

HTML

DEEPER INTO HTML

WHAT WE'LL BE COVERING

- ▶ Semantics
- ▶ `<div>` and ``
- ▶ Text Elements
- ▶ Structure
- ▶ Links

SEMANTICS

COOL WORD, WHAT DOES IT MEAN?

- ▶ se·man·tics

the meaning of a word, phrase, sentence, or text.

- ▶ Example:

Oh man, they're on fire

There is a bird on the TV

Dude

- ▶ Without context, these phrases are hard to understand.

HOW DOES IT APPLY TO HTML?

- ▶ A computer interpreting code, doesn't understand ambiguous context.

- ▶ **Bad Non-Semantic Headline**

```
<font size="6"><b>My Page Headline</b></font>
```

- ▶ **Good Semantic Headline**

```
<h1>My Page Headline</h1>
```

WHY ARE SEMANTICS IMPORTANT?

- ▶ Without explaining what a piece of content is, a computer has no way of identifying it.
- ▶ Semantic code aids accessibility
- ▶ Search engines need to understand what your content is about in order to rank you properly on search engines.

WHY ARE SEMANTICS IMPORTANT?

- ▶ Semantic code is shorter and downloads faster.
- ▶ Easier to understand and read.
- ▶ Good semantic code re-enforces separation of style from content.
 - ▶ Allows easier visual changes through CSS
 - ▶ You can change the visual style without changing the HTML, so... less coding.

EXAMPLES OF SEMANTIC HTML ELEMENTS

```
<html>
```

```
<address>
```

```
<article>
```

```
<header>
```

```
<footer>
```

```
<nav>
```

```
<section>
```


<div> and

DIV AND SPAN ELEMENTS

- ▶ Have no semantic value.
- ▶ Used as containers for styling.
- ▶ We'll use them
(especially <div>) very often.

BLOCK ELEMENTS AND INLINE ELEMENTS

- ▶ <div> are block elements
- ▶ are in-line elements.

BLOCK LEVEL ELEMENTS

- ▶ Begin on a new line
- ▶ Stack on top of each other
- ▶ Take up any available width.
- ▶ Example:
<http://codepen.io/ianshea/pen/XXaYqZ>

INLINE ELEMENTS

- ▶ Do not begin on a new line
- ▶ Line up one after another right *inline*
- ▶ Maintain the width of their content within the tag
- ▶ Example:
<http://codepen.io/ianshea/pen/BjdVOj>

USAGE

- ▶ Use `<div>` to identify groupings of content
- ▶ Use `` to style pieces of content
- ▶ We can control the visual styling of these elements using classes and ids
- ▶ Example:
<http://codepen.io/ianshea/pen/vLJrQo?editors=110>

TEXT ELEMENTS

CONTENT CONTENT CONTENT

- ▶ Most content on the web is text.
- ▶ Content is searchable or 'indexable' by search engines.
- ▶ We've seen and used these text elements already!
`<h1>` tags and `<p>` tags.

TEXT ELEMENT EXAMPLES

```
<h1>Headline Level 1</h1>
```

```
<h2>Headline Level 2</h2>
```

```
<h3>Headline Level 3</h3>
```

```
<h4>Headline Level 4</h4>
```

```
<h5>Headline Level 5</h5>
```

```
<h6>Headline Level 6</h6>
```

```
<p>Some rad paragraph text</p>
```

SEMANTICS FOR TEXT ELEMENTS

- ▶ We use `` tags to place strong importance on text.
- ▶ We use `` tags to semantically emphasize text.
- ▶ Remember! ``s will not provide semantic value

```
<p><em>Pizza</em> is the <strong>best</strong>.</p>
```

STRUCTURE

BUILDING PAGE STRUCTURE

- ▶ We can use semantic HTML elements to build structure.
- ▶ We can use <div>s inside these elements to provide additional “structural support”.
- ▶ Structural elements are usually block level elements
- ▶ Structural based elements:

```
<header> <nav> <article> <section>  
<aside> <footer>
```

BUILDING PAGE STRUCTURE

- ▶ Structural example

<http://codepen.io/ianshea/pen/QyMBwM?editors=110>

`<header>`

- ▶ Identify the top of a page, article, section or any other element on your page.
- ▶ Not to be confused with the `<head>` or `<h1>` elements.
- ▶ `<head>` elements provide instructions to the browser.
ex: The `<title>` element and link to a CSS file
- ▶ `<h1>` may live within a `<header>` and denote levels of headings.

`<nav>`

- ▶ Identifies navigation on your page.
- ▶ Used for global navigations, or large sections of navigation.
- ▶ For example:
It would not be used to wrap `<a>` tags in your list of sites from Assignment #1.

<article>

- ▶ Identifies independent self-contained content.
- ▶ Examples:
 - ▶ A blog post
 - ▶ User comments
- ▶ Article does not explicitly imply an “article” like in a newspaper. Articles are pieces of **self-contained content**.

`<section>`

- ▶ Identifies grouping of thematic content
- ▶ Usually has a `<header>` but not always.
- ▶ Is somewhat generic, but should not be used instead of a `<div>`
- ▶ Remember, it should identify a thematic grouping of content.
- ▶ Example - A comments section on a blog.
<http://codepen.io/ianshea/pen/XXaBMo?editors=110>

`<article>` or `<section>` or `<div>`

- ▶ To decide which to use, look at your content
- ▶ `<div>` are used primarily for controlling visual style. We'll use these A LOT.
- ▶ `<article>` are used for independent content
- ▶ `<section>` are used for groups of thematic content

`<aside>`

- ▶ Used for content that is related to other content
- ▶ Example:

```
<article> ... Blog Post ... </article>
```

```
<aside> About the Author </aside>
```

- ▶ It is named “**aside**” but it’s a block level element and it will not be naturally positioned to the side of anything.

`<footer>`

- ▶ Identifies closing of an article or page.
- ▶ Usually found at the bottom of it's parent.
- ▶ Contains relative content to what it's closing.

`<footer>`

- ▶ Identifies closing of an article or page.
- ▶ Usually found at the bottom of it's parent.
- ▶ Contains relative content to what it's closing.

LINKS

<A> TAGS

- ▶ Moving around the internet entails clicking on <a> tags.
- ▶ <a> tags are an inline element.
- ▶ <a> tags require an href element so it knows where to send a user when clicked.

RELATIVE VS. ABSOLUTE

- ▶ Relative links are for linking to items that reside on the same server or file system as our document that contains the link.
- ▶ Absolute links direct someone to an explicit location.

RELATIVE VS. ABSOLUTE

▶ Relative Links

- ▶ href attribute value does not contain a domain

Example: *href="about.html"*

- ▶ for linking to **local** assets within the same website

▶ Absolute Links

- ▶ href attribute value Will contain a full domain

Example: *href="http://example.com/about.html"*

- ▶ For linking to **external** sites or assets

RELATIVE VS. ABSOLUTE

/ your / website / files /

```
<link href="main.css">  
<a href="about.html">About</a>
```

index.html

about.html

```
<link href="main.css">  
<a href="index.html">Home</a>  
<a href="http://github.com">Github</a>
```

