

## Lesson 2: Getting to Know HTML

### Semantics Overview

- **semantics** (w/in HTML) = practice of giving content on the page meaning & structure by using the proper element describes the value of content on a page regardless of style or appearance of that content
- benefits of using semantic elements: enabling ... Computers ... to adequately read & understand content on a web page  
screen readers  
search engines  
makes HTML easier to manage & work w/

### Identifying Divisions & Spans

- HTML elements that act as containers solely for styling purposes (no semantic value)
- apply targeted styles to a contained set of content
- **<div>** = block-level element commonly used to identify large groupings of content → helps build web page's layout & design
- **<span>** = inline-level element commonly used to identify smaller groupings of text w/in a block-level element
- **block-level elements** = begin on a new line, stacking one on top of the other & occupy any available width  
commonly used for larger pieces of content (eg. paragraphs)  
may be nested inside one another & wrap inline-level elements
- **inline-level elements** = fall into normal flow of a document (ie. don't begin on a new line), lining up one after the other  
& only maintain width of their content  
commonly used for smaller pieces of content (eg. a few words)  
may be nested inside one another  
can't wrap block-level elements
- want class or id name (attribute value) that refers to content of an element > appearance of element

### Comments w/in HTML & CSS

- HTML comment format: `<!-- ... -->`
- CSS comment format: `/* ... */`

### Using Text-Based Elements

- popular text elements: headings  
paragraphs  
bold text  
italics
- **headings (<h1>-<h6>)** = used to designate multiple levels of text headings throughout a page  
block-level elements  
key identifiers for users that help break up content & establish hierarchy  
help search engines index & determine content on a page
- **paragraphs (<p>)** = block-level element  
often follow headings

- **bold text (<strong>)** = inline-level element

  - (eg. <strong>): element used to give strong importance to text

  - (eg. <b>): element used to stylistically offset text

- **italicize text (<em>)** = inline-level element

  - (eg. <em>): element used to place a stressed emphasis on text

  - (eg. <i>): element used to convey text in an alternate tone or voice

## Building Structure

- **structural elements** = identify groupings of content, give meaning to organization of pages & improve structural semantics
  - block-level elements
  - no implied position or style

- structurally-based elements: <header>

  - <nav>

  - <article>

  - <section>

  - <aside>

  - <footer>

- **header (<header>)** = element used to identify top of a page, article, section, or other segment of a page
  - falls w/in the <body> element

  - may include: heading
  - introductory text
  - navigation

- **navigation (<nav>)** = element used to identify a section of major navigational links on a page
  - included links → link to other pages w/in same website or parts of the same web page

  - should be reserved for primary navigation sections only: global navigation
  - table of contents
  - previous/next links
  - other noteworthy groups of navigational links

  - misc. one-off links → anchor element (<a>)

- **article (<article>)** = element used to identify a section of independent content that may be independently distributed
  - Content could be replicated somewhere else w/o any confusion
  - or reused

- **section (<section>)** = element used to identify a thematic grouping of content
  - generally includes a heading
  - commonly used to break up & provide hierarchy to a page

- **aside (<aside>)** = element that holds content tangentially related to the content surrounding it: sidebars  
inserts  
brief explanations  
  
→ (eg. w/in <article>): identify content related to author of the article
- **footer (<footer>)** = element used to identify closing/end of a page, article, section, or other segment of a page  
content should be relative info & shouldn't diverge from document or section it's included w/in

→ (eg. w/in <article>): identify content related to author of the article

- **footer (<footer>)** = element used to identify closing/end of a page, article, section, or other segment of a page  
content should be relative info & shouldn't diverge from document or section it's included w/in

## Encoding Special Characters

- special characters can be misunderstood in HTML ∴ they need to be encoded
- encoded character format: & ... ;

→ (eg. &copy;): copyright symbol ©

## Creating HyperLinks

- **hyperlink** = provides the ability to link from one web page or resource to another established via anchor (<a>) element & hyperlink reference (href attribute)

- format: `<a href="url">text</a>` → `<a href="http://shayhowe.com">Shay</a>`

- **anchor (<a>)** = inline-block element w/ specific permission to wrap any level elements

- paths = href attribute values

- **relative path** = what links pointing to other pages of the same website will have  
doesn't include domain in path

- **absolute path** = what links to other websites outside of the current one requires path must include full domain

- **email link**: when clicked → opens user's default email client & pre-populates parts of the email

- format: `<a href="mailto:emailaddress">text</a>` → `<a href="mailto:shay@awesome.com">Email me</a>`

- optional subject & body parameters format: `<a href="mailto:emailaddress?subject=...&body=...">text</a>`



`<a href="mailto:shay@awesome.com? subject=Reaching%20out&body=How%20are%20you">Email me</a>`

- open link in new window

- format: `<a href="url" target="_blank">text</a>` → `<a href="http://shayhowe.com/" target="_blank">Shay</a>`

- link to part of the same page (e.g. Back to top)

```
- format: <body id="top">                -> <body id="top">
          <a href="#top">text</a>           <a href="#top">Back to top</a>
        </body>                            </body>
```