



CSS






Part #2

Topics

- History
- Basic Syntax
- Selectors
- Fonts
- Lists
- Colors
- Alignments
- Background images
- Borders



Properties in seven categories

- Fonts 
- Lists 
- Alignment of text 
- Margins 
- Colors 
- Backgrounds
- Borders



How to give property values

- *Keywords* - `left`, `small`, ...
 - Not case sensitive
- *Length* - numbers, maybe with decimal points
 - Units:
 - `px` - pixels
 - `in` - inches
 - `cm` - centimeters
 - `mm` - millimeters
 - `pt` - points
 - `pc` - picas (12 points)
 - `em` - height of the letter 'M'
 - `ex` - height of the letter 'X'
 - No space between the number and the unit
 - e.g., `1.5 in` is illegal!



How to give property values (2)

- Percentage

- a number followed immediately by a percent sign

- URL values

- `url(protocol://server/pathname)`

- Colors

- Color name
- `rgb(n1, n2, n3)`
 - Numbers can be decimal or percentages
- Hexadecimal form: `#XXXXXX`



Font Property: `font-family`

```
font-family: Arial, Helvetica, Futura
```

- Value is a list of font names
 - browser uses the first in the list that is available
- Generic fonts: `serif`, `sans-serif`, `cursive`, `fantasy`, and `monospace` (defined in CSS)
 - Browser has a specific font for each



Font Property: `font-size`

```
font-size: 10pt
```

- Possible values: a length number or a name, such as `xx-small`, `x-small`, `small`, `medium`, `large`, `x-large` and `xx-large`, etc

Font Property: `font-variant`

```
font-variant: small-caps
```

- Default is `normal`, but can be set to `small-caps`



Font Property: `font-style`

```
font-style: italic
```

- Possible values: `italic`, `oblique` (useless), `normal`

Font Property: `font-weight`

```
font-weight: bolder
```

- Possible values: `bolder`, `lighter`, `bold`, `normal`
- Could specify as a multiple of 100 (100 – 900)



Font Shorthand: `font`

```
font: italic bolder 14pt Arial
```

- Order must be: style, weight, size, name(s)

Font Property: `text-decoration`

```
text-decoration: line-through
```

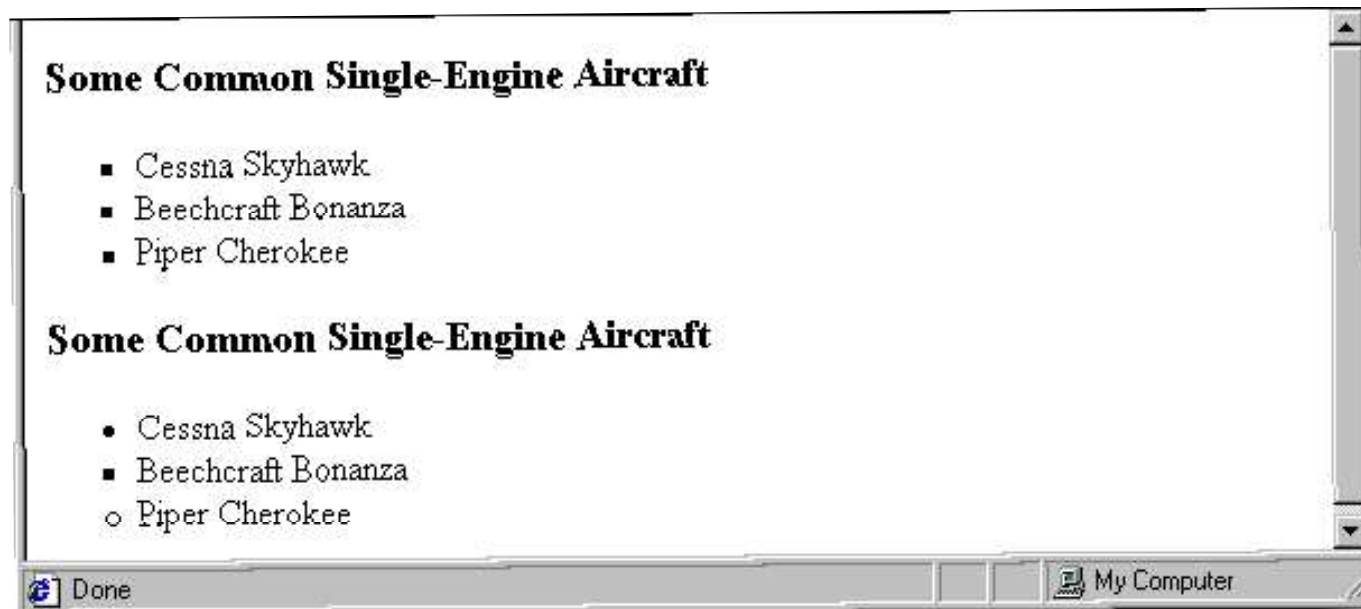
- Possible values: `line-through`, `overline`, `underline`, `none`



List Properties: list-style-type

```
<h3> Some Common Single-Engine Aircraft </h3>
<ul style = "list-style-type: square">
  <li> Cessna Skyhawk </li>
  <li> Beechcraft Bonanza </li>
  <li> Piper Cherokee </li>
</ul>
```

- Possible values: disc, square, circle, none



List Properties: `list-style-type` (2)

```
<ul>
  <li style = "list-style-image: url(bird.jpg)">
    Cessna Skyhawk </li>
  <li style = "list-style-type: square">
    Beechcraft Bonanza </li>
  <li style = "list-style-type: circle">
    Piper Cherokee </li>
</ul>
```



List Properties: `list-style-type` (3)

```
<!DOCTYPE html>
<html>

<head>
  <style>
    ul {
      list-style-image: url('somepicture.gif');
    }
  </style>
</head>

<body>
  <h2>The list-style-image Property</h2>
  <p>The list-style-image property specifies an image as the list item
marker:</p>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Coca Cola</li>
  </ul>
</body>
</html>
```



List Properties: `list-style-type` (3)

- For ordered lists

<code>decimal</code>	Arabic numerals	1, 2, 3, 4
<code>upper-alpha</code>	uc letters	A, B, C, D
<code>lower-alpha</code>	lc letters	a, b, c, d
<code>upper-roman</code>	uc Roman	I, II, III, IV
<code>lower-roman</code>	lc Roman	i, ii, iii, iv



Think about this

Aircraft Types

- I. General Aviation (piston-driven engines)
 - A. Single-Engine Aircraft
 - 1. Tail wheel
 - 2. Tricycle
 - B. Dual-Engine Aircraft
 - 1. Wing-mounted engines
 - 2. Push-pull fuselage-mounted engines
- II. Commercial Aviation (jet engines)
 - A. Dual-Engine
 - 1. Wing-mounted engines
 - 2. Fuselage-mounted engines
 - B. Tri-Engine
 - 1. Third engine in vertical stabilizer
 - 2. Third engine in fuselage

Try the CSS properties seen earlier



Horizontal list -- menu

- http://www.w3schools.com/html/tryit.asp?filename=tryhtml_lists_menu



Colors: {color: red}

```
<style>
  th.red {color: red}
  th.orange {color: orange}
</style>
```

- Recall: three ways to represent colors
 - Color name
 - `rgb (n1, n2, n3)`
 - Numbers can be decimal or percentages
 - Hexadecimal form: `#XXXXXX`



Alignment: `text-indent`

```
<style>  
  p.indent {text-indent:0.5in}  
</style>
```

- Possible values: a length or a % value



Alignment: `text-align`

```
<style>
h1 {text-align:center}
h2 {text-align:left}
h3 {text-align:right}
</style>
```

- Possible values: `left` (the default), `center`, `right`, or `justify`



Alignment: `float`

```
<style>
  img {float: right}
</style>
```

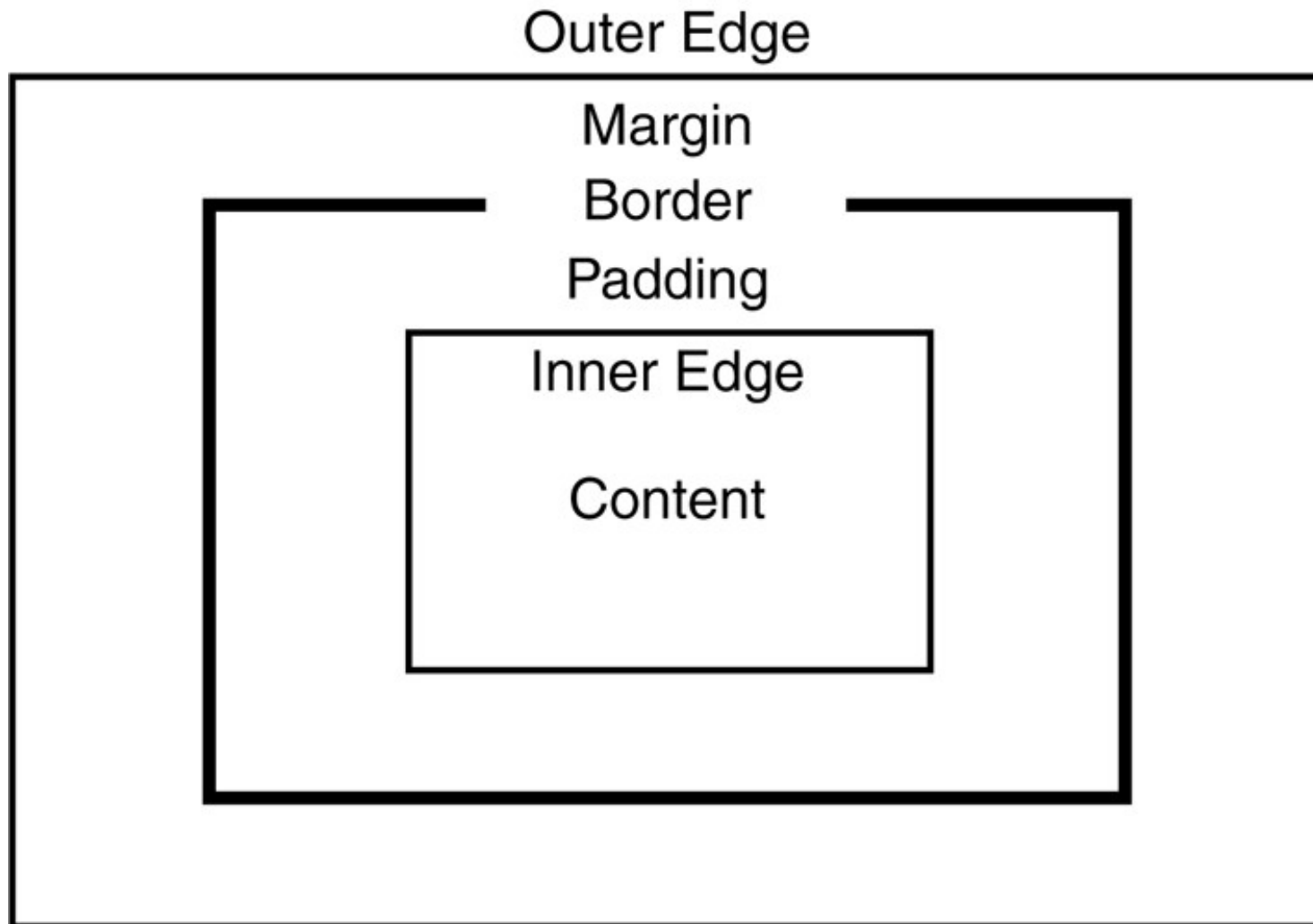
- Text flows around another element
- Possible values: `left`, `right`, and `none` (the default)

Turn off `float`: `Clear`



The Box Model

- For borders and outlines



Margin: {margin: value}

- `margin` values: number values
 - `margin -top`
 - `margin -bottom`
 - `margin -left`
 - `margin -right`



Border: style/color/width

- `border-style` **values:** `none`, `dotted`, `dashed`, `and` `double`
 - `border-top-style`
 - `border-bottom-style`
 - `border-left-style`
 - `border-right-style`
- `border-width` **values:** `thin`, `medium` (default), `thick`, or a length value in pixels
 - `border-top-width`, `border-bottom-width`, `border-left-width`, `border-right-width`
- `border-color` **values:** any color
 - `border-top-color`, `border-bottom-color`, `border-left-color`, `border-right-color`



Padding: {padding: value}

○padding values: number values

- padding-top
- padding-bottom
- padding-left
- padding-right



Background image: `background-image`

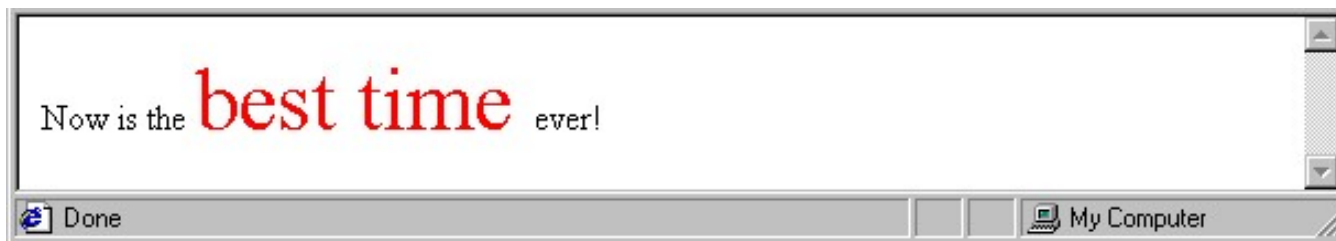
- `background-repeat` **values:** `repeat (default)`, `no-repeat`, `repeat-x`, **or** `repeat-y`
- `background-position` **values:** `top`, `center`, `bottom`, `left`, **or** `right`



: a small span of style

```
<style>
  .bigred {font-size: 24pt;
           font-family: Ariel; color: red}
</style>

...
<p>
  Now is the
    <span class = "bigred">
      best time </span> ever!
</p>
```



<div>: apply to several elements

```
<style>
  .bigred {font-size: 24pt;
           font-family: Arial; color: red}
</style>

...
<div class = "bigred">
<p>
  ...
</p>
<p>
  ...
</p>
</div>
```



Conflict Resolution Rule

- **Selector Rules: Calculating Specificity:**
 - a. `id` selectors
 - b. Class and pseudo-class selectors
 - c. Contextual and simple selectors
 - d. Universal selectors
 - e. Default
- **Order of Specification**
 - when two rules have the same weight, the last rule specified wins.



Conflict Resolution (2)

```
h3 {color: red}  
body h3 {color: blue}
```

```
p.special {font-style:italic !important; font-size:  
14}
```

- **Origin of Rules (Author's vs. Reader's)**

- Important declarations with user origin
- Important declarations with author origin
- Normal declarations with author origin
- Normal declarations with user origin
- Any declarations with browser (or user agent) origin



(Almost) all CSS references

- <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>



Practice example

- Reformat the course outline
- Be creative!
- Exchange your CSS with your friends
- Try your friends' CSS
- Make sure you use the same ids/classnames as your friends

