

# Introduction to Web Programming

## Lecture 3: CSS Properties

### 3.1: Basic CSS

- 3.1: Basic CSS
- 3.2: CSS Properties

# The bad way to produce styles

```
<p>  
  <font face="Arial">Welcome to Greasy Joe's.</font>  
  You will <b>never</b>, <i>ever</i>, <u>EVER</u> beat  
  <font size="+4" color="red">OUR</font> prices!  
</p>
```

Welcome to Greasy Joe's. You will never, *ever*, EVER beat **OUR** prices!

- tags such as b, i, u, and font are discouraged in strict HTML
  - Why is this bad?

## Cascading Style Sheets (CSS): [link](#)

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

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

- **CSS** describes the appearance and layout of information on a web page
  - (as opposed to HTML, which describes the content of the page)
- can be embedded in HTML or placed into separate .css file (preferred)

---

# Basic CSS rule syntax

---

```
selector {  
  property: value;  
  property: value;  
  ...  
  property: value;  
}
```

```
p {  
  font-family: sans-serif;  
  color: red;  
}
```

- a CSS file consists of one or more **rules**
- a rule's **selector** specifies HTML element(s) and applies style **properties**
  - a selector of **\*** selects all elements

---

## CSS comments: `/* ... */`

---

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

- CSS (like HTML) is usually not commented as much as code such as Java
- the `//` single-line comment style is NOT supported in CSS
- the `<!-- ... -->` HTML comment style is also NOT supported in CSS

# CSS properties for colors

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

This paragraph uses the style above.

property	description
color	color of the element's text
background-color	color that will appear behind the element

## Specifying colors

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

This paragraph uses the first style above.

This h2 uses the second style above.

This h4 uses the third style above.

- **color names:** aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (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

---

```
p, h1, h2 {  
  color: green;  
}  
h2 {  
  background-color: yellow;  
}
```

This paragraph uses the above style.

This h2 uses the above styles.

- 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**

---

property	description
<a href="#">font-family</a>	which font will be used
<a href="#">font-size</a>	how large the letters will be drawn
<a href="#">font-style</a>	used to enable/disable italic style
<a href="#">font-weight</a>	used to enable/disable bold style
<a href="#">Complete list of font properties</a>	

---

## font-family

---

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

This paragraph uses the first style above.

This h2 uses the second style above.

- enclose multi-word font names in quotes

---

## More about font-family

---

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

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)


## 3.2: CSS Properties

- 3.1: Basic CSS
- **3.2: CSS Properties**

---

### CSS properties for text

---

property	description
<a href="#">text-align</a>	alignment of text within its element
<a href="#">text-decoration</a>	decorations such as underlining
<a href="#">text-indent</a>	indents the first letter of each paragraph
<a href="#">text-shadow</a>	a colored shadow near an existing piece of text 
<a href="#">line-height</a> , <a href="#">word-spacing</a> , <a href="#">letter-spacing</a>	gaps between the various portions of the text
<a href="#">Complete list of text properties</a>	



---

## 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 unarmed. Take your weapon. Strike me down with all of your hatred and your journey towards the dark side will be complete.

- 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;
```

## text-shadow

```
p {  
  font-weight: bold;  
  text-shadow: -2px 5px gray;  
}
```

This paragraph uses the style above.

- shadow is specified as an X-offset, a Y-offset, and an optional color

## The list-style-type property

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

- Possible values:
  - i. none : No marker
  - ii. disc (default), circle, square
  - iii. decimal : 1, 2, 3, etc.
  - iv. decimal-leading-zero : 01, 02, 03, etc.
  - v. lower-roman : i, ii, iii, iv, v, etc.
  - vi. upper-roman : I, II, III, IV, V, etc.
  - vii. lower-alpha : a, b, c, d, e, etc.
  - viii. upper-alpha : A, B, C, D, E, etc.
  - ix. lower-greek : alpha, beta, gamma, etc.
  - x. others: hebrew, armenian, georgian, cjk-ideographic, hiragana, katakana, hiragana-iroha, katakana-iroha

# CSS properties for backgrounds

property	description
<code>background-color</code>	color to fill background
<code>background-image</code>	image to place in background
<code>background-position</code>	placement of bg image within element
<code>background-repeat</code>	whether/how bg image should be repeated
<code>background-attachment</code>	whether bg image scrolls with page
<code>background-size</code>	how large the background appears behind the element
<code>background</code>	shorthand to set all background properties

## background-image

```
body {  
  background-image: url("images/draft.jpg");  
}
```

This is the first paragraph

This is the second paragraph...  
It occupies 2 lines

- background image/color fills the element's content area

---

## background-repeat

---

```
body {  
  background-image: url("images/draft.jpg");  
  background-repeat: repeat-x;  
}
```

This is the first paragraph

This is the second paragraph...  
It occupies 2 lines

- can be repeat (default), repeat-x, repeat-y, or no-repeat

---

## background-position

---

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

This is the first paragraph

This is the second paragraph...  
It occupies 2 lines

- value consists of two tokens, each of which can be top, left, right, bottom, center, a percentage, or a length value in px, pt, etc.
- value can be negative to shift left/up by a given amount

---

# Body styles

---

```
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

---

## Styles that conflict

---

```
body { color: green; }  
p, h1, h2 { color: blue; font-style: italic; }  
h2 { color: red; background-color: yellow; }
```

*This paragraph uses the first style above.*

***This heading uses both styles above.***

- when two styles set conflicting values for the same property, the latter style takes precedence
- (later we will learn about more specific styles that can override more general styles)

---

## Embedding style sheets: `<style>` (BAD!)

---

```
<head>
  <style type="text/css">
    p { font-family: sans-serif; color: red; }
    h2 { background-color: yellow; }
  </style>
</head>
```

- CSS code can be embedded within the head of an HTML page
- this is *bad style*; DO NOT DO THIS (why?)

---

## Inline styles: the `style` attribute (BAD!)

---

```
<p style="font-family: sans-serif; color: red;">
This is a paragraph</p>
```

This is a paragraph

- higher precedence than embedded or linked styles
- used for one-time overrides and styling a particular element
- this is *bad style*; DO NOT DO THIS (why?)

---

# Content vs. presentation

---

- HTML is for **content**; *what* is on the page (heading; list; code; etc.)
- CSS is for **presentation**; how to display the page (bold; centered; 20px margin; etc.)
- keeping content separate from presentation is a very important web design principle
- If the HTML contains no styles, its entire appearance can be changed by swapping .css files
- see also: [CSS Zen Garden](#)

---

## *Cascading* style sheets

---

- it's called Cascading Style Sheets because the properties of an element *cascade* together in this order:
  - browser's [default styles](#) ([reference](#))
  - external style sheet files (in a <link> tag)
  - internal style sheets (in a <style> tag in the page header)
  - inline style (the style attribute of an HTML element)

# Inheriting styles (explanation)

```
body { font-family: sans-serif; background-color: yellow; }
p { color: red; background-color: aqua; }
a { text-decoration: overline underline; }
h2 { font-weight: bold; text-align: center; }
```

This is a heading.

A styled paragraph. [Previous slides](#) are available on the web site.

- a bulleted list

- when multiple styles apply to an element, they are **inherited**
- a more tightly matching rule can override a more general inherited rule
- not all properties are inherited (notice link's color above)

## CSS pseudo-classes

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

[Buy early, buy often!](#)

class	description
:active	an activated or selected element
:focus	an element that has the keyboard focus
:hover	an element that has the mouse over it
:link	a link that has not been visited
:visited	a link that has already been visited
:first-letter	the first letter of text inside an element
:first-line	the first line of text inside an element
:first-child	an element that is the first one to appear inside another
:nth-child( <i>N</i> )	applies to every Nth child of a given parent



# W3C CSS Validator

---

```
<p>  
  <a href="http://jigsaw.w3.org/css-validator/check/referer">  
    </a>  
</p>
```



- [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