

1 HTML5 and CSS3

STRUCTURING WEB PAGES WITH HTML

3 Basic Structure of an HTML Document

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
  </head>
```

```
  <body>
```

```
  </body>
```

```
</html>
```

5 The "Head" Element—Page Title

- The one required element (and often the first item) inside the <head> area is the page <title> element
- The "page title" is not displayed on the web page but rather on the page's browser tab
- Format:
 - <title>*document title*</title>
- Example:
 - <title>My Web Page</title>

7 Comments

- Notes or comments about the code
- Can be spread out over several lines
- Format:
 - <!-- *comment* -->
- Example:
 - <!-- This is my comment
 Author: Carl B. Struck -->

9 HTML5 Structural Elements

- HTML5 structural elements are a set of new tags that help define the "sections" within the body of a web page
- Used along with CSS3, can enhance structural layout
- The elements include:
 - article—subsection covering a single topic
 - aside—contains side issues to main topic (perhaps off to the right side on the main text)

- footer—content placed at the bottom of the page
- header—content placed at the top of the page
- nav—navigation list of links (often in left column of page)
- section—major topical area on page

12 The div Element

- The `<div>` tag is used in versions earlier than HTML5 to define the “sections” within the body of a web page
 - Versions that do not use the HTML5 structural elements
- Rarely used in HTML5 (last resort if nothing else works) although there are hundreds of millions of legacy web pages that still use it
- Format:
 - `<div id = "id">`
 content
 `</div>`

13 The Heading Elements

- A set of “grouping elements” that contains text that define “titles” for the main sections of the content of a web page
- Displayed in bold and double-spaced after the heading
- Format:
 - `<hn>content</hn>`
 - ... in which *n* is a value between 1 and 6 denoting a larger to smaller font size
- Example:
 - `<h1>My First Heading</h1>`
 - The `<h1>` tag usually denotes a “major” heading

15 Grouping Headings

- The `<hgroup>` block is used for documentation to possibly to indicate that a main title and subtitle belong together
- Format:
 - `<hgroup>`
 heading elements
 `</hgroup>`
- Example:
 - `<hgroup>`
 `<h1>My Main Title</h1>`
 `<h2>My Subtitle</h2>`
 `</hgroup>`

17 Paragraph Elements

- The `<p>` tag is a “grouping element” that defines a “paragraph” on a web page
- Implements double-spacing after each paragraph

○Format:

✦ `<p>content</p>`

○Example:

✦ `<p>Welcome to HTML. In this unit you will learn about the structure of a web document and some of its basic elements.</p>`

19 **Block Quotes**

- The `<blockquote>` tag is a grouping element used to mark “extended quotes”

- The browser usually indents the text

- Format:

○ `<blockquote>`
 content
`</blockquote>`

○Example:

✦ `<blockquote>`
 `<p>This is an extended quote ...</p>`
`</blockquote>`

21 **Addresses**

- The `<address>` tag is a group element to that renders text in *italics*

○ There are few tags that are used to format for italics

- “Logically” used when the text is an *address*

- Format:

○ `<address>content</address>`

- Example:

○ `<address>Carl B. Struck`
 1000 Crooked Hill Road
 Brentwood, NY 11717
`</address>`

23 **Ordered Lists—Page 1**

- The `` tag creates a list of items (*list items* using a `` block) that are sequential

- By default a numbered list

- Format:

○ ``
 `content`
 `content`
 ...
``

24 **Ordered Lists—Page 2**

- Example:

```
○ <ol>
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ol>
```

- Displays as:

```
1. First item
2. Second item
3. Third item
```

25 **Unordered Lists—Page 1**

- The `` tag creates a list of items (*list items* using a `` block) that are non-sequential
- By default a bulleted list
- Format:

```
○ <ul>
  <li>content</li>
  <li>content</li>
  ...
</ul>
```

26 **Unordered Lists—Page 2**

- Example:

```
○ <ul>
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ul>
```

- Displays as:

```
• First item
• Second item
• Third item
```

27 **Nested Lists—Page 1**

- A sub-list at some point within a list
- Adds an additional level of indentation
- For unordered lists the browser usually uses a different bullet symbol at each level

28 **Nested Lists—Page 2**

- Example:

```
○ <ul>
```

```

<li>First item</li>
<li>Second item
  <ul>
    <li>First sub-item</li>
    <li>Second sub-item</li>
  </ul>
</li>
<li>Third item</li>
</ul>

```

29 **Nested Lists—Page 3**

- Previous page displays as:
 - First item
 - Second item
 - First sub-item
 - Second sub-item
 - Third item

30 **Description Lists—Page 1**

- The <dl> block creates a list of terms <dt> followed by the descriptions <dd>, e.g. the definitions
- By default descriptions <dd> are indented an additional level
- Format:
 - <dl>


```

<dt>term1</dt>
<dd>description1</dd>
<dt>term2</dt>
<dd>description2</dd>
...
</dl>

```

31 **Description Lists—Page 2**

- Example:
 - <dl>


```

<dt>Boy</dt>
<dd>A young male</dd>
<dt>Girl</dt>
<dd>A young female</dd>
<dt>Dog</dt>
<dd>A friend to both</dd>
</dl>

```

32 **Description Lists—Page 3**

- Previous page displays as:
 - Boy
 - A young male
 - Girl
 - A young female
 - Dog
 - A friend to both

34 **External Style Sheets**

- Style sheets are written in CSS (Cascading Style Sheet) language and implement web page formatting and layout
- A web page can link to a separate (external) style sheet file
- The link tag is placed in the head section
- Format:
 - `<link href = "location/filename" rel = "stylesheet" type = "text/css" />`
- Example:
 - `<link href = "mystyles.css" rel = "stylesheet" type = "text/css" />`

36 **HTML5/CSS3 and Browser Compatibility**

- "Modernizer" is a free, open-source library of functions that provides support for HTML5 and newer CSS elements for many older browsers
- Written in the scripting language JavaScript
- A script tag that runs "modernizr-1.5.js" is placed in the head section (script will run when the web page loads)
- The src attribute indicates path and filename of script file
- Format:
 - `<script src = "file"></script>`
- Example:
 - `<script src = "modernizr-1.5.js" ></script>`

38 **Text-Level Elements**

- Text-level elements flow alongside of (inline with) characters (not on a new, separate line)
- Format:
 - `<element>content</element>`
- Example:
 - `<p>Welcome to HTML</p>`
 - ✦ The `` tag formats text to display "emphasized" in italics, e.g. "Welcome to *HTML*"
 - The `<i>` tag also displays italic text

39 **Some Text-Level Elements**

- `b` text in boldface
- `cite` a citation usually in *italics*
- `code` program code usually in fixed width font
- `em` emphasized text usually in *italics*
- `i` text in *italics*
- `q` quoted "text"
- `small` text in a smaller font
- `strong` strongly emphasized text usually in boldface
- `sub` subscripted text₃
- `sup` superscripted text³

40 **“Strong” Text**

- The `` tag is a text-level element that displays text in boldface
- Format:
 - `content`
- Example:
 - `<p>Welcome to HTML </p>`
 - ✦ Displays as “Welcome to HTML”
 - The `` tag also displays boldface text

42 **Line Breaks**

- The `
` tag is a one-sided element that inserts a one-line break within a grouping element such a paragraph or heading
 - Even if the text is not on a new line in the HTML code
- Format:
 - `
`
- Example:
 - `<p>Welcome to
HTML</p>`
 - ✦ Displays as:
 - Welcome to
 - HTML

44 **The `` Tag**

- The `` tag is used to group *inline* block elements
- Provides a method to add formatting to just a part of the text or document
- Format:
 - `<element>content content content</element>`
- Example:
 - `<p>My mother has blue eyes.</p>`

45 **The “style” Attribute—Page 1**

- The style attribute is used for *inline* style sheet formatting
 - Highest of the three levels of formatting in CSS and so overrides any style set globally
 - ✦ E.g. styles specified in the <style> tag in the <head> section, or in an external style sheet
- Can be used with almost any HTML tag element

46 The "style" Attribute—Page 2

- Format:
 - `<element style="property1:value1; property2:value2; ... ">content</element>`
- Example:
 - `<p style="color:blue">My mother has blue eyes.</p>`
 - Renders as:
 - My mother has blue eyes

47 The "style" Attribute—Page 3

- Format:
 - `<element style="property1:value1; property2:value2; ... ">content</element>`
- Example:
 - `<p>My mother has blue eyes.</p>`
 - Renders as:
 - My mother has blue eyes

49 Hyperlinks

- A hyperlink is an element that links to another place in the same document or to an entirely different document
 - Usually the hyperlink is clicked to follow that link
- Created using an <a> (anchor) tag
- Format:
 - `content`
- Example:
 - `content`

50 Absolute and Relative URLs

- An absolute address is the *complete* Web address
- A relative is only a *partial* address, which is then taken as "relative to" the current document's address
 - Within a document you can refer to other documents on the same site using a relative address
- Examples:
 - Absolute address:
 - ✦ "http://www.google.com"
 - Relative address:

- ✦ "myLogo.png"

52 **Inline Images**

- The `` tag is a one-sided element that inserts a graphic image file into the document
- Format:
 - ``
 - ✦ `src`—name and location of the graphic image file
 - ✦ `alt`—alternate text displayed if image is not available (called a broken image)
 - ✦ Also there are optional width and height attributes
- Example:
 - ``

54 **Favicon Images**

- A favicon (meaning "favorite icon") is a small "icon" image file displayed next to the page's title on the browser tab
 - Icon images should be 16 × 16 pixels in size
- Created by using a `<link>` tag in the `<head>` section
- Format:
 - `<link rel="shortcut icon" href="path/filename" />`
 - ✦ `rel`—assigned value "shortcut icon" means a *favicon*
 - ✦ `href`—name and location of the graphic icon file
- Example:
 - `<link rel="shortcut icon" href="jpsfavicon.png" />`

56 **Character Sets and Special Characters**

- Display characters from the Unicode set
 - Most commonly used character set on the web is UTF-8, a compressed version of Unicode
- The value for a character is preceded by an ampersand (&) followed by a hash (#) character and then ends with semicolon (;)
- Format:
 - `&#code;`
- Example:
 - `<p>The © symbol</p>`
 - ✦ Displays as "The © symbol"

57 **Some Unicode Codes**

- The Unicode specification is a 16-bit binary character set of 65,536 values
- Examples:
 - `&169;` — ©
 - `&174;` — ®
 - `&188;` — ¼

○ `&189;` — ½

○ `&190;` — ¾

○ `&960;` — π

○ `&8482;` — ™

58 Character Entity References

- A set of special character “names” that substitute for the Unicode numbers

- Format:

○ `&char;`

✦ `©` — ©

✦ `®` — ®

✦ `™` — ™

✦ ` ` — (non-breaking space, e.g. the spacebar)

- Example:

○ `<p>The © symbol</p>`

✦ Displays as “The © symbol”

60 Specifying the Character Set

- A `<meta>` tag that using the `charset` attribute is designed to make certain that the browser employs the correct character set when using Unicode values

○ A required element in the `<head>` section in HTML5

- Format:

○ `<meta charset="character-set" />`

- Example:

○ `<meta charset="UTF-8" />`

✦ An assigned value of “UTF-8” specifies the encoding that is used on the World Wide Web

62 Communicating with Search Engines—Page 1

- A `<meta>` tag that uses the `name` attribute is designed to help search engines find and index the web page

○ An element within the `<head>` section

63 Communicating with Search Engines—Page 2

- Format:

○ `<meta name="author | description | keywords" [content="keywords"] />`

✦ The `content` attribute provides information about the author or description or keywords

- Examples:

○ `<meta name="description" content="The homepage of J-Prop Enterprises" />`

○ `<meta name="keywords" content="J-Prop, juggling props" />`