**Start with CSS**

**What is Cascading Style Sheets (CSS)?**

Cascading Style Sheets (CSS) specifies the appearance and the formatting of an HTML document.

A CSS style is made up of one or more declarations separated by a semi-colon. Each declaration consists of a CSS property and a value separated by a colon. Property names in CSS are not case-sensitive.

**background**-**color**:grey;

**color**:white;

**Where can CSS Styles be defined?**

1. Inline Declarations

Inside the HTML

e.g.

<h2 **style="color: red; text-transform: uppercase;"**> xxxxxxxxx </h2>

1. Embedded style sheet

Between <style> and </style> tags inside the head element (this is called an)

e.g.

**<head>**

………………………………

**<style type="text/css">**

**h1, h2 {**

**color: green;**

**}**

**h3 {**

**color: blue;**

**}**

**</style>**

**…………………………………**

**</head>**

* Can be used inside the body tag also

<wicket:panel>

<html>

<head></head>

<body>

<style type = *"text/css"* >

**html,body** {overflow: *visible !important*; height: *auto !important*;}

</style>

<table wicket:id=*"datatable"* ></table>

</body>

</wicket:panel>

If you need a certain CSS property to take precedence over all other CSS rules setting the same CSS property for the same HTML elements, you can add the instruction **!important** after the CSS property when you declare it. The **!important** instruction has the highest precedence of all precedence factors.

1. External style sheet

In an external CSS file

e.g.

<head>

…………………………………

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

…………………………………

</head>

myStyle.css

**h1, h2 {**

**color: green;**

**}**

**h3 {**

**color: blue;**

**}**

1. Importing from another stylesheet

You can import styles from one stylesheet into another using the @import statement.

The style.css File:

@import "another.css";

**span** {

**border**: medium black dashed;

**padding**: 10px;

}

You can import more than one stylesheets, using one @import statement per stylesheet. The @import statements must appear at the top of the stylesheet.

The @import statement has the effect of importing the styles and then merging the style.

**Selectors and the Structure of CSS Rules**

Every CSS style rule consists of two parts:

1. Selector

which defines the type(s) of HTML element(s) to which the style rule applies;

1. Series of declarations

which consisting of properties and values, that define the style.

e.g.

h1, h2, h3 {

font-size: 120%;

text-transform: capitalize;

}

**Types of selectors**

1. **Universal Selector**

\* {

color: red;

}

In this example, **all elements in the page** are given a text color of red.

1. **Element Type Selector**

most common selector

a {

color: white;

background-color: green;

}

In this example, we specify the text and background color of **all hyperlinks in the current document**.

1. **Class Selector**

The following style sheet entry **defines a class** named special (must **precede the class name with a period**).

.special {

font-family: Verdana, Helvetica, Arial, sans-serif;

}

**Add class = "special" to the elements** that we want to adopt this style

<h1 class="special">A Special Heading</h1>

<p class="special">This is a special paragraph.</p>

- Write **the class to apply only to a particular type of element**

p.special {

font-family: Verdana, Helvetica, Arial, sans-serif;

}

The association of the class (in this case, special) with an element of any other type, the style rule simply does not apply to that element.

* **An HTML element can belong to multiple classes**:

Simply list those classes (separated by spaces) in the class attribute:

<p **class="special exciting"**>Paragraph! Of! Stuff!</p>

### Classes are NOT unique

* You can use the same class on multiple elements.
* You can use multiple classes on the same element.

1. **ID Selector**
2. ID's are unique
3. Each element can have only one ID
4. Each page can have only one element with that ID
5. The ID selector selects elements by the global id attribute
6. Syntax:

#idvalue

Or

elementType.#idvalue

<!**DOCTYPE** **HTML**>

<**html**>

<**head**>

<**title**>Example</title>

<**style** type="text/css">

#myAnchor {

**border**: thin black solid;

**padding**: 4px;

}

</style>

</head>

<**body**>

<**a** id="myAnchor" href="http://java2s.com"> Visit the java2s.com </a>

</body>

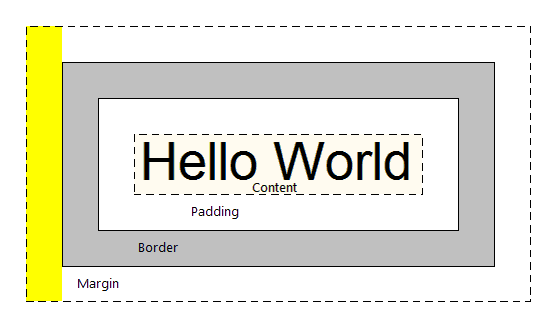
</html>

1. **D**
2. **Z**

In CSS, selectors are patterns used to select the element(s) you want to style.

The "CSS" column indicates in which CSS version the property is defined (CSS1, CSS2, or CSS3).

|  |  |  |  |
| --- | --- | --- | --- |
| **Selector** | **Example** | **Example description** | **CSS** |
| [.*class*](http://www.w3schools.com/cssref/sel_class.asp) | .intro | Selects all elements with class="intro" | 1 |
| [#*id*](http://www.w3schools.com/cssref/sel_id.asp) | #firstname | Selects the element with id="firstname" | 1 |
| [\*](http://www.w3schools.com/cssref/sel_all.asp) | \* | Selects all elements | 2 |
| [*element*](http://www.w3schools.com/cssref/sel_element.asp) | P | Selects all <p> elements | 1 |
| [*element,element*](http://www.w3schools.com/cssref/sel_element_comma.asp) | div,p | Selects all <div> elements and all <p> elements | 1 |
| [*element* *element*](http://www.w3schools.com/cssref/sel_element_element.asp) | div p | Selects all <p> elements inside <div> elements | 1 |
| [*element*>*element*](http://www.w3schools.com/cssref/sel_element_gt.asp) | div>p | Selects all <p> elements where the parent is a <div> element | 2 |
| [*element*+*element*](http://www.w3schools.com/cssref/sel_element_pluss.asp) | div+p | Selects all <p> elements that are placed immediately after <div> elements | 2 |
| [[*attribute*]](http://www.w3schools.com/cssref/sel_attribute.asp) | [target] | Selects all elements with a target attribute | 2 |
| [[*attribute*=*value*]](http://www.w3schools.com/cssref/sel_attribute_value.asp) | [target=\_blank] | Selects all elements with target="\_blank" | 2 |
| [[*attribute*~=*value*]](http://www.w3schools.com/cssref/sel_attribute_value_contains.asp) | [title~=flower] | Selects all elements with a title attribute containing the word "flower" | 2 |
| [[*attribute*|=*value*]](http://www.w3schools.com/cssref/sel_attribute_value_lang.asp) | [lang|=en] | Selects all elements with a lang attribute value starting with "en" | 2 |
| [:link](http://www.w3schools.com/cssref/sel_link.asp) | a:link | Selects all unvisited links | 1 |
| [:visited](http://www.w3schools.com/cssref/sel_visited.asp) | a:visited | Selects all visited links | 1 |
| [:active](http://www.w3schools.com/cssref/sel_active.asp) | a:active | Selects the active link | 1 |
| [:hover](http://www.w3schools.com/cssref/sel_hover.asp) | a:hover | Selects links on mouse over | 1 |
| [:focus](http://www.w3schools.com/cssref/sel_focus.asp) | input:focus | Selects the input element which has focus | 2 |
| [:first-letter](http://www.w3schools.com/cssref/sel_firstletter.asp) | p:first-letter | Selects the first letter of every <p> element | 1 |
| [:first-line](http://www.w3schools.com/cssref/sel_firstline.asp) | p:first-line | Selects the first line of every <p> element | 1 |
| [:first-child](http://www.w3schools.com/cssref/sel_firstchild.asp) | p:first-child | Selects every <p> element that is the first child of its parent | 2 |
| [:before](http://www.w3schools.com/cssref/sel_before.asp) | p:before | Insert content before  the content of every <p> element | 2 |
| [:after](http://www.w3schools.com/cssref/sel_after.asp) | p:after | Insert content after every <p> element | 2 |
| [:lang(*language*)](http://www.w3schools.com/cssref/sel_lang.asp) | p:lang(it) | Selects every <p> element with a lang attribute equal to "it" (Italian) | 2 |
| [*element1*~*element2*](http://www.w3schools.com/cssref/sel_gen_sibling.asp) | p~ul | Selects every <ul> element that are preceded by a <p> element | 3 |
| [[*attribute*^=*value*]](http://www.w3schools.com/cssref/sel_attr_begin.asp) | a[src^="https"] | Selects every <a> element whose src attribute value begins with "https" | 3 |
| [[*attribute*$=*value*]](http://www.w3schools.com/cssref/sel_attr_end.asp) | a[src$=".pdf"] | Selects every <a> element whose src attribute value ends with ".pdf" | 3 |
| [[*attribute*\*=*value*]](http://www.w3schools.com/cssref/sel_attr_contain.asp) | a[src\*="w3schools"] | Selects every <a> element whose src attribute value contains the substring "w3schools" | 3 |
| [:first-of-type](http://www.w3schools.com/cssref/sel_first-of-type.asp) | p:first-of-type | Selects every <p> element that is the first <p> element of its parent | 3 |
| [:last-of-type](http://www.w3schools.com/cssref/sel_last-of-type.asp) | p:last-of-type | Selects every <p> element that is the last <p> element of its parent | 3 |
| [:only-of-type](http://www.w3schools.com/cssref/sel_only-of-type.asp) | p:only-of-type | Selects every <p> element that is the only <p> element of its parent | 3 |
| [:only-child](http://www.w3schools.com/cssref/sel_only-child.asp) | p:only-child | Selects every <p> element that is the only child of its parent | 3 |
| [:nth-child(*n*)](http://www.w3schools.com/cssref/sel_nth-child.asp) | p:nth-child(2) | Selects every <p> element that is the second child of its parent | 3 |
| [:nth-last-child(*n*)](http://www.w3schools.com/cssref/sel_nth-last-child.asp) | p:nth-last-child(2) | Selects every <p> element that is the second child of its parent, counting from the last child | 3 |
| [:nth-of-type(*n*)](http://www.w3schools.com/cssref/sel_nth-of-type.asp) | p:nth-of-type(2) | Selects every <p> element that is the second <p> element of its parent | 3 |
| [:nth-last-of-type(*n*)](http://www.w3schools.com/cssref/sel_nth-last-of-type.asp) | p:nth-last-of-type(2) | Selects every <p> element that is the second <p> element of its parent, counting from the last child | 3 |
| [:last-child](http://www.w3schools.com/cssref/sel_last-child.asp) | p:last-child | Selects every <p> element that is the last child of its parent | 3 |
| [:root](http://www.w3schools.com/cssref/sel_root.asp) | :root | Selects the document's root element | 3 |
| [:empty](http://www.w3schools.com/cssref/sel_empty.asp) | p:empty | Selects every <p> element that has no children (including text nodes) | 3 |
| [:target](http://www.w3schools.com/cssref/sel_target.asp) | #news:target | Selects the current active #news element (clicked on a URL containing that anchor name) | 3 |
| [:enabled](http://www.w3schools.com/cssref/sel_enabled.asp) | input:enabled | Selects every enabled <input> element | 3 |
| [:disabled](http://www.w3schools.com/cssref/sel_disabled.asp) | input:disabled | Selects every disabled <input> element | 3 |
| [:checked](http://www.w3schools.com/cssref/sel_checked.asp) | input:checked | Selects every checked <input> element | 3 |
| [:not(*selector*)](http://www.w3schools.com/cssref/sel_not.asp) | :not(p) | Selects every element that is not a <p> element | 3 |
| [::selection](http://www.w3schools.com/cssref/sel_selection.asp) | ::selection | Selects the portion of an element that is selected by a user |  |



**Understanding the Box Model**

If an element is visible, it will occupy a rectangular region of the page. This is known as the element's box. There are four parts to this box.

Two of the parts can be visible: The content and the border.

* Padding is the amount of space between the content and the border.
* The margin is the space between the border and the other elements.

**CSS Padding**

* The padding clears an area around the content (inside the border) of an element.
* The padding is affected by the background color of the element.
* The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once.
* Possible Values

|  |  |
| --- | --- |
| Value | Description |
| *Length* | Defines a fixed padding (in pixels, pt, em, etc.) |
| *%* | Defines a padding in % of the containing element |

p {  
    padding-top: 25px;  
    padding-right: 50px;  
    padding-bottom: 25px;  
    padding-left: 50px;  
}

* The padding property can have from one to four values.
* padding: 25px 50px 75px 100px; // **clockwise start from top.**
  + - top padding is 25px
    - right padding is 50px
    - bottom padding is 75px
    - left padding is 100px
* padding: 25px 50px 75px; // **clockwise start from top.**
  + - top padding is 25px
    - right and left paddings are 50px
    - bottom padding is 75px
* padding: 25px 50px; // **clockwise start from top.**
  + - top and bottom paddings are 25px
    - right and left paddings are 50px
* padding: 25px; // **clockwise start from top.**
  + - all four paddings are 25px

**CSS Margin**

The CSS margin properties define the space around elements.

The margin clears an area around an element (outside the border). The margin does not have a background color, and is completely transparent.

The top, right, bottom, and left margin can be changed independently using separate properties. A shorthand margin property can also be used, to change all margins at once.

## Possible Values

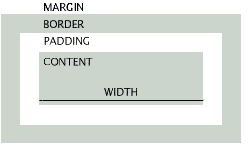
|  |  |
| --- | --- |
| **Value** | **Description** |
| auto | The browser calculates a margin |
| *length* | Specifies a margin in px, pt, cm, etc. Default value is 0px |
| *%* | Specifies a margin in percent of the width of the containing element |
| inherit | Specifies that the margin should be inherited from the parent element |

It is also possible to use negative values, to overlap content.

p {  
    margin-top: 100px;  
    margin-bottom: 100px;  
    margin-right: 150px;  
    margin-left: 50px;  
}

The margin property can have from one to four values.

* **margin: 25px 50px 75px 100px;** 
  + top margin is 25px
  + right margin is 50px
  + bottom margin is 75px
  + left margin is 100px
* **margin: 25px 50px 75px;**
  + top margin is 25px
  + right and left margins are 50px
  + bottom margin is 75px
* **margin: 25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px
* **margin: 25px;**
  + all four margins are 25px



An element can contain other elements. In this case, the parent element's context box is known as the child element's container.

**border-width sets the width of the border**

| Value | Description |
| --- | --- |
| <length> | Sets the border width to a length in CSS measurement units such as em, px, or cm. |
| <perc>% | Sets the border width to a percent of the width of the area around. |
| Thin | Sets the border width to thin, the actual value is defined by browser |
| medium | Sets the border width to medium, the actual value is defined by browser |
| thick | Sets the border width to thick, the actual value is defined by browser |

**border**-**width**: 5px;

**border**-**style**: solid;

**border**-**color**: black;

**Applying a Border to a Single Side**

| Property | Description Values |
| --- | --- |
| border-top-width | Defines the top border, check border-width for its value |
| border-top-style | Defines the top border, check border-style for its value |
| border-top-color | Defines the top border, check border-color for its value |
| border-bottom-width | Defines the bottom border, check border-width for its value |
| border-bottom-style | Defines the bottom border, check border-style for its value |
| border-bottom-color | Defines the bottom border, check border-color for its value |
| border-left-width | Defines the left border, check border-width for its value |
| border-left-style | Defines the left border, check border-style for its value |
| border-left-color | Defines the left border, check border-color for its value |
| border-right-width | Defines the right border, check border-width for its value |
| border-right-style | Defines the right border, check border-style for its value |
| border-right-color | Defines the right border, check border-color for its value |

**Setting Elements Width:**

CSS Syntax

width: auto|*value*|initial|inherit;

Property Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| auto | Default value. The browser calculates the width |
| *length* | Defines the width in px, cm, etc. |
| *%* | Defines the width in percent of the containing block |
| Initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |
| Inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |

**Setting Elements height:**

CSS Syntax

height: auto|*length*|initial|inherit;

Property Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| Auto | The browser calculates the height. This is default |
| *length* | Defines the height in px, cm, etc. |
| *%* | Defines the height in percent of the containing block |
| Initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |
| Inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |

320px (width)  
+ 20px (left + right padding)  
+ 10px (left + right border)  
+ 0px (left + right margin)  
= 350px

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

**Setting Element Backgrounds**

The background Properties

| Property | Description |
| --- | --- |
| [background-color](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Color_and_Image.htm) | Sets the background color. |
| [background-image](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Color_and_Image.htm) | Sets the background images. |
| [background-repeat](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Color_and_Image.htm) | Sets the repeat style for images. |
| [background-size](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Image_Size.htm) | Sets the size of a background image. |
| [background-position](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Image_Position.htm) | Positions the background image. |
| [background-attachment](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Attachment_for_the_Background.htm) | Sets the attachment style for images. |
| [background-clip](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Image_Origin.htm) | Specifies the clipping style for images. |
| [background-origin](http://www.java2s.com/Book/HTML-CSS/CSS/Setting_the_Background_Image_Origin.htm) | Positions the background image. |
| [background](http://www.java2s.com/Book/HTML-CSS/CSS/Using_the_background_Shorthand_Property.htm) | Shorthand element. |

**border**: medium solid black;

**background**-**color**: lightgray;

**background**-image: url(http://java2s.com/Book/**HTML**-CSSImages/star.png);

**background**-size: 40px 40px;

**background**-repeat: repeat-x;

**The background-size Values:**

| Value | Description |
| --- | --- |
| contain | Scales the image, preserving the aspect ratio, to the largest size that can fit. |
| Cover | Scales the image, preserving the aspect ratio, to the smallest size that can fit. |
| Auto | Display the image at full size(default). |

**background**-image: url(http://java2s.com/Book/**HTML**-CSSImages/star.png);

**background**-repeat: repeat-x;

**background**-size:contain;

**Setting the Background Image Position**

The background-position property sets where the background image should be located

The predefined values are:

| Value | Description |
| --- | --- |
| Top | Positions the image at the top edge. |
| Left | Positions the image at the left edge. |
| Right | Positions the image at the right edge. |
| Bottom | Positions the image at the bottom edge. |
| Center | Positions the image at the mid-point. |

The value could be:

* left top
* left center
* left bottom
* right top
* right center
* right bottom
* center top
* center center
* center bottom

The first value controls the vertical position and the second value controls the horizontal position.

**Setting the Attachment for the Background**

When you apply a background to an element that has a viewport(with scrollbar), you can specify how the background is attached to the content.

The background-attachment Values:

| Value | Description |
| --- | --- |
| fixed | fixed to the viewport, the background doesn't move during scrolling. |
| local | attached to the content, the background moves with the content when scrolled. |
| scroll | fixed to the element and does not scroll with the content. |

**Setting the Background Image Origin**

The background-origin specifies where the background color and image are applied.

The background-origin Values

| Value | Description |
| --- | --- |
| border-box | The background color and image are drawn within the border box. |
| padding-box | The background color and image are drawn within the padding box. |
| content-box | The background color and image are drawn within the content box. |

**Setting the Background Image Clipping Style**

The background-clip determines the region where the background color and image are drawn in the element's box.

The background-clip Values

| Value | Description |
| --- | --- |
| border-box | The background color and image are drawn within the border box. |
| padding-box | The background color and image are drawn within the padding box. |
| content-box | The background color and image are drawn within the content box. |

**Using the background Shorthand Property**

The background property sets all of the background values in a single declaration.

Here is the format referencing the individual properties:

**background**: <**background**-**color**> <**background**-**position**> <**background**-size> <**background**-repeat> <**background**-origin> <**background**-clip> <**background**-attachment> <**background**-image>

**e.g.**

**background**: lightgray **top** **right** no-repeat **border**-box content-box local url(background.png);

is equivalent to,

**background**-**color**: lightgray;

**background**-**position**: **top** **right**;

**background**-repeat: no-repeat;

**background**-origin: **border**-box;

**background**-**position**: content-box;

**background**-attachment: local;

**background**-image: url(background.png);

CSS Text

Text Color

The color property is used to set the color of the text.

The default color for a page is defined in the body selector.

body {color:blue;}  
h1 {color:#00ff00;}  
h2 {color:rgb(255,0,0);}

Text Alignment

The text-align property is used to set the horizontal alignment of a text. (Text can be centered, or aligned to the left or right, or justified.)

The vertical-align property is used to set the vertical alignment of a text. (Text can be middle, or aligned to the top or bottom.)

## All CSS List Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [list-style](http://www.w3schools.com/cssref/pr_list-style.asp) | Sets all the properties for a list in one declaration |
| [list-style-image](http://www.w3schools.com/cssref/pr_list-style-image.asp) | Specifies an image as the list-item marker |
| [list-style-position](http://www.w3schools.com/cssref/pr_list-style-position.asp) | Specifies if the list-item markers should appear inside or outside the content flow |
| [list-style-type](http://www.w3schools.com/cssref/pr_list-style-type.asp) | Specifies the type of list-item marker |

CSS Tables

Table Borders

To specify table borders in CSS, use the border property.

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

Notice that the table in the example above has double borders. This is because both the table and the th/td elements have separate borders.

To display a single border for the table, use the border-collapse property.

Collapse Borders

The border-collapse property sets whether the table borders are collapsed into a single border or separated:

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

Table Width and Height

Width and height of a table is defined by the width and height properties.

table   
{  
width:100%;  
}  
th  
{  
height:50px;  
}

Table Text Alignment

The text in a table is aligned with the text-align and vertical-align properties.

The text-align property sets the horizontal alignment, like left, right, or center:

The vertical-align property sets the vertical alignment, like top, bottom, or middle:

td  
{

height:50px;  
vertical-align:bottom;  
text-align:right;  
}

## Table Padding

To control the space between the border and content in a table, use the padding property on td and th elements:

td  
{  
padding:15px;  
}

## Table Color

The example below specifies the color of the borders, and the text and background color of th elements:

table, td, th  
{  
border:1px solid green;  
}  
th  
{  
background-color:green;  
color:white;  
}

<!DOCTYPE html>

<html>

<head>

<style>

table{

border-collapse:collapse;

width:100%

}

table, td, th

{

border:5px solid green;

}

th

{

height:50px;

background-color:green;

color:white;

text-align:left;

vertical-align:middle;

}

tr{

height:50px;

background-color:gray;

color:blue;

text-align:center;

vertical-align:bottom;

}

</style>

</head>

<body>

<table>

<tr>

<th>Firstname</th>

<th>Lastname</th>

<th>Savings</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

<td>$100</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

<td>$150</td>

</tr>

<tr>

<td>Joe</td>

<td>Swanson</td>

<td>$300</td>

</tr>

<tr>

<td>Cleveland</td>

<td>Brown</td>

<td>$250</td>

</tr>

</table>

</body>

</html>

CSS :hover Selector

Select and style a link when you mouse over it:

a:hover  
{

color:red;  
background-color:yellow;  
}

The :hover selector is used to select elements when you mouse over them.

**Tip:** The :hover selector can be used on all elements, not only on links.

**Tip:** Use the [:link](http://www.w3schools.com/cssref/sel_link.asp) selector to style links to unvisited pages, the [:visited](http://www.w3schools.com/cssref/sel_visited.asp) selector to style links to visited pages, and the [:active](http://www.w3schools.com/cssref/sel_active.asp) selector to style the active link.

**Note:** :hover MUST come after :link and :visited (if they are present) in the CSS definition, in order to be effective!

**Question: What is the difference between padding and margins?**

**Answer:**

Padding is the space inside the border between the border and the actual image or cell contents. In the image, the padding is the yellow area around the contents. Note that padding goes completely around the contents: there is padding on the top, bottom, right and left sides.

Margins are the spaces outside the border, between the border and the other elements next to this object. In the image, the margin is the red area outside the entire object. Note that, like the padding, the margin goes completely around the contents: there are margins on the top, bottom, right, and left sides.

## Possible Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| auto | The browser calculates a margin |
| *length* | Specifies a margin in px, pt, cm, etc. Default value is 0px |
| *%* | Specifies a margin in percent of the width of the containing element |
| inherit | Specifies that the margin should be inherited from the parent element |

|  |  |
| --- | --- |
|  | **Note:** It is also possible to use negative values, to overlap content. |

## Margin - Individual sides

In CSS, it is possible to specify different margins for different sides:

## Example

margin-top:100px;  
margin-bottom:100px;  
margin-right:50px;  
margin-left:50px;

Margin - Shorthand property

To shorten the code, it is possible to specify all the margin properties in one property. This is called a shorthand property.

The shorthand property for all the margin properties is "margin":

## Example

margin:100px 50px;

The margin property can have from one to four values.

* **margin:25px 50px 75px 100px;** 
  + top margin is 25px
  + right margin is 50px
  + bottom margin is 75px
  + left margin is 100px
* **margin:25px 50px 75px;**
  + top margin is 25px
  + right and left margins are 50px
  + bottom margin is 75px
* **margin:25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px
* **margin:25px;**
  + all four margins are 25px

## All CSS Margin Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [margin](http://www.w3schools.com/cssref/pr_margin.asp) | A shorthand property for setting the margin properties in one declaration |
| [margin-bottom](http://www.w3schools.com/cssref/pr_margin-bottom.asp) | Sets the bottom margin of an element |
| [margin-left](http://www.w3schools.com/cssref/pr_margin-left.asp) | Sets the left margin of an element |
| [margin-right](http://www.w3schools.com/cssref/pr_margin-right.asp) | Sets the right margin of an element |
| [margin-top](http://www.w3schools.com/cssref/pr_margin-top.asp) | Sets the top margin of an element |

## Padding

The CSS padding properties define the space between the element border and the element content.

The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element.

The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once.

## Possible Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| *length* | Defines a fixed padding (in pixels, pt, em, etc.) |
| *%* | Defines a padding in % of the containing element |

## Padding - Individual sides

In CSS, it is possible to specify different padding for different sides:

## Example

padding-top:25px;  
padding-bottom:25px;  
padding-right:50px;  
padding-left:50px;

## Padding - Shorthand property

To shorten the code, it is possible to specify all the padding properties in one property. This is called a shorthand property.

The shorthand property for all the padding properties is "padding":

## Example

padding:25px 50px;

The padding property can have from one to four values.

* **padding:25px 50px 75px 100px;** 
  + top padding is 25px
  + right padding is 50px
  + bottom padding is 75px
  + left padding is 100px
* **padding:25px 50px 75px;**
  + top padding is 25px
  + right and left paddings are 50px
  + bottom padding is 75px
* **padding:25px 50px;**
  + top and bottom paddings are 25px
  + right and left paddings are 50px
* **padding:25px;**
  + all four paddings are 25px

## All CSS Padding Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [padding](http://www.w3schools.com/cssref/pr_padding.asp) | A shorthand property for setting all the padding properties in one declaration |
| [padding-bottom](http://www.w3schools.com/cssref/pr_padding-bottom.asp) | Sets the bottom padding of an element |
| [padding-left](http://www.w3schools.com/cssref/pr_padding-left.asp) | Sets the left padding of an element |
| [padding-right](http://www.w3schools.com/cssref/pr_padding-right.asp) | Sets the right padding of an element |
| [padding-top](http://www.w3schools.com/cssref/pr_padding-top.asp) | Sets the top padding of an element |

**Attributes of css**

**Color**

color: yellow; => for text’s color

background-color: black; => for background’s color

**Font**

font-family: 'Comic Sans MS', Arial, Helvetica, sans-serif;

font-weight: bold;

font-size: 90%;

line-height: 1.8em;

**Image**

img {

float: left;

}

**Padding**

padding-top: 10px;

padding-right: 20px;

padding-bottom: 10px;

padding-left: 5px;

padding: 10px 20px 10px 5px;

## CSS Comments

Comments are used to explain your code, and may help you when you edit the source code at a later date. Comments are ignored by browsers.

A CSS comment begins with "/\*", and ends with "\*/", like this:

/\*This is a comment\*/  
p  
{  
text-align:center;  
/\*This is another comment\*/  
color:black;  
font-family:arial;  
}

# CSS Float

Elements are floated horizontally, this means that an element can only be floated left or right, not up or down.

A floated element will move as far to the left or right as it can. Usually this means all the way to the left or right of the containing element.

The elements after the floating element will flow around it.

The elements before the floating element will not be affected.

If an image is floated to the right, a following text flows around it, to the left:

If you place several floating elements after each other, they will float next to each other if there is room.

## Turning off Float - Using Clear

Elements after the floating element will flow around it. To avoid this, use the clear property.

The clear property specifies which sides of an element other floating elements are not allowed.

Add a text line into the image gallery, using the clear property:

## Example

.text\_line {  
    clear: both;  
}

## All CSS Float Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Description** | | **Values** | |
| [clear](http://www.w3schools.com/cssref/pr_class_clear.asp) | | Specifies which sides of an element where other floating elements are not allowed | left right both none inherit | |
| [float](http://www.w3schools.com/cssref/pr_class_float.asp) | | Specifies whether or not a box should float | left right none inherit |

**CSS Position:**

|  |  |  |
| --- | --- | --- |
| [position](http://www.w3schools.com/cssref/pr_class_position.asp) | Specifies the type of positioning for an element | absolute fixed relative static (by default) inherit |

static

* HTML elements are positioned static by default.
* An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

div.static {  
    position: static;  
    border: 3px solid #8AC007;  
}

relative

* An element with position: relative; is positioned relative to its normal position.
* relative behaves the same as static unless you add some extra properties.
* Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

div.relative {  
    position: relative;  
    left: 30px;  
    border: 3px solid #8AC007;  
}

fixed

* An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.
* A fixed element does not leave a gap in the page where it would normally have been located.

div.fixed {  
    position: fixed;  
    bottom: 0;  
    right: 0;  
    width: 300px;  
    border: 3px solid #8AC007;  
}

absolute  
  
An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

* This is a very powerful type of positioning that allows you to literally place any page element exactly where you want it.
* You use the positioning attributes top, left bottom and right to set the location. Remember that these values will be relative to the next parent element with relative (or absolute) positioning.
* If there is no such parent, it will default all the way back up to the <html> element itself meaning it will be placed relatively to the page itself.

The trade-off, and most important thing to remember, about absolute positioning is that these elements are removed from the flow of elements on the page. An element with this type of positioning is not affected by other elements and it doesn't affect other elements. This is a serious thing to consider every time you use absolute positioning. It's overuse or improper use can limit the flexibility of your site.

div.relative {  
    position: relative;  
    width: 400px;  
    height: 200px;  
    border: 3px solid #8AC007;  
}   
  
div.absolute {  
    position: absolute;  
    top: 80px;  
    right: 0;  
    width: 200px;  
    height: 100px;  
    border: 3px solid #8AC007;  
}

Overlapping Elements

* When elements are positioned, they can overlap other elements.
* The z-i-ndex property specifies the stack order of an element (which element should be placed in front of, or behind, the others).
* An element can have a positive or negative stack order:
* An element with greater stack order is always in front of an element with a lower stack order.
* If two positioned elements overlap without a z-index specified, the element positioned last in the HTML code will be shown on top

img {  
    position: absolute;  
    left: 0px;  
    top: 0px;  
    z-index: -1;  
}

**All CSS Positioning Properties**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [bottom](http://www.w3schools.com/cssref/pr_pos_bottom.asp) | Sets the bottom margin edge for a positioned box | auto *length %* inherit |
| [clip](http://www.w3schools.com/cssref/pr_pos_clip.asp) | Clips an absolutely positioned element | *shape* auto inherit |
| [cursor](http://www.w3schools.com/cssref/pr_class_cursor.asp) | Specifies the type of cursor to be displayed | *url* auto crosshair default pointer move e-resize ne-resize nw-resize n-resize se-resize sw-resize s-resize w-resize text wait help |
| [left](http://www.w3schools.com/cssref/pr_pos_left.asp) | Sets the left margin edge for a positioned box | auto *length %* inherit |
| [overflow](http://www.w3schools.com/cssref/pr_pos_overflow.asp) | Specifies what happens if content overflows an element's box | auto hidden scroll visible inherit |
| [position](http://www.w3schools.com/cssref/pr_class_position.asp) | Specifies the type of positioning for an element | absolute fixed relative static inherit |
| [right](http://www.w3schools.com/cssref/pr_pos_right.asp) | Sets the right margin edge for a positioned box | auto *length %* inherit |
| [top](http://www.w3schools.com/cssref/pr_pos_top.asp) | Sets the top margin edge for a positioned box | auto *length %* inherit |
| [z-index](http://www.w3schools.com/cssref/pr_pos_z-index.asp) | Sets the stack order of an element | *number* auto inherit |

Font style:

font: *font-style* *font-variant* *font-weight* *font-size/line-height font-family*|caption|icon|menu|message-box|small-caption|status-bar|initial|inherit;

## Property Values

|  |  |
| --- | --- |
| **Property/Value** | **Description** |
| *font-style* | Specifies the font style. Default value is "normal". See [font-style](http://www.w3schools.com/cssref/pr_font_font-style.asp) for possible values |
| *font-variant* | Specifies the font variant. Default value is "normal". See [font-variant](http://www.w3schools.com/cssref/pr_font_font-variant.asp) for possible values |
| *font-weight* | Specifies the font weight. Default value is "normal". See [font-weight](http://www.w3schools.com/cssref/pr_font_weight.asp) for possible values |
| *font-size/line-height* | Specifies the font size and the line-height. Default value is "normal". See [font-size](http://www.w3schools.com/cssref/pr_font_font-size.asp) and [line-height](http://www.w3schools.com/cssref/pr_dim_line-height.asp) for possible values |
| *font-family* | Specifies the font family. Default value depends on the browser. See [font-family](http://www.w3schools.com/cssref/pr_font_font-family.asp) for possible values |
| caption | Uses the font that are used by captioned controls (like buttons, drop-downs, etc.) |
| icon | Uses the font that are used by icon labels |
| menu | Uses the fonts that are used by dropdown menus |
| message-box | Uses the fonts that are used by dialog boxes |
| small-caption | A smaller version of the caption font |
| status-bar | Uses the fonts that are used by the status bar |
| initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |
| inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |

font-style: normal|italic|oblique|initial|inherit;

font-variant: normal|small-caps|initial|inherit;

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** |  |
| normal | The browser displays a normal font. This is default |  |
| small-caps | The browser displays a small-caps font |  |
| initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |  |
| inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |  |

font-weight: normal|bold|bolder|lighter|*number*|initial|inherit;

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** | **Play it** |
| normal | Defines normal characters. This is default | [Play it »](http://www.w3schools.com/cssref/playit.asp?filename=playcss_font-weight) |
| bold | Defines thick characters | [Play it »](http://www.w3schools.com/cssref/playit.asp?filename=playcss_font-weight&preval=bold) |
| bolder | Defines thicker characters |  |
| lighter | Defines lighter characters |  |
| 100 200 300 400 500 600 700 800 900 | Defines from thin to thick characters. 400 is the same as normal, and 700 is the same as bold |  |
| initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |  |
| inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |  |

font-size:medium|xx-small|x-small|small|large|x-large|xx-large|smaller|larger|*length*|initial|inherit;

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** |  |
| medium | Sets the font-size to a medium size. This is default |  |
| xx-small | Sets the font-size to an xx-small size |  |
| x-small | Sets the font-size to an extra small size |  |
| small | Sets the font-size to a small size |  |
| large | Sets the font-size to a large size |  |
| x-large | Sets the font-size to an extra large size |  |
| xx-large | Sets the font-size to an xx-large size |  |
| smaller | Sets the font-size to a smaller size than the parent element |  |
| larger | Sets the font-size to a larger size than the parent element |  |
| *length* | Sets the font-size to a fixed size in px, cm, etc. |  |
| *%* | Sets the font-size to a percent of  the parent element's font size |  |
| initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |  |
| inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |  |

font-family: *font*|initial|inherit;

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** |  |
| *family-name generic-family* | A prioritized list of font family names and/or generic family names |  |
| initial | Sets this property to its default value. [Read about initial](http://www.w3schools.com/cssref/css_initial.asp) |  |
| inherit | Inherits this property from its parent element. [Read about inherit](http://www.w3schools.com/cssref/css_inherit.asp) |  |

References:

<http://www.java2s.com/Book/HTML-CSS/CSS/What_is_Cascading_Style_Sheets_CSS.htm>

 <http://www.codecademy.com/tracks/web>

 <http://learnlayout.com/>

 <https://developer.mozilla.org/en-US/learn/css>

<http://html.net/tutorials/css/lesson14.php>

http://www.iraqtimeline.com/maxdesign/basicdesign/principles/prinposition.html

**Real time example of .css file**

**img***.fullwidth*{width:*100%*;height:*261px*;}

**img***.logo*{width:*179px*;}

*.none*{display:*none*;}

*.block*{display:*block*;}

*.left*{float:*left*;}

*.right*{float:*right*;}

*.center*{margin:*0* *auto*;text-align:*center*;}

*.textleft*{text-align:*left*;}

*.textright*{text-align:*right*;}

*.textcenter*{text-align:*center*;}

*.caps*{text-transform:*uppercase*;}

*.relative*{position:*relative*;}

*.absolute*{position:*absolute*;}

*.fixed*{position:*fixed!important*;}

*.cpointer*{cursor:*pointer*;}

*.cunvailable*{cursor:*not-allowed*;}

*.indent0*{text-indent:*0px*;}

*.ohidden*{overflow:*hidden*;}

*.posright-0*{right:*auto!important*}

*.line1*{background:*#e7e7e7*;height:*1px*;margin:*16px* *0* *6px* *0*;padding:*0*;display:*block*;}

*.line2*{background:*#e8e8e8*;height:*1px*;padding:*0*;display:*block*;}

*.line3*{background:*#e8e8e8*;height:*1px*;margin:*0px* *0* *0px* *0*;padding:*0*;display:*block*;}

*.line4*{background:*#e8e8e8*;height:*1px*;margin:*17px* *0* *15px* *0*;padding:*0*;display:*block*;}

*.line5*{background:*#e8e8e8*;height:*1px*;margin:*4px* *0* *12px* *0*;padding:*0*;display:*block*;}

*.xslim*{font-weight:*100*;}

*.slim*{font-weight:*300*;}

*.normal*{font-weight:*400*;}

*.bold*{font-weight:*700*;}

*.lh1*{line-height:*7px* *!important*;}

*.lh2*{line-height:*15px*;}

*.lh3*{line-height:*60px*;}

*.lh4*{line-height:*17px*;}

*.lh5*{line-height:*36px*;}

*.lato*{font-family:*"Helvetica Neue",* *Helvetica,* *Arial,* *sans-serif*;}

*.opensans*{font-family:*"Helvetica Neue",* *Helvetica,* *Arial,* *sans-serif*;}

*.size8*{font-size:*8px*;}

*.size10*{font-size:*10px*;}

*.size11*{font-size:*11px*;}

*.size12*{font-size:*12px*;}

*.size13*{font-size:*13px*;}

*.size14*{font-size:*14px*;}

*.size16*{font-size:*16px*;}

*.size18*{font-size:*18px*;}

*.size20*{font-size:*20px*;}

*.size22*{font-size:*22px*;}

*.size24*{font-size:*24px*;}

*.size28*{font-size:*28px*;}

*.size26*{font-size:*26px*;}

*.size30*{font-size:*30px*;}

*.size40*{font-size:*40px*;}

*.size48*{font-size:*48px*;}

*.size60*{font-size:*60px*;}

*.size65*{font-size:*65px*;}

*.size100*{font-size:*100px*;}

*.white*{color:*#fff*;}

*.dark*{color:*#333*;}

*.lgrey*{color:*#ccc*;}

*.lightgrey*{color:*#ccc*;}

*.grey*{color:*#999*;}

*.grey2*{color:*#666*;}

*.green*{color:*#72bf66*;}

*.green2*{color:*#339933*;}

*.lblue*{color:*#0099cc*;}

*.yellow*{color:*#ffff00*;}

*.red*{color:*#ff3300*;}

*.lred*{color:*#ff9966*;}

*.lred2*{color:*#e20613*;;}

**a***.dark*{color:*#333*;text-decoration:*none*;}

**a***:hover.dark*{color:*#0099cc*;text-decoration:*none*;}

**a***.grey*{color:*#999*;font-weight:*normal*;}

**a***.orange*{color:*#e30e1c*;}

**a***.whitelink*{color:*#fff*;}

**a***.greylink*{color:*#666*;}

**a***.clblue*{color:*#0099cc*;}

**a***:hover.clblue*{color:*#72bf66*;}

**a***.green2*{color:*#72bf66*;font-weight:*normal*;text-decoration:*none*;-webkit-transition:*.3s*;-moz-transition:*.3s*;transition:*.3s*;}

**a***:hover.green2*{color:*#339933*;font-weight:*normal*;text-decoration:*none*;}

*.shadow*{box-shadow:*0* *0* *10px* *rgba(0,* *0,* *0,* *0.1)*;}

*.fwi*{width:*100%*;eight:*261px*;overflow:*hidden*;position:*relative*;}

*.fwi* **img**{width:*100%*;margin-left:*0px*;}