



TalentRiot

Week 2: HTML + CSS

October 16, 2013



What's ahead

- Goals
- HTML
- CSS
- Connecting CSS to HTML
- Note of Frameworks and Boilerplate
- Weekly Assignment



Anatomy of a website

- HTML: Structure/Markup
- CSS: Look and Feel
- JavaScript: Brains/Logic

Let me show you



HTML: Purpose



```
8      <title>Dependable Software Developers - TalentRiot</title>
9
10     <link rel="stylesheet" href="css/reset.css" type="text/css">
11     <link rel="stylesheet" href="css/style.css" type="text/css">
12     <link rel="stylesheet" href="http://fonts.googleapis.com/css?family=Noto+Serif:400,700,400italic" type="text/css">
13     <link rel="stylesheet" href="webfonts/ss-standard.css" type="text/css">
14     <link rel="author" href="http://www.brandongrant.com"/>
15     <link rel="shortcut icon" href="favicon.ico">
16
17     <!-- enable HTML5 elements in IE7+8 -->
18     <!--[if lt IE 9]>
19         <script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
20     <![endif]-->
21
22     <script src="js/mailto.js"></script>
23
24 </head>
25 <body>
26     <header class="group">
27         <div class="site-width">
28             <div role="navigation" id="nav">
29                 <ul>
30                     <li><a href="#intro" class="nav-button active">Home</a></li>
31                     <li><a href="#software" class="nav-button">Custom Software</a></li>
32                     <li><a href="#developers" class="nav-button">Developers</a></li>
33                     <li><a href="#why" class="nav-button">Why TalentRiot?</a></li>
34                     <li><a href="#values" class="nav-button">Values</a></li>
35                     <li><a href="#careers" class="nav-button">Careers</a></li>
36                     <li><a href="#contact" class="nav-button">Contact</a></li>
```

HTML Terminology



- Element

`<a>`

- Tags

`<a>...`

- Attributes

`Talent Riot`



HTML Structure & Syntax

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Hello World</title>
  </head>
  <body>
    <h1>Hello World</h1>
    <p class="hero">This is a website.</p>
  </body>
</html>
```

HEAD



- `<title>`
- `<meta type="*">`
- `<link src="">`
- `<script></script>`
- `<style></style>`

BODY



- `<div>`
- ``
- `<p>`
- `<table>`
- `<form>`
 - `<input>`
- Many more!



Important Attributes

- `class=""`
- `id=""`
- `on=""`
- `title=""`
- `tabindex=""`



Attribute Order

- class
 - id
 - data-*
 - for, type, href
-
- `Example Link`

Dos & Don'ts



- Use soft tabs with two spaces
- Nested elements should be indented once (2 spaces)
- Always use double quotes, never single quotes
- Lower-case tags
- Use tables only for tabular data
- Know the difference between ``, ``, ``



Note on SEO

- `<h1>`, `<h2>`, `<h3>` tags
- `<p>` tags
- lists
- Sitemap
- Inbound links

Breathe



CSS



- What's the point of CSS
- How does it work?
- Definition order: top->down
- Specificity



Connecting CSS to HTML

- **Inline**

```
<div style="margin: 10px">...</div>
```

- **Embedded**

```
<head>
```

```
...
```

```
  <style>
```

```
    .my-class {
```

```
      margin-left: 10px;
```

```
    }
```

```
  </style>
```

```
</head>
```

- **External**

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


CSS Terms: Selectors



- Universal
- Element Type
- Class
- ID
- Attribute
- Pseudo-classes
- Pseudo-elements

`* { ... }`

`p { ... }`

`.class-name { ... }`

`#idAttr { ... }`

`a[src$=.png] { ... }`

`li:first-child { ... }`

`p:before { ... }`

CSS Terms: Properties



- Box
- Layout
- List
- Table
- Color
- Typography

```
p {  
    padding: ...  
    color: ...  
    font-size: ...  
}
```

CSS Terms: Values



- cssvalues.com
- Lots of em!

```
p {  
  padding: 10px;  
  color: #ff0;  
  font-size: 16px;  
}
```

Combining Selectors



- Descendant Selector

```
.nav li {  
    list-style: none;  
}
```

Combining Selectors



- Child Selector

```
ul>li {  
    list-style-type: disc;  
}
```

Combining Selectors



- Adjacent Sibling

```
h2+p {  
    font-color: yellow;  
}
```

Combining Selectors



- General Sibling (CSS3)

```
h2~p {  
    font-color: yellow;  
}
```

Combining Selectors



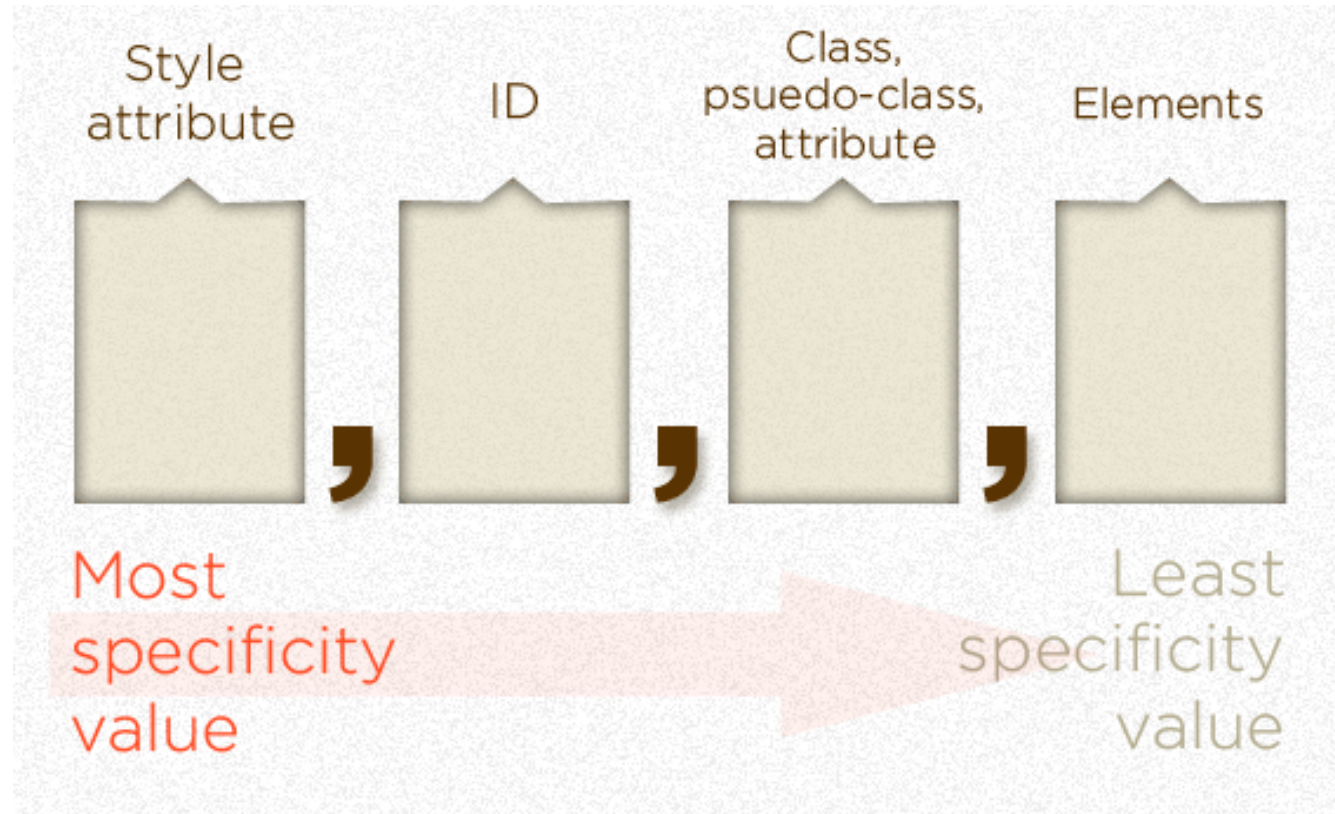
- Combination
 - Don't do this

```
.big.yellow.button {  
    font-color: yellow;  
}
```


Breathe



CSS Specificity



css-tricks.com/specifics-on-css-specificity

Specificity: Style



```
<li style="color: red;">
```

Style
attribute



,

ID



,

Class,
psuedo-class,
attribute



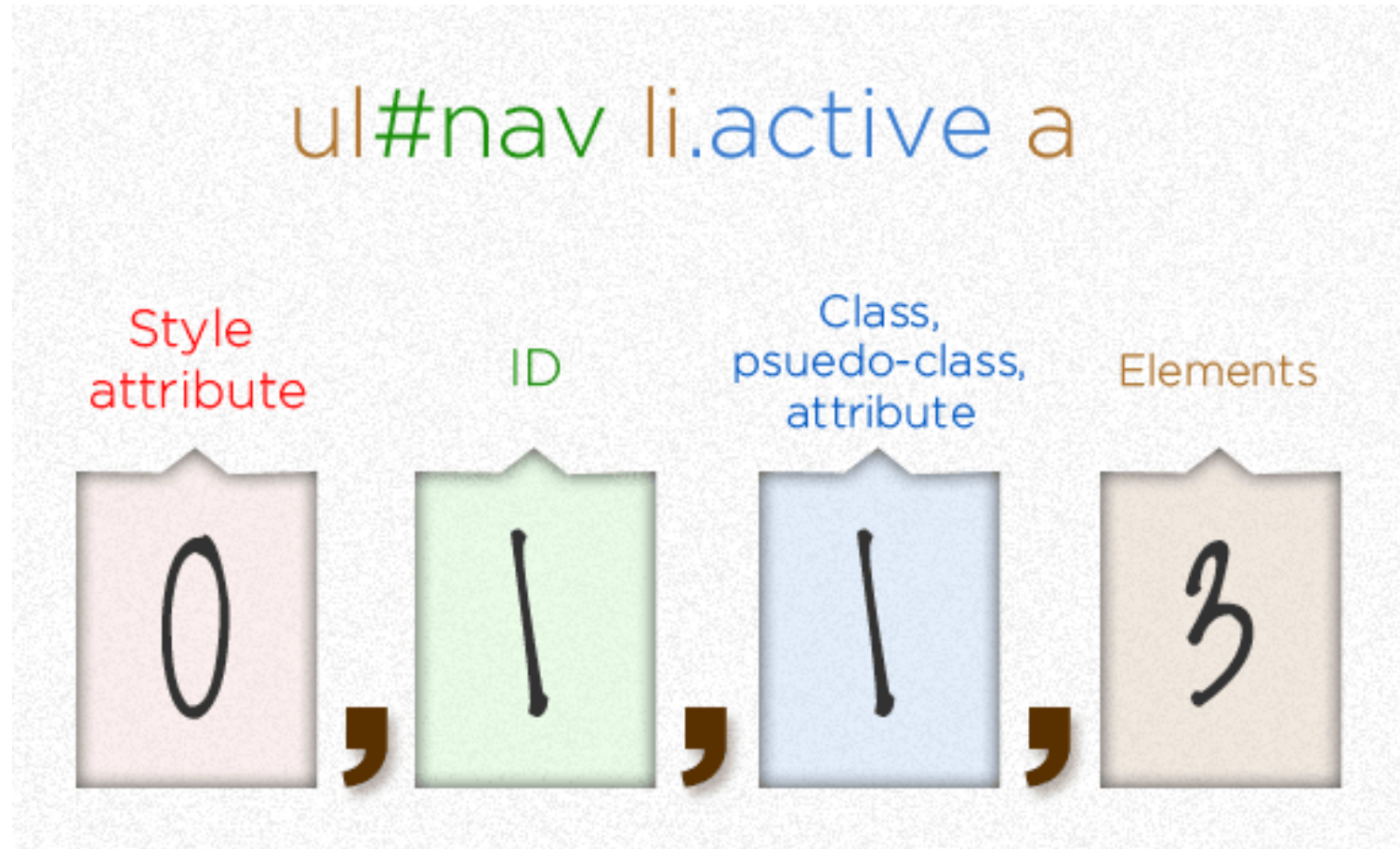
,

Elements





Specificity: ID, class, elements





Specificity Notes

- The universal selector (*) has no specificity value (0,0,0,0)
- Pseudo-elements (e.g. :first-line) get 0,0,0,1 unlike their psuedo-class brethren which get 0,0,1,0
- The pseudo-class :not() adds no specificity by itself, only what's inside it's parentheses.
- The **!important** value appended a CSS property value is an automatic win. It overrides even inline styles from the markup.



Best Practices

- Avoid universal key selector
`* { ... }`
- Don't qualify ID rules with tag names or classes
~~`div#header`~~
- Avoid the descendant selector
`.header` ~~`→ .menu`~~
- Remove redundant qualifiers
`div#nav.item` `->` `#nav li`

Breathe



CSS Preprocessors



- Great, familiar, smart
- LESS
- Stylus
- SASS



Front End Frameworks

- Bootstrap
- Foundations
- Blueprint
- Skeleton
- Front End Frameworks Compared:
[usablica.github.io/front-end-frameworks/
compare.html](https://usablica.github.io/front-end-frameworks/compare.html)

Recap



- HTML: Structure/Markup
- CSS: Look and Feel
- Connecting them together
- Tools and further reading
- “HTML and CSS: Design and Build Websites” by Jon Duckett

Assignment



- Create a personal website that hyperlinks to your github/other places
- Experiment with styling and layout
- Commit your work and push it to the repository you've created

Example: mwop.net



phly, boy, phly: matthew weier o'phinney

HOME

BLOG

RÉSUMÉ

CONTACT

Welcome!

I'm Matthew Weier O'Phinney. I am:

- A husband.
- A father of two wonderful children.
- A PHP developer and framework architect.
- A frequent PHP conference speaker.
- An open source software developer and advocate.
- A (sometimes) brewer of beer.



Recent Github Activity

- [weierophinney pushed to develop at zendframework/zf-web](#)
- [weierophinney pushed to master at zendframework/zf-web](#)
- [weierophinney pushed to master at zfcampus/zf-apigility-skeleton](#)
- [weierophinney pushed to develop at zendframework/Component ZendForm](#)
- [weierophinney pushed to master at zendframework/Component ZendForm](#)

[Read more](#)



About

[Tom MacWright](#) [About](#) [Archive](#) [Talks](#) [Code](#) [@tmcw](#)



Tom MacWright is sort of a coder who works at [MapBox](#). He lives in Washington, DC and plays guitar in [Teen Mom](#).

If you want to get in touch, [send an email](#).

[Tom MacWright](#) [Blog](#) [Projects](#) [About](#) [Archive](#) [@tmcw](#) [RSS](#)



zeno rocha
FRONT-END ENGINEER

Blog

Talks

Projects



Hi, I'm *Zeno*. I am a 22 year old front-end engineer who lives in *Rio* and works for an awesome company based in *LA* :)

Also a co-founder of [BrazilJS Foundation](#)¹, host on [Zone of Front-enders](#)² and graduated in Information Systems at [UNIRIO](#)³.

I used to work as a front-end developer at [Globo](#)⁴, software developer at [Petrobras](#)⁵ and consultant mentor at [Papaya Ventures](#)⁶. Now I'm at [Liferay, Inc.](#)⁷ working on a open source project called [AlloyUI](#)⁸.

sberkmada.com



AMK

Adam Krebs builds things.

- ✉ email
- 📁 github
- ⚙️ dev blog
- 📁 projects
- 📍 4sq
- 🌐 linkedin
- 📄 resume
- 👉 diversions

Don't do this



Cya Next Week!

