

HTML/CSS 2

Uchicago Techteam

Semantics

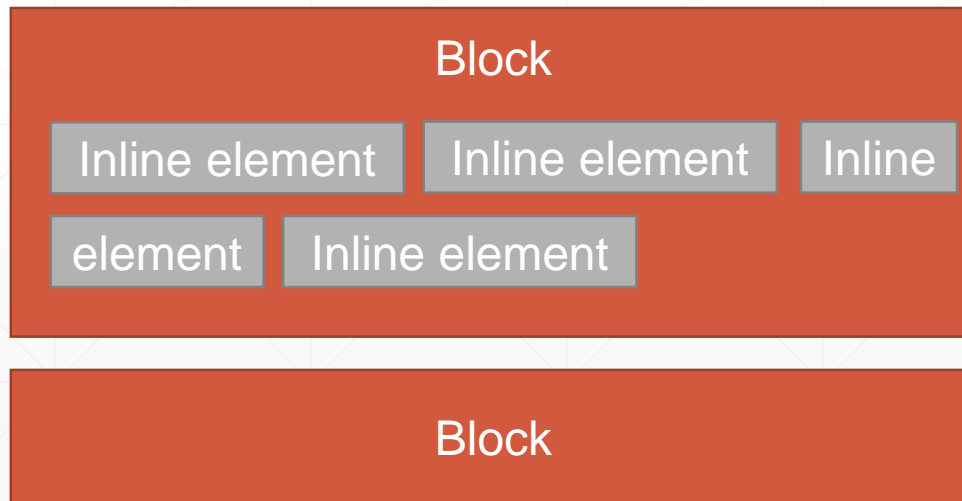
- Describes the value of page content, not appearance
 - These two elements have same appearance with different semantics:
 - `This is the page title`
 - `<h1>This is a heading</h1>`
-

Why care about semantics?

- Usually shorter
 - Easier for search engines to understand the content
 - Easier for other web designers to pick up the code
 - Easier to make changes to design (design – content separation)
 - HTML elements can be grouped by:
 - Non-semantic elements
 - Text-based
 - Structure
 - Hyperlinks, etc.
-

Non-semantic: Divisions & Spans

- HTML elements solely used for styling purposes
- `<div></div>`: block-level styling element. Used for large groupings of contents
- ``: inline-level styling element. Used for small groupings



Blocks: begin in new line, spans the entire width

Inlines: fall into normal flow, maintains content width

Inlines cannot wrap blocks!!

Example of Div/Span's usage

```
<div class="social">
```

```
<p>I may be found on...</p>
```

```
<p>Additionally, I have a profile on...</p>
```

```
<p>Soon we'll be <span id="tooltip">writing HTML</span> with the best of  
them.</p>
```

```
</div>
```

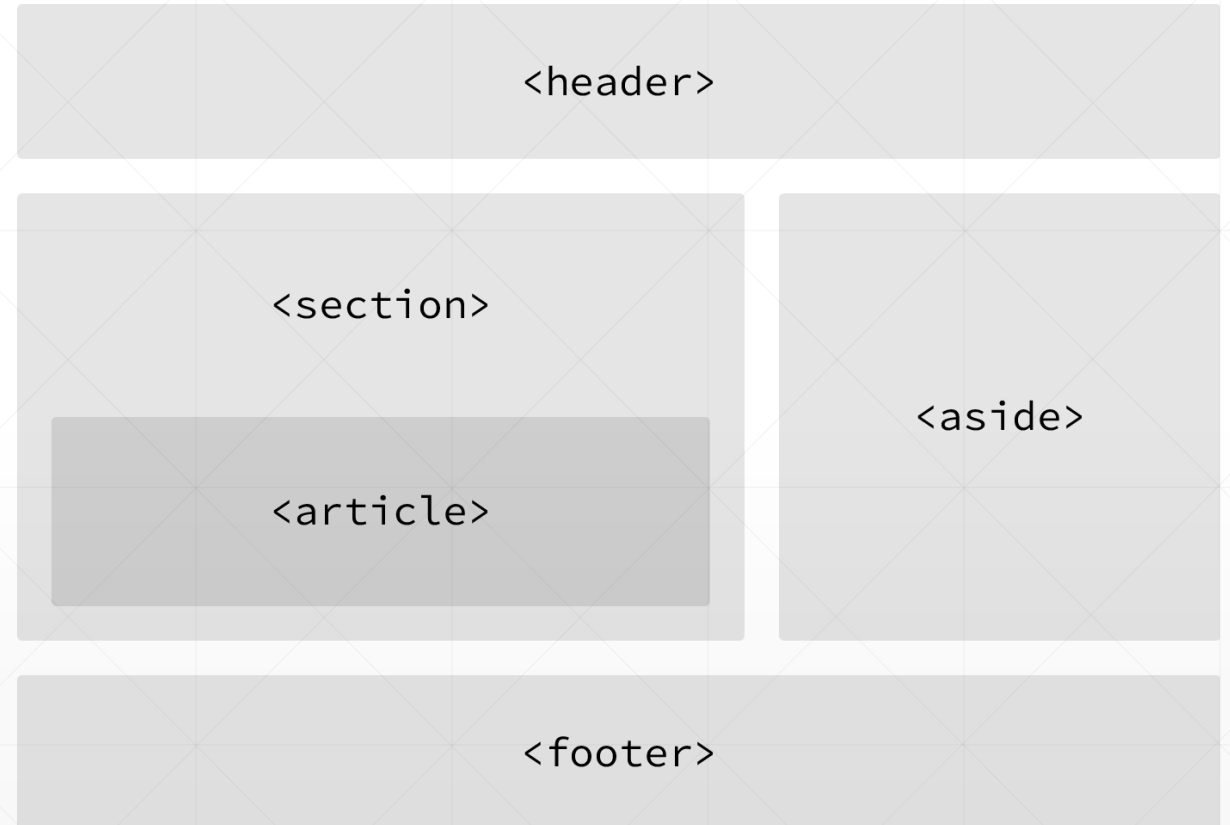
Note the use of class and id attributes for div and span!

Text-based elements

- `<h1>` ~ `<h6>`: breaks up contents and establishes hierarchy
 - `<p>`: supporting paragraphs
 - `` and ``: for bold texts
 - `` for style + semantics, `` for style only
 - `<p>`This recipe calls for ``bacon`` and ``baconnaise```</p>`
 - `` and `<i>`: for italicized texts
 - `` for stressed emphasis, `<i>` for alternative voice/tone
 - `<p>`I ``love`` Chicago!`</p>` `<p>`I `<i>`love`</i>` Chicago!`</p>`
-

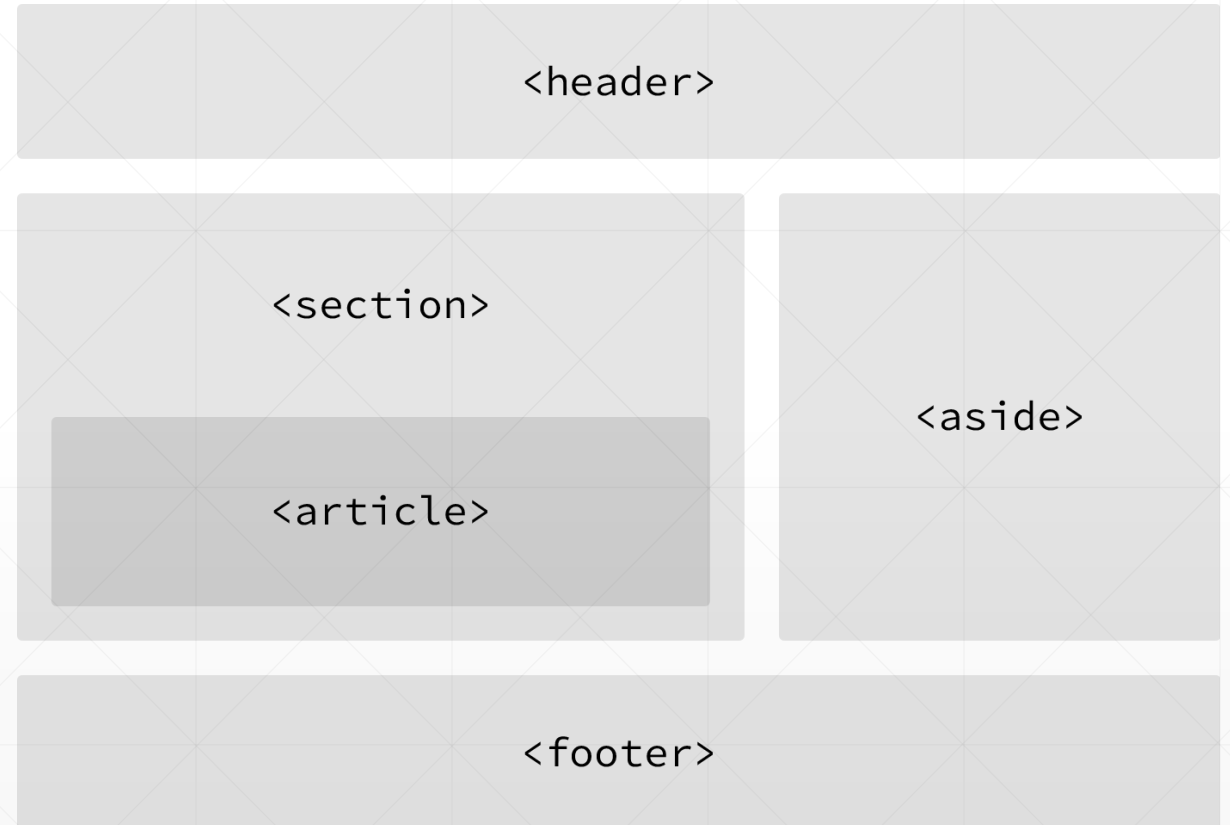
Building structure

- Gives meaning to the organization of each page
- All block elements
 - But different from `<div>`s
 - Note that `<section>` and `<aside>` are also block-level, spanning the entire width



Building structure

- `<header>`: heading, logos, introductory texts, navigations...
- `<section>`: for thematic grouping of a content (could be multiple)
- `<aside>`: content related to surrounding elements
 - Sidebar, inserts, brief explanation, author info...
- `<footer>`: closing of a page
- `<article>`: self-contained content that could be replicated in other pages



Try HTML/CSS on your own!

- (Follow the instructions on <http://learn.shayhowe.com/html-css/getting-to-know-html/#practice-1>)
 - Step 0: First, comment out what you have written last week on your .html and .css
 - Step 1: Add a <header> with <h1>, <h3> included
 - Step 2: Add a <section> with <h2>, <p> included
 - Step 3: Add a nested <section> (Try to include all 3 sections! Copy from downloaded file)
 - Step 4: Add a footer with encoded special character
-

Try HTML/CSS on your own!

- Step 5: Add red borders to all the structure-based elements and black borders to all the text-based elements (border: 2px solid red; & border: 2px solid black;)
 - Step 6: Set the width of <section> elements to 200px
 - Step 7: Pick a few words and bold/italicize the text by inserting tags + try changing their background color to yellow.
-

Creating Hyperlinks

- anchor `<a>` element with href attribute
 - `Click me!`
 - Only inline element that could wrap block elements
 - Try yourself!
 - Wrap the first `<section>` element with the `<a>` `` tags
 - Wrap the `<h5>Speakers</h5>` element with the `<a>` `` tags
 - If you want the link to lead nowhere, you can put href='#'
-

Relative vs. Absolute path

- The value of the “href” attribute could be...
 - Relative to the path of your current .html file or
 - Absolute URL assigned to your .html page

`About`

`Events`

`About`

Other hyperlink functionalities

- Email links

```
<a href="mailto:aa@gmail.com?subject=Reaching%20Out&body=How%20are%20you?">Email me!</a>
```

- Opening link in new window

```
<a href="about.html" target="_blank">About</a>
```

- Linking to parts of the same page

```
<body id="top">
```

...

```
<a href="#top">Back to top</a>
```

```
</body>
```

Try:

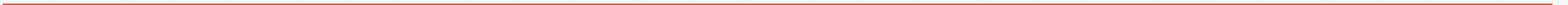
- Adding an <a> element within <footer> that leads to the first section of your same page
- For the <a> element that you wrapped <small> with, add a new attribute target="_blank" and see what happens

Back to CSS: The Cascade

```
p {  
  background: orange;  
  font-size: 24px;  
}
```

```
p {  
  background: green;  
}
```

```
p {  
  background: orange;  
  background: green;  
}
```



Specificity weights

- Determines which style takes precedence
 - $X(\text{num of id selectors}) - X(\text{class}) - X(\text{type})$
- $p \{ \} : 0 - 0 - 1$
- $.\text{food} \{ \} : 0 - 1 - 0$
- $\#\text{hotdog} \{ \} : 1 - 0 - 0$

```
#hotdog {  
  color: red;  
}
```

1 – 0 – 0

```
.food {  
  color: blue;  
}
```

0 – 1 – 0

```
p {  
  color: black;  
}
```

0 – 0 – 1

Combining Selectors

```
<div class="hotdog">
  <p>...</p>
  <p>...</p>
  <p class="mustard">...</p>
</div>
```

```
.hotdog p {
  background: brown;
}
.hotdog p.mustard {
  background: yellow;
}
```

- Multiple selectors are read from right to left
 - Rightmost selector: key selector
 - The rest: pre-qualifiers
 - .hotdog p
 - Applies style to “p element WITHIN an element of class .hotdog”
 - p.mustard
 - If there is no space in between, this means “.mustard class that is ALSO p”
 - div a p
 - “p element WITHIN a WITHIN div”

Thanks!

You can review the materials at <http://learn.shayhowe.com/html-css>