yoonjeewoo / **WebDevelopmentNote**

No description, website, or topics provided.

4 commits | **1** branch | **0** releases | **1** contributor

Branch: **master** | New pull request | Create new file | Upload files | Find file | Clone or download

yoonjeewoo done — Latest commit 6628b17 2 days ago

readme.md — done — 2 days ago

readme.md

Web Application Development

2. Basic HTML

HTML is a **H**yper**T**ext **M**arkup **L**anguage

Structure of an HTML Page

```
<!DOCTYPE html>
<html>
  <head>
      information about the page
  </head>
  <body>
      page contents
  </body>
</html>
```

Page title: `<title>`

Describes the title of the web page

```
<title>Chapter 2: HTML Basics</title>
```

Paragraph: `<p>`

```
<p>You're not your job.
You're not how much money you have in the bank.
You're not the car you drive.   You're not the contents
of your wallet. You're not your         khakis.  You're
   the all-singing, all-dancing crap of the world.</p>
```

Headings: `<h1>,<h2>,<h3>,,,<h6>`

headings to separate major areas of the page (block)

```
<h1>University of Whoville</h1>
<h2>Department of Computer Science</h2>
<h3>Sponsored by Micro$oft</h3>
```

Horizontal rule: `<hr>`

```
<p>First paragraph</p>
<hr />
```

```
<p>Second paragraph</p>
```

Links: `<a>`

```
<p>
  Search
  <a href="http://www.google.com/">Google</a> or our
  <a href="lectures.html">Lecture Notes</a>.
</p>
```

Block & Inline

- Block Element: elements contain an entire large region of content
  - paragraphs
  - lists
  - table cells
- Inline Element: elements affect a small amount of content
  - bold text
  - code fragments
  - images
  - **must be nested inside a block element**

Images: `<img>`

HTML5 also requires an `alt` attribute describing the image

```
<img src="images/gollum.jpg" alt="Gollum from LOTR" />
```

`title` attribute is an optional tooltip (*on ANY element*)

```
<a href="http://theonering.net/">
  <img src="images/gandalf.jpg" alt="Gandalf from LOTR"
   title="You shall not pass!" />
</a>
```

Line break: `<br>`

Warning: Don't over-use br (guideline: >= 2 in a row is bad)

```
<p>Teddy said it was a hat, <br /> So I put it on.</p>
<p>Now Daddy's sayin', <br /> Where the
heck's the toilet plunger gone?</p>
```

Phrase elements: `<em>, <strong>`

```
<p>
  HTML is <em>really</em>,
  <strong>REALLY</strong> fun!
</p>
```

Comments: `<!-- -->`

```
<!-- My web page, by Suzy Student
     CSE3026, 2nd Semester 2025    -->
<p>CSE courses are <!-- NOT --> a lot of fun!</p>
```

Unordered list: `<ul>, <li>`

```
<ul>
  <li>No shoes</li>
  <li>No shirt</li>
  <li>No problem!</li>
</ul>
<!-- More List -->
<ul>
  <li>Simpsons:
    <ul>
      <li>Homer</li>
      <li>Marge</li>
    </ul>
  </li>
  <li>Family Guy:
    <ul>
      <li>Peter</li>
      <li>Lois</li>
    </ul>
  </li>
</ul>
```

Ordered list: `<ol>`

```
<p>RIAA business model:</p>
<ol>
  <li>Sue customers</li>
  <li>???</li>
  <li>Profit!</li>
</ol>
```

Definition list: `<dl>, <dt>, <dd>`

```
<dl>
  <dt>newbie</dt> <dd>one who does not have mad skills</dd>
  <dt>own</dt> <dd>to soundly defeat
    (e.g. I owned that newbie!)</dd>
  <dt>frag</dt> <dd>a kill in a shooting game</dd>
</dl>
```

Quotations: `<blockquote>`

```
<p>As Lincoln said in his famous Gettysburg Address:</p>
<blockquote>
  <p>Fourscore and seven years ago, our fathers brought forth
    on this continent a new nation, conceived in liberty, and
    dedicated to the proposition that all men are created equal.</p>
</blockquote>
```

Inline quotations: `<q>`

```
<p>Quoth the Raven, <q>Nevermore.</q></p>
```

HTML Character Entities

```
< >      &lt; &gt;
é è ñ    &eacute; &egrave; &ntilde;
™ ©      &trade; &copy;
π δ Δ    &pi; &delta; &Delta;
И        &#1048;
" &      &quot; &amp;
```

HTML-encoding text

```
&lt;p&gt;
  &lt;a href=&quot;http://google.com/search?q=marty&amp;ie=utf-8&quot;&gt;
    Search Google for Marty
  &lt;/a&gt;
&lt;/p&gt;
```

Deletions and insertions: `<del>, <ins>`

```
<p>
  <del>Final Exam</del> <ins>Midterm</ins> is on
  <del>Dec 10-15</del> <ins>Oct 10-11</ins>.
</p>
```

Abbreviations: `<abbr>`

```
<p>
  Safe divers always remember to check their
  <abbr title="Self-Contained Underwater Breathing Apparatus">SCUBA</abbr> gear.
</p>
```

Computer code: `<code>`

```
<p>
  The <code>ul</code> and <code>ol</code>
  tags make lists.
</p>
```

Preformatted text: `<pre>`

```
<pre>
     Steve Jobs speaks loudly
        reality distortion
       Apple fans bow down
</pre>
```

Web page metadata: `<meta>`

```
<meta charset="utf-8" />
<meta name="description"
    content="Authors' web site for Building Java Programs." />
<meta name="keywords" content="java, textbook" />
<meta http-equiv="refresh" content="10, www.naver.com">
<meta http-equiv="content-language" content="ko" />
```

Favorites icon: `favicon`

```
<link href="yahoo.gif" type="image/gif" rel="shortcut icon" />
```

HTML tables: `<table>, <tr>, <td>`

```
<table>
  <tr><td>1,1</td><td>1,2 okay</td></tr>
  <tr><td>2,1 real wide</td><td>2,2</td></tr>
</table>
```

Table headers, captions: `<th>, <caption>`

```
<table>
  <caption>My important data</caption>
  <tr><th>Column 1</th><th>Column 2</th></tr>
  <tr><td>1,1</td><td>1,2 okay</td></tr>
  <tr><td>2,1 real wide</td><td>2,2</td></tr>
</table>
```

Linking to multimedia files

```
<a href="video.avi">My video</a>
```

> MPG, MOV, WMV, RM, SWF, WAV, MID

HTML 5 embedding a video

```
<video src="video.ogv" width="425" height="350"></video>
```

Embedded objects: `<object>`

```
<object data="video.avi" type="video/avi"></object>
```

> attributes: archive, classid, codebase, codetype, data, declare, height, name, standby, type, usemap, width

Parameters: `<param>`

```
<object data="duckhunt.mp4" width="100" height="50">
  <param name="autoplay" value="true" />
</object>
```

Embedding a Youtube video

```
<iframe width="width" height="height" src="videoURL">
</iframe>
```

3. CSS for Styling

Cascading Style Sheets (CSS): `<link>`

```
<link href="style.css" type="text/css" rel="stylesheet" />
```

CSS properties for colors

```
p {
  color: red;
  background-color: yellow;
}
```

Specifying colors

```
p { color: red; }
h2 { color: rgb(128, 0, 196); }
h4 { color: #FF8800; }
```

font-family

```
p {
  font-family: Georgia;
}
h2 {
  font-family: "Courier New";
}
```

More about font-family

```
p {
  font-family: Garamond, "Times New Roman", serif;
}
```

font-size

```
p {
  font-size: 14pt;
}
```

font-weight, font-style

```
p {
  font-weight: bold;
  font-style: italic;
}
```

font

```
p {
  font: italic bold 14px "Arial", cursive;
}
```

CSS comments: /* ... */

```
/* This is a comment.
  It can span many lines in the CSS file. */
p {
  color: red;
  background-color: aqua;
}
```

CSS properties for text

```
/* left, right, center, or justify */
blockquote { text-align: justify; }
h2 { text-align: center; }
/* overline, line-through, blink, or none */
p {
  text-decoration: underline;
}
/* text-shadow */
p {
  font-weight: bold;
  text-shadow: -2px 5px gray;
}
```

CSS properties for backgrounds

```
body {
  background-image: url("images/draft.jpg");
  background-repeat: no-repeat;
  background-position: 370px 20px;
}
```

The list-style-type property

```
ol { list-style-type: lower-roman; }
```

Styling tables

```
table { border: 2px solid black; caption-side: bottom; }
tr { font-style: italic; }
td { background-color: yellow; text-align: center; width: 30%; }
```

The border-collapse property

```
table, td, th { border: 2px solid black; }
table { border-collapse: collapse; }
```

The rowspan and colspan attributes

```
<table>
  <tr><th>Column 1</th><th>Column 2</th><th>Column 3</th></tr>
  <tr><td colspan="2">1,1-1,2</td>
    <td rowspan="3">1,3-3,3</td></tr>
  <tr><td>2,1</td><td>2,2</td></tr>
  <tr><td>3,1</td><td>3,2</td></tr>
</table>
```

Column styles: `<col>, <colgroup>`

```
<table>
  <col class="urgent" />
  <colgroup class="highlight" span="2"></colgroup>

  <tr><th>Column 1</th><th>Column 2</th><th>Column 3</th></tr>
  <tr><td>1,1</td><td>1,2</td><td>1,3</td></tr>
  <tr><td>2,1</td><td>2,2</td><td>2,3</td></tr>
</table>
```

CSS pseudo-classes

```
a:link    { color: #FF0000; }      /* unvisited link */
a:visited { color: #00FF00; }      /* visited link */
a:hover   { color: #FF00FF; }      /* mouse over link */
```

Page Layout

CSS context selectors

```
selector1 selector2 {
  properties
}
```

- applies the given properties to selector2 only if it is inside a selector1 on the page

```
selector1 > selector2 {
  properties
}
```

- applies the given properties to selector2 only if it is directly inside a selector1 on the page (selector2 tag is immediately inside selector1 with no tags in between)

CSS properties for borders

```
h2 { border: 5px solid red; }
h2 {
  border-left: thick dotted #CC0088;
  border-bottom-color: rgb(0, 128, 128);
  border-bottom-style: double;
}
```

- border-color, border-width, border-style
- border-bottom, border-left, border-right, border-top
- border-bottom-color, border-bottom-style, border-bottom-width, border-left-color, border-left-style, border-left-width, border-right-color, border-right-style, border-right-width, border-top-color, border-top-style, border-top-width

Rounded corners with border-radius

```
p {
  border: 3px solid blue;
  border-radius: 12px;
  padding: 0.5em;
}
```

CSS properties for padding

```
p { padding: 20px; border: 3px solid black; }
h2 { padding: 0px; background-color: yellow; }

p {
  padding-left: 200px; padding-top: 30px;
  background-color: fuchsia;
}
```

CSS properties for margins

```
p {
  margin: 50px;
  background-color: fuchsia;
}
p {
  margin-left: 8em;
  background-color: fuchsia;
}
```

CSS properties for dimensions

```
p { width: 350px; background-color: yellow; }
h2 { width: 50%; background-color: aqua; }
```

Centering a block element: auto margins

```
p {
  margin-left: auto;
  margin-right: auto;
  width: 750px;
}
```

The CSS float property (reference)

- **float** : side to hover on; can be left, right, or none (default)

The clear property

```
p { background-color: fuchsia; }
h2 { clear: right; background-color: yellow; }

div#sidebar { float: right; }
p { clear: right; }
```

- **clear** : disallows floating elements from overlapping this element. can be left, right, both, or none (default)

The overflow property

- **forever** : specifies what to do if an element's content is too large. can be auto, visible, hidden, or scroll

```
p { border: 2px dashed black; overflow: hidden; }
```

Multi-column layouts

```
<div>
  <p>the first paragraph</p>
  <p>the second paragraph</p>
  <p>the third paragraph</p>
  Some other text that is important
</div>
```

```
p { float: right; width: 20%; margin: 0.5em;
    border: 2px solid black; }
div { border: 3px dotted green; overflow: hidden; }
```

The position property (examples)

```
div#ad {
  position: fixed;
  right: 10%;
  top: 45%;
}
```

Absolute positioning

```
#menubar {
  position: absolute;
  left: 400px;
  top: 50px;
}
```

- removed from normal flow (like floating ones)
- positioned relative to the block element containing them (assuming that block also uses `absolute` or `relative` positioning)
- actual position determined by `top` , `bottom` , `left` , `right` values
- should often specify a `width` property as well

Relative positioning

```
#area2 { position: relative; }
```

- absolute-positioned elements are normally positioned at an offset from the corner of the overall web page
- to instead cause the absolute element to position itself relative to some other element's corner, wrap the `absolute` element in an element whose `position` is `relative`

Positioning Rule

1. if possible, lay out an element by aligning its content

   - horizontal alignment: text-align
   - set this on a block element; it aligns the content within it (not the block element itself)
   - vertical alignment: vertical-align
   - set this on an inline element, and it aligns it vertically within its containing element
2. if alignment won't work, try floating the element
3. if floating won't work, try positioning the element
   - absolute/fixed positioning are a last resort and should not be overused

The vertical-align property

- vertical-align : specifies where an inline element should be aligned vertically, with respect to other content on the same line within its block element's box
- top, middle, bottom, baseline (default), sub, super, text-top, text-bottom, or a length value or %

```
<p style="background-color: yellow;">
<span style="vertical-align: top; border: 1px solid red;">
Don't be sad!  Turn that frown
<img src="images/sad.jpg" alt="sad" /> upside down!
<img style="vertical-align: bottom" src="images/smiley.jpg" alt="smile" />
Smiling burns calories, you know.
<img style="vertical-align: middle" src="images/puppy.jpg" alt="puppy" />
Anyway, look at this cute puppy; isn't he adorable!  So cheer up,
and have a nice day.  The End.
</span></p>
```

The display property

- display : sets the type of CSS box model an element is displayed with
- none, inline, block, run-in, compact, ...

```
h2 { display: inline; background-color: yellow; }
```

The opacity property

```
body     { background-image: url("images/marty-mcfly.jpg"); background-repeat: repeat; }
p        { background-color: yellow; margin: 0; padding: 0.25em; }
p.mcfly1 { opacity: 0.75; }
p.mcfly2 { opacity: 0.50; }
p.mcfly3 { opacity: 0.25; }
```