




Umadevi Balakrishnan





WHAT IS CSS?
APPLYING STYLES
SYNTAX
COLORS
PIXELS AND EMS
FONT
BORDER
LINK DECORATION
SELECTOR
 CLASS
 ID
 PSEUDO
CASCADING

WHAT IS CSS?

- **Cascading Style Sheets**
- Language used to describe the appearance and formatting of HTML
- A **style sheet** is a file that describes how an HTML file should look.
- Power of CSS
 - http://www.w3schools.com/css/demo_default.htm
 - <http://www.csszengarden.com/214/page1>

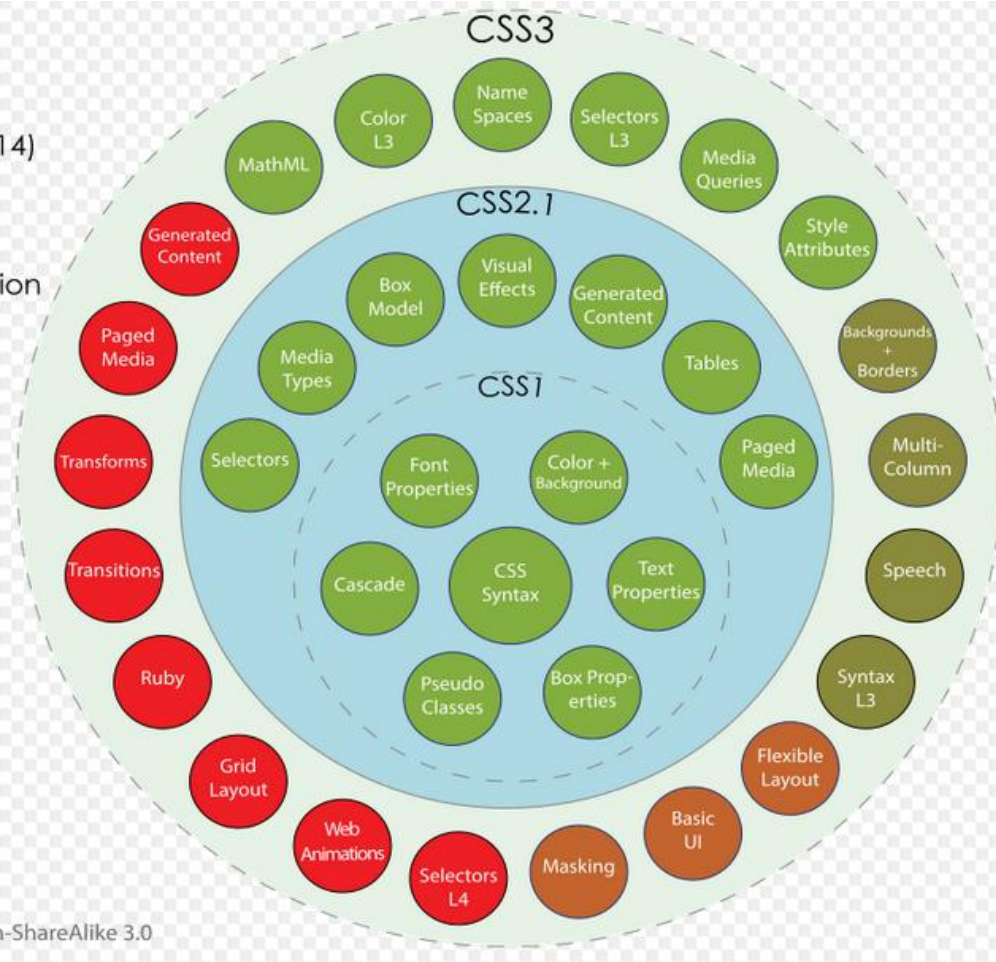
CASCADING?????

- More than one style sheet rule could apply to a particular piece of HTML
- There has to be a known way of determining which specific rule applies to which piece of HTML
- The rule used is chosen by cascading down from the more general rules to the specific rule required.
- The most specific rule is chosen.

CSS3

Taxonomy & Status (October 2014)

- W3C Recommendation
- Candidate Recommendation
- Last Call
- Working Draft
- Obsolete or inactive



By Sergey Mavrody 2011-14 | CC Attribution-ShareAlike 3.0



APPLYING STYLES

7

Inline
<style>
stylesheet

INLINE STYLING

```
<p style="color:red">Red font!</p>
```



○ Cons

- Write the same code over and over
- To make a big stylistic change ,change every single style tag

<STYLE> TAG

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style>
5       p {
6         color: purple;
7       }
8     </style>
9     <title>Result</title>
10  </head>
11  <body>
12    <p>Check it out! I'm purple!</p>
13  </body>
14 </html>
```



LINK IT UP!

- CSS in a totally separate file
 - Cssfilename.css
- Use a link tag in the HTML file

```
index.html
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link rel="stylesheet" type="text/css" href="stylesheet.css">
5     <title>Result</title>
6   </head>
7   <body>
8     <p>I want to be SIZE 44 font!</p>
9   </body>
10 </html>
```

```
stylesheet.css
1 p {
2   font-size: 44px;
3 }
```



<LINK> TAG ATTRIBUTES

1. A type attribute that should always be equal to "text/css"
2. A rel attribute that should always be equal to "stylesheet"
3. A href attribute that should point to the web address of your CSS file

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

WHY SEPARATE FORM FROM FUNCTION?

- Can apply the same formatting to several HTML elements without rewriting code over and over (e.g. `style="color:red":)`
- Can apply similar appearance and formatting to several HTML pages from a single CSS file



CSS SYNTAX

13

CSS SYNTAX IS DIFFERENT FROM THE HTML

- The general format looks like this:

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

- Each property-value pair should end with a semi-colon (;)

A **selector** can be any HTML element, such as <p>, , or <table>
To make a paragraph's text red with CSS

Selector

```
p {  
  color: red;  
}
```

Value

Property

A **value** is a possible setting for a property.
color can be red, blue, black, or almost any color;
font-family can be a whole bunch of different fonts; and so on.

A **property** is an aspect of a selector.
For instance, the font-family, color, and font-size of the text on web pages

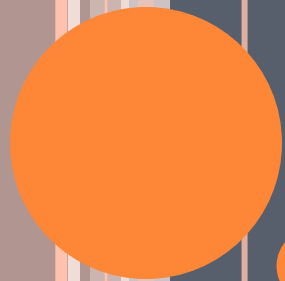
ONE SELECTOR, MANY PROPERTIES

- Set many properties for one selector
- To set a paragraph's font, font color, and font size

```
p {  
    font-family: Arial;  
    color: blue;  
    font-size: 24px;  
}
```


LET'S DO SOME CODING....

- **Sample1 folder**
- Make all the h3 headings red.
- Set all the paragraphs to the Courier font-family. (Make sure to capitalize "Courier" as shown!)
- The second paragraph contains text between `` tags. Set the background-color of that `` to 'yellow'.



COMMENTS

18

COLOR COMMENTARY

- Good comments will help to remind why something is done in a certain way

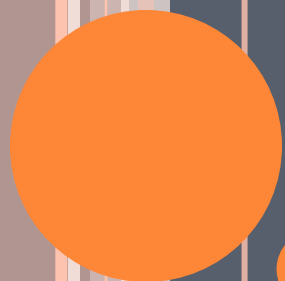
```
<!--I'm a comment!-->
```

```
/*I'm a comment!*/
```

LETS CHECK BEFORE WE WRECK

Instructions

- Add a `<link>` to `stylesheet.css` between your `<head></head>` tags.
- Change the `<h1>` header's font-family to Verdana. (Make sure Verdana is capitalized as shown!)
- Change the `<h3>` header's font-family to Courier. (Make sure Courier is capitalized as shown!)
- Make the paragraph text color purple.
- Add a CSS comment .It can say anything you like. (Remember: HTML comments and CSS comments are written differently.)



COLORS



21





HOW DOES CSS KNOW ALL COLORS?

- CSS understands millions of colors in the form of **hexadecimal values**
- Hexadecimal counting is **base-16**.
- Each digit can be the numbers 0 through 9 **or the letters a through f**

<http://www.color-hex.com/color-palettes/>

- Hex values always start with a pound sign (#), are up to six "digits" long, and are **case-insensitive**
 - #FFC125 and #ffc125 are the same color



PIXELS AND EMS

24

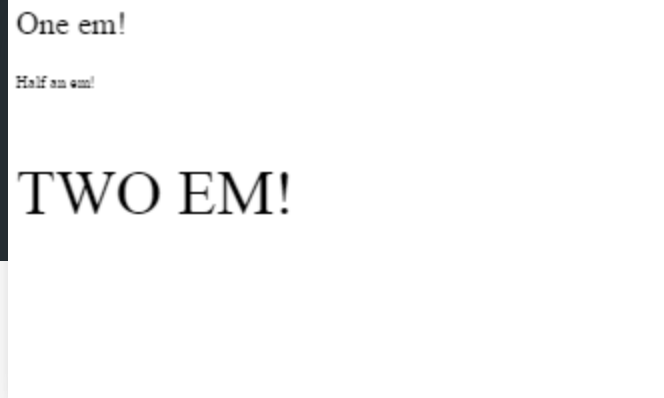
PIXELS

- To adjust font size, the unit used is px (for "pixels")
- A pixel is a dot on your computer screen.
- Specifying font sizes in pixels helps the user to see exactly on their screen what was designed on designers system
 - Though it is assumed that the screens are of similar size.

USING SMARTPHONE - EM SHOULD HELP!!!!

- The font-size unit **em** is a **relative** measure
- One em is equal to the default font size on whatever screen the user is using.
- It doesn't try to tell the smartphone *exactly* how big to make a font: it just says,
"Hey, 1em is the font size that you normally use, so 2em is twice as big and 0.5em is half that size!"

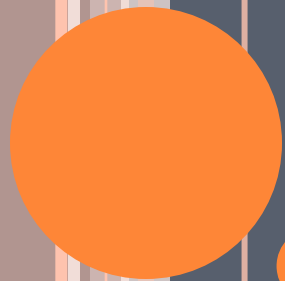
```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Result</title>
5   </head>
6   <body>
7     <p style="font-size: 1em">One em!</p>
8     <p style="font-size: 0.5em">Half an em!</p>
9     <p style="font-size: 2em">TWO EM!</p>
10  </body>
11 </html>
```



One em!

Half an em!

TWO EM!



A FONT OF KNOWLEDGE



28



HOW MANY FONTS DOES CSS KNOW?

- Most computers will understand popular fonts like Verdana, Courier, and Garamond
- Each individual computer has different fonts installed on it.
- CSS has some built-in defaults meant to ensure users see what designers intended.
 - **serif**
 - **sans-serif**
 - **cursive**

```
1 ▾ /*Add your CSS below!*/  
2   h1  
3 ▾ {  
4   font-family:serif;  
5   }  
6   h2  
7 ▾ {  
8   font-family:sans-serif;  
9   }  
10  h3  
11 ▾ {  
12  font-family:cursive;  
13  }  
14  |
```

BACKUP VALUES

- We can tell CSS to try several fonts, going from one to the next if the one mentioned isn't available.

```
1 p {  
2   font-family: Tahoma, Verdana, sans-serif;  
3 }
```



SELECTING HTML ELEMENTS

31

DECORATE <DIV> TAG

- background-color : which you set to a color or hex value
- Height : which you set to a value in pixels
- Width : which is also measured in pixels

LETS DO SOME DESIGNING

Instructions

- Set the background-color to #cc0000
- Set the height to 100px
- Set the width to 100px



LINKS AND TEXT DECORATION

34

LINKS

- Links have a lot of the same properties as regular text
 - font, color, size
- Links also have a property, text-decoration
 - To give more custom flair to the links

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
   .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <p>The below link goes to Google!</p>
9     <a href="http://www.google.com/">Google</a>
10  </body>
11 </html>
```

The below link goes to Google!


[Google](http://www.google.com/)

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <p>The below link goes to Google!</p>
9     <a href="http://www.google.com/">Google</a>
10  </body>
11 </html>
```

The below link goes to Google!

Google

```
1 /*Add your CSS below!*/
2 a
3 {
4   color:#cc0000;
5   text-decoration:none;
6 }
7 |
```



TEXT DECORATION PROPERTY

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  </head>
5  <body>
6
7  <h1>This is heading 1</h1>
8  <h2>This is heading 2</h2>
9  <h3>This is heading 3</h3>
10
11 </body>
12 </html>
```

```
1  h1 {
2      text-decoration: overline;
3  }
4
5  h2 {
6      text-decoration: line-through;
7  }
8
9  h3 {
10     text-decoration: underline;
11 }
```

This is heading 1

~~This is heading 2~~

This is heading 3



CSS SELECTORS

38


SELECTOR

- **Any** HTML element can be a CSS selector!
- ``s, `<table>`s, and even the entire `<body>` can be modified by selecting `ul`, `table`, and `body`

How to???

- So what if you want to grab `<p>`s that are inside two `<div>`s, and not all `<p>`s?

```
<div>  
  <div>  
    <p>I like tacos!</p>
```



```
div div p {  
  /*CSS stuff!*/  
}
```


MULTIPLE SELECTORS

- It's possible to nest HTML elements inside one another

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
   .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <h3>I'm plain old font!</h3>
9     <div>
10      <h3>Me, too!</h3>
11      <div>
12        <h3>Me three!</h3>
13        <div>
14          <h3>Forget you guys. I'm about to be red!</h3>
15        </div>
16      </div>
17    </div>
18  </body>
19 </html>

```

```

1 div div div h3
2 {
3   color:red;
4 }

```

I'm plain old font!

Me, too!

Me three!

Forget you guys. I'm about to be red!

ONE SELECTOR TO RULE THEM ALL

- A very special selector that can be used to apply CSS styling to *every element* on the page
 - the * selector

```
* {  
    border: 2px solid black;  
}
```

- create a two-pixel wide solid black border around *every* element on the HTML page

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet.css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <h3>Boxes within boxes!</h3>
9     <div>
10      <p>Paragraph One</p>
11    </div>
12    <div>
13      <p>Paragraph Two</p>
14    </div>
15    <div>
16      <p>Paragraph Three</p>
17    </div>
18  </body>
19 </html>

```

```

1 *
2 {
3   border:1px solid #3a5fcd;
4   border-radius:5px;
5 }

```

Boxes within boxes!

Paragraph One

Paragraph Two

Paragraph Three

ROCK YOUR SELECTORS – LETS CODE

- **Instructions**
- Set all paragraph text to the hexadecimal color #00E5EE.
- Set all paragraph text for paragraphs nested inside div tags to the hex color #CC0000.
- Put a border with the hex color #3A5FCD around every HTML element. It can be solid, dotted, dashed, 2px, 3px, whatever you like!
- **border : 1px solid #3a5fcd;**

The left side of the slide features a series of vertical stripes in shades of brown, tan, and grey. Overlaid on these stripes are several orange circles of varying sizes. One large circle is positioned near the top, and several smaller circles are scattered below it, some overlapping the stripes.

"C" IS FOR "CASCADING"

45

CASCADE

- Within CSS, all styles cascade from the top of a style sheet to the bottom, allowing different styles to be added or overwritten as the style sheet progresses.

```
p {  
    background: orange;  
    font-size: 24px;  
}  
  
p {  
    background: green;  
}
```

```
p {  
    background: orange;  
    background: green;  
}
```

BRANCHING

- HTML document as a tree
 - Elements "branch out" from the main trunk :
 - `<html>` `</html>` - **parent**
 - The first two big branches :
 - `<head>` and `<body>` are **siblings** and are **children**
- Branches multiply and become finer as we get to elements
 - `<div>`s, `<table>`s, and text

PARENTS, CHILDREN, AND SIBLINGS

```
1 ▾ <!DOCTYPE html>
2 ▾ <html> <!--The trunk of the tree!-->
3 ▾   <head> <!--Child of html, parent of
      title,
4       |       sibling of body-->
5       |       <title></title> <!--Immediate child
      of head,
6       |       |       |       |       child of head AND
      html-->
7       </head>
8 ▾   <body> <!--Child of html, parent of p,
9       |       sibling of head-->
10      |       <p></p> <!--Immediate child of body,
11      |       |       |       child of body AND html-->
12      </body>
13 </html>
```


SWINGING FROM BRANCH TO BRANCH

- **Instructions**
- On the CSS, target ONLY the <a>s that are children ofs and
- Set their text-decoration to none
- Set their font-family to cursive
- Don't change the link that's not part of the unordered list!

NAVIGATION USING “>”

- To grab *direct children* - that is, an element that is *directly* nested inside another element, with no elements in between

```
div > p { /* Some CSS */ }
```

- Works on <p>s that are nested *directly* inside of <div>s
- Will not affect any paragraphs that are, say, nested inside another tag that are in turn nested inside <div>s.

```

1- <!DOCTYPE html>
2- <html>
3-   <head>
4-     <link type="text/css" rel="stylesheet" href="stylesheet.css"/>
5-     <title>Ultimate Text Challenge</title>
6-   </head>
7-   <body>
8-     <p>Introduction: Cascading with CSS</p>
9-     <div>
10-      <p>Synopsis: When you set a property of a selector like 'p' to a
      certain value, that value applies to <em>all</em> p tags.
11-      If, however, you change that same property to a different value
      for a more specific instance of p,
12-      that change will <em>override</em> the 'general rule'.
13-      </p>
14-      <ul>
15-        <li><p>If you say p { font-family: Garamond}, all 'p's will
        have the font Garamond.</p></li>
16-        <li><p>BUT if you say li p {font-family: Verdana}, 'p's
        outside of 'li's will be
17-          in Garamond, and 'p's INSIDE 'li's will be in Verdana.
18-        </p></li>
19-        <li><p>The more specific your selectors are, the higher
        importance CSS gives to the styling you apply!</p></li>
20-      </ul>
21-    </div>
22-    <p>Summary: Greater specificity makes CSS prioritize that particular
    styling.</p>
23-  </body>
24- </html>

```

- Make all `<p>` tags have a font-family of Garamond.
- Make the introduction paragraph and the summary paragraph have a font-weight of bold
- Make the synopsis paragraph have the color #7AC5CD.
- Make the paragraphs in the unordered list have the color #000000 and text-decoration underline.

SOLUTION

```
1 ▾ /*Add your CSS below!*/
2   p
3   {
4     font-family:Garamond;
5     font-weight:bold;
6   }
7   div > p
8   {
9     color:#7ac5cd;
10  }
11  div ul p
12  {
13    color:#000000;
14    text-decoration:underline;
15  }
16
17 |
```



CLASS AND ID SELECTORS

54

BEYOND HTML ELEMENTS

- All HTML elements can be CSS selectors
- * is the universal selector
- There are two important selectors in addition to the universal selector and HTML elements
 - **Class**
 - **ID**

BARCODES AND SERIAL NUMBERS



CLASS



ID




```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <h3 class="red">I'm an h3 header in the red class!</h3>
9     <h3>I'm just a regular old h3 header.</h3>
10    <p class="red">I'm a paragraph in the red class!</p>
11    <p>I'm just a regular old paragraph.</p>
12    <p id="rogue">I'm a rogue paragraph! I do what I want!</p>
13  </body>
14 </html>

```

```

1
2 .red {
3   color: red;
4 }
5
6 #rogue {
7   color: #FF00FF;
8   font-weight: bold;
9   font-family: cursive;
10 }
11

```

I'm an h3 header in the red class!

I'm just a regular old h3 header.

I'm a paragraph in the red class!

I'm just a regular old paragraph.

I'm a rogue paragraph! I do what I want!

KEEPING IT CLASSY

- Classes are useful when you have a bunch of elements that should all receive the same styling.
- Rather than applying the same rules to several selectors, simply apply the same class to all those HTML elements
- Classes are assigned to HTML elements with the word class and an equals sign

```
<div class="square"></div>
<img class="square"/>
<td class="square"></td>
```

```
.square {
    height: 100px;
    width: 100px;
}
```

ID, PLEASE!

- Used when exactly *one* element should receive a certain kind of styling
- IDs are assigned to HTML elements with the word id and an equals sign
- IDs are identified in CSS with a pound sign (#)
- This allows you to apply style to a single instance of a selector, rather than *all* instances

USING ID

```
<div id="first"></div>  
<div id="second"></div>  
<p id="intro"></p>
```

```
#first {  
    height: 50px;  
}  
  
#second {  
    height: 100px;  
}  
  
#intro {
```

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <!--Add your HTML elements with the ID "serious" below!
      -->
9     <div id="serious">Welcome to the serious block...</div>
10    <div id="cool">Iam a cool block :P </div>
11  </body>
12 </html>
```

Welcome to the serious block...

Iam a cool block :P

```
1 /*Define your CSS id #serious below!*/
2 #serious
3 {
4   color:#cc0000;
5   font-family:Courier;
6   font-weight:bold;
7 }
8
```

PUTTING IT ALL TOGETHER

Instructions

- **In HTML:**
 - Give the h2 header an ID of "intro".
 - Give the first h3 and first p a class of "standout". Don't do anything to the second h3 and p!
- **In CSS:**
 - Inside the #intro selector, set color to #B83C3A.
 - Inside the .standout selector, setcolor to #F7AC5F and font-family to Verdana.



PSEUDO-CLASS SELECTORS

63

FINER CONTROL

- A **pseudo-class selector** is a way of accessing HTML items that aren't part of the document tree
- It's very easy to see where a link is in the tree. But where would you find information about whether a link had been clicked on or not?
- Pseudo-class selectors are used to style these kinds of changes in HTML document
- For example
 - Control the appearance of unvisited and visited links & even links the user is hovering over but hasn't clicked!

- The CSS syntax for pseudo selectors is

```
selector:pseudo-class_selector {  
    property: value;  
}
```

```
1 a:hover {  
2     color: #cc0000;  
3     font-weight: bold;  
4     text-decoration: none;  
5 }
```

LINKS

- There are a number of useful pseudo-class selectors for links, including:
- `a:link`: An unvisited link
- `a:visited`: A visited link
- `a:hover`: Hovering mouse over the link

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
   .css"/>
5     <title></title>
6   </head>
7   <body>
8     <div>
9       <a href="http://www.google.com">Google</a><br>
10      <a href="http://www.yahoo.com">Yahoo</a><br>
11      <a href="http://www.csscorp.com">CSS Corp</a>
12    </div>
13  </body>
14 </html>
```

[Google](http://www.google.com)
[Yahoo](http://www.yahoo.com)
[CSS Corp](http://www.csscorp.com)

LETS CODE

- **Instructions**
- Add three links between the div tags. They can link to any websites you like!
- In the CSS , set all a:links to have no text decoration and a color of #008B45.
- Set all a:hovers to have a color of #00FF00.
- Set all a:visiteds to have a color of #EE9A00.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
.css"/>
5     <title></title>
6   </head>
7   <body>
8     <div>
9       <a href="http://www.google.com">Google</a><br>
10      <a href="http://www.yahoo.com">Yahoo</a><br>
11      <a href="http://www.csscorp.com">CSS Corp</a>
12    </div>
13  </body>
14 </html>
```

```
1 a
2 {
3   text-decoration:none;
4 }
5 a:links
6 {
7   color:#008b45;
8   text-decoration:none;
9 }
10
11 a:hover
12 {
13   color:#cc0000;
14   text-decoration:none;
15 }
16
17 a:visited
18 {
19   color:#ee9a00;
20   text-decoration:none;
21 }
```

Google
Yahoo
CSS Corp

Google
Yahoo
CSS Corp

Google
Yahoo
CSS Corp



PSEUDO SELECTORS – CHILD

70

FIRST CHILD

- Another useful pseudo-class used to apply styling to *only* the elements that are the first children of their parents

```
p:first-child {  
    color: red;  
}
```

- Would make all paragraphs that are the first children of their parent elements red

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title></title>
6   </head>
7   <body>
8     <div>
9       <p>I'm the first child!</p>
10      <p>We're not.</p>
11      <p>We're not.</p>
12      <p>We're not.</p>
13      <p>We're not.</p>
14      <p>We're not.</p>
15      <p>We're not.</p>
16    </div>
17  </body>
18 </html>

```

I'm the first child!

We're not.

We're not.

We're not.

We're not.

We're not.

```

1 /*Add your CSS below!*/
2 p:first-child
3 {
4   color:red;
5   font-family:cursive;
6 }
7

```


NTH CHILD

- Can select *any* child of an element after the first child
- Add the child's number in parentheses after the pseudo-class selector

```
p:nth-child(2) {  
    color: red;  
}
```

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title></title>
6   </head>
7   <body>
8     <div>
9       <p>I'm the first child!</p>
10      <p>We're not.</p>
11      <p>We're not.</p>
12      <p>We're not.</p>
13      <p>We're not.</p>
14      <p>We're not.</p>
15      <p>We're not.</p>
16    </div>
17  </body>
18 </html>

```

```

1 /*Add your CSS below!*/
2 p:first-child
3 {
4   color:red;
5   font-family:cursive;
6 }
7 p:nth-child(2)
8 {
9   font-family:Tahoma;
10 }
11 p:nth-child(3)
12 {
13   color:#cc0000;
14 }
15 p:nth-child(4)
16 {
17   background-color:#00ff00;
18 }
19 p:nth-child(5)
20 {
21   font-size:22px;
22 }
23

```

I'm the first child!

We're not.

We're not.

We're not.

We're not.

We're not.

SHOW IT IF YOU KNOW IT!

○ Instructions

- Add three links to the body of the HTML document. They can go anywhere and the text between the tags can say whatever you like.
- In the CSS, set the all a:hovers to have no text-decoration.
- Set the first link to the color#CDBE70. (Remember: in this case, the first link also happens to be the first child of the body element.)
- Set the third link to the color#FFC125.



CSS POSITIONING

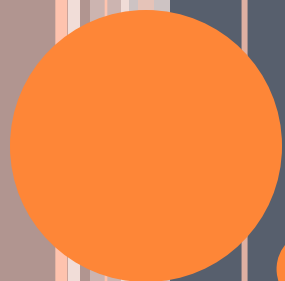
76

POSITION

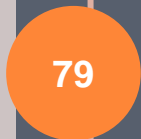
- Controlling the position of HTML elements allows incredibly fine control over how the pages look
- No longer will <div>s sit directly on top of one another!
- Elements populate the page in the **CSS box model**

TAKING UP SPACE

- Each HTML element gets its own box to live in.
- The outermost box of each element goes all the way across the page.
- HTML elements by default ,take up the full width of the page.
- This can be handled with the **display** property



DISPLAY



79



DISPLAY PROPERTY

- **block**: This makes the element a block box. It won't let anything sit next to it on the page! It takes up the full width.
- **inline-block**: This makes the element a block box, but will allow other elements to sit next to it on the same line.
- **inline**: This makes the element sit on the same line as another element, but without formatting it like a block. It only takes up as much width as it needs (not the whole line).
- **none**: This makes the element and its content disappear from the page entirely!


```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
    .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div id="one"></div>
9     <div id="two"></div>
10    <div id="three"></div>
11    <div id="four"></div>
12  </body>
13 </html>
```

```
* {
  border: 1px dashed blue;
}

#one {
  background-color: #FF0000;
}

#two {
  background-color: #0000FF;
}

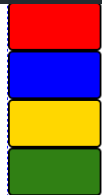
#three {
  background-color: #FFD700;
}

#four {
  background-color: #308014;
}
```

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div id="one"></div>
9     <div id="two"></div>
10    <div id="three"></div>
11    <div id="four"></div>
12  </body>
13 </html>

```



```

1 * {
2   border: 1px dashed blue;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 2px solid black;
9   border-radius: 5px;
10  /*Add your CSS here!*/
11  display: block;
12 }
13
14 #one {
15   background-color: #FF0000;
16 }
17
18 #two {
19   background-color: #0000FF;
20 }
21
22 #three {
23   background-color: #FFD700;
24 }
25
26 #four {
27   background-color: #308014;
28 }

```

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div id="one"></div>
9     <div id="two"></div>
10    <div id="three"></div>
11    <div id="four"></div>
12  </body>
13 </html>

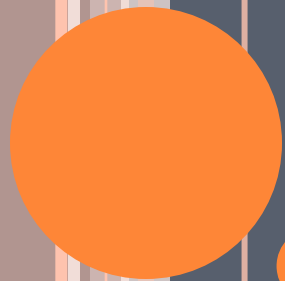
```



```

1 * {
2   border: 1px dashed blue;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 2px solid black;
9   border-radius: 5px;
10  /*Add your CSS here!*/
11  display:inline-block;
12 }
13
14
15 #one {
16   background-color: #FF0000;
17 }
18
19 #two {
20   background-color: #0000FF;
21 }
22
23 #three {
24   background-color: #FFD700;
25 }
26
27 #four {
28   background-color: #308014;

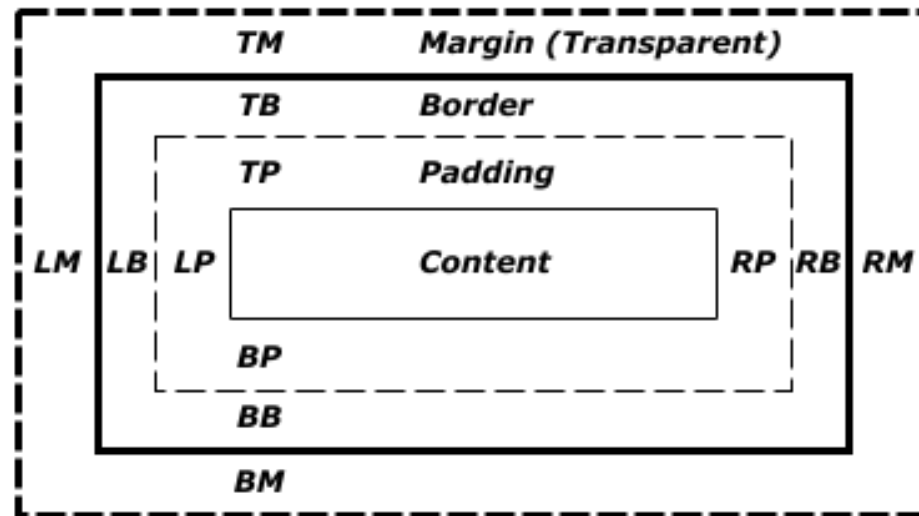
```



BOX MODEL

84

BOX MODEL

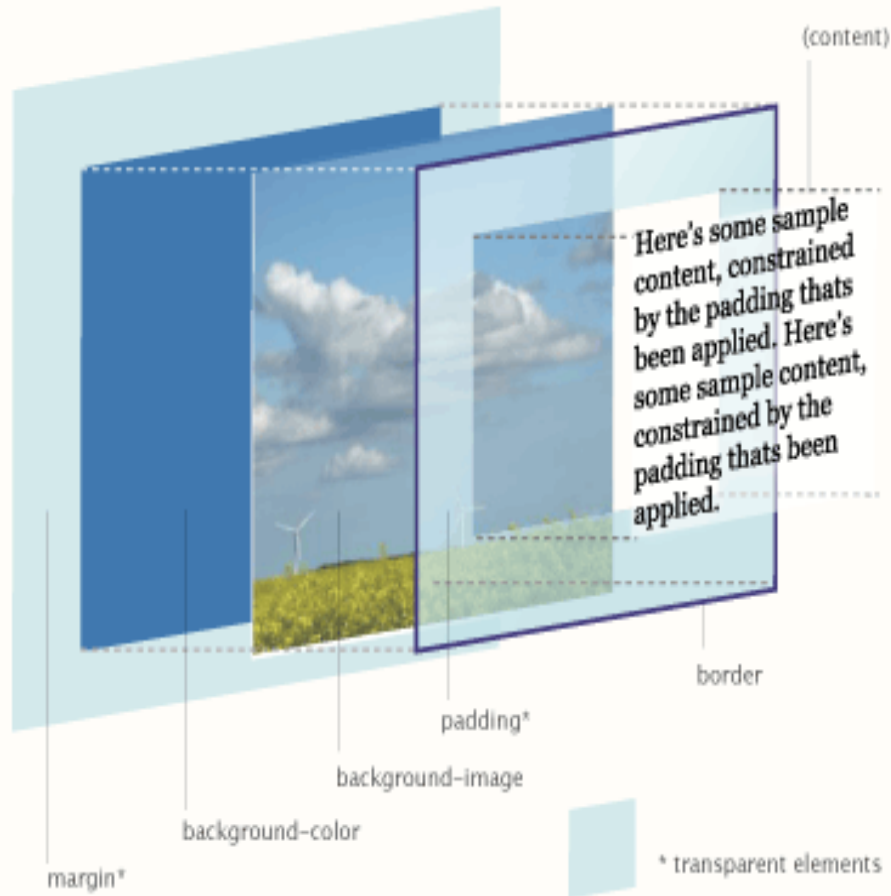


- Margin edge
- Border edge
- - - Padding edge
- Content edge

- Every element on a page is a rectangular box

PROPERTIES OF BOX MODEL

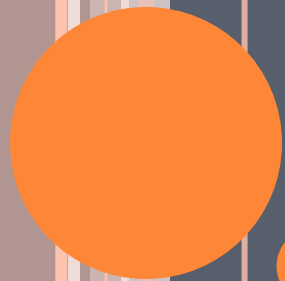
- The **margin** is the space around the element.
 - The larger the margin, the more space between element and the elements around it.
 - Can adjust the margin to move HTML elements closer to or farther from each other.
- The **border** is the edge of the element.
 - It's what we've been making visible every time we set the border property.
- The **padding** is the spacing between the content and the border.
 - Can adjust this value with CSS to move the border closer to or farther from the content.
- The **content** is the actual "stuff" in the box.
 - If we're talking about a `<p>`element, the "stuff" is the text of the paragraph.



```
div {
  border: 6px solid #949599;
  height: 100px;
  margin: 20px;
  padding: 20px;
  width: 400px;
}
```

total width of an element

$\text{margin-right} + \text{border-right} + \text{padding-right} + \text{width} + \text{padding-left} + \text{border-left} + \text{margin-left}$



MARGIN



MARGIN & PADDING DECLARATIONS

- There is more than one way to declare values for certain properties.
 - Longhand - listing multiple properties and values one after the other, in which each value has its own property.
 - Shorthand - listing multiple values with one property.
- The margin and padding properties come in both longhand and shorthand form

!!!!Not all properties have a shorthand alternative!!!

MARGIN

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
   .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9   </body>
10 </html>

```

```

1 * {
2   border: 1px dashed black;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 2px solid black;
9   border-radius: 5px;
10  background-color: #308014;
11  /* margin:auto;*/
12 }

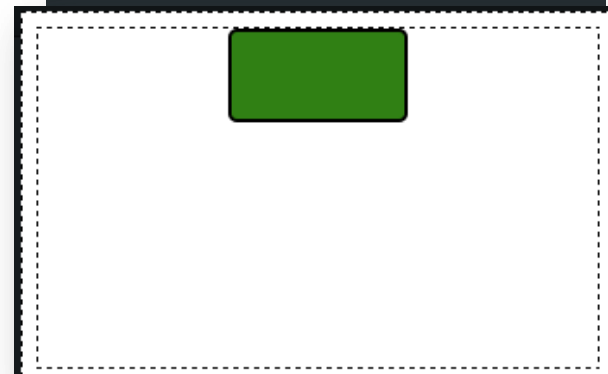
```



```

1 * {
2   border: 1px dashed black;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 2px solid black;
9   border-radius: 5px;
10  background-color: #308014;
11  margin:auto;
12 }

```



MARGIN

```
margin-top: /*some value*/  
margin-right: /*some value*/  
margin-bottom: /*some value*/  
margin-left: /*some-value*/
```

MARGIN

- To set the same value for all four sides of an element

```
div {  
  margin: 20px;  
}
```

- To set one value for the top and bottom and another value for the left and right sides of an element

```
div {  
  margin: 10px 20px;  
}
```

- To set unique values for all four sides of an element

```
div {  
  margin: 10px 20px 0 15px;  
}
```

top

bottom

right

left

MARGIN

```

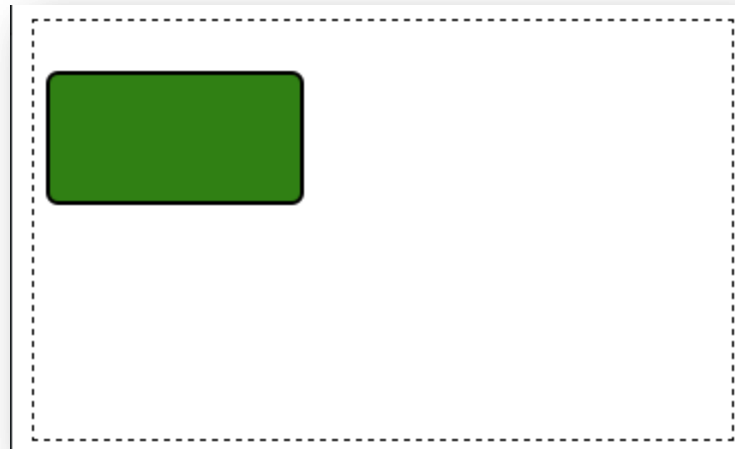
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9   </body>
10 </html>

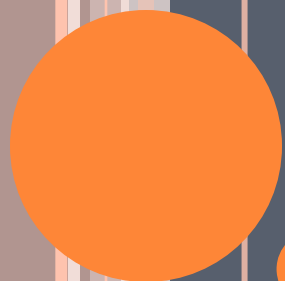
```

```

1 * {
2   border: 1px dashed black;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 2px solid black;
9   border-radius: 5px;
10  background-color: #308014;
11  margin: 20px 50px 10px 5px;
12 }

```





BORDER



94



BORDERING ON INSANITY

- Many HTML elements support the border property.
- Especially useful with tables.
- The border property supports several values.
 - For example, for a border 2 pixels thick, solid, and red

```
selector {  
    border: 2px solid red;  
}
```

```

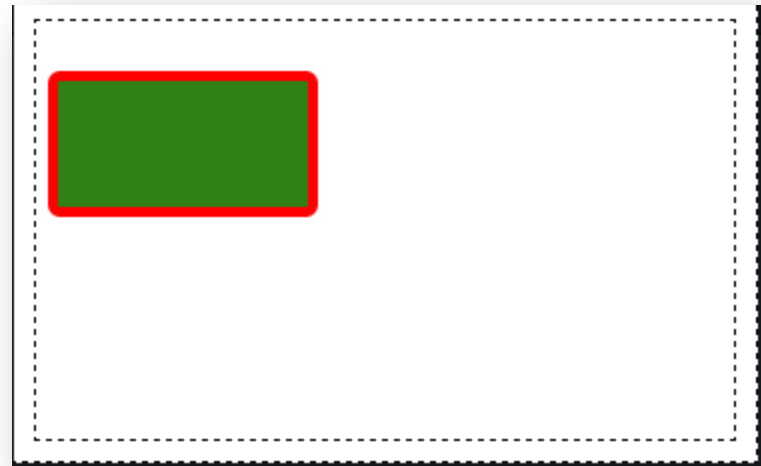
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9   </body>
10 </html>

```

```

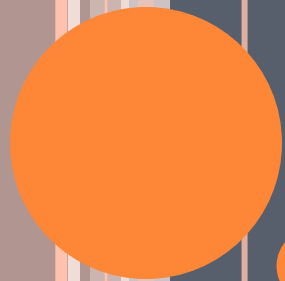
1 * {
2   border: 1px dashed black;
3 }
4
5 div {
6   height: 50px;
7   width: 100px;
8   border: 4px solid #ff0000;
9   border-radius: 5px;
10  background-color: #308014;
11  margin: 20px 50px 10px 5px;
12 }

```



LETS DO SOME CODING

- In the stylesheet.css
- Set your td(table data cells) to have a height of 50px
- Give your td a border of 1px dashed blue.
- Give your table a border of 1px solid black.



PADDING



PADDING

- The space between the border and the innermost layer: the actual content.
- Padding can be set in two ways

```
padding-top: /*some value*/  
padding-right: /*some value*/  
padding-bottom: /*some value*/  
padding-left: /*some-value*/
```

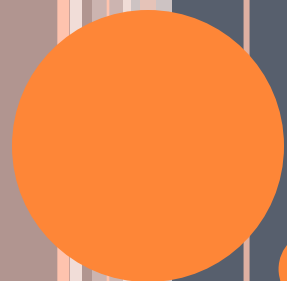
```
padding: value
```

```
padding: value value value value;
```

LETS DO SOME PADDING!

○ Instructions

- Give that div a border of 1px solid black.
- Give it a background color of #CC0000.
- Set its top margin to 10px, its right margin to 5px, its bottom margin to 5px, and its left margin to 50px.
- Set its top padding to 0px, its right padding to 30px, its bottom padding to 0px, and its left padding to 10px.



FLOATING



TO THE RIGHT!

- How do we determine where the html elements go on the page?
- One way to position elements on a page is with the float property.

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet.css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9   </body>
10 </html>

```

```

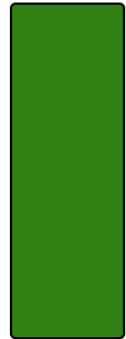
1 div {
2   height: 300px;
3   width: 100px;
4   border: 2px solid black;
5   border-radius: 5px;
6   background-color: #308014;
7   /*Add your CSS here!*/

```



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet.css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9   </body>
10 </html>
```

```
1 div {
2   height: 300px;
3   width: 100px;
4   border: 2px solid black;
5   border-radius: 5px;
6   background-color: #308014;
7   /*Add your CSS here!*/
8   float:right;
9
10 }
```



FLOAT PROPERTY VALUE

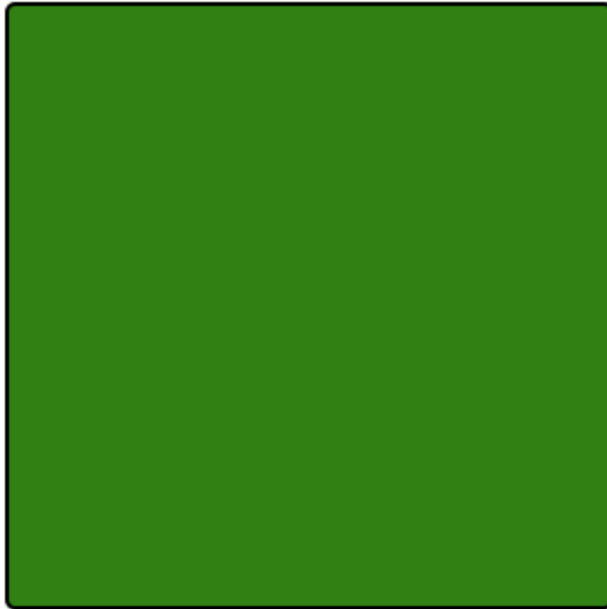
Value	Description
none	The element is not floated, and will be displayed just where it occurs in the text. This is default
left	The element floats to the left
right	The element floats the right

FLOAT FOR TWO

- Can use floated elements to naturally divide our pages into different sections

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" rel="stylesheet" href="stylesheet
      .css"/>
5     <title>Result</title>
6   </head>
7   <body>
8     <div></div>
9     <p>Check it out! I'm a block of text, but the &lt;div>
      and I can live in harmony. Boats on a sea, man. Boats on a sea
    </p>
10  </body>
11 </html>
```

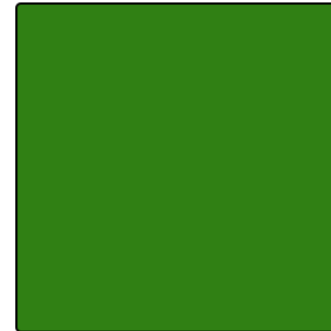
```
1 div {
2   height: 300px;
3   width: 300px;
4   border: 2px solid black;
5   border-radius: 5px;
6   background-color: #308014;
7   /*Add your CSS here!
8   float:right;*/
9 }
10
11 p {
12   font-family: Verdana, sans-serif;
13   font-size: 20px;
14   width: 280px;
15   /*Add your CSS here!
16   float:left;*/
17 }
```



Check it out! I'm a block of text, but the `<div>` and I can live in harmony. Boats on a sea, man. Boats on a sea.

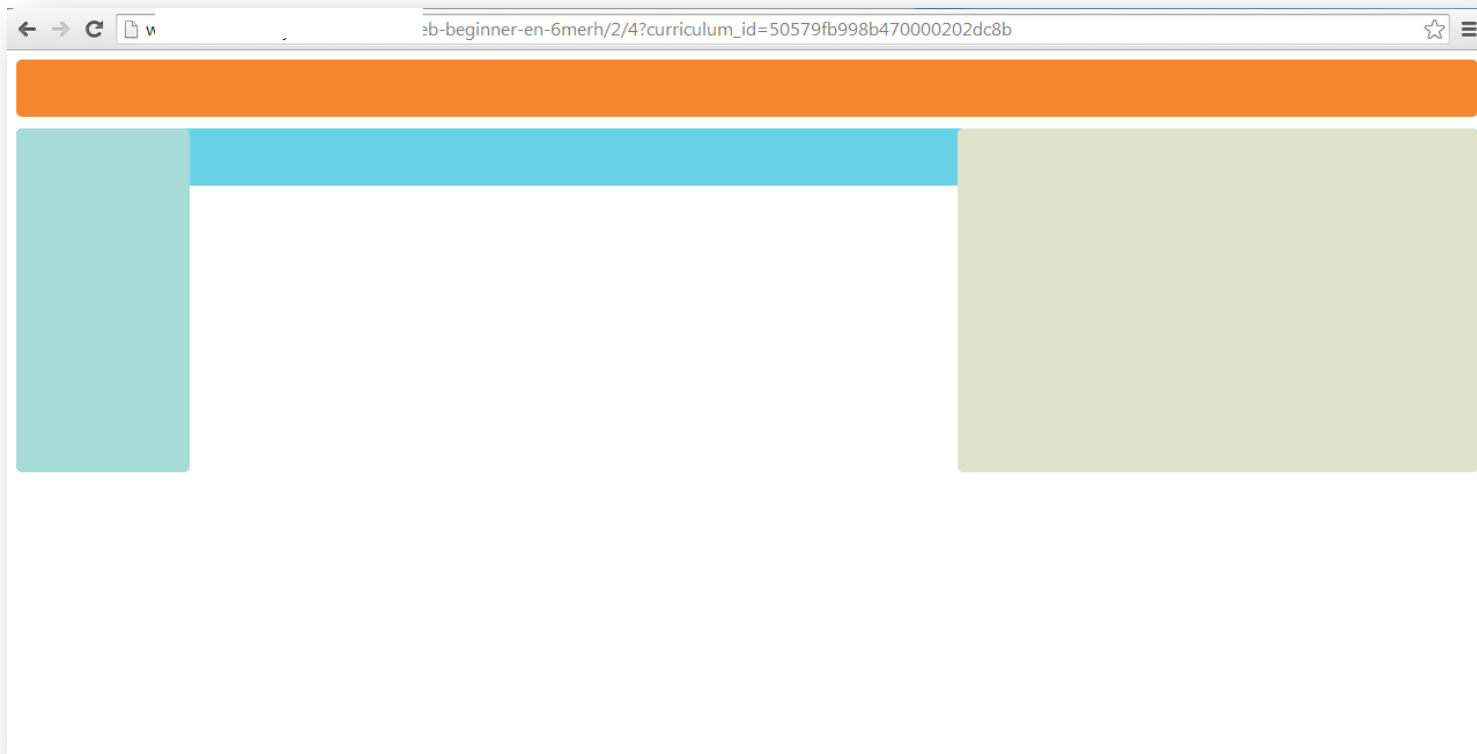
```
1 ▾ div {  
2     height: 300px;  
3     width: 300px;  
4     border: 2px solid black;  
5     border-radius: 5px;  
6     background-color: #308014;  
7 ▾     /*Add your CSS here!*/  
8     float:right;  
9 }  
10  
11 ▾ p {  
12     font-family: Verdana, sans-serif;  
13     font-size: 20px;  
14     width: 280px;  
15 ▾     /*Add your CSS here!*/  
16     float:left;  
17 }
```

Check it out! I'm a block of text, but the <div> and I can live in harmony. Boats on a sea, man. Boats on a sea.



CLEARING ELEMENTS

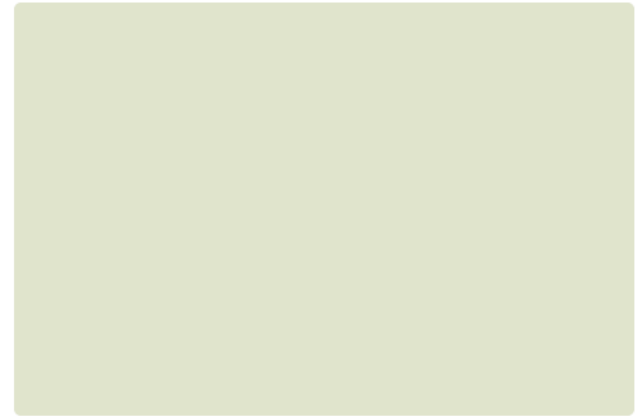
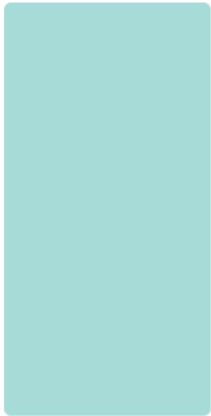
- Sometimes elements *do* end up on top of each other



CLEAR PROPERTY

- Left - will immediately move below any floating elements on the left side of the page
- Right - can also clear elements on the right.
- Both - will get out of the way of elements floating on the left *and* right!

```
element {  
    clear: /*right, left, or both*/  
}
```



```

1  ,
2
3
4
5  #header {
6      height: 50px;
7      background-color: #F38630;
8      margin-bottom: 10px;
9  }
10
11  .left {
12      height: 300px;
13      width: 150px;
14      background-color: #A7DBD8;
15      float: left;
16      margin-bottom: 10px;
17  }

```

```

18  .right {
19      height: 300px;
20      width: 450px;
21      background-color: #E0E4CC;
22      float: right;
23
24      margin-bottom: 10px;
25  }
26  #footer {
27      height: 50px;
28      background-color: #69D2E7;
29      /*Add your CSS here!*/
30      clear: both;
31  }

```

POSITIONING

○ Static

- A static positioned element is always positioned according to the normal flow of the page.
- Static positioned elements are not affected by the top, bottom, left, and right properties.

○ Fixed

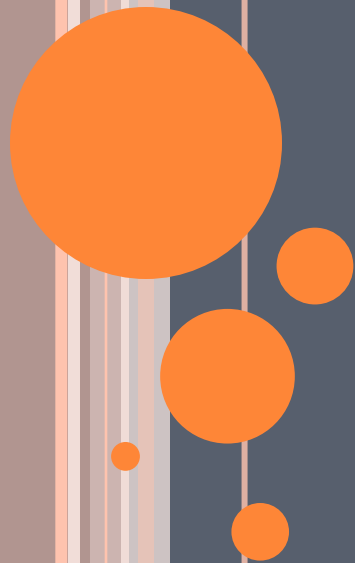
- An element with a fixed position is positioned relative to the browser window, and will not move even if the window is scrolled

○ Relative

- A relative positioned element is positioned relative to its normal position






○ Absolute

- An absolute position element is positioned relative to the first parent element that has a position other than static.



TEXT EFFECTS

BROWSER SUPPORT

Property					
text-shadow	10.0	4.0	3.5	4.0	9.5
word-wrap	5.5	23.0	3.5	6.1	12.1



TEXT SHADOW

```
text-shadow: h-shadow v-shadow blur-radius color|none|initial|inherit;
```

```
<!DOCTYPE html>

<html>
<head>
  <link rel="stylesheet" href="text_style.css" type="text/css">
  <title>Page Title</title>
</head>

<body>
<h3>Lorem Ipsum is simply dummy text of the printing and typesetting industry.
Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,
when an unknown printer took a galley of type and scrambled it to make a type
specimen book. It has survived not only five centuries, but also the leap into
electronic typesetting, remaining essentially unchanged. It was popularised in
the 1960s with the release of Letraset sheets containing Lorem Ipsum passages,
and more recently with desktop publishing software like Aldus PageMaker including
versions of Lorem Ipsum.</h3>

</body>
</html>
```

```
h3
{
  color: white;
  text-shadow: 2px 2px 2px black;
}
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.



MULTIPLE SHADOWS

```
h3
{
  color: white;
  text-shadow: 0px 0px 5px green, 0 0 10px red;
}
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. A
 unknown printer took a galley of type and scrambled it to make a type speci
 remaining essentially unchanged. It was popularised in the 1960s with the re
 publishing software like Aldus PageMaker including versions of Lorem Ipsum



TEXT-ALIGN

```
text-align: left|right|center|justify|initial|inherit;
```

```
h1 {  
    text-align: center;  
}  
  
p.date {  
    text-align: right;  
}  
  
p.main {  
    text-align: justify;  
}
```



TEXT-DECORATION

```
text-decoration: none|underline|overline|line-through|initial|inherit;
```

Property Values

Value	Description
none	Defines a normal text. This is default
underline	Defines a line below the text
overline	Defines a line above the text
line-through	Defines a line through the text

```
h1 {  
  text-decoration: overline;  
}  
  
h2 {  
  text-decoration: line-through;  
}  
  
h3 {  
  text-decoration: underline;  
}
```

This is heading 1

~~This is heading 2~~

This is heading 3



TEXT-INDENT

- Indent the first line of element

```
p {  
  text-indent: 50px;  
}
```

`text-indent: length|initial|inherit;`

Property Values

Value	Description
<i>length</i>	Defines a fixed indentation in px, pt, cm, em, etc. Default value is 0
%	Defines the indentation in % of the width of the parent element



TEXT-OVERFLOW

```
text-overflow: clip|ellipsis|string|initial|inherit;
```

Property Values

Value	Description
clip	Default value. Clips the text
ellipsis	Render an ellipsis ("...") to represent clipped text
<i>string</i>	Render the given string to represent clipped text



TEXT-TRANSFORM

```
text-transform: none|capitalize|uppercase|lowercase|initial|inherit;
```

Property Values

Value	Description
none	No capitalization. The text renders as it is. This is default
capitalize	Transforms the first character of each word to uppercase
uppercase	Transforms all characters to uppercase
lowercase	Transforms all characters to lowercase

```
p.uppercase {  
    text-transform: uppercase;  
}  
  
p.lowercase {  
    text-transform: lowercase;  
}  
  
p.capitalize {  
    text-transform: capitalize;  
}
```

THIS IS SOME TEXT.

this is some text.

This Is Some Text.



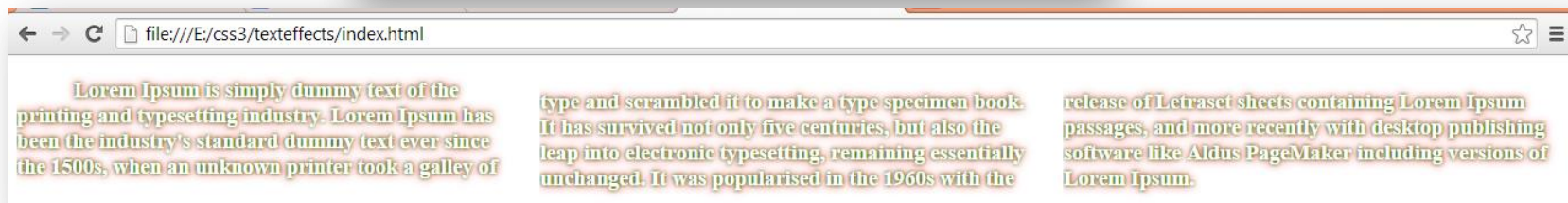
COLUMN COUNT

```
column-count: number|auto|initial|inherit;
```

Property Values

Value	Description
<i>number</i>	The optimal number of columns into which the content of the element will be flowed
auto	Default value. The number of columns will be determined by other properties, like e.g. "column-width"

```
h3
{
    color: white;
    text-shadow: 0px 0px 5px green, 0 0 10px red;
    text-indent: 50px;
    -webkit-column-count: 3;
}
```



COLUMN GAP & RULE

- Pixel gap between the columns

```
div {  
  -webkit-column-gap: 40px; /* Chrome, Safari, Opera */  
  -moz-column-gap: 40px; /* Firefox */  
  column-gap: 40px;  
}
```

- Specify the width, style and color of the rule between columns

```
div {  
  -webkit-column-rule: 4px outset #ff00ff; /* Chrome, Safari, Opera */  
  -moz-column-rule: 4px outset #ff00ff; /* Firefox */  
  column-rule: 4px outset #ff00ff;  
}
```

COLUMN SPAN

```
column-span: 1|all|initial|inherit;
```

Property Values

Value	Description
1	Default value. The element should span across one column
all	The element should span across all columns



COLUMN SPAN

```
h3
{
    color: white;
    text-shadow: 0px 0px 5px green, 0 0 10px red;
    text-indent: 50px;
    -webkit-column-count: 5;
    -webkit-column-gap: 40px;
    -webkit-column-rule: 4px dotted teal ;
}
h2
{
    -webkit-column-span:all;
}
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown

printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five

centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was

popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently

with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.



Sample

```
#msg
{
  color: white;
  text-shadow: 0px 0px 5px green, 0 0 10px red;
  text-indent: 50px;
  -webkit-column-count: 3;
  -webkit-column-gap: 40px;
  -webkit-column-rule: 4px dotted teal ;
}
h3
{
  color: black;
  -webkit-column-span: 1;
}
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Lorem Ipsum has been the industry's standard dummy

text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem

Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker, including versions of Lorem Ipsum.



