Recall: Basic CSS rule syntax

- a CSS file consists of one or more rules
- each rule starts with a selector that specifies an HTML element(s) and then applies style properties to them
Attaching a CSS file: `<link>`

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

<link rel="stylesheet" type="text/css"
  href="http://www.google.com/uds/css/gsearch.css" />
```

- A page can link to multiple style sheet files
  - In case of a conflict (two sheets define a style for the same HTML element), the latter sheet's properties will be used

CSS properties for colors

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

This paragraph uses the style above.

- `color`: Color of the element's text
- `background-color`: Color that will appear behind the element
Specifying colors

This paragraph uses the first style above. This heading uses the second style above. This heading uses the third style above.

- **color names**: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, (white), yellow
- **RGB codes**: red, green, and blue values from 0 (none) to 255 (full)
- **hex codes**: RGB values in base-16 from 00 (0, none) to FF (255, full)

Grouping styles

This paragraph uses the above style. This heading uses the above style.

- A style can select multiple elements separated by commas
- The individual elements can also have their own styles (like h2 above)

CSS properties for fonts

- **font-family**: which font will be used
- **font-size**: how large the letters will be drawn
- **font-style**: used to enable/disable italic style
- **font-weight**: used to enable/disable bold style
- Complete list of font properties
font-family

This paragraph uses the first style above.

This heading uses the second style above.

- enclose multi-word font names in quotes

More about font-family

This paragraph uses the above style.

- can specify multiple fonts from highest to lowest priority
- generic font names:
  serif, sans-serif, cursive, fantasy, monospace
- if the first font is not found on the user's computer, the next is tried
- generally should specify similar fonts
- placing a generic font name at the end of your font-family value ensures that every computer will use a valid font
font-size

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

This paragraph uses the style above.

- units: pixels (px) vs. point (pt) vs. m-size (em)
  - 16px, 16pt, 1.16em
- vague font sizes: xx-small, x-small, small, medium, large, x-large, xx-large,
  smaller, larger
- percentage font sizes, e.g.: 90%, 120%

- pt specifies number of point, where a point is 1/72 of an inch onscreen
- px specifies a number of pixels on the screen
- em specifies number of m-widths, where 1 em is equal to the font’s current size

font-weight, font-style

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

This paragraph uses the style above.

- either of the above can be set to normal to turn them off (e.g. headings)
Spending on styles with Firebug

- install Firebug Firefox add-on to inspect XHTML/CSS of any page
- can see what styles apply to which elements, and change them dynamically

Why `<strong>`, `<em>` and not `<b>`, `<i>`?

```javascript
strong { font-weight: normal; color: red; }
em { font-style: normal; background-color: #DDDDDD; }
```

Now if I want to strongly emphasize something or just emphasize it, it doesn't necessarily have to be bold or italic.

- `strong` and `em` describe attributes of the content (it is something important in the document that you want to emphasize)
- `b` and `i` describe formatting and presentation ("I want this to be bold.")

CSS properties for text

- `text-align`: alignment of text within its element
- `text-decoration`: decorations such as underlining
- `line-height`, `word-spacing`, `letter-spacing`: gaps between the various portions of the text
- `text-indent`: indents the first letter of each paragraph
- Complete list of text properties
text-align

blockquote { text-align: justify; }
h2 { text-align: center; }

The Emperor's Quote

[TO LUKE SKYWALKER] The alliance... will die. As will your friends. Good, I can feel your anger. I am unarmmed. Take your weapon. Strike me down with all of your hatred and your journey towards the dark side will be complete.

- text-align can be left, right, center, or justify (which widens all full lines of the element so that they occupy its entire width)

text-decoration

p {
  text-decoration: underline;
}

This paragraph uses the style above.

- can also be overline, line-through, blink, or none
- effects can be combined:

  text-decoration: overline underline;

CSS comments: /* ... */

/* This is a comment.
   It can span many lines in the CSS file. */

- CSS (like HTML) is usually not commented as rigorously as programming languages such as Java
- the // single-line comment style is NOT supported in CSS
- the <!-- ... --> HTML comment style is also NOT supported in CSS
Body styles

```css
body {
  font-size: 16px;
}
```

- to apply a style to the entire body of your page, write a selector for the `body` element
- saves you from manually applying a style to each element

W3C CSS Validator

- [jigsaw.w3.org/css-validator/](http://jigsaw.w3.org/css-validator/)
- checks your CSS to make sure it meets the official CSS specifications
- more picky than the web browser, which may render malformed CSS correctly
More XHTML elements
lists, quotations, and source code

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

`ul` represents a bulleted list of items (block)
`li` represents a single item within the list (block)

```html
<ul>
  <li>No shoes</li>
  <li>No shirt</li>
  <li>No problem!</li>
</ul>
```

- No shoes
- No shirt
- No problem!

More about unordered lists

- a list can contain other lists

```html
<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>
```

- Simpsons:
  - Homer
  - Marge
- Family Guy:
  - Peter
  - Lois
Ordered list: `<ol>`

`<ol>` represents a numbered list of items (block)

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

RIAA business model:

1. Sue customers for copying music
2. ???
3. Profit!

- we can make lists with letters or Roman numerals using CSS (later)

Common Error: Not closing a list

```
<ul>
  <li>No shoes</li>
  <li>No shirt</li>
  <li>No problem!</li>
</ul>
<p>Paragraph after list...</p>
```

- No shoes
- No shirt
- No problem!

Paragraph after list...

- if you leave a list open, subsequent contents will be indented
Common Error: Improper nested list placing

```html
<ul>
  <li>Simpsons:
      <ul>
        <li>Bart</li>
        <li>Lisa</li>
      </ul>
  </li>
  <li>Family Guy:
      <ul>
        <li>Peter</li>
        <li>Lois</li>
      </ul>
  </li>
</ul>
```

- closing the outer `li` too early (or not at all) will render correctly in most browsers, but it is incorrect XHTML

The `list-style-type` property

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

- Possible values:
  1. `none`: No marker
  2. `disc` (default), `circle`, `square`
  3. `decimal`: 1, 2, 3, etc.
  4. `decimal-leading-zero`: 01, 02, 03, etc.
  5. `lower-roman`: i, ii, iii, iv, v, etc.
  6. `upper-roman`: I, II, III, IV, V, etc.
  7. `lower-alpha`: a, b, c, d, e, etc.
  8. `upper-alpha`: A, B, C, D, E, etc.
  9. `lower-greek`: alpha, beta, gamma, etc.
  10. `hebrew`, `armenian`, `georgian`, `cjk-ideographic`, `hiragana`, `katakana`, `hiragana-iroha`, `katakana-iroha`
Quotations: `<blockquote>`

*a lengthy quotation (block)*

```html
<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>
```

As Lincoln said in his famous Gettysburg Address:

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.

Quotations: `<q>`

*a short quotation (inline)*

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

Quoth the Raven, "Nevermore."

- Why not just write the following?
  ```html
  <p>Quoth the Raven, "Nevermore."</p>
  ```

We don't use " marks for two reasons:

1. XHTML shouldn't contain literal quotation mark characters; they should be written as &quot;
2. using `<q>` allows us to apply CSS styles to quotations (seen later)

### HTML Character Entities

*a way of representing any Unicode character within a web page*

<table>
<thead>
<tr>
<th>character(s)</th>
<th>entity</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;&gt;</code></td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>é è ë ê</td>
<td>é è ñ</td>
</tr>
<tr>
<td>™ ©</td>
<td>™ ©</td>
</tr>
<tr>
<td>π δ Α</td>
<td>π δ Δ</td>
</tr>
</tbody>
</table>
• Complete list of HTML entities
• How would you display the text &amp; on a web page?

HTML-encoding text

To display the above text in a web page, its special characters must be encoded as shown below:

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