CSS - WHAT

- CSS is an acronym for Cascading Style Sheets.
- CSS is a style language that defines visual styles of HTML documents.
- CSS can solve many design problems without adding images or changing the HTML

WHY?

- Separation of presentation from markup. Cleaner code.
- Easier to manage style changes.

History

1994

CSS was first proposed by Håkon Wium Lie.

1996

CSS level 1 was published as a W3C Recommendation.

1998

CSS level 2 was published as a W3C Recommendation.

2011

CSS level 2.1 was published as a W3C Recommendation.

CSS level-1

CSS was proposed in 1994 as a web styling language, to solve some of the problems of Html 4. There were other styling languages proposed at this time, such as Style Sheets for Html and JSSS but CSS won

- Font properties such as typeface and emphasis
- Color of text, backgrounds, and other elements
- Text attributes such as spacing between words, letters, and lines of text
- Alignment of text, images, tables and other elements
- Margin, border, padding, and positioning for most elements
- Unique identification and generic classification of groups of attributes

NOTE:- The W3C no longer maintains the CSS 1 Recommendation

CSS level-2

CSS level -2 superset of CSS 1, CSS 2 includes a number of new capabilities like absolute, relative, and fixed positioning of elements and z-index, the concept of media types, support for aural style sheets (which were later replaced by the CSS 3 speech modules)[44] and bidirectional text, and new font properties such as shadows.

NOTE:- The W3C no longer maintains the CSS 1 Recommendation

CSS level-2.1

CSS level 2 revision 1, referred as CSS 2.1, fixes errors in CSS 2, removes poorly supported or not fully interoperable features and adds already implemented browser extensions to the specification.

CSS 2.1 went to Proposed Recommendation on 12 April 2011. After being reviewed by the W3C Advisory Committee, it was finally published as a W3C Recommendation on 7 June 2011.

CSS level-3

- Unlike CSS 2, which is a large single specification defining various features, CSS 3 is divided into several separate documents called "modules".
- Each module adds new capabilities or extends features defined in CSS 2, preserving backward compatibility.
- Work on CSS level 3 started around the time of publication of the original CSS 2 recommendation.
- The earliest CSS 3 drafts were published in June 1999.

CSS level-4 ??

- There is **no single, integrated CSS4 specification**, because the specification has been split into many separate modules which level independently.
- Modules that build on things from CSS Level 2 started at Level 3. Some of them have already reached Level 4 or are already approaching Level 5.
- Other modules that define entirely new functionality, such as Flexbox,have been designated as Level 1 and some of them are approaching Level 2.
- So, No more any CSS 4, 5… as in future.

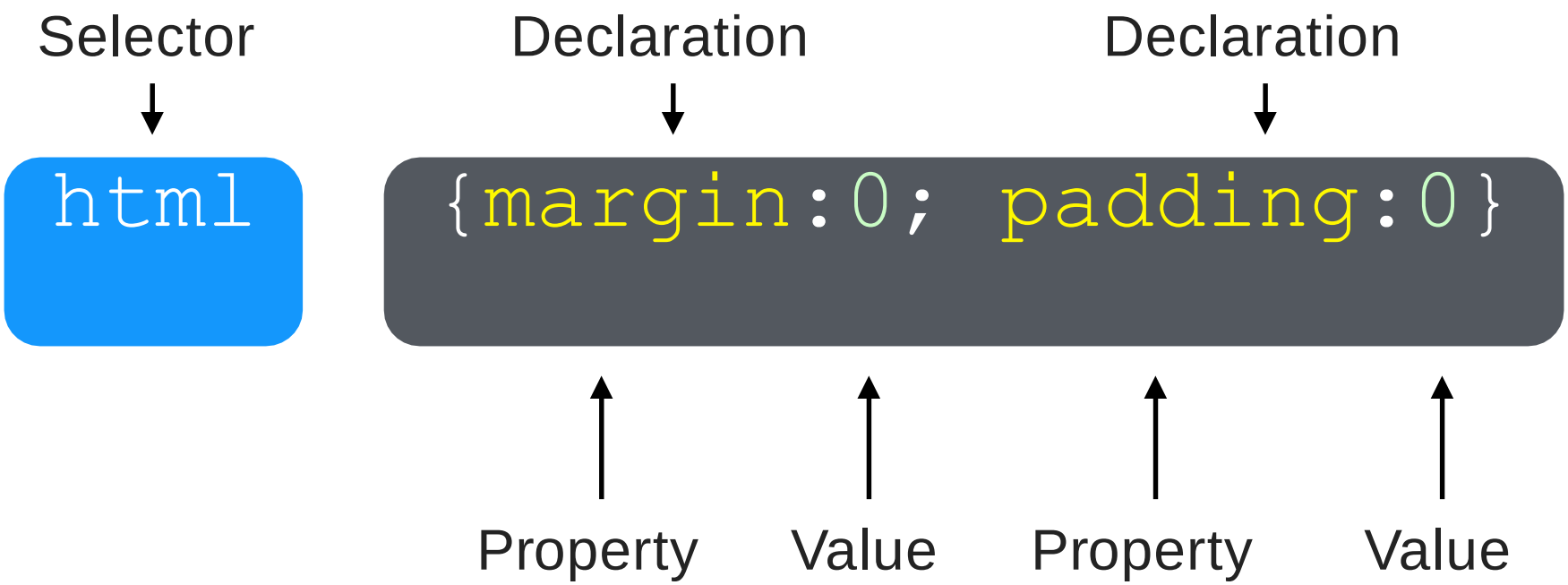
CSS Syntax

A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document.

A style rule is made of three parts −

- **Selector** − A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.
- **Property** - A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be color, border etc.
- **Value** - Values are assigned to properties. For example, color property can have value either red or #F1F1F1 etc.

CSS Syntax




```
body {background-color:#d20c0c;}
```

```
h1 {font-size:12px;}
```

CSS Selectors

A CSS selector is the part of a CSS rule set that actually selects the content.

Pattern	Meaning	Level
*	any element	2
E	an element of type E	1
E.warning	an E element whose class is "warning"	1
E#myid	an E element with ID equal to "myid"	1

CSS Selectors

1-Element Type Selector

An element type selector matches every instance of the element in the document tree with the corresponding element type name.

Example

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

2 Id Selectors

- The id selector is used to define style rules for a *single* or *unique* element.
- The id selector is defined with a '#' (hash) sign immediately followed by the id value.

Example

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

3 Class Selector

- The class selector selects elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the name of the class.

Example

```
.center {
      text-align: center;  color: red;
   }
```

Grouping Declarations

```
p {
    color: red;
}

p {
    fontsize: 12px;
}

p {
    lineheight: 15px;
}
```

```
p {
    color: red;  font-
    size: 12px;  line-
    height: 15px;
}
```

Grouping Selectors

```
h1 {
    color: red;  font-
    weight: bold;
}

h2 {
    color: red;  font-
    weight: bold;
}

h3 {
    color: red;  font-
    weight: bold;
}
```

```
h1, h2, h3 {
     color: red;  font-
     weight: bold;
 }
```

Comments

```
/* Comment here */
p {
          margin: 1em; /* Comment here */
          padding: 2em;
          /* color: white; */
          backgroundcolor: blue;
 }
 /*multiline
 comment here*/
```

Ways to use CSS

Three Ways to Insert CSS

- External style sheet
- Internal style sheet
- Inline style

External Style sheet

- With an external style sheet, you can change the look of an entire website by changing just one file!
- Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:
- The style sheet file must be saved with a .css extension.

For Example:

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

Internal Style sheet

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

For Example:

```
<head>
<style>
body {
background-color: linen;
}
h1 {
color: maroon;  margin-left: 40px;
}
</style>
</head>
```

Inline Style sheet

- An inline style may be used to apply a unique style for a single element.
- To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
- The example below shows how to change the color and the left margin of a <h1> element:

For Example

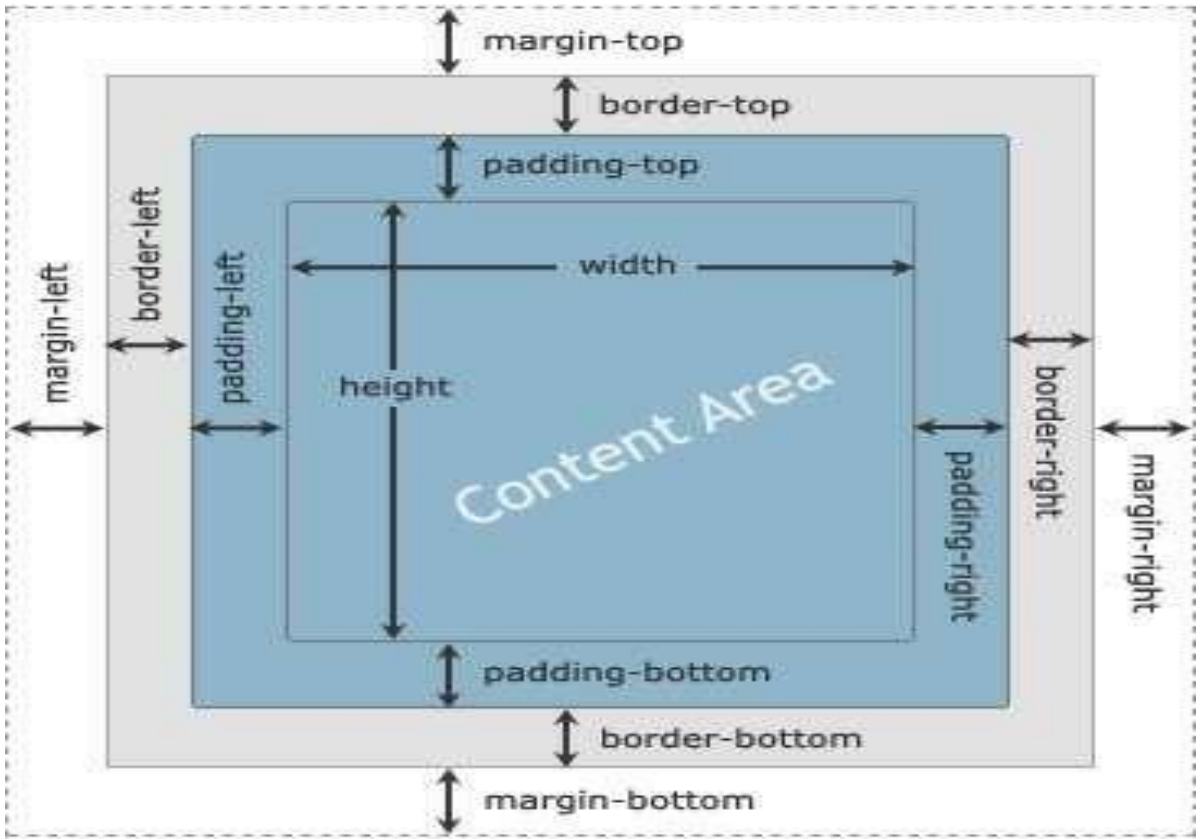
```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```

Measurement units

Unit	Description	Example
%	Defines a measurement as a percentage relative to another value, typically an enclosing element.	p {font-size: 16pt; line-height: 125%;}
cm	Defines a measurement in centimeters.	div {margin-bottom: 2cm;}
em	A relative measurement for the height of a font in em spaces. Because an em unit is equivalent to the size of a given font, if you assign a font to 12pt, each "em" unit would be 12pt; thus, 2em would be 24pt.	p {letter-spacing: 7em;}
ex	This value defines a measurement relative to a font's x-height. The x-height is determined by the height of the font's lowercase letter x.	p {font-size: 24pt; line-height: 3ex;}
in	Defines a measurement in inches.	p {word-spacing: .15in;}
mm	Defines a measurement in millimeters.	p {word-spacing: 15mm;}
pc	Defines a measurement in picas. A pica is equivalent to 12 points; thus, there are 6 picas per inch.	p {font-size: 20pc;}
pt	Defines a measurement in points. A point is defined as 1/72nd of an inch.	body {font-size: 18pt;}
px	Defines a measurement in screen pixels.	p {padding: 25px;}

BOX Model

The CSS box model describes the **rectangular boxes** that are generated for elements in the document tree and laid out according to the **visual formatting model**.



CSS Margin

- The CSS margin properties are used to generate space around elements.
- The margin properties set the size of the white space outside the border.
  - margin-top
  - margin-right
  - margin-bottom
  - margin-left
- All the margin properties can have the following values:
  - auto - the browser calculates the margin
  - l*ength* - specifies a margin in px, pt, cm, etc.
  - *%* - specifies a margin in % of the width of the containing element
  - inherit - specifies that the margin should be inherited from the parent element

CSS Margin

For Example :

```
p {
    margin-top: 100px;
    margin-bottom: 100px;
    margin-right: 150px;
    margin-left: 80px;
  }
```

For Example (Shorthand): (clock-wise rule)

```
p {
    margin: 100px 150px 100px 80px;
    margin: 25px 50px 75px 100px;
  }
```

CSS (Shortend Rules)

- If the margin property has four values:

  margin: 25px 50px 75px 100px;
  - top margin is 25px
  - right margin is 50px
  - bottom margin is 75px
  - left margin is 100px
- If the margin property has three values:

  margin: 25px 50px 75px;
  - top margin is 25px
  - right and left margins are 50px
  - bottom margin is 75px
- If the margin property has two values:

  margin: 25px 50px;
  - top and bottom margins are 25px
  - right and left margins are 50px
- If the margin property has one value:

  margin: 25px;
  - all four margins are 25px

CSS Padding

- The CSS padding properties are used to generate space around content.
- The padding clears an area around the content (inside the border) of an element.
- CSS has properties for specifying the padding for each side of an element:
  - padding-top
  - padding-right
  - padding-bottom
  - padding-left

For Example:

```
p {
    padding-top: 50px;
    padding-right: 30px;
    padding-bottom: 50px;
    padding-left: 80px;
}
```

Padding shorthand

```
p {
    padding: 50px 30px 50px 80px;
}
```

CSS Height and Width

- The height and width properties are used to set the height and width of an element.
- The height and width can be set to auto (this is default. Means that the browser calculates the height and width), or be specified in *length values*, like px, cm, etc., or in percent (%) of the containing block.

For Example

```
div {
    height: 200px;
    width: 50%;
    background-color: powderblue;
    }
```

CSS Height and Width

All CSS Dimension Properties

- Height Sets the height of an element
- max-height Sets the maximum height of an element
- max-width Sets the maximum width of an element
- min-height Sets the minimum height of an element
- min-width Sets the minimum width of an element
- Width Sets the width of an element

CSS Fonts

CSS font properties define the font family, boldness, size, and the style of a text.

Not all computers have the same fonts installed

- **font-family** Specifies the font family for text
- **font-size** Specifies the font size of text
- **font-style** Specifies the font style for text (e.g. normal or italic)
- **font-variant** Specifies if text should be small-caps
- **font-weight** Specifies the weight of a font
- **font** Sets all the font properties in one declaration

CSS Fonts

'font-family'	[[<family-name> | <generic-family>] [, <family-name>| <generic-family>]*] | inherit Example: font-family: "Times New Roman", Times, serif;
'font-size'	<absolute-size> | <relative-size> |<length> | keywords: xx-small, x-small, small, medium, large, x-large, xx-large. size in pixel values (e.g. 12px, 16px, etc.)
'font-style'	normal | italic | oblique | inherit
'font-variant'	normal | small-caps | inherit
'font-weight'	normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | inherit
'font'	[['font-style' || 'font-variant' || 'font-weight']? 'font-size' [/ 'line-height']? 'font-family'] | caption | icon | menu | message-box | small-caption | status-bar | inherit
'height'	<length> | <percentage> | auto |inherit
'letter-spacing'	normal | <length> | inherit
'line-height'	normal | <number> | <length> |<percentage> | inherit

CSS Text

'text-align'	left | right | center | justify | inherit	a nameless value that acts as 'left' if 'direction' is 'ltr', 'right' if 'direction' is 'rtl'
'text-decoration'	none | [underline || overline || line-through || blink] | inherit	none
'text-indent'	<length> | <percentage> | inherit	0
'text-transform'	capitalize | uppercase | lowercase | none | inherit	none
'text-align'	left | right | center | justify | inherit	a nameless value that acts as 'left' if 'direction' is 'ltr', 'right' if 'direction' is 'rtl'

CSS Background

CSS background properties are used to define the background effects of an element.

- **background-color** sets the background color of an element
- **background-image** sets the background image for an element
- **background-repeat** property is used to control the repetition of an image in the background.
- **background-attachment** sets whether a background image is fixed or scrolls with the rest of the page
- **background-position** controls the position of a background image
- **background** is used as shorthand to sets all the background properties in one declaration

CSS Background

'background-attachment'	scroll | fixed | inherit	scroll
'background-color'	<color> | transparent | inherit	transparent
'background-image'	<url> | none | inherit	none
'background-position'	[[<percentage> | <length> | left | center | right] [<percentage> |<length> | top | center | bottom]?] | [[left | center | right] || [top | center | bottom]] | inherit	0% 0%
'background-repeat'	repeat | repeat-x | repeat-y | no-repeat | inherit	repeat
'background'	['background-color' || 'background-image' || 'background-repeat' ||'background-attachment' ||'background-position'] | inherit	see individual properties

CSS Border

'border-color'	[<color> |	see individual properties
'border-spacing'	<length> <length>?	0
'border-style'	None,solid,dotted,dashed,double,groove,ridge, inset,outset,hidden	see individual properties
'border-top''border-right''border-bottom''border-left'	[<border-width> || <border-style> ||'border-top-color']	see individual properties
'border-top-color' 'border-right-color''border-bottom-color' 'border-left-color'	<color>	the value of the 'color' property
'border-top-style' 'border-right-style''border-bottom-style' 'border-left-style'	<border-style>	None
'border-top-width' 'border-right-width''border-bottom-width' 'border-left-width'	<border-width>	medium
'border-width'	length in px, pt or cm or it should beset to *thin, medium or thick*.	see individual properties
'border'	[<border-width> || <border-style> ||'border-color']	see individual properties

CSS Pseudo Classes

CSS pseudo-classes are used to select the **current condition** of an element.

The most common pseudo-classes are **links**
Links can be **styled differently depending** on what **state** they are in.

a:link Selects an unvisited link
a:visited Selects visited links
a:active Selects the active link
a:hover Selects links on mouse over

CSS Positioning

The CSS positioning properties allow you to position an element. Elements can be positioned using the top, bottom, left, and right properties.

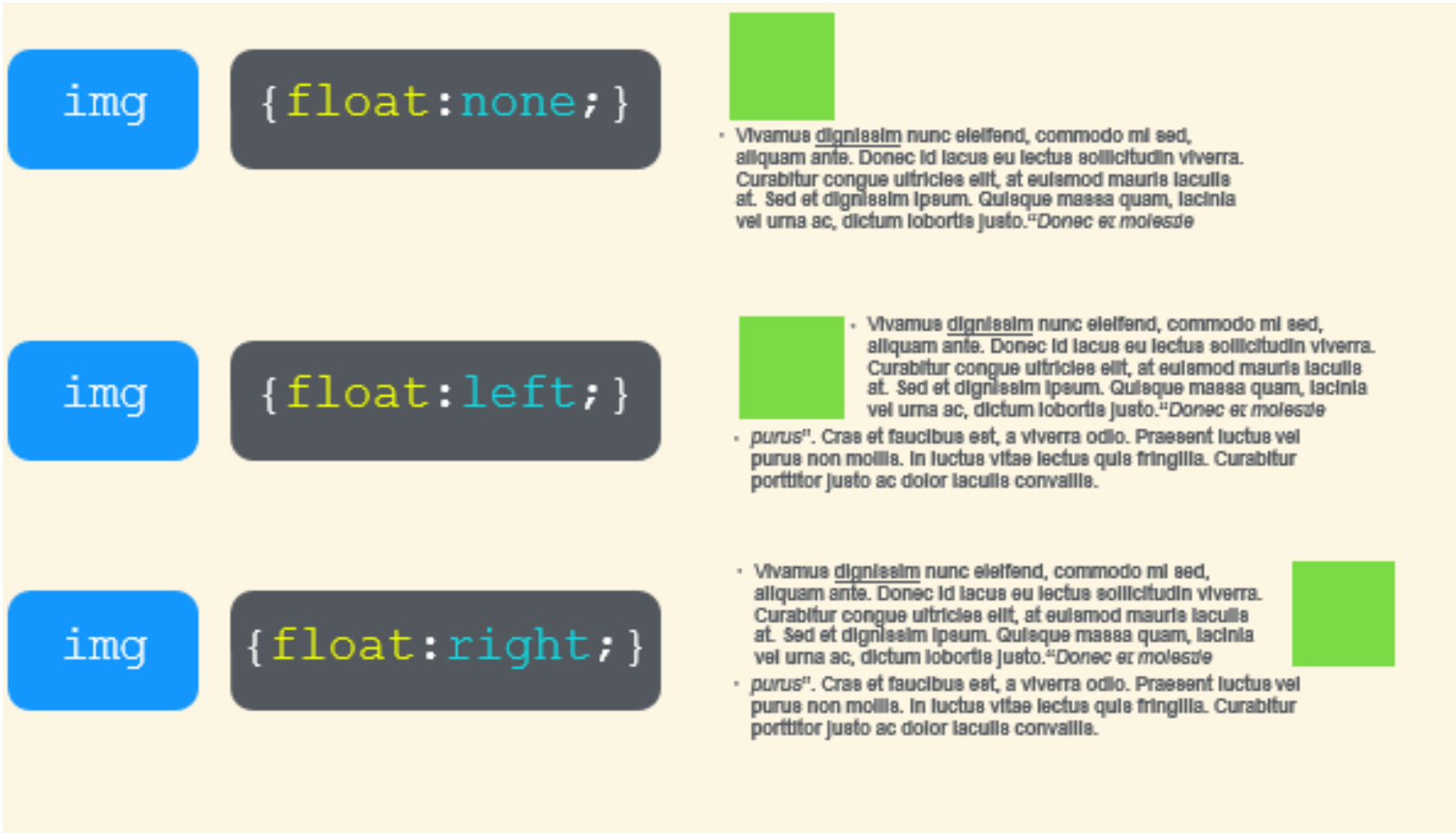
- **position:static** is the default position of an element
- **position:fixed** tells an element to be positioned relative to the browser window. It will not move even if the window is scrolled
- **position:relative** tells an element to be positioned relative to its normal position.
- **position:absolute** tells an element where to be positioned relative to its parent element.

CSS Float

With CSS float, an element can be pushed to the left or right, allowing other elements to wrap around it.

- **clear** Specifies which sides of an element where other floating elements are not allowed.
- **float** Specifies whether or not a box should float to the left or right.

CSS Float


img
{float:none;}
Vivamus dignissim nunc eleifend, commodo mi sed, aliquam ante. Donec id lacus eu lectus sollicitudin viverra. Curabitur congue ultricies elit, at euismod mauris iaculis at. Sed et dignissim ipsum. Quisque massa quam, lacinia vel urna ac, dictum lobortis justo.“Donec et molestie
img
{float:left;}
Vivamus dignissim nunc eleifend, commodo mi sed, aliquam ante. Donec id lacus eu lectus sollicitudin viverra. Curabitur congue ultricies elit, at euismod mauris iaculis at. Sed et dignissim ipsum. Quisque massa quam, lacinia vel urna ac, dictum lobortis justo.“Donec et molestie
purus”. Cras et faucibus est, a viverra odio. Praesent luctus vel purus non mollis. In luctus vitae lectus quis fringilla. Curabitur porttitor justo ac dolor iaculis convallis.
img
{float:right;}
Vivamus dignissim nunc eleifend, commodo mi sed, aliquam ante. Donec id lacus eu lectus sollicitudin viverra. Curabitur congue ultricies elit, at euismod mauris iaculis at. Sed et dignissim ipsum. Quisque massa quam, lacinia vel urna ac, dictum lobortis justo.“Donec et molestie
purus”. Cras et faucibus est, a viverra odio. Praesent luctus vel purus non mollis. In luctus vitae lectus quis fringilla. Curabitur porttitor justo ac dolor iaculis convallis.

CSS Display and Visibility

The display property specifies if/how an element is displayed, and the visibility property specifies if an element should be visible or hidden.

visibility:hidden hides an element, but it will still take up the same space as before.

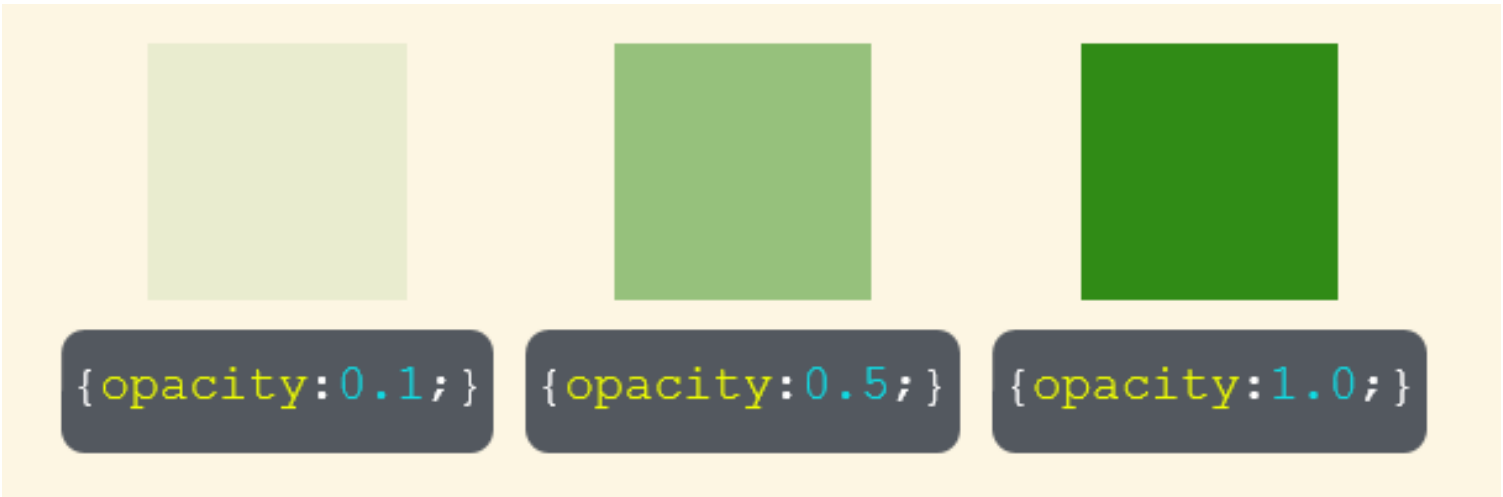
display:none hides an element, and it will not take up any space.

display:inline tells an element to only take up as much width as necessary, and will not force line breaks.

display:block tells an element will takes up the full width available, and will have a line break before and after it

CSS Image Opacity / Transparency

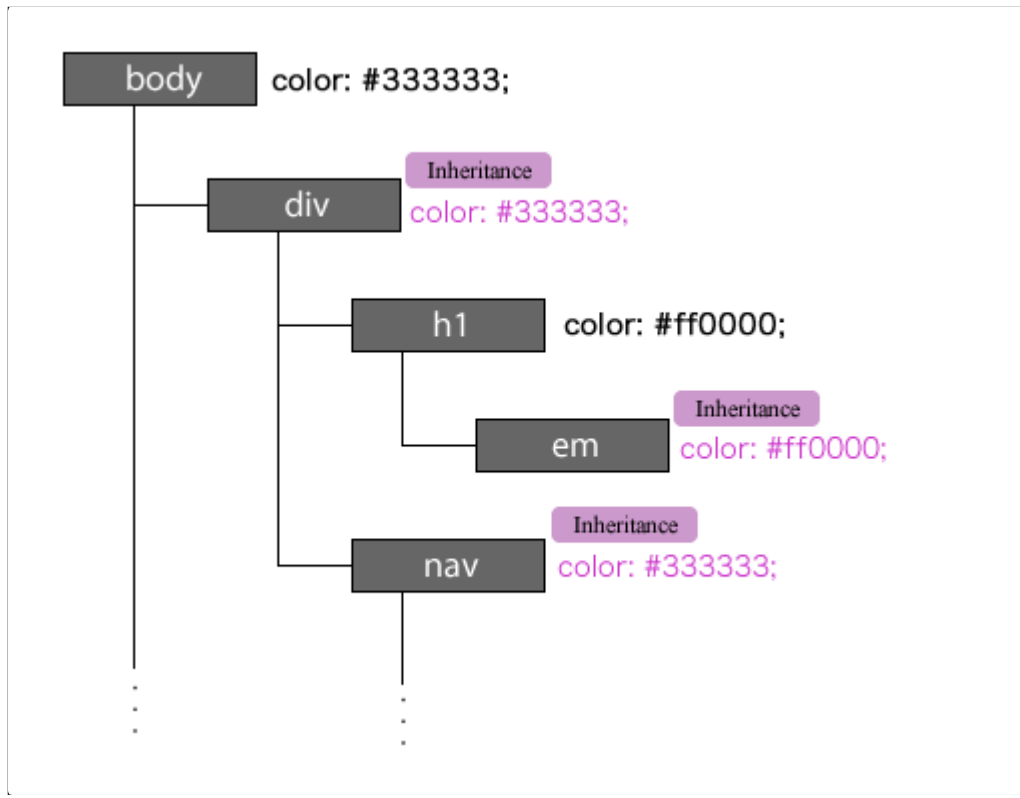
- The CSS3 property for transparency is **opacity**.
- The opacity property can take a value from 0.0 - 1.0. A **lower** value makes the element **more transparent**.



DEFAULTING

When the cascade does not result in a value, the specified value must be found some other way:

- Inherited properties draw their defaults from their parent element through inheritance.
- Other properties take their initial value



CSS Combinators

A CSS selector can contain more than one simple selector.
Between the simple selectors, we can include a combinator.
There are four different combinators in CSS:

- descendant selector (space)
- child selector (>)
- adjacent sibling selector (+)
- general sibling selector (~)

Explicit Defaulting

- Resetting a Property: the **initial** keyword.
- Explicit Inheritance: the **inherit** keyword.
- Erasing All Declarations: the **unset** keyword.

Style Sheet Origins

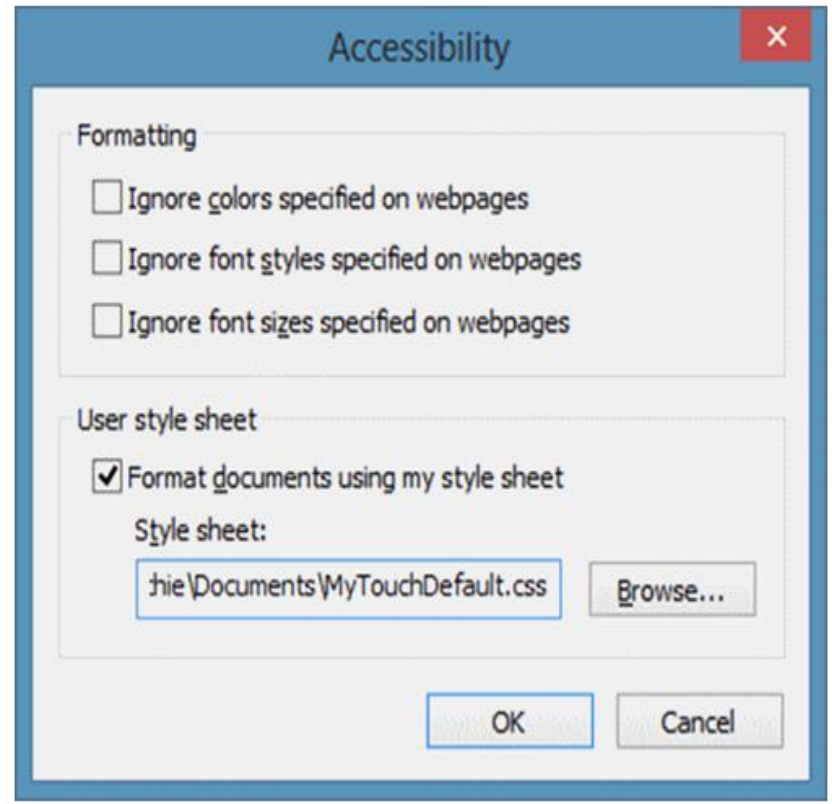
- Author
- User
- Agent

User Agent & User Stylesheet

User Agent - Browser apply style sheets to all web documents. These are referred to as a "default" browser style sheet.

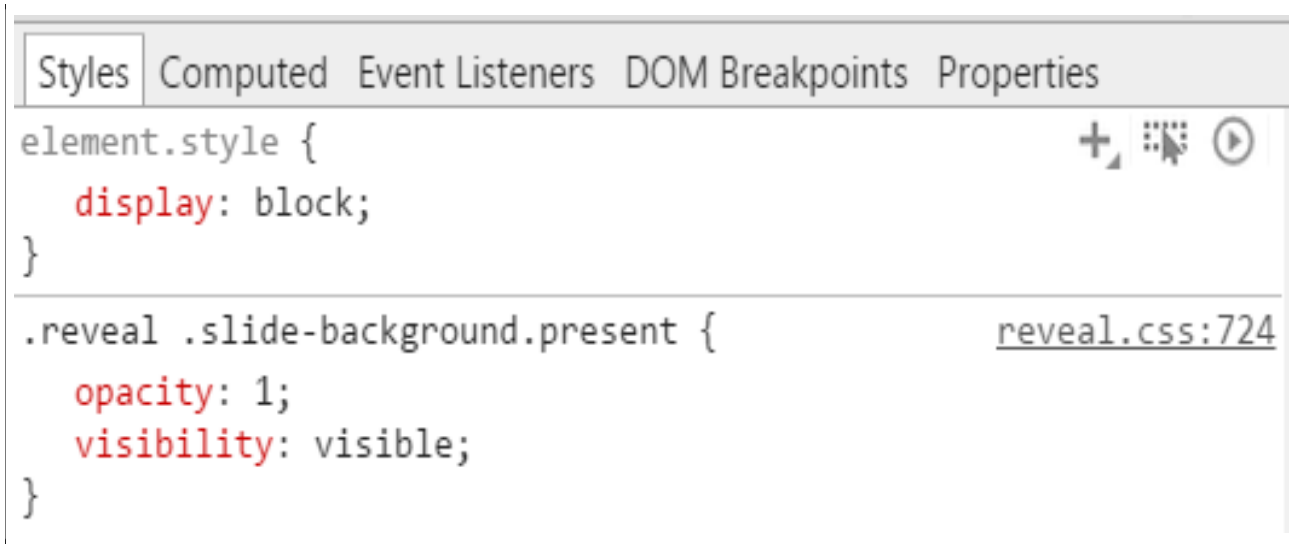
User - Most browsers allow user to apply their own style sheets within the browser.

```
input, textarea, keygen, select, button {      user agent stylesheet
   margin:▶0em;
   font:▶-webkit-small-control;
   color: initial;
   letter-spacing: normal;
   word-spacing: normal;
   text-transform: none;
   text-indent: 0px;
   text-shadow: none;
   display: inline-block;
   text-align: start;
}
```



Author Stylesheet

Web authors can apply one or more style sheets to an HTML document.



CSS Tables

Adding Borders to Tables

- The CSS border property is the best way to define the borders for the tables.
- The following example will set a black border for the <table>, <th>, and <td> elements.

Example

```
table, th, td
{
        border: 1px solid black;
}
```

CSS Tables

Striped Tables

For zebra-striped tables, use the **nth-child()** selector and add a **background-color** to all even (or odd) table rows:

For Example:

```
tr:nth-child(even) {background-color: #f2f2f2}
```

First Name	Last Name	Savings
Peter	Griffin	$100
Lois	Griffin	$150
Joe	Swanson	$300