

JMS 302: HACKING THE MEDIA

CLASS 9: 9/16/2014

TODAY'S TASKS



- Intro to CSS
- Writing Ruby Methods

REVIEW

- Flow Control
 - comparisons, looping, branching
 - ==, !=, <=, if/else, while, &&, ||, !
- Arrays

MISC HTML

- HTML5 (what it is vs. what most people mean)
- Always specify basic doctype
- `<!DOCTYPE html>`
- Comments:
- `<!-- This is a comment -->`

B

What HTML5 is: the latest specification (candidate) for the HTML standard

What most people mean when they use the term: A website (not a native app) that will run on any standard browser (especially mobile ones) and doesn't use flash. Often, they are contrasting with flash-based sites (especially with video) or native apps.

MISC HTML

- C** • id & class attributes
- B** • block & inline elements
- C** • div & span
- B** • iframe
- A** • meta:
 - description, keywords, robots, author, pragma, expires
- A** • escape characters
 - & > " ™ ...

THE POWER OF CSS



THE SAME HTML, DIFFERENT CSS FILES
CSSZENGARDEN.COM

INTRO TO CSS(3)

- selectors & declarations
- external css (link href type rel)
- internal style



```
h1, h2, h3 { font-family: Arial; color: yellow; }
```

better style ((note the lack of quotes, also note that you can specify an element many times)

```
h1, h2, h3 {  
  font-family: Arial;  
  color: yellow;  
}  
h1 {  
  font-size: 30px;  
}
```

SELECTORS

SELECTOR	MEANING	EXAMPLE
UNIVERSAL SELECTOR	Applies to all elements in the document	* <code>h1</code> Targets all elements on the page
TYPE SELECTOR	Matches element names	<code>h1</code> , <code>h2</code> , <code>h3</code> <code>h1</code> Targets the <code>h1</code> , <code>h2</code> and <code>h3</code> elements
CLASS SELECTOR	Matches an element whose class attribute has a value that matches the one specified after the period (or full slash symbol)	<code>.color</code> <code>h1.color</code> <code>h2</code> Targets any element whose class attribute has a value of <code>color</code> Targets only <code>h1</code> elements whose class attribute has a value of <code>color</code>
ID SELECTOR	Matches an element whose id attribute has a value that matches the one specified after the pound or hash symbol	<code>#color</code> <code>h1#color</code> <code>h2</code> Targets the element whose id attribute has a value of <code>color</code>
CHILD SELECTOR	Matches an element that is a direct child of another	<code>h1 > h2</code> Targets any <code>h2</code> elements that are children of an <code>h1</code> element (but not other <code>h2</code> elements in the page)
DESCENDANT SELECTOR	Matches an element that is a descendant of another specified element (not just a direct child of that element)	<code>h1 * h2</code> Targets any <code>h2</code> elements that are inside a <code>h1</code> element, even if there are other elements nested between them
ADJACENT SIBLING SELECTOR	Matches an element that is the next sibling of another	<code>h1 + h2</code> <code>h2</code> Targets the first <code>h2</code> element after any <code>h1</code> element (but not other <code>h2</code> elements)
GENERAL SIBLING SELECTOR	Matches an element that is a sibling of another, although it does not have to be the directly preceding element	<code>h1 ~ h2</code> <code>h2</code> If you had two <code>h2</code> elements that are siblings of an <code>h1</code> element, this rule would apply to both

CASCADING & INHERITANCE

- Last, Specific
- !important overrides **B**
- Inheritance
- Some do; some don't. Can specify `inherit`.

A

TESTING ACROSS BROWSERS, DEVICE

- saucelabs.com
- browserstack.com
- browsershots.org
- crossbrowsertesting.com

FYI

YOUR OWN RUBY METHODS

- Defining vs. calling (def)
- Parameters
- Local variables & scope*
- Return values, nil



You can always put parentheses around parameters. Sometimes they aren't needed. Without extra efforts, parameters defined on a method become required.

*This trips up many people.