Dynamic HTML
Cascading Style Sheets™
(CSS)
CSS allows you to specify the style of your page elements (spacing, margins, etc.) separately from the structure of your document (headers, body text, links, etc.). This separation of structure from content allows greater manageability and makes changing the style of your document easier.

1) Define how documents should be presented on screens or in print
2) Put a structure on formatting
3) Defined once and used many times
4) Nested styles are permitted for flexibility
5) HTML tags can be redefined with the desired presentation
6) Custom styles can be defined as needed
Inline Styles (Local)

Inline styles

1. Individual element’s style declared using the **STYLE** attribute
2. Each CSS property followed by a colon and the value of that attribute
3. Multiple properties separated by semicolons
4. Inline styles override any other styles

```html
<P STYLE = "font-size: 20 pt; color: #0000FF">
```
<p>inline styles</p>
<p style="color: green; font-size:20pt">inline styles</p>
<p style="color: red; font-size:30pt">inline styles</p>
<H1 style="color:blue; font-size:40pt">inline styles</H1>
<DIV style="color: lime; font-size:50pt">inline styles</DIV>
Font Lengths

There are lots of style properties whose value is specified as some kind of length, including font sizes, indentation, spacing, etc. Lengths are specified using a number followed by a 2 letter code that indicates the units. Some of the common unit codes are:

<table>
<thead>
<tr>
<th>Unit Code</th>
<th>Units</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>em</td>
<td>Overall height of current font</td>
<td>Hello &lt;DIV style=&quot;font-size:2em&quot;&gt;World&lt;/DIV&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hello</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>World</strong></td>
</tr>
<tr>
<td>px</td>
<td>Pixels</td>
<td>Hello &lt;DIV style=&quot;font-size:8px&quot;&gt;World&lt;/DIV&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hello</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>World</strong></td>
</tr>
<tr>
<td>in</td>
<td>Inches</td>
<td>Hello &lt;DIV style=&quot;font-size:.5in&quot;&gt;World&lt;/DIV&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hello</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>World</strong></td>
</tr>
<tr>
<td>pt</td>
<td>Points size (72.27 points/inch)</td>
<td>Hello &lt;SPAN style=&quot;font-size:8pt&quot;&gt;World&lt;/SPAN&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hello</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>World</strong></td>
</tr>
</tbody>
</table>
## Font Properties

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Possible Values (examples)</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>font-family</code></td>
<td>Sets the font used</td>
<td>Courier, Times, monospace, sans-serif, cursive, fantasy</td>
</tr>
<tr>
<td><code>font-size</code></td>
<td>Sets the size of characters</td>
<td>12pt (12 point size), 200% (twice as big as normal), +5pt (5 points bigger than normal), small, medium, large, larger</td>
</tr>
<tr>
<td><code>font-style</code></td>
<td>italic vs. normal</td>
<td>italic, normal</td>
</tr>
<tr>
<td><code>font-weight</code></td>
<td>Sets the boldness of characters</td>
<td>normal, bold, bolder (bolder than current setting), lighter (lighter than current setting)</td>
</tr>
</tbody>
</table>
## Color

Each element of a document has both a foreground and background color. These can be changed with the following properties:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Possible Values (examples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>color</td>
<td>Sets the foreground color</td>
<td>Red, Green, Orchid</td>
</tr>
<tr>
<td></td>
<td>This sets the text color!</td>
<td>#CECECE, #0000FF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>any valid color specifier</td>
</tr>
<tr>
<td>background-color</td>
<td>Sets the background color</td>
<td>any valid color specifier</td>
</tr>
<tr>
<td>background-image</td>
<td>puts an image behind part of a document</td>
<td>url (<a href="http://www.yahoo.com/images/blah.gif">http://www.yahoo.com/images/blah.gif</a>) url(cookie.gif)</td>
</tr>
</tbody>
</table>
Text Properties

Font properties determine the size, color and style of characters. Text properties determine the spacing and alignment. Here are some of the text properties:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Possible Values (examples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>letter-spacing</td>
<td>adds space between characters.</td>
<td>normal 2px (2 pixels)</td>
</tr>
<tr>
<td>line-height</td>
<td>space between lines</td>
<td>12pt 120% (1.2 times current value)</td>
</tr>
<tr>
<td>text-align</td>
<td>where to align the text</td>
<td>left (this is the default)</td>
</tr>
</tbody>
</table>


Box Properties

Box properties control how things are placed within a box by the browser. All HTML elements are placed in a box by the browser to determine where to start and stop drawing them. These (invisible) boxes control the layout of an entire page. Here are some of the box properties:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Possible Values (examples)</th>
</tr>
</thead>
<tbody>
<tr>
<td>border-color</td>
<td>controls the color of a border around a (otherwise invisible) box.</td>
<td>any color specifier</td>
</tr>
<tr>
<td>border-width</td>
<td>controls the border thickness</td>
<td>any length specifier 1px .25in</td>
</tr>
</tbody>
</table>
| margin        | space between box and any containing elements | any length specifier 10px  
Note: also check out margin-left, margin-right, margin-top, margin-bottom |
# Relative Length Values

<table>
<thead>
<tr>
<th>Name</th>
<th>Type of Unit</th>
<th>What It Is</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>em</td>
<td>EM dash</td>
<td>Width of the letter M for that font (and height)</td>
<td>3em</td>
</tr>
<tr>
<td>ex</td>
<td>x-height</td>
<td>Height of the UpperCase letter is the same as of the lowercase x of that font</td>
<td>5ex</td>
</tr>
<tr>
<td>px</td>
<td>Pixel</td>
<td>Based on the monitor's resolution</td>
<td>125px</td>
</tr>
</tbody>
</table>

xx-small  x-small  small  medium  large  x-large  xx-large
## Absolute Length Values

<table>
<thead>
<tr>
<th>Name</th>
<th>Type of Unit</th>
<th>What It Is</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>cm</td>
<td>Centimeters</td>
<td></td>
<td>5.1cm</td>
</tr>
<tr>
<td>pc</td>
<td>Picas</td>
<td>Generally used to describe font size. 1pc = 12pt</td>
<td>3pc</td>
</tr>
<tr>
<td>mm</td>
<td>Millimeters</td>
<td></td>
<td>25mm</td>
</tr>
<tr>
<td>pt</td>
<td>Point</td>
<td>Generally used to describe font size. 36pt = 1/2 inch</td>
<td>12pt</td>
</tr>
<tr>
<td>in</td>
<td>Inches</td>
<td>1 inch = 2.54cm</td>
<td>2.25in</td>
</tr>
</tbody>
</table>
This is sans-serif, bold, 20pt. But, I'm about to change that.

Now I'm monospace, 12pt (and still bold!)

Now I'm back to where I was before!
This is my paragraph. It should start out with an aqua background and purple text. I'm thinking about changing colors soon. Once I change I'll start a span with some other stuff, perhaps a red background. OK here goes.

This part of the paragraph should have a red background and the text should be in white.

This text is after the end of the span, so it should not have a red background.
WIDER is better.
This paragraph has a thick green border around it and is 2 inches wide.

This paragraph has a thin red border around it a whopping 1/2 inch margin.
We all need to **LEARN** and to understand how to use **CSS**.
<table>
<thead>
<tr>
<th>Cursor Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>auto</td>
<td>Shows it according to how the viewer has it set (plain)</td>
</tr>
<tr>
<td>crosshair</td>
<td>Looks like a cross</td>
</tr>
<tr>
<td>default</td>
<td>Makes the cursor stay the same</td>
</tr>
<tr>
<td>move</td>
<td>Looks like you should be moving something</td>
</tr>
<tr>
<td>hand</td>
<td>The hand you usually see over links</td>
</tr>
<tr>
<td>help</td>
<td>A question mark beside the arrow</td>
</tr>
<tr>
<td>text</td>
<td>The bar you see when the mouse is over text</td>
</tr>
<tr>
<td>wait</td>
<td>The &quot;waiting&quot; hourglass!</td>
</tr>
<tr>
<td>n-resize</td>
<td>An arrow- North</td>
</tr>
<tr>
<td>s-resize</td>
<td>An arrow- South</td>
</tr>
<tr>
<td>e-resize</td>
<td>An arrow- East</td>
</tr>
<tr>
<td>w-resize</td>
<td>An arrow- West</td>
</tr>
<tr>
<td>ne-resize</td>
<td>An arrow- NorthEast</td>
</tr>
<tr>
<td>nw-resize</td>
<td>An arrow- NorthWest</td>
</tr>
<tr>
<td>se-resize</td>
<td>An arrow- SouthEast</td>
</tr>
<tr>
<td>sw-resize</td>
<td>An arrow- SouthWest</td>
</tr>
</tbody>
</table>
What will this do?

<A HREF="#" style="cursor:hand">A handLink</A><br>
<A HREF="#" style="cursor:crosshair">A Cross Link</A><br>
<A HREF="#" style="cursor:help">A help Link</A><br>
<A HREF="#" style="cursor:wait">A wait Link</A><br>
<A HREF="#" style="cursor:s-resize">A south resize Link</A><br>
<A HREF="#" style="cursor:e-resize">A east resize Link</A><br>
<A HREF="#" style="cursor:sw-resize">A south west Link</A><br>
<A HREF="#" style="cursor:se-resize">A south east Link</A><br>
<A HREF="#" style="cursor:text">A text Link</A><br>
<A HREF="#" style="cursor:move">A move Link</A><br>
<A HREF="#" style="cursor: default">A default Link</A>
Embedded Styles (Global)

Embedded CSS (the Intended way)
1. Style Rules defined only once
2. Create one place to store all Styles
3. Linked to many Web pages
4. Embedded Styles require less work!
   - Set Background colors or images
   - Set one or more margins
   - Draw Borders in different sizes and colors.
5. Embedded Styles are different sizes and colors.

To embed your <STYLE> block.
In the <HEAD> after the <TITLE>, type:

```html
<STYLE TYPE="text/css">  Here I embed my styles with joy </STYLE>
```
Embedded Styling

Figure 1. Three Parts to a style sheet rule.
What do the following do?

1) A {text-decoration: none;}

This tag would allow you to eliminate the underscore on the hypertext links

2) B {font-family: arial; color: red; font-size: 125%}

The Bold tag produces darker and thicker text, now you can add a few more attributes

3) H1 {font-family: arial black; color: blue; font-size: 200%}

It adds Family, Color and Size at 100% larger than the base text for the page.

4) P {font-family: arial; font-size: 13pt; color: blue; margin-left: 1in; margin-right: 1in; text-align: right; }

The paragraph tag has both margins by 1 inch.
5) P B { background: yellow; }
   Now *if, and only if*, the BOLD tag occurs within a paragraph, a yellow background will appear behind the text marked between the <B> and </B> tags.

6) H1, H2, H3 {text-align: center}
   does exactly the same thing as:
   H1 {text-align: center}
   H2 {text-align: center}
   H3 {text-align: center}

7) OL LI {list-style: upper-alpha}
   OL OL LI {list-style: upper-roman}
   OL OL OL LI {list-style: lower-alpha}
   OL OL OL OL LI {list-style: decimal}
8) P UL STRONG {color: blue}
and use the sequence <p><ul><strong> in the document, the second, more specific rule applies, coloring the contents of the <strong> tag blue.

9) .italic {font-style: italic}
creates a generic class named italic. To use it, simply include its name with the class attribute. So, for instance, use <p class=italic> or <h1 class=italic> to create an italic paragraph or header.

10) H1#blue { color : rgb(0,0,255)}
#yellow { color : yellow}

Within the document, we use <h1 id=blue> to create a blue heading, or add id=yellow to almost any tag to turn it yellow.

11) table {color:blue}// tr {color:red}// td {color:green}
12) Drop down list

```css
select {
    border: thin dotted #FF0000;
    font-family: "Comic Sans MS";
    background-color: #FFCCCC;
}
```

13) Text Field / Button

```css
input {
    background-color: #FFFFCC;
    border: thin dotted #FF9900;
    font-family: "Comic Sans MS";
    background-image: url(picture.jpg);
}
```

14) TextArea

```css
textArea{
    background-color: #FFFFCC;
    font-family: "Comic Sans MS";
    background-image: url(picture.gif);
}
```
see my styles ....

This should show up as green and 24 pt.

This should show up as red, bold, Arial, and 24pt.
Hi there..............
how is your day............
Well, I am ok.
We need to focus on many issues to be able to understand the concept of CSS and its likeable properties.
<form><input type="button" value="ClickMe" name="buttonX" onClick="alert('Hi there');"></form>
<style>
select {font-family:Arial; background-color: #FFAA00; width:200px; font-size:2em; color:#00bb22; } 
</style>
<form>
select size=3>
<option>F5
<option>F15
<option>F16
<option>C5
<option>C141
<option>Cobra
</select></form>
<style>
p {color:rgb(45,100,255); font-size:20pt}
TEXTAREA {width: 500px; height: 400px; color:#f8f888;background-color:blue; font-size:30pt;font-weight:bold; margin-left: 150px;}
input{color:red; font-size:25pt;}
</style>
<script>function change()
{myForm.txtArea.value=myForm.txt.value;}
</script>
<form name="myForm">
<p>Type here:  <input type="text" name="txt"  size=20>
<input type="button" name="theButton" value="clickHere" onClick="change();">
<TEXTAREA name="txtArea">you will see your writing here..
</TEXTAREA>
</form>
CSS lets you flow text around graphics or text plus text around tables. It uses a Float property.
CSS lets you flow text around graphics or text plus text around tables. It uses a Float property.

CSS lets you flow text around graphics or text plus text around tables. It uses a Float property.
Hey! Wow!! Amazing!!! Impressive!!!! k00l!!!!!

Fantastic!!!!!!!
CSS Positioning (CSS-P) is more accurate than either graphics or tables and the results are displayed much faster. CSS introduces the position property and a capability called *absolute positioning*, which gives us greater control over how documents are displayed.

In other words, CSS-P allows us to position elements on the screen either exactly where we want them or in relation to other elements on the screen.
Left: defines the amount of space between the element and the left edge of the browser window. Top defines the space between the element and the top of the window.
A style does not go out of style as long as it adapts itself to its period. When there is an incompatibility between the style and a certain state of mind, it is never the style that triumphs.
A style does not go out of style as long as it adapts itself to its period. When there is an incompatibility between the style and a certain state of mind, it is never the style that triumphs.

Our words have wings, but fly now where we would. There are aphorisms that, like airplanes, stay up only while they are in motion.
A style does not go out of style as long as it adapts itself to its period. When there is an incompatibility between the style and a certain state of mind, it is never the style that triumphs.

Our words have wings, but fly now where we would. There are aphorisms that, like airplanes, stay up only while they are in motion.
<p>Place any text you want here.....</p>
This is Jordan's Flag
Browsers that don’t support images will still be able to read text that has been added as CSS. This also benefits your search engine ranking—search engines can’t index text that’s part of an image. It isn’t possible to add more than one background image to your document. Making site navigation easy is one area in which CSS really comes into its own.
Browsers that don’t support images will still be able to read text that has been added as CSS.

This also benefits your search engine ranking—search engines can’t index text that’s part of an image.

It isn’t possible to add more than one background image to your document. Making site navigation easy is one thing, tricky.
Hi there............

How do you do?...

How is things?..
Hi there...........

How do you do?...

How is things?..
The **z-index** attribute allows you to properly layer overlapping elements. Elements that have higher **z-index** values are displayed in front of elements with lower **z-index** values.
output
<style TYPE="text/css">
BODY{ background-color:#CCFFDDDD}
DIV   {text-align: center; margin-bottom: 0.3em; 
   width: 30%; position: relative; 
   left:25%;       padding:0.3em}
</style>

<BODY>

<DIV STYLE="border-style:solid"> Solid border</DIV>
<DIV STYLE="border-style:double"> Double border</DIV>
<DIV STYLE="border-style:groove"> Groove border</DIV>
<DIV STYLE="border-style:ridge"> Ridge border</DIV>
<DIV STYLE="border-style:inset"> Inset border</DIV>
<DIV STYLE="border-style:outset"> Outset border</DIV>
<DIV STYLE="border-style:outset double; color:red"> Outset Double Blue border</DIV>

</BODY>
output

Solid border
Double border
Groove border
Ridge border
Inset border
Outset border
Outset Double red border
<title>Dynamic Positioning</title>

<script>
var speed=5;
var count=10;
var firstLine="Text Growing";
var fontStyle=["monospace", "arial","serif"];  
var fontStyleCount=0;

function start()
{
    window.setInterval ("run()",100);
}
</script>
function run() {
  count +=speed;

  if ((count%400)==0) {
    speed*=-1;
    pText.style.color=(speed<0)? "red" : "blue";
    firstLine=(speed<0)? "Text Shrinking" : "Text Growing";
    pText.style.fontFamily = fontStyle[++fontStyleCount % 3];
  }

  pText.style.fontSize=count/3;
  pText.style.left=count;
  pText.innerHTML=firstLine+ "<BR>  Font Size: "+count+"px";
}
</SCRIPT>
Welcome!
Dynamic HTML: Filters

```html
<style>
BODY{  background-color:#CCFFDDDD}
TABLE {font-size:2em; font-family:Arial;
background-color:#FFCCDDCC; border-style:outset;
border-collapse:collapse;  }
TD { border-style:inset; text-align:center; padding:2ex; }
</style>
<BODY><TABLE>
<TR> <TD STYLE="filter:fliph">Text</TD><TD>Text</TD></TR>
<TR> <TD STYLE="filter:flipv fliph">Text</TD>
<TD STYLE="filter:flipv">Text</TD></TR>
</TABLE>
</BODY>
```
output
Good Luck