

# CSS Basics

## Cascading Style Sheets

Camosun College - ICS111-CSS - Section 2

## HTML: Looking Back

- HTML dictates order, structure, and function
- Does very little to specify layout or visual rendering

Camosun College - ICS111- CSS - Section 2

2

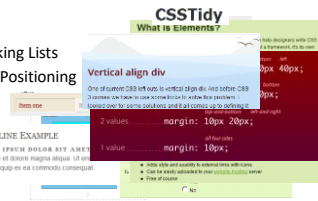
## Overview

- What is CSS?
- CSS & HTML
- The Box Model
- Style Sheet Implementation
- CSS Rule Structure
- HTML & DIVs
- Common CSS properties
- CSS Cascade and Inheritance
- Resources

Camosun College - ICS111-CSS - Section 2

## What Web Designers Need to Know About CSS

- Page Layouts
- Styling and Making Lists
- Box model and Positioning
- Typography
- Styling Forms
- Using CSS Flexbox
- Best Practices
- CSS Hacks



Camosun College - ICS111- CSS - Section 2

4

## The Purpose of CSS

- HTML is the content and meaning (*structure*)
- CSS helps to convey that meaning (*style*)
- Allows developers to separate the content from layout and design
  - Content and design inherently different in nature
  - Change in content does not require change in design

Carosoun College - IC3111-CSS - Section 2

5

## Origin of CSS

- W3C has produced three style recommendations CSS1, CSS2, and CSS3 – each level builds on the previous version - <http://www.w3.org/TR/CSS21/>
- Not all browsers implement these changes consistently
  - <http://tools.css3.info/selectors-test/test.html>
- Early browsers (MS Internet Explorer 3 and 4) did not fully support CSS
- As of July 2015, no browser has yet fully implemented CSS3 – some more than others
- [http://en.wikipedia.org/wiki/Comparison\\_of\\_layout\\_engines\\_\(CSS\)](http://en.wikipedia.org/wiki/Comparison_of_layout_engines_(CSS))

Carosoun College - IC3111-CSS - Section 2

6

## Style Sheets

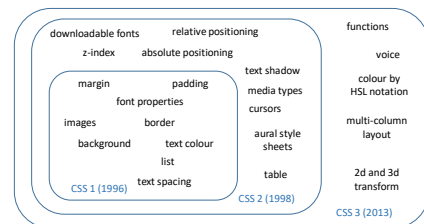


- With HTML v4 (published 1998) a new approach to web page definition was developed using style sheets
- A style sheet is a set of defined presentation instructions that is separate from the content
- The concept of style sheets had been around since SGML days (1980s)

Carosoun College - IC3111-CSS - Section 2

7

## Evolution of CSS Features from W3C



Carosoun College - IC3111-CSS - Section 2

8

## CSS Zen Garden

- Site using static HTML content
- Differing external CSS files create dramatically different layout
- Support for multiple browsers
  - link: <http://www.csszengarden.com>
  - hint: change the styles on the page

## An Example

Consider the **boldface** font in the following examples:

Thick bacon is delicious.

Today is a good day.

- Meaning is conveyed by the styling
- Attention is given to the information
- Remove the style and meaning is lost
- No additional meaning is lost when removed

Say it out loud. The "thick" is meant to be said strongly. If you lose the intonation and expression, some of the meaning is lost along with it.

## Why CSS?

- CSS skills are essential for web page design
- HTML skill allows you to understand how to structure the HTML content but not how to present it effectively
- Easily change the presentation of an entire web site by modifying a single CSS style sheet
- CSS skills are vital in web projects
- If there is a problem with your CSS, usually there are no error messages – check with Firebug on Firefox or Internet Tools on IE (ver 8)

## What is CSS ?

- Like HTML, CSS is simple human-readable text
- CSS is not a programming language like Java or PHP
- CSS is not the same as HTML – CSS cannot be present without HTML or XSL
- CSS will help simplify your web content presentation and make it more manageable
- CSS will also work with XSL (Extensible Stylesheet Language), another web technology

## What is CSS?

- Typical CSS file is a text file with an extension **.css** and contains a series of commands or rules. These rules tell the HTML how to display.

```

Style.css
/* Styles for sitename.com */
body {
  font-family: Arial;
  background: #0000;
}
#container {
  text-align: left;
  width: 1020px;
}
#header {
  height: 232px;
}
#footer {
  width: 100%;
  padding: 0 10px;
  margin-bottom: 10px;
}
And soon....

```

Camosun College - IC3111 - CSS - Section 2

13

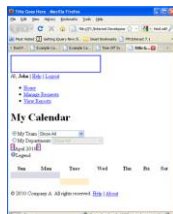
## CSS Benefits

- Separates structure from presentation
- Provides advanced control of presentation
- Easy maintenance of multiple pages
- Faster page loading
- Better accessibility for disabled users
- Easy to learn

Camosun College - IC3111 - CSS - Section 2

14

## HTML Without CSS



*"HTML without CSS is like a piece of candy without a pretty wrapper."*

- Without CSS, HTML elements typically flow from top to bottom of the page and position themselves to the left by default.
- With CSS help, we can create containers or DIVs to better organize content and make a Web page visually appealing.

Camosun College - IC3111 - CSS - Section 2

15

## CSS Module 2: Syntax

- We explore the following:
  - How does CSS work in the web page?
  - What is a selector?
  - What is a declaration?
  - How do CSS properties and values work?
  - How are CSS errors handled?
  - What the different types of values in CSS?

Camosun College - IC3111 - CSS - Section 2

16

## The Box Model



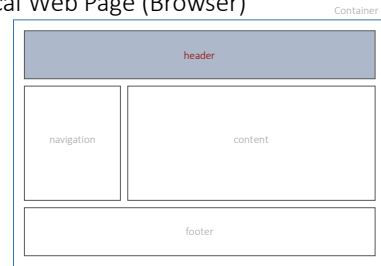
- CSS works on the box model. A typical Web page consists of many boxes joined together from top to bottom. These boxes can be stacked, nested, and can float.

- **Header**  
**Navigation**  
**Content**  
**Footer**

Carson College - IC3111 - CSS - Section 2

17

## Typical Web Page (Browser)



Carson College - IC3111 - CSS - Section 2

18

## Attaching a Style Sheet

- Attach a style sheet to a page by adding the code to the <head> section of the HTML page.
- There are 3 ways to attach CSS to a page:
  1. External Style Sheet: Best used to control styling on multiple pages. You can have more than one
 

```
<link rel="stylesheet" type="text/css" media="all" href="css/styles.css" />
<link href="http://www.google.ca/uds/css/qsearch.css" rel="stylesheet" type="text/css" />
```

Carson College - IC3111 - CSS - Section 2

19

## Attaching a Style Sheet

1. Internal Style Sheet: Best used to control styling on one page. Embedded within the head of the HTML page.
 

```
<style type="text/css">
  h1 {color: red}
</style>
```
2. Inline Style Sheet: CSS is not attached in the <header> but is used directly within HTML tags. Higher precedence than embedded or linked styles. Avoid this if at all possible.
 

```
<p style="color: red">Some Text</p>
```

Carson College - IC3111 - CSS - Section 2

20

## CSS Rule Structure

- A CSS RULE is made up of a selector and a declaration. A declaration consists of property and value.

selector { **property: value;** }

↙      ↘

selector      declaration

## CSS Style Attribute and Tag (selector)

- CSS has a style attribute and style tag (selector)

<h1 style = "property : value;  
property : value; ... ">

<style>  
selector { **property : value;**  
property : value; ... }  
</style>

↙      ↘

selector      declaration

the selector is usually an HTML tag name.  
The final ; is optional.

## Selectors

- A selector, here in **gray**, is often an element of HTML.

```
body { property: value; }
h1 { property: value; }
em { property: value; }
p { property: value; }
```

## Properties and Values

- Properties and values tell an HTML element how to display.

```
body {background: purple;}
h1 {color: green;}
h2 {font-size: large;}
p {color: #ff0000;} /*hexadecimal for red*/
```

body {  
background: purple;  
color: green;  
}

\*CSS code can be  
written in a linear  
format or in a block  
format.

## Grouping Selectors

- Group the same selector with different declarations together on one line.

```
h1 {color: black;}
h1 {font-weight: bold;}
h1 {background: white;}
```

- Example of grouping selectors (both are correct):

```
h1 {
  color: black;
  font-weight: bold;
  background: white;
}
```

Carouson College - IC3111 - CSS - Section 2

25

## Grouping Selectors

- Group different selectors with the same declaration on one line.

```
h1 {color: yellow;}
h2 {color: yellow;}
h3 {color: yellow;}
```

- Example of grouping selectors (both are correct):

```
h1, h2, h3 {color: yellow;}
```

Carouson College - IC3111 - CSS - Section 2

26

## Comments in CSS

- Explain the purpose of the coding
- Help others read and understand the code
- Serve as a reminder to you for what it all means
- Starts with `/*` and ends with `*/`

```
p {color: #ff0000;} /* Company Branding */
```

Carouson College - IC3111 - CSS - Section 2

27

## Paragraph

- To start off our understanding of cascading style sheets, we're going to use a special line of CSS code that does something HTML alone could never do right... we're going to indent every paragraph automatically.
- Here's the CSS code:

```
p { text-indent: 3em; }
```

Carouson College - IC3111 - CSS - Section 2

28

## Working with Paragraph

- Then, right before your `</head>` tag in each page add a line similar to this:  

```
<link rel="stylesheet" type="text/css" href="main.css" title="Default">
```
- This will link a new style sheet, location `main.css`, to each page that has the one line in it.
  - The title and type attributes are optional.

## import

- `@import` can be used in conjunction with the other methods. Imagine you want 2 pages out of your initial 10 pages to have, in addition to the normal indent, each and every paragraph in blue text.
- You could write a second style sheet, call it `oohblue.css`, and inside that sheet you have:  

```
p { color: blue; }
```

## import

- Then, in those 2 special pages, you place the normal CSS link, but you'll add special code, `@import`, to add the extra color.

```
<link rel="stylesheet" type="text/css" href="main.css" title="Default">
<style type="text/css">
<!--
  @import url(oohblue.css);
-->
</style>
```

## Basis overview

- Those are the basics. Let's review the ways you can include a style sheet:
- Write it inline inside each and every tag
- Place a `<style> </style>` at the beginning of the web page
- Dedicate and link a CSS file and write it inside that file
- use `@import` to include it as portion of a page's CSS



## Exercise 1

- Use `<style>` to make all paragraphs have 10 spaces indentation (hint: 6em) and make the text red.
- Hint: Combine both into one line of code using the `;` separator. Remember to create a paragraph in the `<body>` to see the style in action! Generic text below.

This is the first paragraph  
with the red text and large indent.

This is the second paragraph  
with the red text and large indent.

Carnouson College - IC3111 - CSS - Section 2

33

## Exercise 2

- Do the same as the last slide except link to an external style sheet!

Carnouson College - IC3111 - CSS - Section 2

34

## CSS properties for colors

```
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

Carnouson College - IC3111 - CSS - Section 2

36

## Headers

- You can apply the same CSS declaration to multiple selectors by using a comma to separate the selectors
- For example: If you want to make all H1, H2, and H3 red, and all H4, H5, H6 yellow, your style could look like this:
 

```
h1, h2, h3 { color: red; }
h4, h5, h6 { color: yellow; }
```
- You can use different representations of colour:
 

```
h2 { color: rgb(128, 0, 196); }
h4 { color: #FF8800; }
```
- You can set a style for nearly all HTML elements.

Carnouson College - IC3111 - CSS - Section 2

37

## Grouping styles

- A style can select multiple elements separated by commas
- The individual elements can also have their own styles

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

This paragraph uses the above style.

This h2 uses the above styles.

Carouson College - IC3111 - CSS - Section 2

38

## Selector a class of a current element

- Every paragraph is now indented. But what if you want a few paragraphs without an indent? We can define an extra selector.
- You can pick a name for these.
- Here's the original code with an added *noindent* selector:
 

```
p { text-indent: 3em; }
p.noindent { text-indent: 0em; }
```
- This says that any p that is part of a class called *noindent* we should use 0em indentation. To call that in code, we use the class attribute.

Carouson College - IC3111 - CSS - Section 2

39

## Normal paragraph

- A normal paragraph looks like this:
 

```
<p> I'm using a style with an indent. </p>
```
- A paragraph with the *noindent* class attribute looks like this:
 

```
<p class="noindent"> I'm using a style without an indent. </p>
```
- If you are going to only indent some paragraphs, but you probably won't indent most, you can just define a special paragraph called indent.
 

```
p.indent { text-indent: 3em; }
```
- If that's our only style, regular `<p>` `</p>` will have no indent, while `<p class="indent">` `</p>` will have a 3em indentation.

Carouson College - IC3111 - CSS - Section 2

40

## This h1 has an indent.

- Imagine a selector *.g*, defined as `{ color: green; }`. Every time you use `class="g"` in an element the text color would be shown in green.
- ID selectors are used for special formatting of only a few elements. ID selectors use a `#` instead of a `.`
- Imagine 3 paragraphs, we'll call them `par1`, `par2`, `par3`. We'll make one red, one orange, and one blue.

Carouson College - IC3111 - CSS - Section 2

41

This h1 has an indent con't

- We could use these styles:

```
p#par1 { color: red; }
p#par2 { color: orange; }
p#par3 { color: blue; }
```

```
<p id="par1">I'm in red</p>
<p id="par2">I'm in orange</p>
<p id="par3">I'm in blue</p>
```

- ID Selectors can also be element-less:

```
#par1 { color: red; }
```

...would apply to all tags that specify id="par1".

Carouson College - IC3111-CSS - Section 2

42

CSS properties for fonts

property	description
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

Carouson College - IC3111-CSS - Section 2

43

font-family

- Enclose multi-word font names in quotes

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

This paragraph uses the first style above.

**This h2 uses the second style above.**

Carouson College - IC3111-CSS - Section 2

44

font-family

- We 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
- Placing a generic font name at the end of your font-family value, ensures that every computer will use a valid font

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

Carouson College - IC3111-CSS - Section 2

45

## font-size

- 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%

```
p {
  font-size: 24pt;
}
```

Carnous College - IC3111 - CSS - Section 2

46

## font-size

- **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

```
p {
  font-size: 24pt;
}
```

Carnous College - IC3111 - CSS - Section 2

47

## font-weight, font-style

- **Bolding and Italics:**

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

***This paragraph uses the style above.***

Carnous College - IC3111 - CSS - Section 2

48

## CSS properties for text

property	description
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

Carnous College - IC3111 - CSS - Section 2

49

## text-align

- text-align can be left, right, center, or justify

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

Carouson College - IC3111 - CSS - Section 2

50

## text-decoration

- can also be overline, line-through, blink, or none
- effects can be combined:
- text-decoration: overline underline;

```
p {
  text-decoration: underline;
}
```

Carouson College - IC3111 - CSS - Section 2

51

## The list-style-type property

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

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

Carouson College - IC3111 - CSS - Section 2

52

## PSEUDO-ELEMENTS

- There are two important pseudo-elements that are built into CSS capable web browsers. (There are also common pseudo-classes which you'll learn in the links chapter.)
- These two elements are *:first-letter* and *:first-line*. Notice that pseudo-elements are defined with a *:* instead of a *.* or *#* (this is because they have special meanings to a web browser).
- Here's an example for each: Imagine you want the first letter of each paragraph to be red, or the first-line of each paragraph to be green.
 

```
p:first-letter { color: red; }
p:first-line { color: green; }
```

Carouson College - IC3111 - CSS - Section 2

53

## CSS properties for backgrounds

property	description
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bg image scrolls with page
background	shorthand to set all background properties

Carnouson College - IC3111-CSS - Section 2

54

## CSS Background, Image and Color Styles

- You are familiar with the <body> tag.

A typical <body> tag looks something like this:

```
<body background="graphic.jpg" text="FFFFFF"
bgcolor="#000000">
```

- To convert that into CSS, it looks like this:

```
body { background-image: url(graphic.jpg);
color: #FFFFFF; background-color: #000000; }
```

Carnouson College - IC3111-CSS - Section 2

55

## background-image

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

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

This is the first paragraph

This is the second paragraph

It occupies 2 lines

Carnouson College - IC3111-CSS - Section 2

56

- CSS adds some special features. One of the most important is the *background-repeat* property.

- It has these values:

```
repeat,
repeat-x,
repeat-y, or
no-repeat.
```

- A regular web page has a default of background-repeat: repeat, which means the image is repeated both horizontally and vertically.
- With CSS, you can set the background to repeat horizontally (repeat-x), repeat vertically (repeat-y), or not repeat at all (no-repeat).

Carnouson College - IC3111-CSS - Section 2

57

## background-repeat

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

DRAFT DRAFT DRAFT DRAFT

Carouson College - IC3111 - CSS - Section 2

58

## Images

- We can edit the style mentioned above to have the body's background never repeat by adding background-repeat: no-repeat:
 

```
body { background-image: url(graphic.jpg);
color: #FFFFFF; background-color: #000000;
background-repeat: no-repeat; }
```
- If you want to include the repeat in your standard background tag (for example, if are not using CSS for the rest of your page), you can add style="background-repeat: no-repeat;" , so it looks like this:
 

```
<body background="graphic.jpg" text="#FFFFFF"
bgcolor="#000000" style="background-repeat: no-repeat;">
```

Carouson College - IC3111 - CSS - Section 2

59

## There are two more important background

- background-attachment: merely allows you to decide if you want the background to scroll or not. If you want it to scroll, use background-attachment: scroll. If you want it to not scroll, use background-attachment: fixed.
- background-position: allows you to position the background. It takes two values, the first is the vertical position (in px [pixels], % [percent], or top, center, bottom) and the second value is the horizontal position (in px [pixels], % [percent], or left, center, right).

Carouson College - IC3111 - CSS - Section 2

60

## background-position

- 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 {
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

DRAFT

Carouson College - IC3111 - CSS - Section 2

61

## Key features

- If you want a background to be at the top right, use: background-position: top right.
- If you want it to be at the bottom center, use background-position: bottom center.
- This is typically most useful used with background-repeat: no repeat.

Carnous College - IC3111 - CSS - Section 2

62

## Background Property

- As you can see, the coding for the background can get pretty long. CSS lets you combine it all into a single property statement, known as background. It follows this format:  
background: background-color || background-image || background-repeat || background-attachment || background-position
- Example:
  - If you want a background color of white, a background image lightPattern.jpg, the background to never repeat, and never scroll, you could use:  
body { background: #FFFFFF url(lightPattern.jpg) no-repeat fixed; }

Carnous College - IC3111 - CSS - Section 2

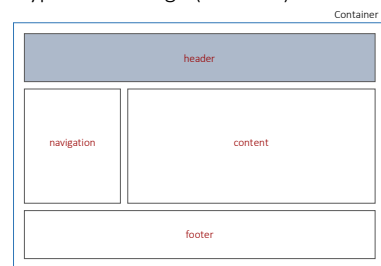
63

- Remember, you'll also need to set the text color, so add color: #000000 (if you want black text)  
body { background: #FFFFFF url(lightPattern.jpg) no-repeat fixed; color: #000000; }
- Notice that the browser is smart enough to realize that a value (in this case: background-position) is missing.

Carnous College - IC3111 - CSS - Section 2

64

## Typical Web Page (Browser)



Carnous College - IC3111 - CSS - Section 2

65



## Typical Web Page (HTML)

- Typical HTML Web page is made up of containers (boxes) or DIVs. Each DIV is assigned an ID or a Class.

```
<div id="container">
  <div id="header">Insert Title</div>
  <div id="main">content
    <div id="menu">content</div>
  </div>
  <div id="footer">content</div>
</div>
```

Carosoun College - IC3111 - CSS - Section 2

66

## Typical Web Page (CSS)

- The CSS file uses the same DIV/ID/Class names as the HTML and uses them to style the elements.

```
• #container {property: value;}
• #menu {property: value;}
• #main {property: value;}
• #footer {property: value;}
```

Carosoun College - IC3111 - CSS - Section 2

67

## Naming HTML elements

- There are two naming options for an HTML element: assigning "ID" names and "class" names.
- When you give an HTML element a class or id name, you need to use that name when making the corresponding style declaration
  - These two options are very similar, and the "class name" approach is more popular, so we focus on that.
- Aside: An id declaration is the same as a class declaration, except that it should only be used specifically once per web page
  - The syntax for id vs. class is also nearly identical, the only difference being the use of a pound sign (#) instead of the period (.) you will see in a couple slides.

Carosoun College - IC3111 - CSS - Section 2

68

## IDs and Classes

- **IDs (#)** are unique and can only be used once on the page
- **Classes (.)** can be used as many times as needed
- HTML Code:
 

```
<h1 id="mainHeading">Names</h1>
<p class="name">Bob</p>
```
- CSS Code:
 

```
#mainHeading {color: green}
.name {color: red}
```

Carosoun College - IC3111 - CSS - Section 2

69

## Example: naming HTML elements

- The following HTML block gives the "class name" bigblue to the following specific <h1> tag in this (very) simple webpage.

```
<html>
<body>
  <h1 class="bigblue"> Introduction </h1>
</body>
</html>
```

Carouson College - IC3111-CSS - Section 2

70

## Connecting a declaration to class name

- To connect a style declaration to a particular class name you wrote into your HTML document, you simply precede the class declaration with: .theclassname

```
.bigblue
{
  font-weight: bold;
  color: blue;
}
```

if you want this style to be used only once in the web page, then specify it as an ID style with this slight syntax change:

```
#bigblue
{
  font-weight: bold;
  color: blue;
}
```

Carouson College - IC3111-CSS - Section 2

71

## Styling hyperlinks

- The associated style selectors are called the hyperlink (or "anchor") pseudo-class selectors:
  - :link, :visited, :hover, :active { }
- Example:
 

```
a:link {color:#FF0000;} /* color to apply to link before it's
visited */
a:visited {color:#00FF00;} /* color to apply to link after it's
visited*/
a:hover {color:#FF00FF;} /* color to apply to link while mouse
pointer is over it*/
a:active {color:#0000FF;} /* color to apply while left mouse
button is held down on link */
```
- Note: a:hover MUST be listed after a:link and a:visited!
- Note: a:active MUST be listed after a:hover!

Carouson College - IC3111-CSS - Section 2

72

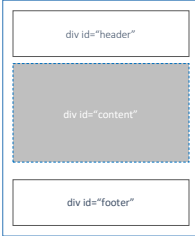
## CSS Border Properties

- Background-color
- Width
- Padding
- Margin
- Border-width
- Border-color
- Border-style

Carouson College - IC3111-CSS - Section 2

73

### HTML



### CSS

```
#content {
  background-color: #ccc;
  margin-bottom: 10px;
  border: 1px dashed blue;
  color: #fff;
  width: auto;
}
```

Carson College - IC3111-CSS - Section 2 74

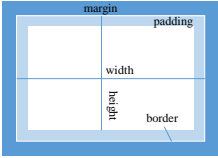
## The <div> tag

- The <div> tag is nothing more than a container unit that encapsulates other page elements and divides the HTML document into sections.
- Web developers use <div> elements to group together HTML elements and apply CSS styles to many elements at once. For instance, by wrapping a set of paragraph elements into a <div> element, we can take advantage of CSS styles and apply a font to all paragraphs at once by applying a font style to the <div> tag instead of coding the same style for each paragraph element.

Carson College - IC3111-CSS - Section 2 75

## Common CSS Layout Properties

- Width
- Height
- Float
- Clear
- Border
- Padding
- Margin



Carson College - IC3111-CSS - Section 2 76

## Width & Height

- Width and height define the width and height of an element.

```
#box {width="50px"}
#box {width="50em"}
#box {width="100%"}
#box {width="auto"}
```

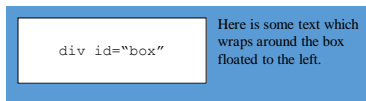
```
#box {height="auto"}
```

Width and height can be specified in pixels, ems, percentages or set to auto

Carson College - IC3111-CSS - Section 2 77

## Float: (left, right)

- Float property makes elements float to the right or left of the screen, positioned where they are in the HTML. Floating allows word wrapping.



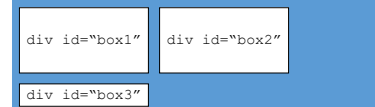
```
#box {float:left; margin-right: 10px;}
```

Carnous College - IC3111 - CSS - Section 2

78

## Clear: (left, right, both)

- When elements are floated, they wrap around each other to form a "caravan." The **clear** property detaches an element from the "caravan" and allows it to start on a new line.

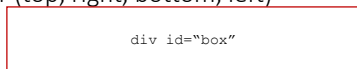


```
#box3 { background-color: white; border: 1px solid #000; clear: both;}
```

Carnous College - IC3111 - CSS - Section 2

79

## Border (top, right, bottom, left)



You can define the entire border or only the top, bottom, left, or right. You can also define the border using one declaration. The code could be any of the following:

```
#box {
border-top: red dotted 1px;
border-bottom: red dotted 1px;
border-left: red dotted 1px;
border-right: red dotted 1px;
}
```

```
#box {
border: red dotted 1px;
}
```

```
#box {
border-color: red;
border-style: dotted;
border-width: 2px;
}
```

Carnous College - IC3111 - CSS - Section 2

80

## Padding (top, right, bottom, left)



The code could be any of the following:

```
padding: 10px;
```

```
padding: 10px 10px;
```

```
padding: 10px 10px 10px 10px;
```

```
padding-left: 10px;
padding-right: 10px;
padding-bottom: 10px;
padding-top: 10px;
```

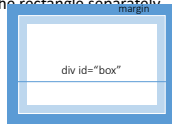
- Padding is the space between the text/content and the border. You can use padding for all around the element or specify each side of the rectangle separately.

Carnous College - IC3111 - CSS - Section 2

81

## Margin (top, right, bottom, left)

- Margin is the space outside the text/content and the border. You can use margin for all around the element or specify each side of the rectangle separately.



- The code could be any of the following:

```
margin: 10px;
```

```
margin: 10px 10px;
```

```
margin: 10px 10px 10px 10px;
```

```
margin-left: 10px;
margin-right: 10px;
margin-bottom: 10px;
margin-top: 10px;
```

Camosun College - IC3111- CSS - Section 2

82

## Text Properties

### MAIN HEADING

Gravida lacinia velit. Vivamus tortor enim, tincidunt at, pellentesque ut, iaculis eu, quam.

- To style the main heading in the paragraph above, we assigned a class the HTML tag.

```
<h3
class="mainHeading">
Main Heading</h3>
```

```
.mainHeading {
color: red;
letter-spacing: 5px;
text-transform: uppercase;
word-spacing: 15px;
text-align: left;
font-family: Times;
text-decoration: underline;
font-size: 12px;
font-style: italic;
font-weight: bold;
}
```

Camosun College - IC3111- CSS - Section 2

83

## Example

```
<!DOCTYPE html>
<html>
<head>
<style>
div.container {
width: 100%;
border: 1px solid gray;
}

header, footer {
padding: 1em;
color: white;
background-color: navy;
clear: left;
text-align: center;
}
```

Camosun College - IC3111- CSS - Section 2

84

```
nav {
float: left;
max-width: 160px;
margin: 0;
padding: 1em;
}

nav ul a {
text-decoration: none;
}

article {
margin-left: 170px;
border-left: 3px solid magenta;
padding: 1em;
overflow: hidden;
}

nav ul {
list-style-type: none;
padding: 0;
}

</style>
```

Camosun College - IC3111- CSS - Section 2

85

## CSS Colors

- Standard
  - White
  - Black
  - Blue
  - Fuchsia
  - Gray
  - Green
  - Lime
  - Aqua

Carnouson College - IC3111 - CSS - Section 2

86

## Styling Links

- The `links` property defines how inactive, hovered, active, and visited [link](#) states appear to the user.

```
a:link {color: red; text-decoration: none; border-bottom: 1px dashed red; background: white;}
a:visited {color: yellow;}
a:active {color: green;}
a:hover {color: orange;}
```

Carnouson College - IC3111 - CSS - Section 2

87

## The Power of Cascade

- When multiple styles or style sheets are used, they start to cascade and sometimes compete with one another due to CSS's inheritance feature.
- Any tag on the page could potentially be affected by any of the tags surrounded by it.
- So, which one wins? Nearest Ancestor Wins.
  - Inline style or directly applied style
  - The last style sheet declared in the `<header>` section

Carnouson College - IC3111 - CSS - Section 2

88

## Saving Time with Inheritance

- In a nutshell, inheritance (not the money you get from your grandma) is the process by which CSS properties applied to one tag are passed on to nested tags.
- For example, the paragraph tag will inherit the same styling as the body tag because `<p>` is always located inside `<body>`.
 

```
<body style="font-family: Arial">
  <p>This text will be Arial as well</p>
</body>
```
- So, instead of styling each paragraph separately, you can define the font color in the `<body>`, and everything inside will have that color.

Carnouson College - IC3111 - CSS - Section 2

89

## Resources

- <https://www.w3schools.com/cssref/> (list of all CSS properties)
- <http://www.w3schools.com/css/>
- <http://www.html.net/tutorials/css/>
- <https://css-tricks.com/>