINTEREST</span>
12 <span class="symbol">New York Times</span>
```

Comma = OR

```
1 h1, h2, h3 {  
2   color: #B3D4FC; /* light blue */  
3   text-align: center;  
4 }
```

all h1 or h2 or h3 elements

Descendant

- Think nested tags
- Let's write a table in HTML
- What is an example of a descendant?

Descendant CSS = (space)

```
1 table a {  
2     font-weight: bold;  
3 }
```

Cascading & Inheritance

- Some properties are passed down to descendants, like font-family
- Some elements may be affected by multiple rules, which "cascade"
- Most specific wins

Cascading

```
1 table {
2   font-family: sans-serif;
3   color: blue
4 }
5
6 .data {
7   border: 1px solid rgba(0, 0, 0, 0.2);
8   color: red
9 }
10 <table>
11 <tr><td>Color blue</td></tr>
12 <tr><td class="data">Color red</td></tr>
13 </table>
```


What's the trade-off?

- Now we can specify styles in another file
- Can select groups of tags or tags with IDs
- But how can we style individual tags?

Style Attribute

- Change the style of individual elements in-line in HTML
- Value of the style attribute is the same format as the definition block
- Only for very special cases!

```
1 <h1 style="color: brown;  
2     background-color: yellow;">  
3     Individual</h1>
```

The power of CSS

- CSS Zen Garden: One site, many designs
- [Garden](#)
- [Under the Sea](#)

Sneak Peak

- JavaScript can manipulate CSS too!
- Composability: they play well together, but don't require each other

Overview

