

# Csci 4131 Internet Programming

Fall 2018

Lecture 5, Sept 19<sup>th</sup>

**Instructor: Dr. Dan Challou**

# Logistics

- HW 2 due next Friday (Sept 28<sup>th</sup>) at 2 PM (afternoon)
- My office hours this Friday (9/21) are cancelled.
  - I'll try and slip some in to replace them on Tuesday of next week
- Reading:
  - Chapter 4 – Sebesta, JavaScript
  - HTML, CSS, and JavaScript tutorials available at:  
<http://www.w3schools.com/>  
[https://www.w3schools.com/js/js\\_htmlDOM.asp](https://www.w3schools.com/js/js_htmlDOM.asp)  
[https://www.w3schools.com/tags/ref\\_eventattributes.asp](https://www.w3schools.com/tags/ref_eventattributes.asp)

# Agenda

- Last Time:
  - CSS (Chapter 3, Sebesta); and CSS tutorial at [www.w3schools.com](http://www.w3schools.com)
- Today
  - CSS Wrap UP
  - DOM
  - JavaScript?

# Helpful Homework Tip

- Section 4.12, 4.13, and 5.7 are essential reading to get the error checking in place and working for the form required for HW 2

# Questions?

# Today

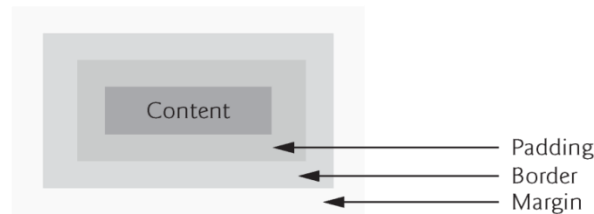
- CSS
  - Units of Measurement
  - Box Model
  - Background for JavaScript (DOM)

# Specifying Measurements

- Generally, Four Units of Measurement Are Used with CSS
  - Two absolute
    - Pixels
    - Points
  - Two relative
    - em (one em is equal to the font size for the current font) – What is *usually* the default font size for most browsers???
    - % - percent specifies a value relative to the current value
  - There are other units of measurement available, see:  
[https://www.tutorialspoint.com/css/css\\_measurement\\_units.htm](https://www.tutorialspoint.com/css/css_measurement_units.htm)

# The Box Model

- When a browser displays a web page, it places each HTML **block-level** element in a Box



**Fig. 4.13** | Box model for block-level elements.



# What Elements Have a Box Around Them?

- The following is a complete list of all HTML block level elements (although "block-level" is not technically defined for elements that are new in HTML5).

[<address>](#) Contact information.  
[<article>](#) [HTML5](#) Article content.  
[<aside>](#) [HTML5](#) Aside content.  
[<audio>](#) [HTML5](#) Audio player.  
[<blockquote>](#) Long ("block") quotation.  
[<canvas>](#) [HTML5](#) Drawing canvas.  
[<dd>](#) Definition description.  
[<div>](#) Document division.  
[<dl>](#) Definition list.  
[<fieldset>](#) Field set label.  
[<figcaption>](#) [HTML5](#) Figure caption.  
[<figure>](#) [HTML5](#) Groups media content with a caption (see [<figcaption>](#)).  
[<footer>](#) [HTML5](#) Section or page footer.  
[<form>](#) Input form.  
[<h1>](#), [<h2>](#), [<h3>](#), [<h4>](#), [<h5>](#), [<h6>](#) Heading levels 1-6.  
[<header>](#) [HTML5](#) Section or page header.  
[<hgroup>](#) [HTML5](#) Groups header information.  
[<hr>](#) Horizontal rule (dividing line).  
[<noscript>](#) Content to use if scripting is not supported or turned off.  
[<ol>](#) Ordered list.  
[<output>](#) [HTML5](#) Form output.  
[<p>](#) Paragraph.  
[<pre>](#) Preformatted text.  
[<section>](#) [HTML5](#) Section of a web page.  
[<table>](#) Table.  
[<tfoot>](#) Table footer.  
[<ul>](#) Unordered list.  
[<video>](#) [HTML5](#) Video player.

© Dan Challou, 2018, All Rights Reserved. Do not share or reproduce without the express written consent of the author

# Configuring the Box with CSS Attributes

# Calculating width and height for elements rendered with the box model

- The total width of an element should be calculated as follows:
- Total element width = width + left padding + right padding + left border + right border + left margin + right margin
- The total height of an element should be calculated like this:
- Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

# Configuring Padding and Margin

## ► Padding

- The padding property determines the distance between the content inside an element and the edge of the element
- Padding be set for each side of the box by using `padding-top`, `padding-right`, `padding-left` and `padding-bottom`

## ► Margin

- Determines the distance between the element's edge and any outside text
- Margins for individual sides of an element can be specified by using `margin-top`, `margin-right`, `margin-left` and `margin-bottom`

# Configuring the Border

## ► Border

- The border is controlled using the properties:

- `border-width`

- May be set to any of the CSS relative or absolute values or to the predefined value of `thin`, `medium` or `thick`

- `border-color`

- Sets the color used for the border

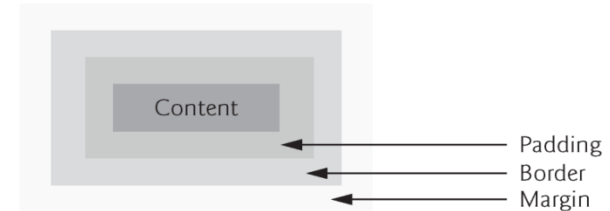
- `border-style`

- Options are: `none`, `hidden`, `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset` and `outset`

# CSS File for Styling the BOX

/\* A CSS Style file for formatting "the box" around some block elements \*/

```
body {  
  border: 3px dotted black;  
  margin: 10px;  
}  
  
section{  
  border: 2px solid black;  
  width: 500px;  
  margin: 20px; /* all four sides */  
  padding: 10px; /* all four sides */  
}  
  
h1,p {  
  border: 1px dashed black;  
  padding: 10px;  
}  
  
h1{  
  margin: .5em 0 .25em; /* .5em top,0 right and left, .25em bottom */  
  padding-left: 15px;  
}  
  
p{  
  margin: 0; /* all four sides */  
  padding-left: 15px;  
}
```



**Fig. 4.13** | Box model for block-level elements.

# HTML File That Uses our CSS Block Element Style File

```
<!DOCTYPE html>

<!-- Box Model Example -->
<html>
  <head>
    <meta charset = "utf-8">
    <title>Box Model Example</title>

    <!-- this begins the style sheet section -->
    <link rel="stylesheet" type="text/css" href="mystyle8.css">

  </head>
  <body>
    <section>
      <h1>The University of Minnesota</h1>
      <p>Educating the Leaders of Tomorrow for over 100 years.
        There is always something happening at the U!</p>
    </section>

  </body>
</html>
```

[Box example](#)

# Drawbacks of our Approach?



# Exercise 1: You can work in groups

- On a piece of paper, update the CSS file that I just did so it seamlessly scales to the window size  
[Rel Box Example.html](#)
- You can use the book, or, as a last resort, your phone or computer **for reference** (but put them away when you are done)
- One key definition 1em (a relative unit of measure) = 16px (an absolute unit of measure) in most browsers
- In your neatest and most legible handwriting, put your name(s) and x.500 id(s) at the top of the page
- Hand it in at the end of class

# A Word on Web Accessibility

- According to the World Health Organization, 253 million people have vision impairments
- *As web applications have grown rich and creative, some have become **less** accessible to these users.*
- If you are not building your pages with accessibility in mind, ***you are shutting out a big potential market !!!!***

# So, ...

- If you are not building your Web Pages / Web Sites with visual and interactive accessibility in mind, ***you are shutting out a big potential market !!!!***

# Web Accessibility Guidelines

- W3C has a complete list of Guidelines and Techniques which can be found here:

<http://www.w3.org/WAI/>

- The U of M does as well, and they can be found here:

<https://accessibility.umn.edu/>

# Summary: Specifying Measurements

- Generally, four Units of Measurement Are Used With CSS
  - Two absolute
    - Pixels
    - Points
  - Two relative
    - em (one em is equal to the font size for the current font) – What is *usually* the default font size???
    - % - percent specifies a value relative to the current value
  - Others available, see link on earlier slide

# Note: the following inline elements have boxes around them as well

## Summary

- Differences between inline and block-level elements:
  - Content model: Generally, inline elements may contain only data and other inline elements.
  - Formatting: By default, inline elements do not begin with new line.

## Elements

- Below elements are "inline":

[b](#), [big](#), [i](#), [small](#), [tt](#)

[abbr](#), [acronym](#), [cite](#), [code](#), [dfn](#), [em](#), [kbd](#), [strong](#), [samp](#), [var](#)

[a](#), [bdo](#), [br](#), [img](#), [map](#), [object](#), [q](#), [script](#), [span](#), [sub](#), [sup](#)

[button](#), [input](#), [label](#), [select](#), [textarea](#)

# Using Units of Measure

Used to specify a variety of CSS properties including:

font-size

line-height

width

margin

padding

# Style Sheet for Common Units of Measure

`/* measurement examples */`

```
body {  
    font-size: 87.5%;  
    margin-left: 2em;}  
  
header {  
    padding-bottom .75em;  
    border-bottom: 3px solid black;  
    margin-bottom:0;}  
  
h1{  
    font-size: 200%;  
    margin-bottom: 0;}
```



# Example Web Page

```
<head>
  <meta charset = "utf-8">
  <title>Units of Measure Example</title>

  <!-- this begins the style sheet section -->
  <link rel="stylesheet" type="text/css" href="mystyle2.css">

</head>
<body>

  <header>
    <h1> This Week At The University of Minnesota</h1>
  </header>

  <section>
    <p> Music, Sports, and lots more, just check the event calendars!
    </p>
  </section>

</body>
```

# Specifying Colors

- 3 ways

**1) Name** color: red;

**2,3) RGB value** either 0-255, or a hexadecimal number

Examples:

- color: rgb(10%,20%,30%)
- color: rgb(254,128, 4)
- color: #ffffff /\* white \*/
- color: #000000 /\* black \*/

# CSS3 Adds Two More Ways to Specify Colors

- `rgba` (red, green, blue, opacity)
  - Percent values 0 – 100% for RGB, or 0 -255. `a` is opacity – think of it as transparency – 0 is totally transparent, 1 is fully opaque
- `hsl` and `hsla` (hue, saturation, lightness) – `a` is opacity. Same values as for `rgba`, but will yield colors in a different way.

# CSS File for Colors

```
/* measurement and color examples */

body {
    font-size: 87.5%;
    margin-left: 2em;}

header {
    padding-bottom: .75em;
    border-bottom: 3px solid black;
    margin-bottom: 0;}

h1{
    font-size: 200%;
    margin-bottom: 0;}

h2{
    color: rgba(0,0,255,0.2);
}

h3{
    color: hsla(240,100%,50%,0.5);
}
```

# Measure and Color Example

```
<!DOCTYPE html>

<!-- External style sheets. -->
<html>
  <head>
    <meta charset = "utf-8">
    <title>Units of Measure Example</title>

    <!-- this begins the style sheet section -->
    <link rel="stylesheet" type="text/css" href="mystyle3.css">

  </head>
  <body>

    <header>
      <h1> This Week At The University of Minnesota</h1>
    </header>

    <section>
      <p>Music, Sports, and lots more, just check the event calendars!
    </p>
    </section>

    <h2> An RGBA color example </h2>

    <h3>An HSLA color example </h3>

    </body>
  </html>
```

[measure\\_n\\_color\\_ex.html](#)

# Working with Text

- 5 generic font families

serif – e.g. times new roman

sans-serif – e.g. Arial

monospace – e.g. Courier-New

*cursive - e.g. Lucida-Handwriting*

**fantasy – e.g. Impact**

# Using the fonts

e.g., `font-family: Arial, Helvetica, sans-serif;`  
`font-family: "Times New Roman", Times, serif;`

`font-size: 150%;`

`font-size: 1.5em; /* same as 150% */`

`font-size: 12pt;`

Which font-size specification is preferred?

# Examples of use in CSS

```
body {  
    font-family: Arial, Helvetica, sans-serif;  
    font-size: 87.5%  
}
```

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

If p is used in the body of an html file using the declarations above, what font will it display?

Why?



# CSS Rule Sets That Select by Element, Type, Id, and Class

`/* Element Type ID and Class Selector Examples */`

`/* All Elements */`

`* {margin: .5em; 1em;}`

`/*Elements by Type */`

`h1{ font-family: Arial, sans-serif, serif;}`

`/*One Element by ID */`

```
#main{  
    border: 2px solid red;  
    padding: 1em;  
}
```

`/* Elements by Class */`

`.blue {color: blue;}`

`.right {text-align: right;}`

# Example

```
<head>
  <meta charset = "utf-8">
  <title>Element Type ID and Class</title>

  <!-- this begins the style sheet section -->
  <link rel="stylesheet" type="text/css" href="mystyle4.css">

</head>
<body>

  <header>
    <h1> This Week At The University of Minnesota</h1>
  </header>

  <section id="main">
    <h1>Events</h1>
    <p class="blue">Music, Sports, and lots more, just check the event calendars!
  </p>
    <p class="blue right">
      There is another week of classes...
    </p>
  </section>

</body>
```

[Elem type id class.html](#)

# Can code multiple selectors

```
h2 {color: green;}
/* Multiple Selectors */
h3,h4 { color: maroon;}

/* All elements with href attributes */
*[href] { font-size: 95%;}

/* All <a> elements with href attributes */
a[href] {font-family: Arial, sans-serif;}
```

```
/* CSS File - Element Type ID and Class Selector Examples */
```

```
/* All Elements */
```

```
* {margin: .5em; 1em;}
```

```
/*Elements by Type */
```

```
h1{ font-family: Arial, sans-serif, serif;}
```

```
/*One Element by ID */
```

```
#main{  
    border: 2px solid red;  
    padding: 1em;  
}
```

```
/* Elements by Class */
```

```
.blue {color: blue;}
```

```
.right {text-align: right;}
```

```
/* Multiple Selectors */
```

```
h2 {color: green;}
```

```
h2,h3,h4 { color: maroon;}
```

```
/* All elements with href attributes */
```

```
*[href] { font-size: 95%;}
```

```
/* All <a> elements within with href attributes */
```

```
a[href] {color: red; font-family: Arial, sans-serif;}
```

```
<!DOCTYPE html>
```

```
<!-- External style sheets. -->
```

```
<html>
```

```
  <head>
```

```
    <meta charset = "utf-8">
```

```
    <title>Multiple Selectors</title>
```

```
    <!-- this begins the style sheet section -->
```

```
    <link rel="stylesheet" type="text/css" href="mystyle5.css">
```

```
  </head>
```

```
  <body>
```

```
    <header>
```

```
      <h1> This Week At The University of Minnesota</h1>
```

```
    </header>
```

```
    <section id="main">
```

```
      <h1>Events</h1>
```

```
      <p class="blue">Music, Sports, and lots more, just check the event calendars!
```

```
    </p>
```

```
      <p class="blue right">
```

```
        There are another 12 weeks of classes...
```

```
      </p>
```

```
    </section>
```

```
    <h2> Here are some examples of multiple selectors, etc. </h2>
```

```
    <h3> The quick brown fox </h3>
```

```
    <h4> jumped over the lazy dog </h4>
```

```
    <a href = "http://www.google.com">Click here for Google</a>
```

```
  </body>
```

```
</html>
```

[Multiple selectors ex.html](#)

# Summary and Recap

- Code a selector for all elements by using the Universal Selector (\*)
- Code a selector for all elements of a specific type by naming the type  
(h1, p, ul, etc.)
- Code a selector for an element with a class attribute by coding a period follow by the class name – then the rule applies to all elements with that class name.
- Code a selector for an element with an id attribute by coding a pound sign (#) followed by an id value

# Remember HW and Reading

- HW 3, due next Friday at 2 pm
- Reading:
  - Today Finish Chapters 3 in Sebesta
  - Reading For Today was Chapter 4 in Sebesta, JavaScript
  - ❑ Especially helpful Chapter & sections for doing the form validation required by your homework:  
Sebesta: 4.12, 4.13, and 5.7
  - HTML, CSS, JavaScript tutorials available at:  
<http://www.w3schools.com/>  
[https://www.w3schools.com/js/js\\_htmlDOM.asp](https://www.w3schools.com/js/js_htmlDOM.asp)  
[https://www.w3schools.com/tags/ref\\_eventattributes.asp](https://www.w3schools.com/tags/ref_eventattributes.asp)