CSS 3

Cascading Style Sheet

1.What is CSS?

CSS allows to create rules that specify how HTML elements should appear.

CSS is the language we use to style an HTML document.

CSS describes how HTML elements should be displayed.

1.1. Applications of CSS

CSS saves time - You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

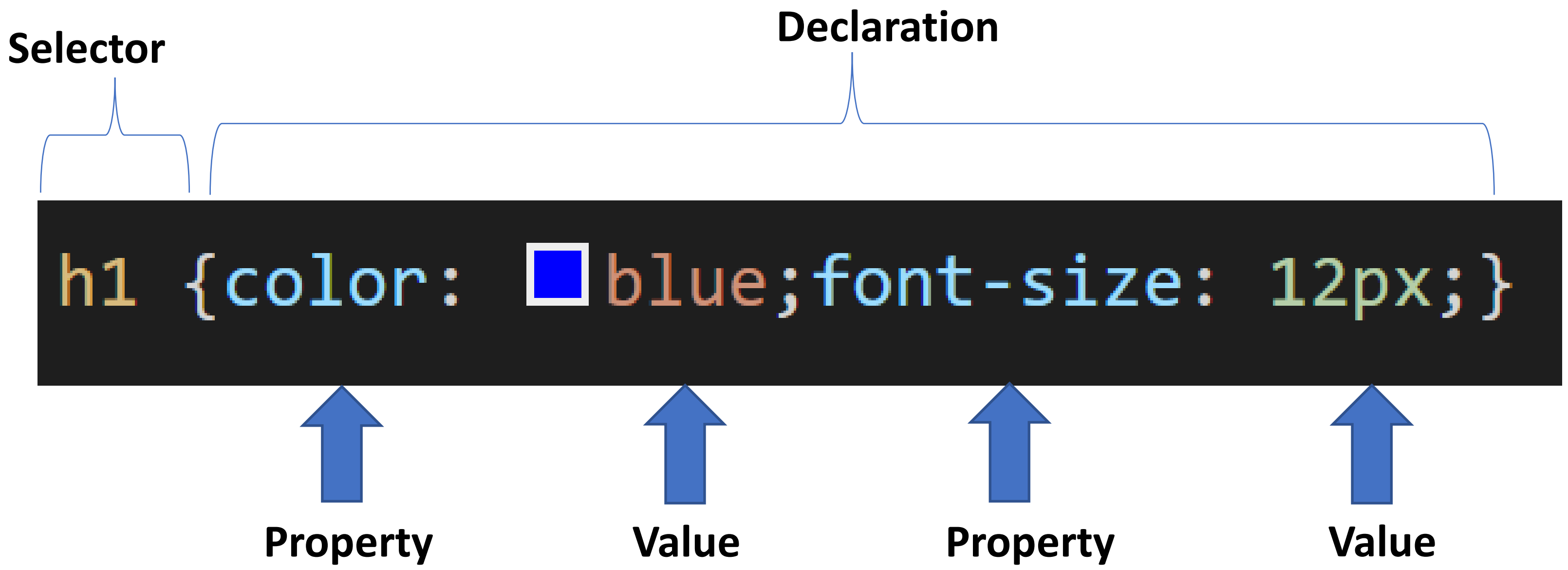
Pages load faster - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So, less code means faster download times.

Easy maintenance - To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

Superior styles to HTML - CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

Multiple Device Compatibility - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as laptops/mobile/tablets.

1.2. CSS Syntax



Selector

Declaration

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

Property

Value

Property

Value

- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a CSS property name and a value, separated by a colon.
- Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

1.3. Three Styling methods

1. Inline Styling method
   - 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.

2. Internal Styling method
   - An internal style sheet may be used if one single HTML page has a unique style.
   - The internal style is defined inside the <style> element, inside the head section.

3. External Styling method
   - An external style sheet may be used if you want to use common styling across multiple web pages
   - Having an external style sheet, improves the code reusability and maintenance.
   - Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.
   - An external style must be saved with a .css extension.

1.3.1 Cascading Order Styling methods

Cascading Order:

What style will be used when there is more than one style specified for an HTML element?

All the styles in a page will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

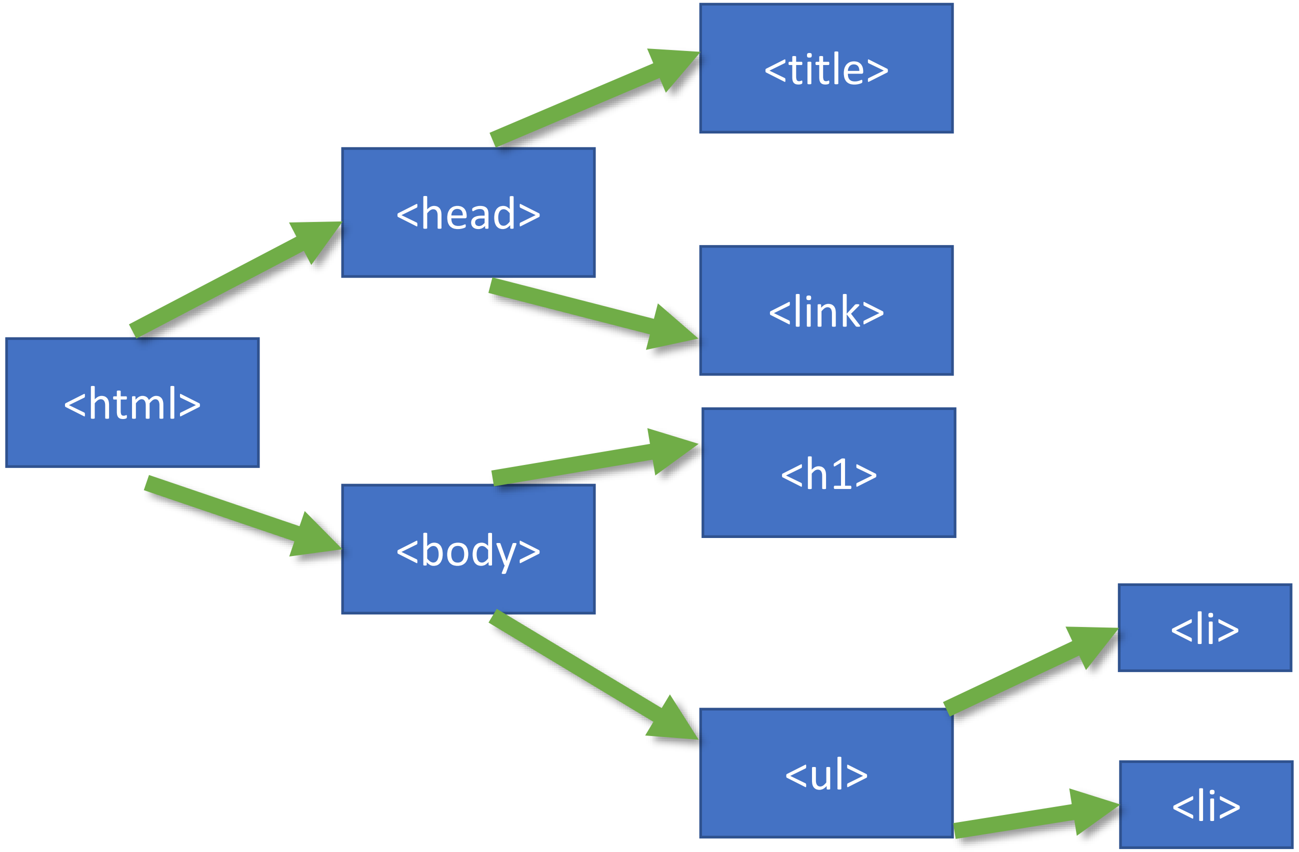
So, an inline style has the highest priority, and will override external and internal styles and browser defaults.

<u>NOTE:</u>

A CSS comment is placed inside the <style> element, and starts with /* and ends with */

1.4. HTML Element Tree

HTML Element Tree



1.4.1 HTML Element Tree

Each HTML document can be referred to as a document tree. We describe the elements in the tree like we would describe a family tree.

There are ancestors, descendants, parents, children and siblings.

It is important to understand the document tree because CSS selectors use the document tree.

Ancestor

An ancestor refers to any element that is connected but further up the document tree - no matter how many levels higher.

Descendant

A descendant refers to any element that is connected but lower down the document tree - no matter how many levels lower.

Parent and Child

A parent is an element that is directly above and connected to an element in the document tree.

A child is an element that is directly below and connected to an element in the document tree.

Sibling

A sibling is an element that shares the same parent with another element.

1.4.1 HTML Element Tree

Each HTML document can be referred to as a document tree. We describe the elements in the tree like we would describe a family tree.

There are ancestors, descendants, parents, children and siblings.

It is important to understand the document tree because CSS selectors use the document tree.

Ancestor

An ancestor refers to any element that is connected but further up the document tree - no matter how many levels higher.

Descendant

A descendant refers to any element that is connected but lower down the document tree - no matter how many levels lower.

Parent and Child

A parent is an element that is directly above and connected to an element in the document tree.

A child is an element that is directly below and connected to an element in the document tree.

Sibling

A sibling is an element that shares the same parent with another element.

1.5 CSS Selectors

CSS selectors are used to select the HTML elements you want to style.

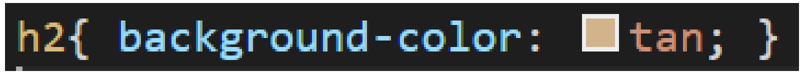
We can divide CSS selectors into five categories:

- Simple selectors (select elements based on element name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

1.5.1 Simple Selectors

The CSS element Selector

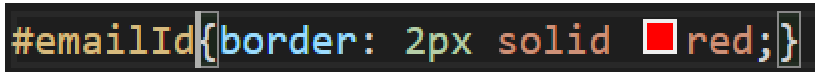
- The element selector selects HTML elements based on the element name.



```css
h2{ background-color: tan; }
```

The CSS id Selector

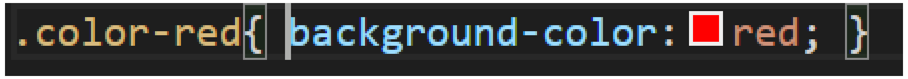
- The id selector uses the id attribute of an HTML element to select a specific element.
- The id of an element is unique within a page, so the id selector is used to select one unique element!
- To select an element with a specific id, write a hash (#) character, followed by the id of the element.



```css
#emailId{border: 2px solid red;}
```

The CSS class Selector

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



```css
.color-red{ background-color: red; }
```

1.5.2 Combinator Selectors

There are four different combinators in CSS:

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

Descendant Selector (space)
The descendant selector matches **all elements** that are descendants of a specified element.

Child Selector (>)
The child selector selects **all elements** that are the immediate children of a specified element.

1.5.2 Combinator Selectors

Adjacent sibling selector (+)

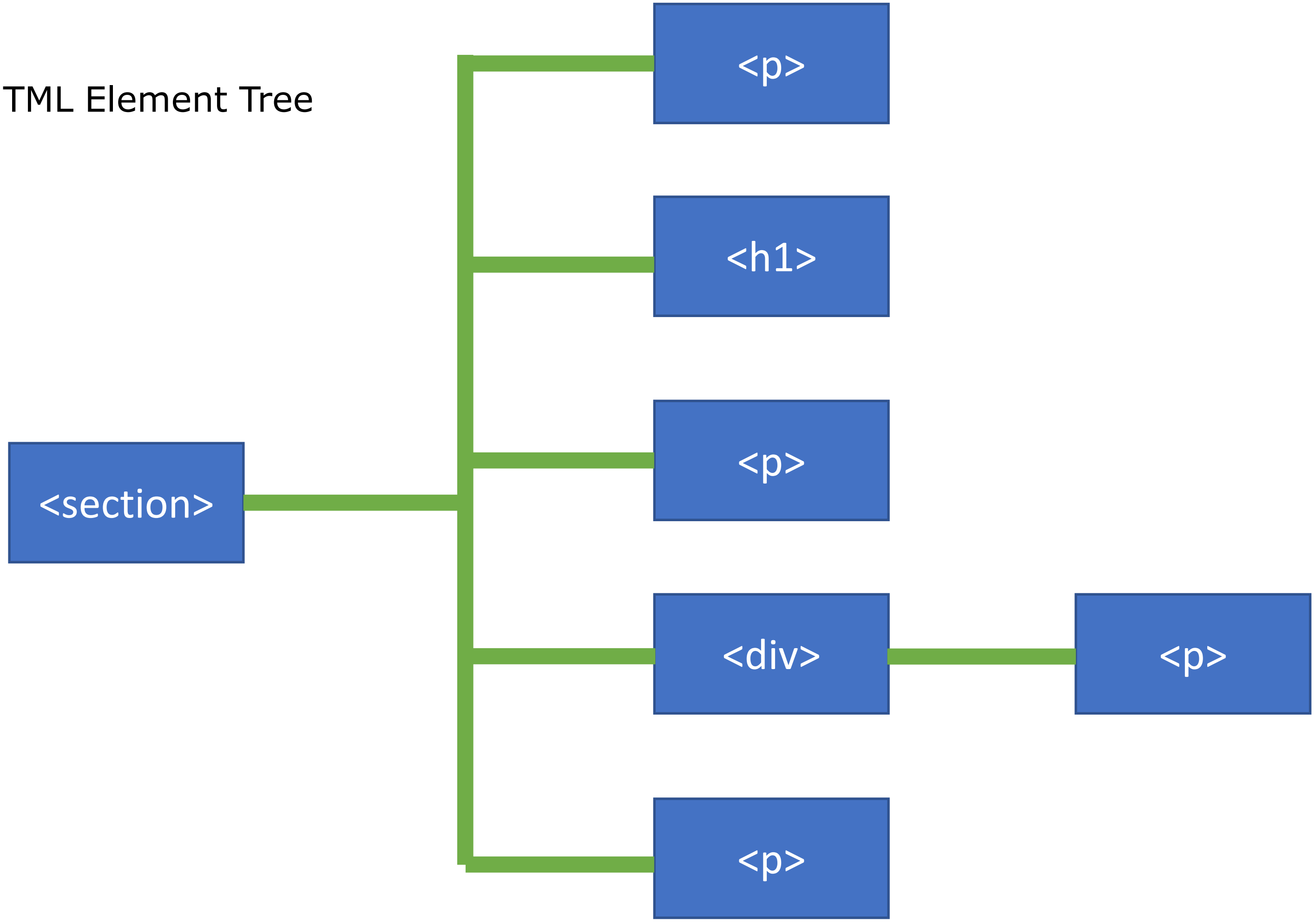
- Selects immediately **following** sibling of the specified element
- Sibling elements must have the same parent element, and "adjacent" means "immediately following".

General sibling selector (~)

- Selects all the sibling **after** the specified element

1.5.2 Element Tree

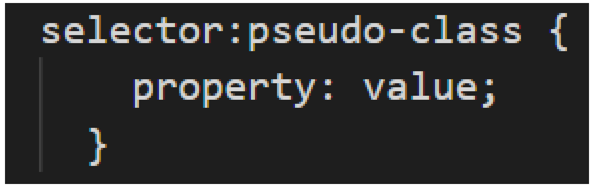
HTML Element Tree



1.5.3 Pseudo-class selectors

A pseudo-class is used to define a special state of an element.

Syntax:



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

1.5.3.1 Anchor pseudo elements

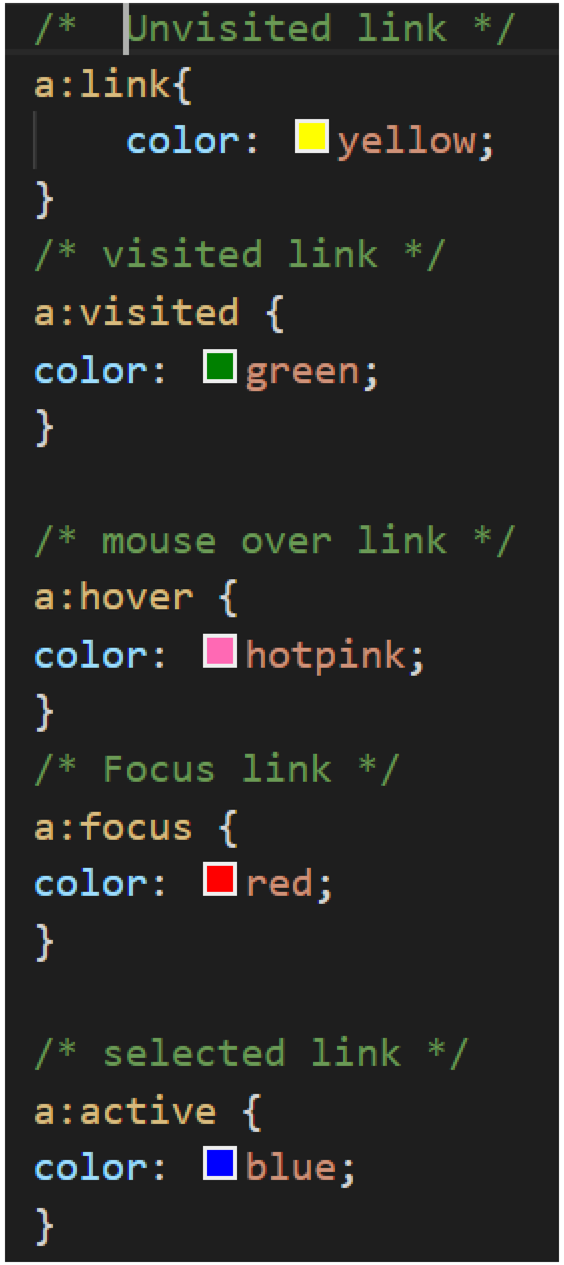
Link Selector:
The :link selector is used to select unvisited links.

Visited Selector:
The :visited selector to style links to visited pages.

Hover Selector:
The :hover selector to style links when you mouse over them.

Active Selector:
The :active selector to style links when you click on them.

Focus Selector:
The :focus selector to style links when you focus on them.

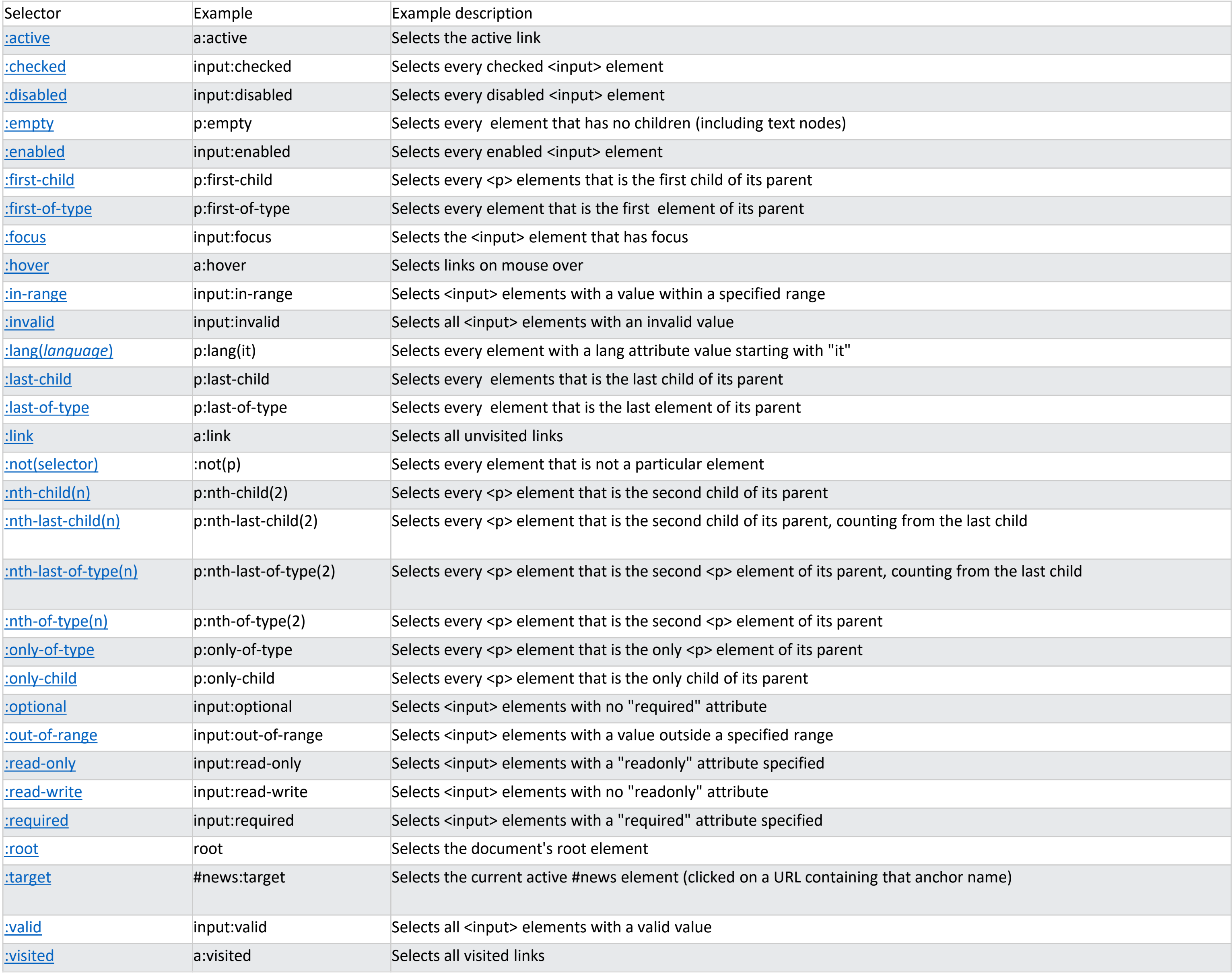


```css
/*  Unvisited link */
a:link{
    color: yellow;
}
/* visited link */
a:visited {
color: green;
}

/* mouse over link */
a:hover {
color: hotpink;
}
/* Focus link */
a:focus {
color: red;
}

/* selected link */
a:active {
color: blue;
}
```

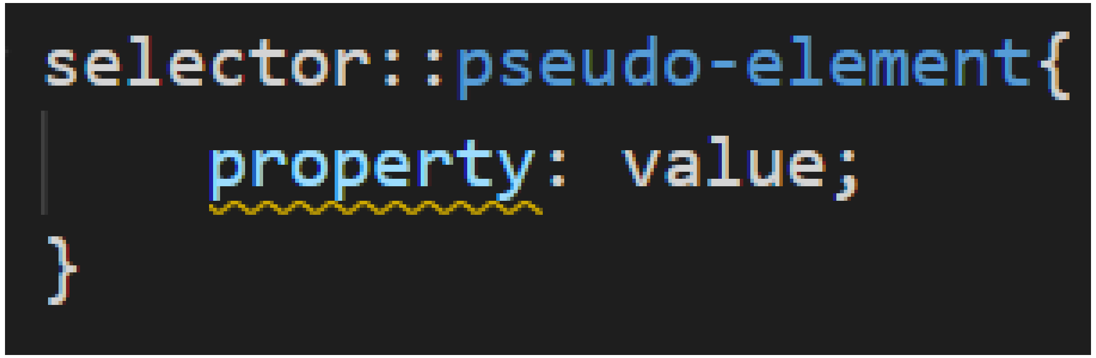
1.5.3.2 Other pseudo elements



Selector	Example	Example description
:active	a:active	Selects the active link
:checked	input:checked	Selects every checked <input> element
:disabled	input:disabled	Selects every disabled <input> element
:empty	p:empty	Selects every element that has no children (including text nodes)
:enabled	input:enabled	Selects every enabled <input> element
:first-child	p:first-child	Selects every <p> elements that is the first child of its parent
:first-of-type	p:first-of-type	Selects every element that is the first element of its parent
:focus	input:focus	Selects the <input> element that has focus
:hover	a:hover	Selects links on mouse over
:in-range	input:in-range	Selects <input> elements with a value within a specified range
:invalid	input:invalid	Selects all <input> elements with an invalid value
:lang(*language*)	p:lang(it)	Selects every element with a lang attribute value starting with "it"
:last-child	p:last-child	Selects every elements that is the last child of its parent
:last-of-type	p:last-of-type	Selects every element that is the last element of its parent
:link	a:link	Selects all unvisited links
:not(selector)	:not(p)	Selects every element that is not a particular element
:nth-child(n)	p:nth-child(2)	Selects every <p> element that is the second child of its parent
:nth-last-child(n)	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child
:nth-last-of-type(n)	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the last child
:nth-of-type(n)	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
:only-of-type	p:only-of-type	Selects every <p> element that is the only <p> element of its parent
:only-child	p:only-child	Selects every <p> element that is the only child of its parent
:optional	input:optional	Selects <input> elements with no "required" attribute
:out-of-range	input:out-of-range	Selects <input> elements with a value outside a specified range
:read-only	input:read-only	Selects <input> elements with a "readonly" attribute specified
:read-write	input:read-write	Selects <input> elements with no "readonly" attribute
:required	input:required	Selects <input> elements with a "required" attribute specified
:root	root	Selects the document's root element
:target	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)
:valid	input:valid	Selects all <input> elements with a valid value
:visited	a:visited	Selects all visited links

1.5.4 Pseudo-Element selector

- A CSS pseudo-element is used to style specified parts of an element.

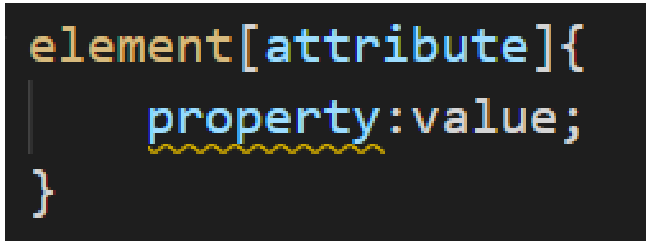


```
selector::pseudo-element{
    property: value;
}
```

Selector	Example	Example description
::after	p::after	Insert something after the content of each <p> element
::before	p::before	Insert something before the content of each <p> element
::first-letter	p::first-letter	Selects the first letter of each <p> element
::first-line	p::first-line	Selects the first line of each <p> element
::marker	::marker	Selects the markers of list items
::selection	p::selection	Selects the portion of an element that is selected by a user

1.5.5 Attribute Selectors

- The [attribute] selector is used to select elements with a specified attribute.



```
element[attribute]{
    property:value;
}
```

Selector	Example	Example description
[attribute]	[target]	Selects all elements with a target attribute
[attribute=value]	[target=_blank]	Selects all elements with target="_blank"
[attribute~=value]	[title~=flower]	Selects all elements with a title attribute containing the word "flower"
[attribute|=value]	[lang|=en]	Selects all elements with a lang attribute value starting with "en"
[attribute^=value]	a[href^="https"]	Selects every <a> element whose href attribute value begins with "https"
[attribute$=value]	a[href$=".pdf"]	Selects every <a> element whose href attribute value ends with ".pdf"
[attribute=value]*	a[href*="w3schools"]	Selects every <a> element whose href attribute value contains the substring "w3schools"

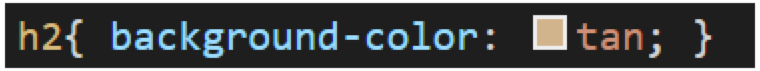
1.6 CSS Colors

Colors are specified using

- predefined color names,
- RGB,
- RGBA,
- HEX,
- HSL,
- HSLA.

1.6.1 Color Names:

In CSS, a color can be specified by using a predefined color name.

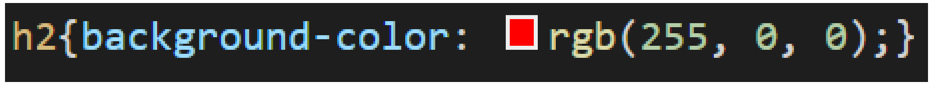


```css
h2{ background-color: tan; }
```

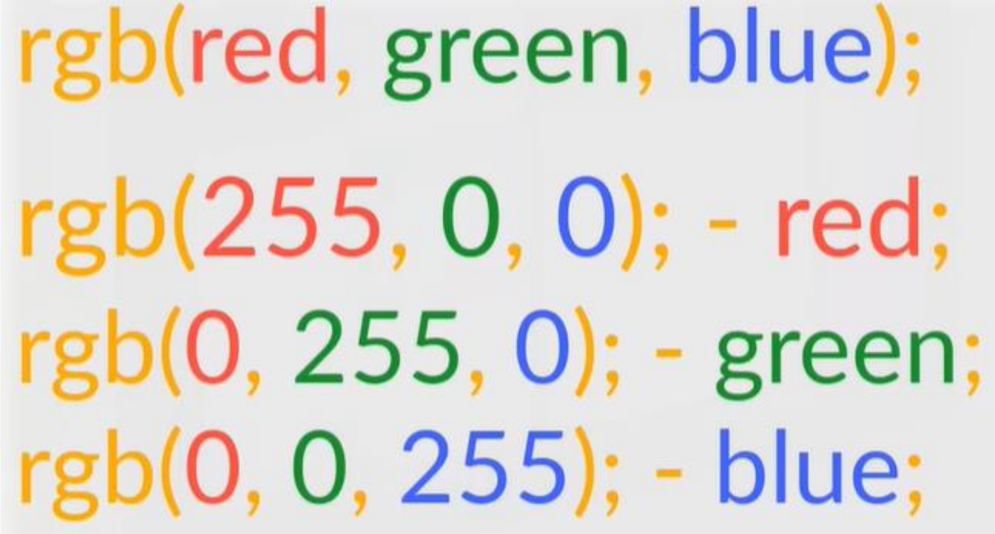
1.6.2 RGB Value

An RGB color value represents RED, GREEN, and BLUE light values.
Range of values between 0 – 255(0 being lowest, 255 being highest)

Syntax:
rgb(*red*, *green*, *blue*)



```
h2{background-color: 🟥rgb(255, 0, 0);}
```



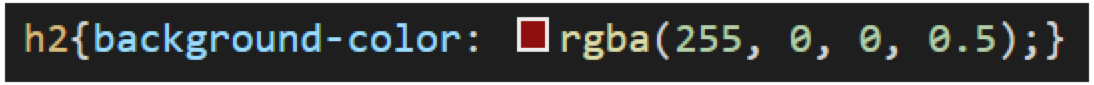
```
rgb(red, green, blue);

rgb(255, 0, 0); - red;
rgb(0, 255, 0); - green;
rgb(0, 0, 255); - blue;
```

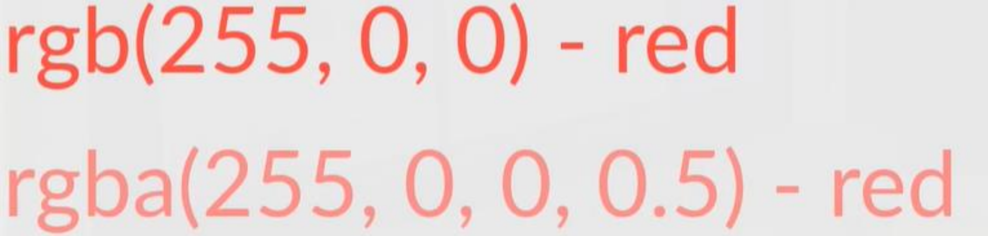
1.6.3 RGBA Value

An RGBA color value represents RED, GREEN,BLUE,ALPHA values.
Range of RGB values between 0 – 255(0 being lowest, 255 being highest)
Range of Alpha values between 0 – 1 (0 being full transparent, 1 being zero transparent)

Syntax:
rgba(*red, green, blue, alpha*)



```
h2{background-color: rgba(255, 0, 0, 0.5);}
```



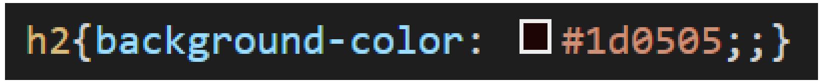
rgb(255, 0, 0) - red
rgba(255, 0, 0, 0.5) - red

1.6.4 HEX Value

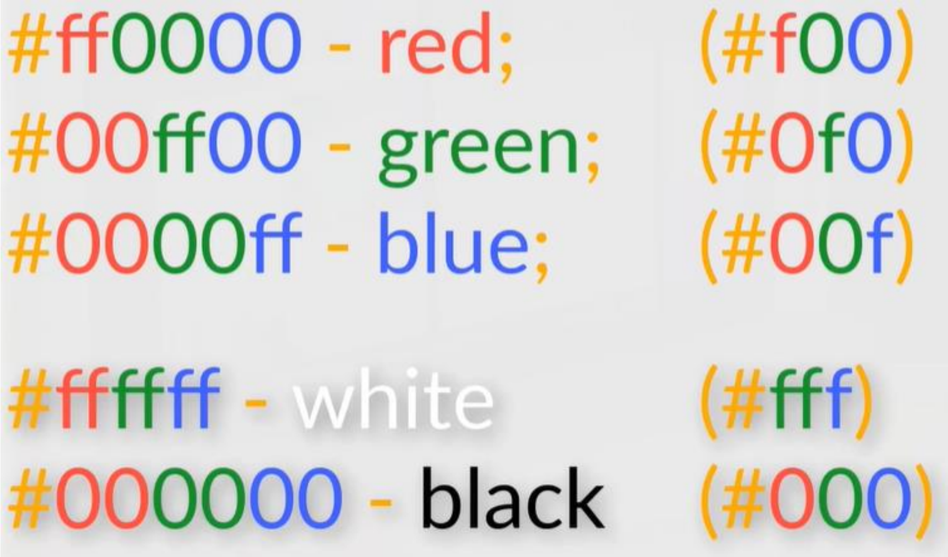
A hexadecimal color is specified with: #RRGGBB, where the RR (red), GG (green) and BB (blue) hexadecimal integers specify the components of the color.
Range is 0 to FF(same as 0-255)

Syntax:
#rrggbb

3 Digit HEX Value
The 3-digit hex code is a shorthand
for some 6-digit hex codes.
The 3-digit hex code has the following form:
#rgb



```
h2{background-color: ■#1d0505;;}
```



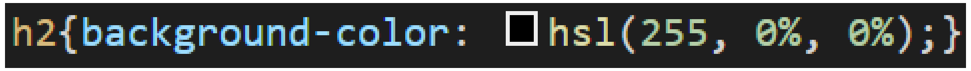
#ff0000 - red; (#f00)
#00ff00 - green; (#0f0)
#0000ff - blue; (#00f)

#ffffff - white (#fff)
#000000 - black (#000)

1.6.5 HSL Values

In CSS, a color can be specified using hue, saturation, and lightness (HSL) .
Hue is a degree on the color wheel from 0 to 360. 0 is red, 120 is green, and 240 is blue.
Saturation is a percentage value, 0% means a shade of gray, and 100% is the full color.
Lightness is also a percentage, 0% is black, 50% is neither light or dark, 100% is white

Syntax:
hsl(*hue*, *saturation*, *lightness*)



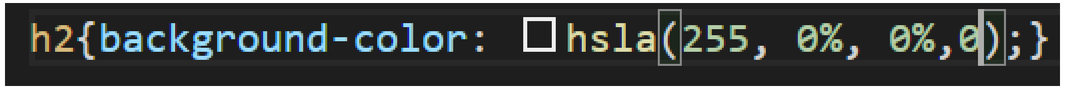
```
h2{background-color: ■hsl(255, 0%, 0%);}
```

1.6.6 HSLA Values

HSLA color values are an extension of HSL color values with an alpha channel - which specifies the opacity for a color.
The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all)

Syntax:

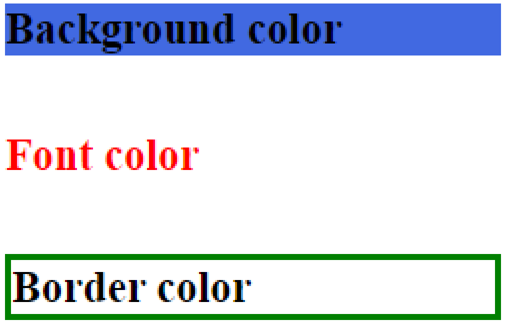
hsla(*hue, saturation, lightness, alpha*)



```
h2{background-color: □hsla(255, 0%, 0%,0);}
```

1.6.7 CSS Colors

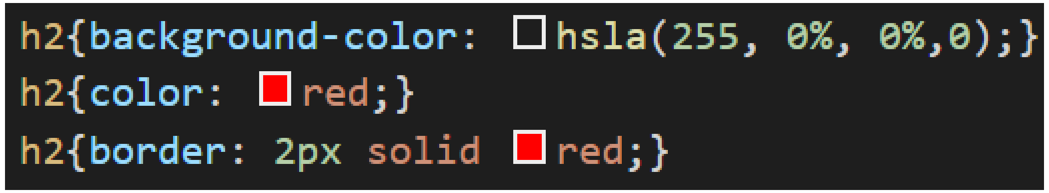
1.6.7.1 Background colors
1.6.7.2 Text colors
1.6.7.3 Border colors



Background color

Font color

Border color



```
h2{background-color: □hsla(255, 0%, 0%,0);}
h2{color: 🟥red;}
h2{border: 2px solid 🟥red;}
```

1.7 CSS Specificity

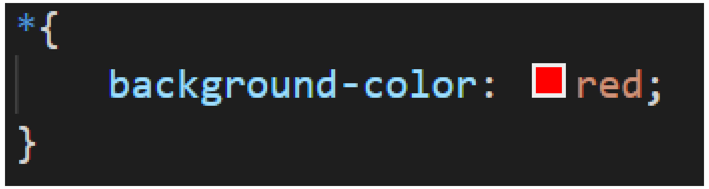
is the means by which browsers decide which CSS property values are the most relevant to an element and, therefore, will be applied.

1.7.1 Specificity scoring

Each selector rule gets a scoring. You can think of specificity as a total score and each selector type earns points towards that score. The selector with the highest score wins.

1.7.2.Universal selector

A universal selector (*) has no specificity and gets **0** points. This means that any rule with 1 or more points will override it

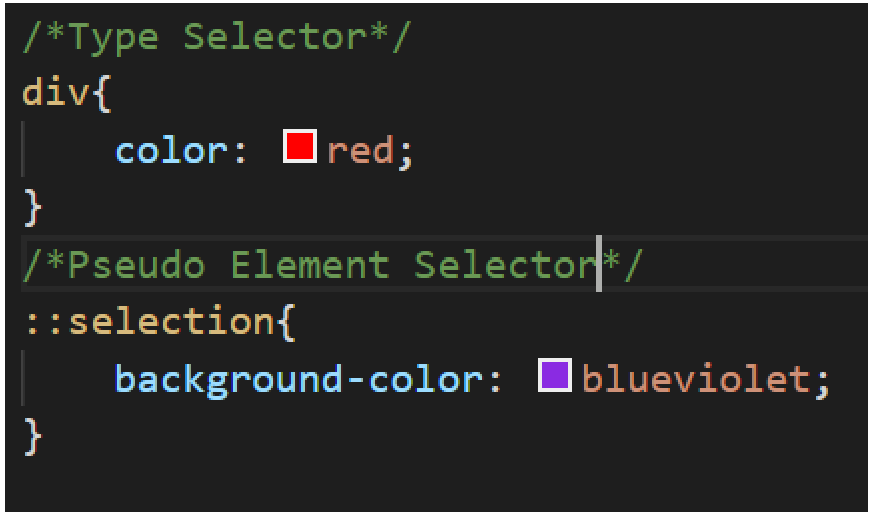


```css
*{
    background-color: red;
}
```

1.7 CSS Specificity

1.7.3 Element or pseudo-element selector

An element (type) or pseudo-element selector gets **1** point of specificity .

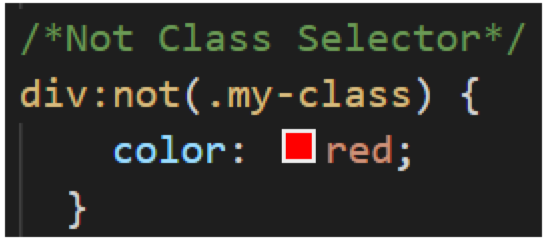


```css
/*Type Selector*/
div{
    color: red;
}
/*Pseudo Element Selector*/
::selection{
    background-color: blueviolet;
}
```

1.7 CSS Specificity

1.7.5 Not Selector

The :not() pseudo-class itself adds nothing to the specificity calculation. However, the selectors passed in as arguments do get added to the specificity calculation.



```
/*Not Class Selector*/
div:not(.my-class) {
    color: 🟥red;
  }
```

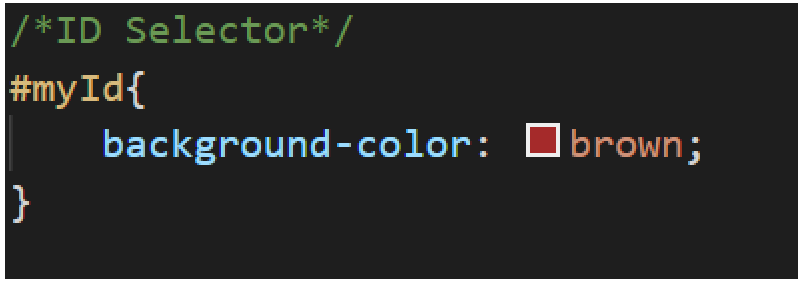
Example: The above sample would have **11** points of specificity because it has

- one type selector (div) and
- one class inside the :not()

1.7 CSS Specificity

1.7.6 ID Selector

An ID selector gets **100** points of specificity.

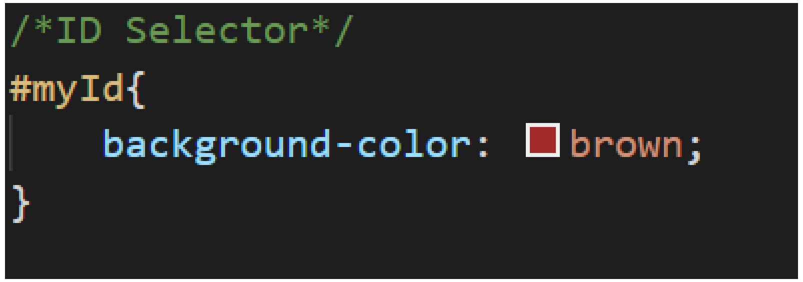


```
/*ID Selector*/
#myId{
    background-color: brown;
}
```

1.7 CSS Specificity

1.7.6 ID Selector

An ID selector gets **100** points of specificity.



```
/*ID Selector*/
#myId{
    background-color: brown;
}
```

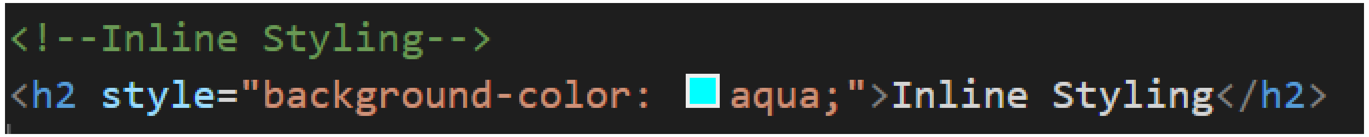
1.7.6 Class/Pseudo Class/Attribute Selector

All class , pseudo class , attribute selector gets **10** points of specificity.

1.7 CSS Specificity

1.7.7 Inline style attribute

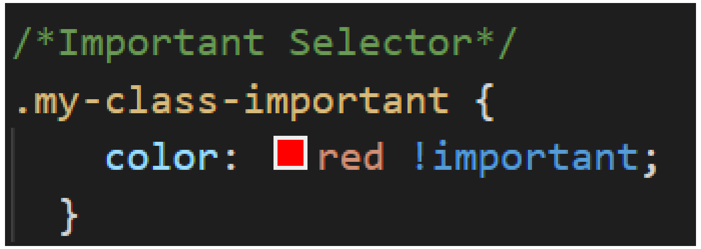
CSS applied directly to the style attribute of the HTML element, gets a specificity score of **1,000** points.



```
<!--Inline Styling-->
<h2 style="background-color: aqua;">Inline Styling</h2>
```

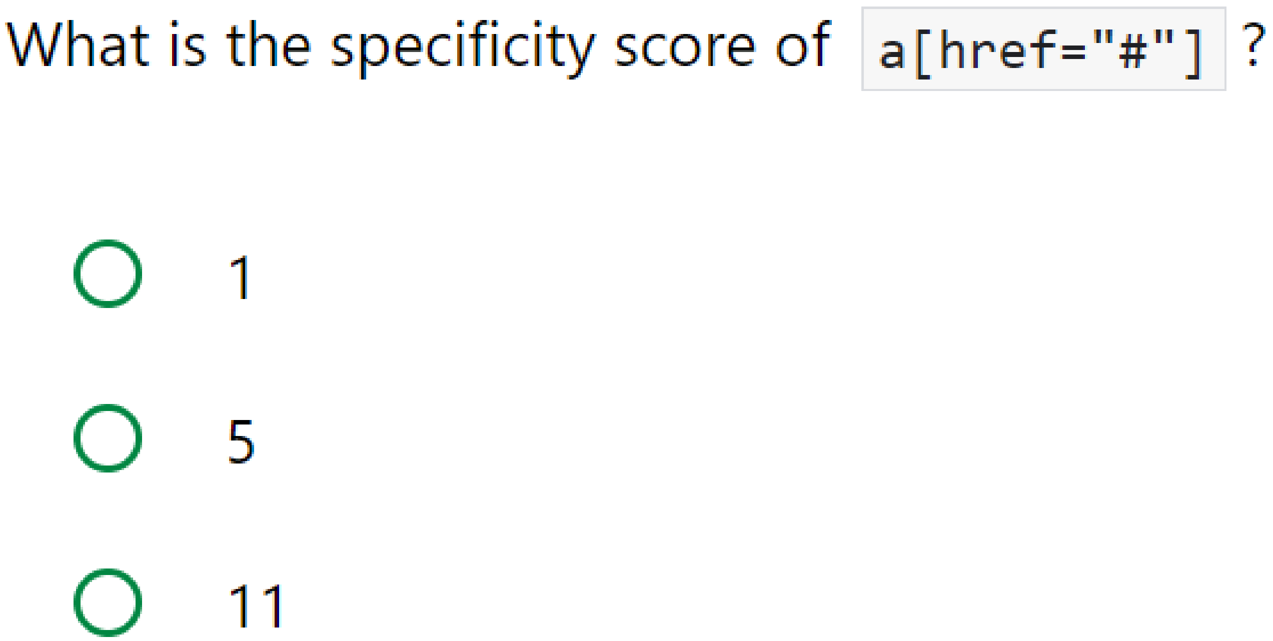
1.7.8 !important attribute

- !important at the end of a CSS value gets a specificity score of **10,000** points.
- This is the highest specificity that one individual item can get.



```
/*Important Selector*/
.my-class-important {
    color: red !important;
  }
```

1.7.9 CSS Specificity - Quiz



What is the specificity score of `a[href="#"]` ?

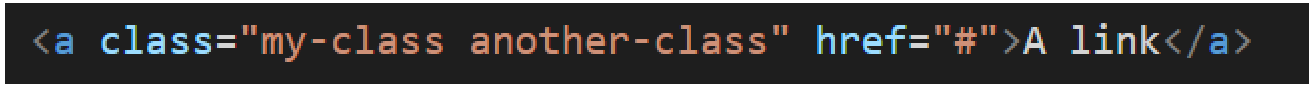
○ 1

○ 5

○ 11

1.7.10 CSS Specificity - Quiz

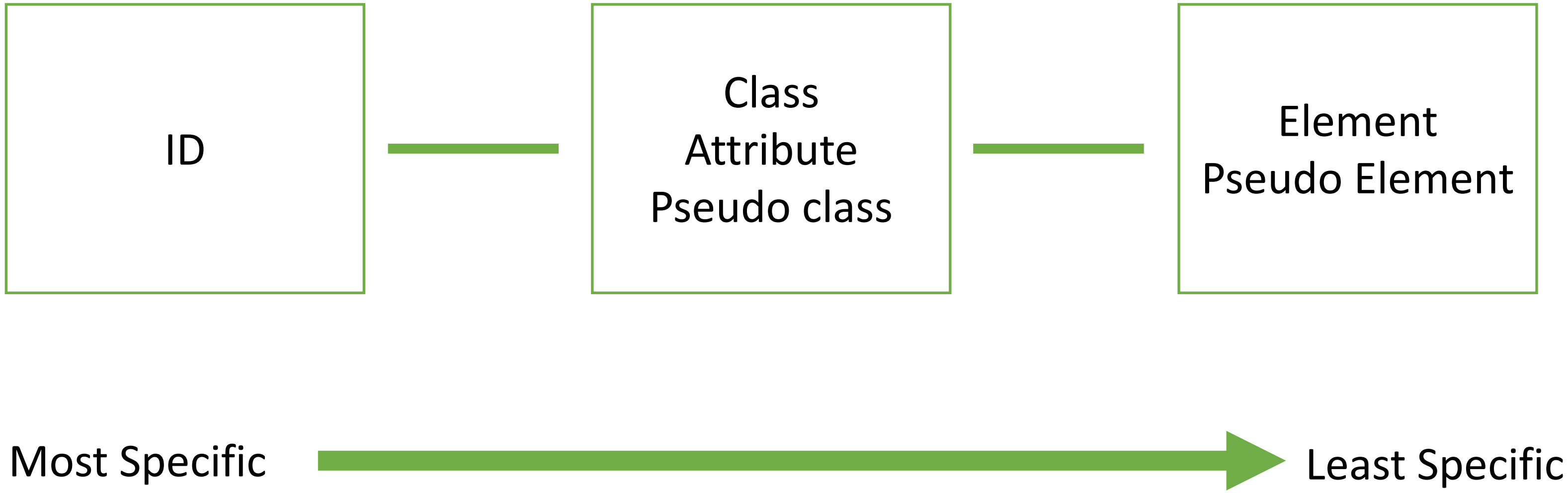
Write a class with highest number of specificity score for the given element



```
<a class="my-class another-class" href="#">A link</a>
```

1.8 Visualizing specificity

In diagrams and specificity calculators, the specificity is often visualized like this:



ID

Class
Attribute
Pseudo class

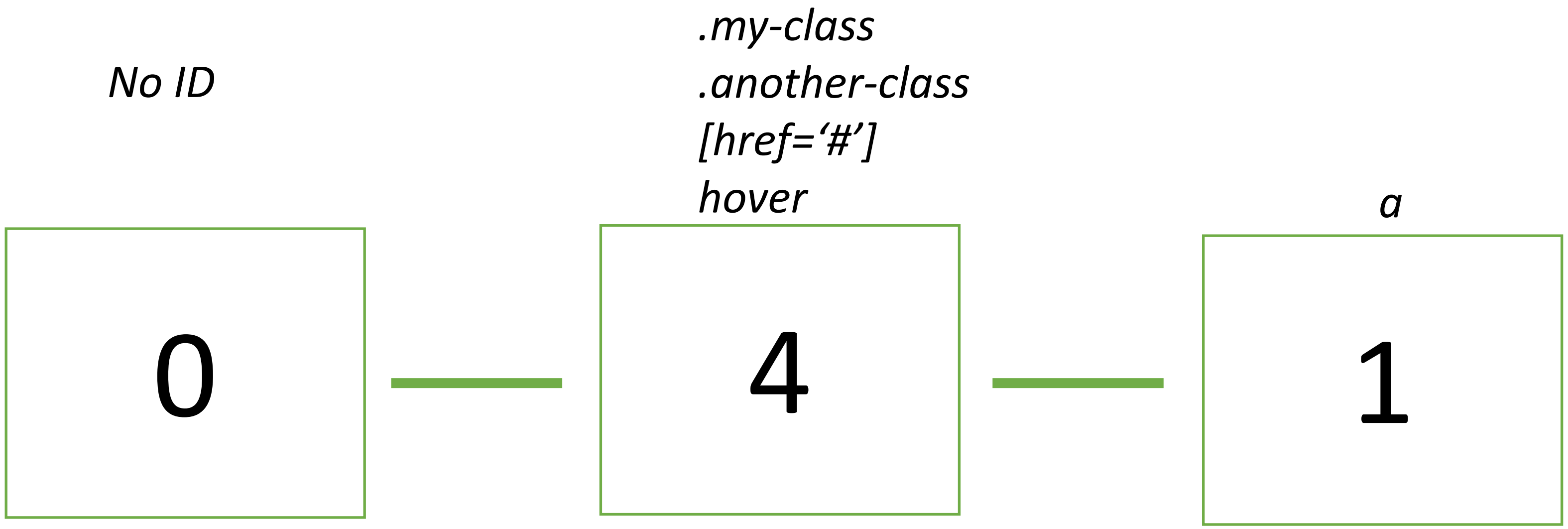
Element
Pseudo Element

Most Specific

Least Specific

1.8 Visualizing specificity

In diagrams and specificity calculators, the specificity is often visualized like this:



No ID

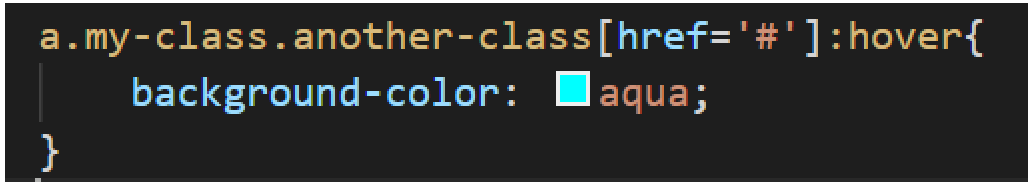
0

.my-class
.another-class
[href=‘#’]
hover

4

a

1



```
a.my-class.another-class[href='#']:hover{
    background-color: aqua;
}
```

1.8.1 Matching specificity

If multiple rules have identical specificity, the newest one is selected by CSS to have its rules applied.

1.8.2 Visualizing Specificity Quiz

Write the specificity for the below CSS classes.

section#specialty.dark

#specialty:hover li.dark

[data-state-rad].dark#specialty:hover

li#specialty section.dark

1-1-1,1-2-1,1-3-0,1-1-2

1.9 CSS Borders

The CSS border properties allow you to specify the style, width, and color of an element's border.

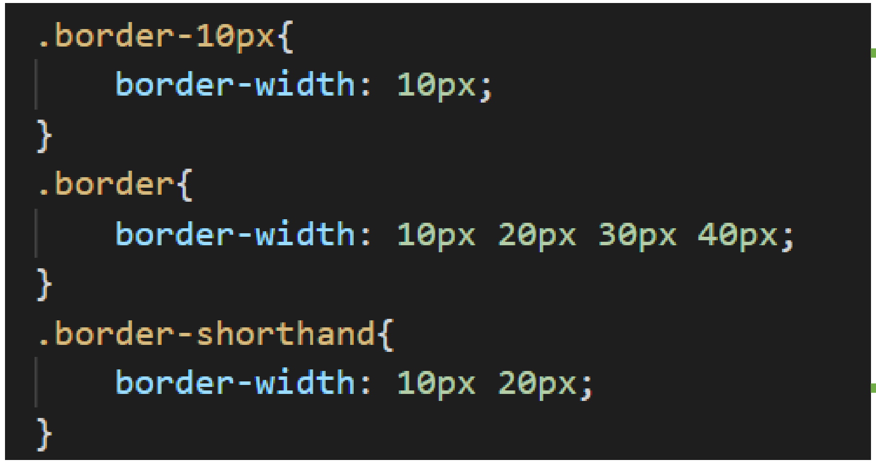
CSS Border Style

The border-style property specifies what kind of border to display.

CSS Border Width

The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick



```css
.border-10px{
    border-width: 10px;
}
.border{
    border-width: 10px 20px 30px 40px;
}
.border-shorthand{
    border-width: 10px 20px;
}
```

Top , right , bottom , left all same

Top ,right ,bottom , left all different

Top & Bottom are same, Left & right are same

1.9 CSS Borders

CSS Border Color

The border-color property is used to set the color of the four borders.

CSS Rounded Borders

The border-radius property is used to add rounded borders to an element

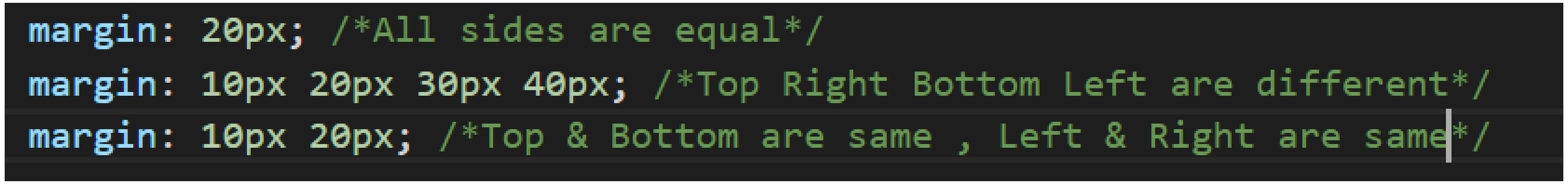
1.10 CSS Margins

- The CSS margin properties are used to create space around elements, outside of any defined borders.

Property	Description
margin	A shorthand property for setting the margin properties in one declaration
margin-bottom	Sets the bottom margin of an element
margin-left	Sets the left margin of an element
margin-right	Sets the right margin of an element
margin-top	Sets the top margin of an element

1.10.1 CSS Margin Collapse

The CSS margin properties are used to create space around elements, outside of any defined borders.



```
margin: 20px; /*All sides are equal*/
margin: 10px 20px 30px 40px; /*Top Right Bottom Left are different*/
margin: 10px 20px; /*Top & Bottom are same , Left & Right are same*/
```

1.11 CSS Padding

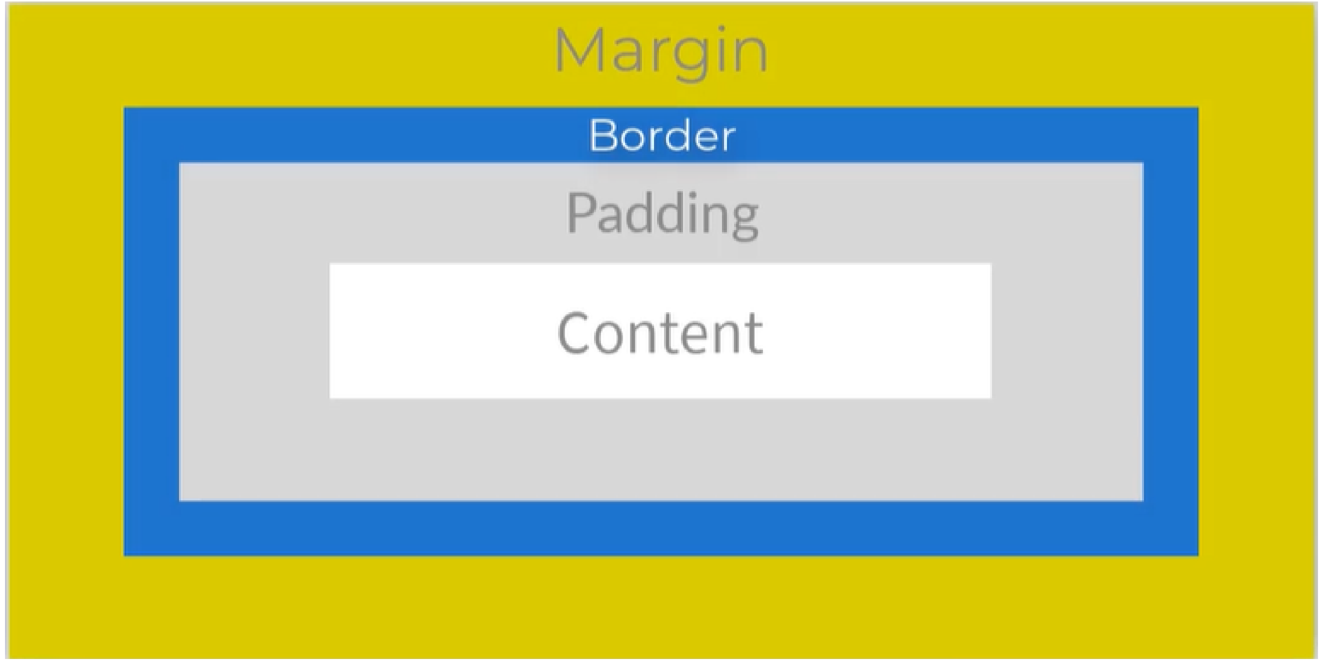
- Padding is used to create space around an element's content, inside of any defined borders.

Property	Description
padding	A shorthand property for setting all the padding properties in one declaration
padding-bottom	Sets the bottom padding of an element
padding-left	Sets the left padding of an element
padding-right	Sets the right padding of an element
padding-top	Sets the top padding of an element

1.9 CSS Box Model

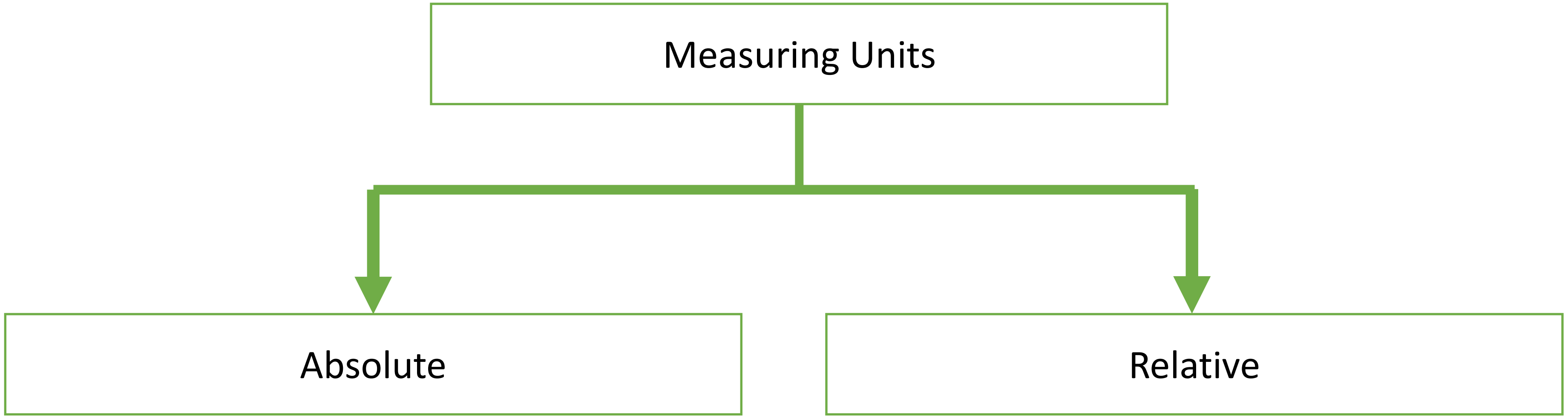
The CSS box model is essentially a box that wraps around every HTML element.

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content
- **Margin** - Clears an area outside the border. The margin is transparent



1.10 Measurement Units

CSS has several different units for expressing a length.



1.10.1 Absolute Units

- The absolute length units are fixed, and a length expressed in any of these will appear as exactly that size.
- Absolute length units are not recommended for use on screen, because screen sizes vary so much. However, they can be used if the output medium is known, such as for print layout.

Unit	Description
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px *	pixels (1px = 1/96th of 1in)
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

1.10.2 Relative Units

- Relative length units specify a length relative to another length property. Relative length units scale better between different rendering medium.

Unit	Description
em	Relative to the font-size of the element (2em means 2 times the size of the current font)
rem	Relative to font-size of the root element
vw	Relative to 1% of the width of the viewport*
vh	Relative to 1% of the height of the viewport*
%	Relative to the parent element

1.11 Inheritance

Inheritance is when the child elements inherits CSS properties from the Parent.

- **Not all CSS properties are inheritable.**
- **Some CSS properties inherit if you don't specify a value for them.**

1.11 Inheritance

1.11.1 Inherit

You can make any property inherit its parent's computed value with the inherit keyword.

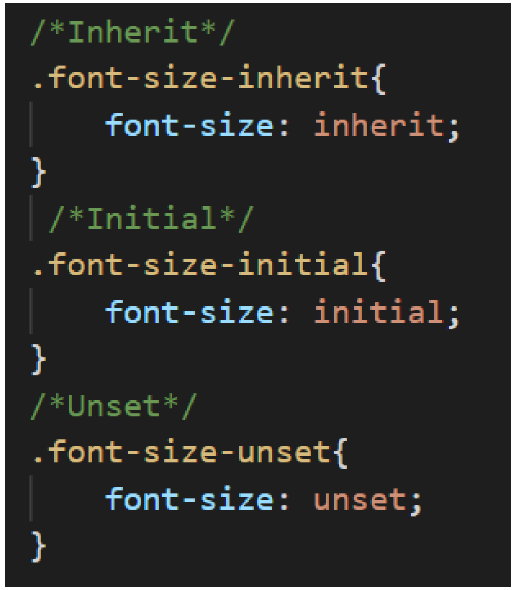
1.11.2 Initial

Inheritance can cause problems with your elements and initial provides you with a powerful reset option.

You learned earlier that every property has a default value in CSS. The initial keyword sets a property back to that initial, default value.

1.11.3 Unset

- The unset property behaves differently if a property is inheritable or not.
- If a property is inheritable, the unset keyword will be the same as inherit.
- If the property is not inheritable, the unset keyword is equal to initial.



```css
/*Inherit*/
.font-size-inherit{
    font-size: inherit;
}
/*Initial*/
.font-size-initial{
    font-size: initial;
}
/*Unset*/
.font-size-unset{
    font-size: unset;
}
```

1.11.4 Inheritable Properties

border-collapse

border-spacing

caption-side

color

cursor

direction

empty-cells

font-family

font-size

font-style

font-variant

font-weight

font

letter-spacing

line-height

list-style-image

list-style-position

list-style-type

list-style

orphans

quotes

text-align

text-indent

text-transform

visibility

white-space

widows

word-spacing

1.12 CSS Positioning

The position property specifies the type of positioning method used for an element.

There are five different position values:

- static
- relative
- fixed
- absolute
- sticky

Elements are then positioned using the

- top
- bottom
- left
- right properties.

1.12 CSS Positioning

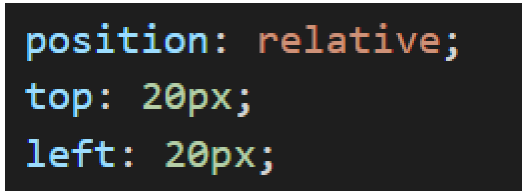
1.12.1 Static

- HTML elements are positioned static by **default**.
- Static positioned elements are not affected by the top, bottom, left, and right properties.
- An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page

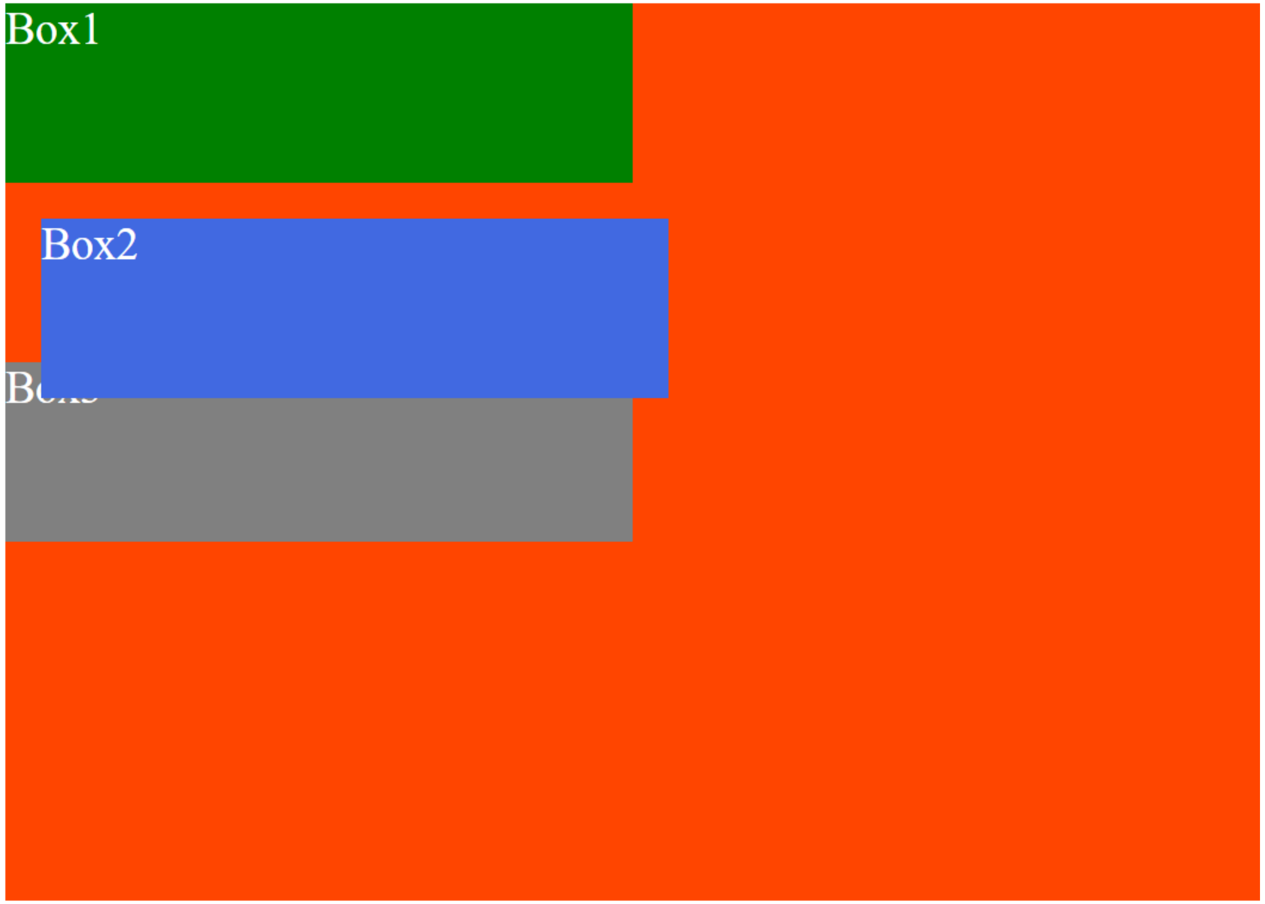
1.12 CSS Positioning

1.12.2 Relative

- HTML elements are **positioned relative to its normal position**.
- 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.
- Relative can also have negative value.
  - For e.g., bottom:-20px ===> top:20px



```
position: relative;
top: 20px;
left: 20px;
```

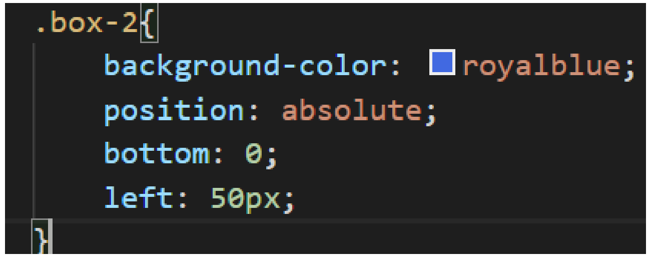


1.12 CSS Positioning

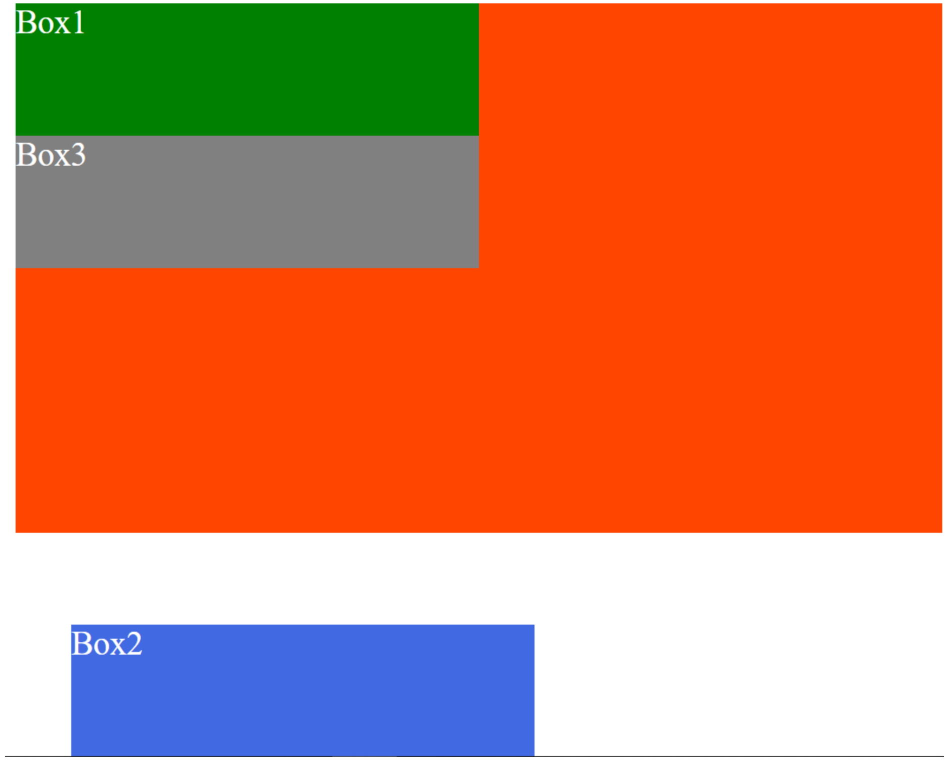
- **1.12.3 Absolute**
- An element with position: absolute; is positioned relative to the nearest positioned ancestor.

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

- Note: **Absolute positioned elements are removed from the normal flow and can overlap elements.**

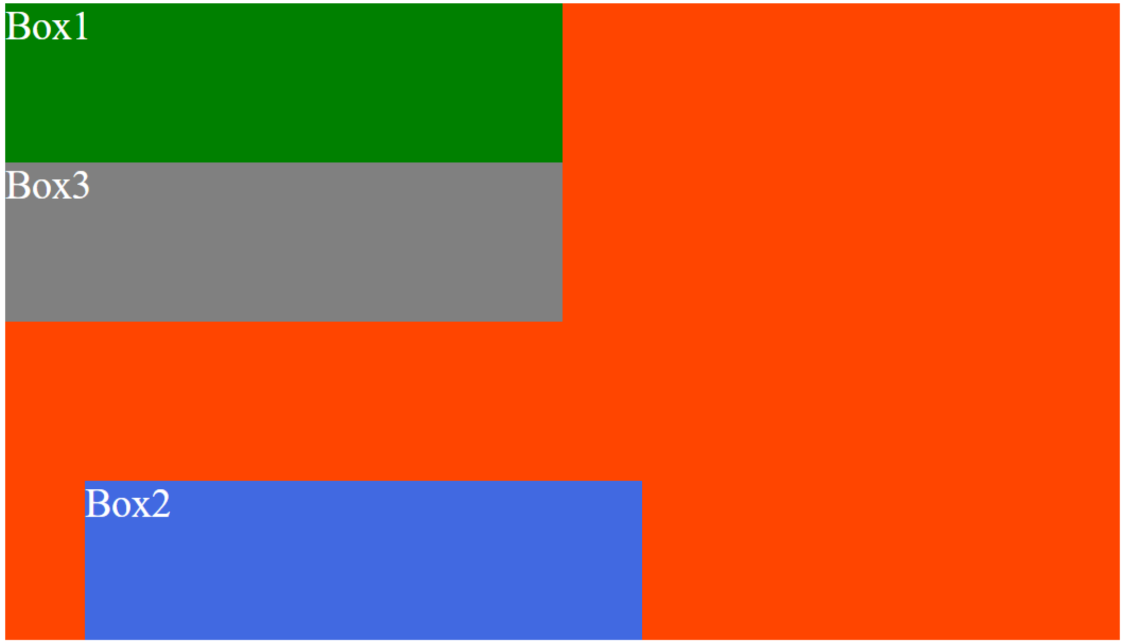


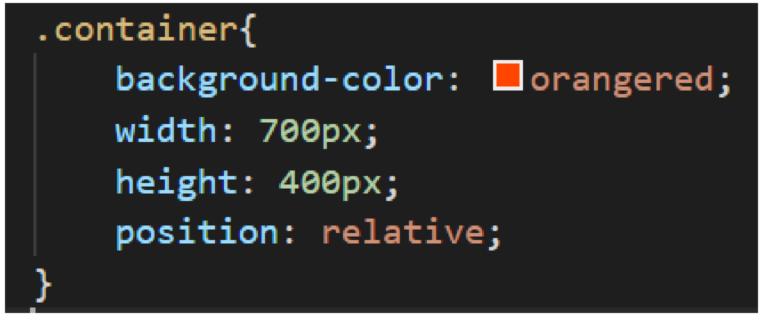
```css
.box-2{
    background-color: royalblue;
    position: absolute;
    bottom: 0;
    left: 50px;
}
```



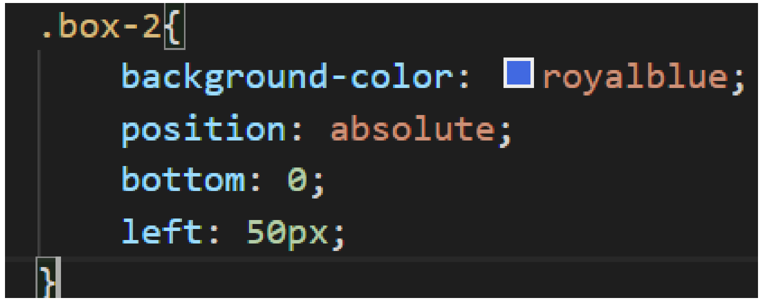
1.12 CSS Positioning

- **1.12.3 Absolute:** If the position of the parent is not static (relative, absolute , fixed, sticky) , then the child absolute element is positioned absolute to the parent element and not the body.





```
.container{
    background-color: 🟥orangered;
    width: 700px;
    height: 400px;
    position: relative;
}
```

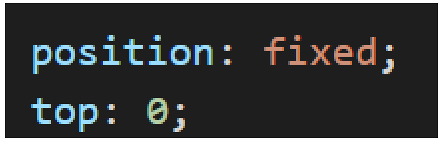


```
.box-2{
    background-color: 🟦royalblue;
    position: absolute;
    bottom: 0;
    left: 50px;
}
```

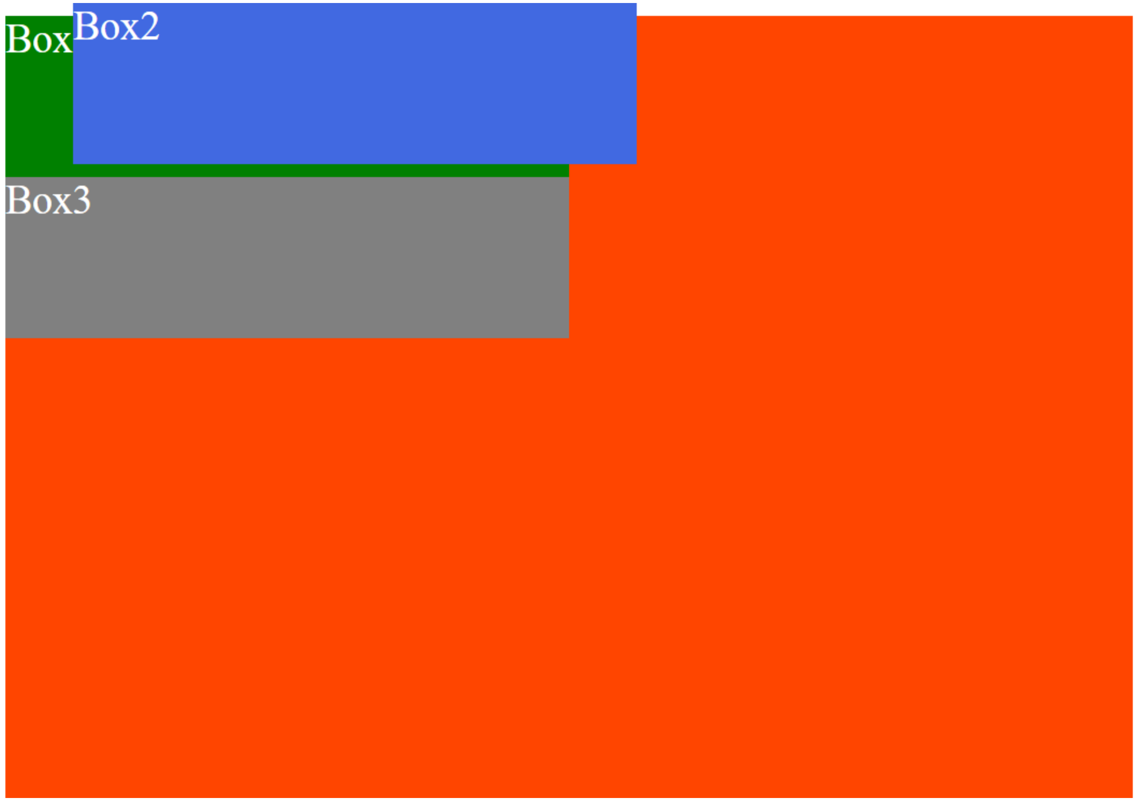
1.12 CSS Positioning

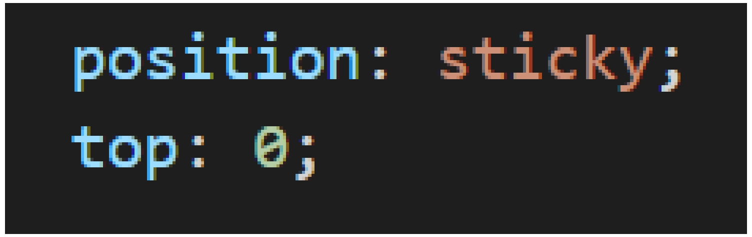
1.12.4 Fixed

- 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.

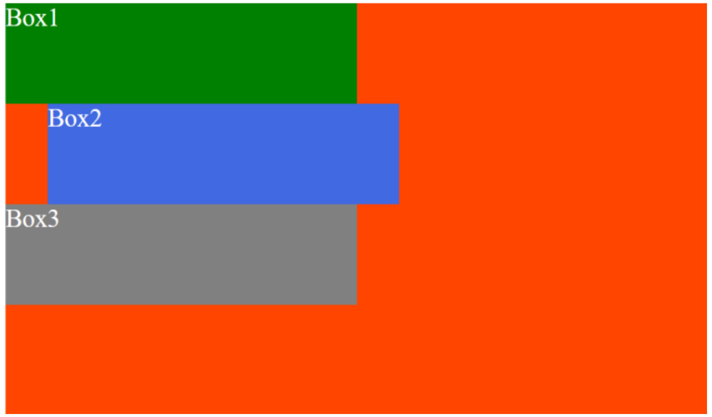


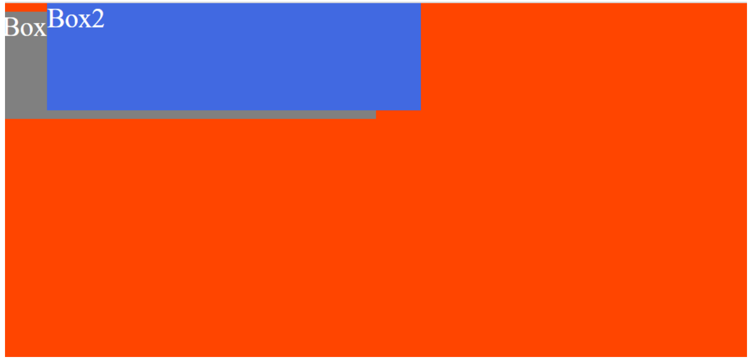
```
position: fixed;
top: 0;
```





```
position: sticky;
top: 0;
```





1.12 CSS Positioning

- **1.12.4 Sticky**
- An element with position: sticky; is positioned based on the **user's scroll position**.

- A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then its "sticks" in place (like position:fixed).

1.12.5 CSS Position Summary

- Static is default
- Relative works just like static but you can move it based on itself
- Absolute works just like relative but it is moved based on the parent element that is using position relative, absolute, fixed , sticky .
- Sticky works just like relative and fixed while scrolling.
- Fixed which allows you to put an element on a page and stay there no matter where the user moves through the scroll wheel

1.13 CSS Overflow

- The overflow property specifies whether to clip the content or to add scrollbars when the content of an element is too big to fit in the specified area.
- The overflow property has the following values:

- visible - Default. The overflow is **not clipped**. The content renders outside the element's box
- hidden - The overflow is clipped, and the rest of the content will be **invisible**
- scroll - The overflow is clipped, and a **scrollbar is always added** to see the rest of the content
- auto - Like scroll, but it adds **scrollbars only when necessary**

NOTE:

The overflow property only works for block elements with a specified height.

1.14 CSS Float and Clear

1.14.1 Float

- The CSS float property specifies how an element should float.
- The float property is used for positioning and formatting content

The float property can have one of the following values:

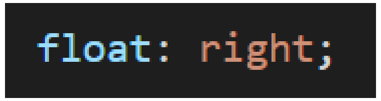
- left - The element floats to the left of its container
- right - The element floats to the right of its container
- none - The element does not float (will be displayed just where it occurs in the text). This is default
- inherit - The element inherits the float value of its parent

1.14 CSS Float and Clear

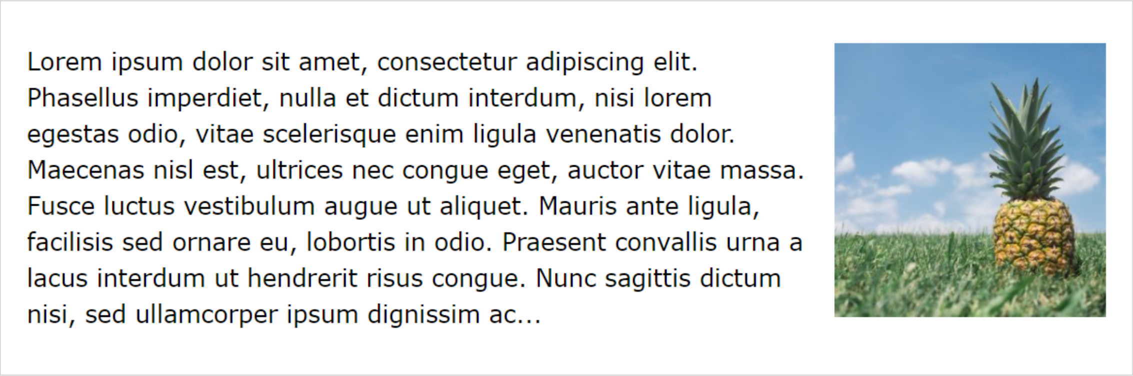
1.14.2 Clear

- The clear property specifies what should happen with the element that is next to a floating element.
- The clear property can have one of the following values:
  - none - The element is not pushed below left or right floated elements. This is default
  - left - The element is pushed below left floated elements
  - right - The element is pushed below right floated elements
  - both - The element is pushed below both left and right floated elements
  - inherit - The element inherits the clear value from its parent
- When clearing floats, you should match the clear to the float: If an element is floated to the left, then you should clear to the left. Your floated element will continue to float, but the cleared element will appear below it on the web page.

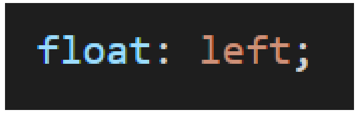
1.14.1.1 Float right/Left



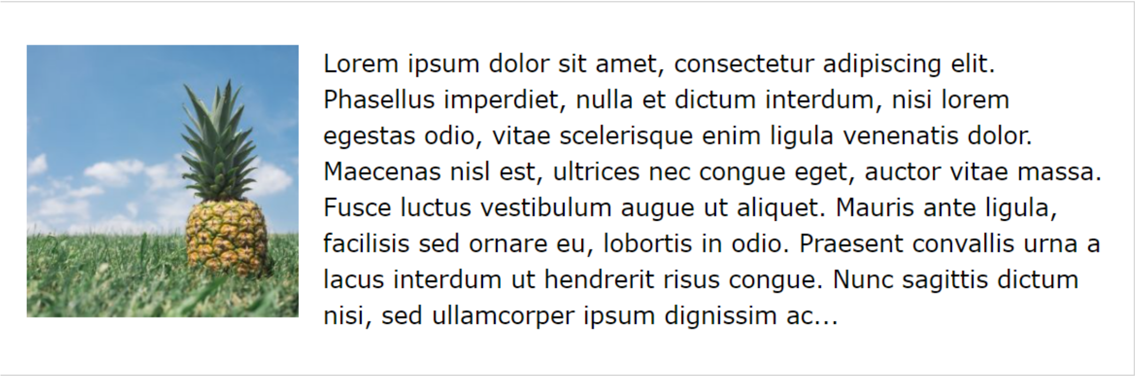
```css
float: right;
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

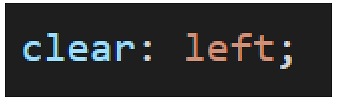


```css
float: left;
```

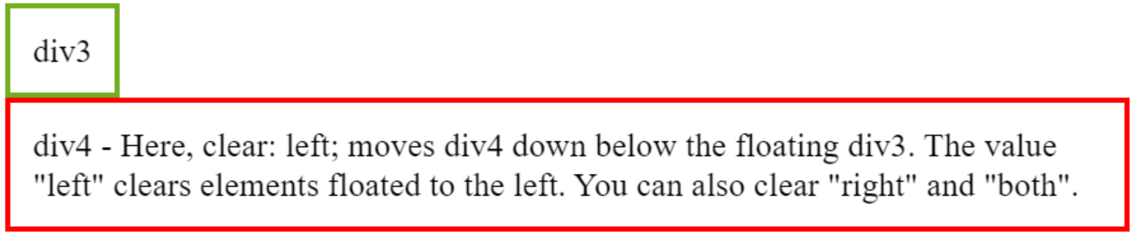


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

1.14.2 Clear Left/Right/Both

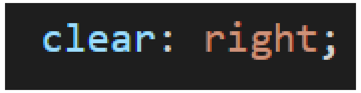


```
clear: left;
```

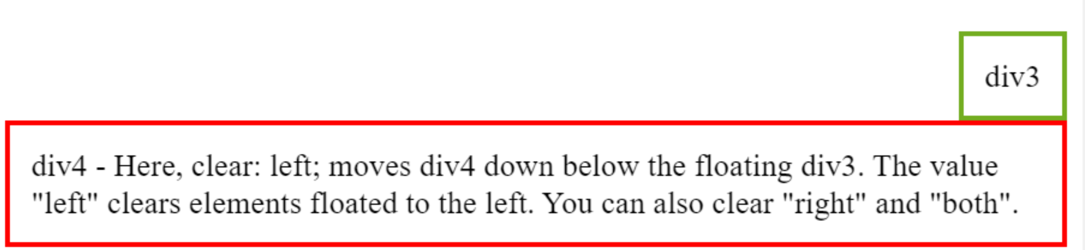


div3

div4 - Here, clear: left; moves div4 down below the floating div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".



```
clear: right;
```



div3

div4 - Here, clear: left; moves div4 down below the floating div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".

1.15 CSS Backgrounds

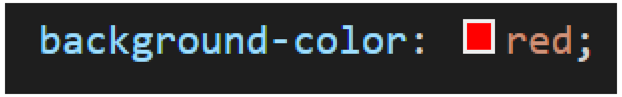
The CSS background properties are used to add background effects for elements.

- background-color
- background-image
- background-repeat
- background-attachment
- background-position
- background (shorthand property)

1.15 CSS Backgrounds

1.15.1 CSS Background color

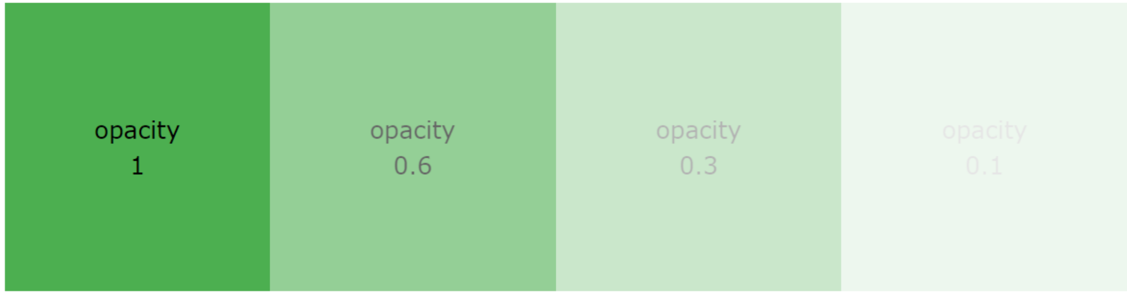
The background-color property specifies the background color of an element.

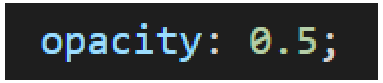


```
background-color: 🟥red;
```

1.15.2 Opacity

The opacity property specifies the opacity/transparency of an element. It can take a value from 0.0 - 1.0. The lower value, the more transparent:





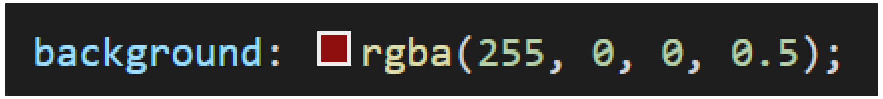
```
opacity: 0.5;
```

1.15 CSS Backgrounds

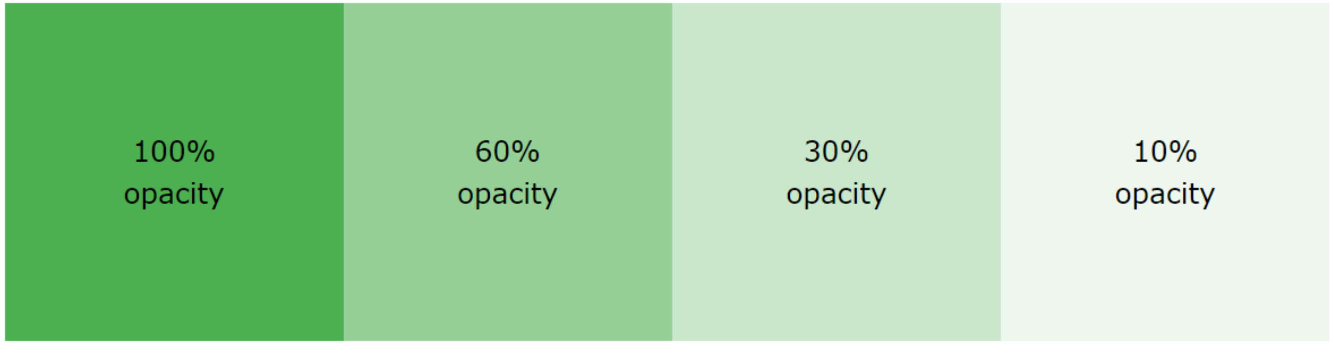
Note: When using the opacity property to add transparency to the background of an element, all its child elements inherit the same transparency. This can make the text inside a fully transparent element hard to read.

1.15.3 Transparency using RGBA

If you do not want to apply opacity to child elements, like in our example above, use **RGBA** color values.



```
background: 🟥rgba(255, 0, 0, 0.5);
```

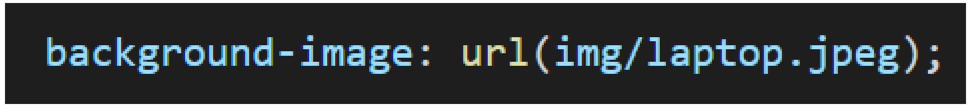


1.15 CSS Backgrounds

1.15.2 Background Image

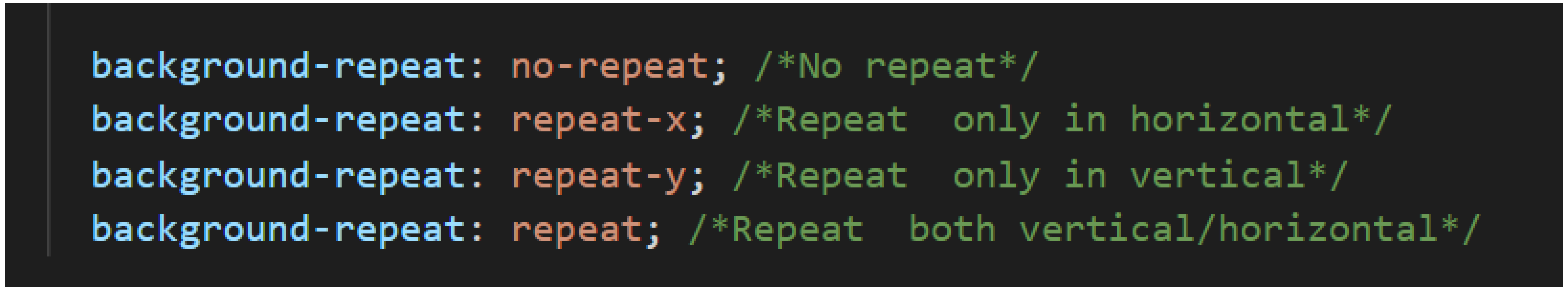
The background-image property specifies an image to use as the background of an element.

By default, the image is repeated so it covers the entire element.



```
background-image: url(img/laptop.jpeg);
```

1.15.2 Background Repeat

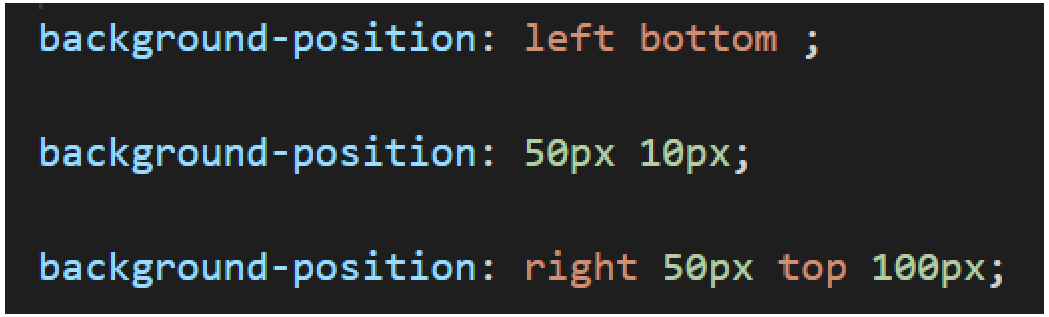


```
background-repeat: no-repeat; /*No repeat*/
background-repeat: repeat-x; /*Repeat  only in horizontal*/
background-repeat: repeat-y; /*Repeat  only in vertical*/
background-repeat: repeat; /*Repeat  both vertical/horizontal*/
```

1.15 CSS Backgrounds

1.15.3 Background position

The background-position property is used to specify the position of the background image.



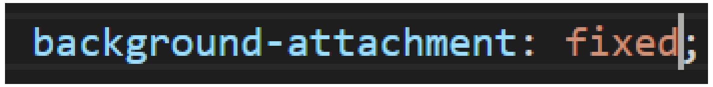
```
background-position: left bottom ;

background-position: 50px 10px;

background-position: right 50px top 100px;
```

1.15.4 Background Attachment

The background-attachment property specifies whether the background image should scroll or be fixed (will not scroll with the rest of the page).

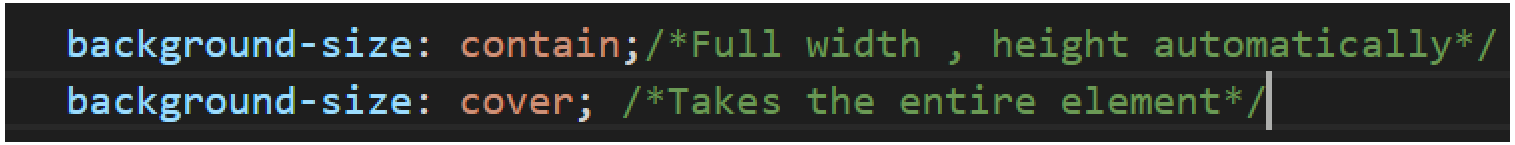


```
background-attachment: fixed;
```

1.15 CSS Backgrounds

1.15.5 Background size

The background-size property is used to specify the size of the background image.



```
background-size: contain;/*Full width , height automatically*/
background-size: cover; /*Takes the entire element*/
```

1.15.6 Background Origin

The background-origin CSS property sets the background's origin: from the border start, inside the border, or inside the padding.

1.15.7 Background Clip

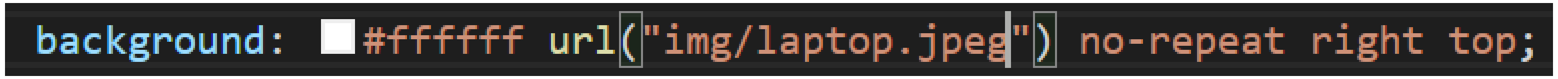
The background-clip CSS property sets whether an element's background extends underneath its border box, padding box, or content box.

1.15 CSS Backgrounds

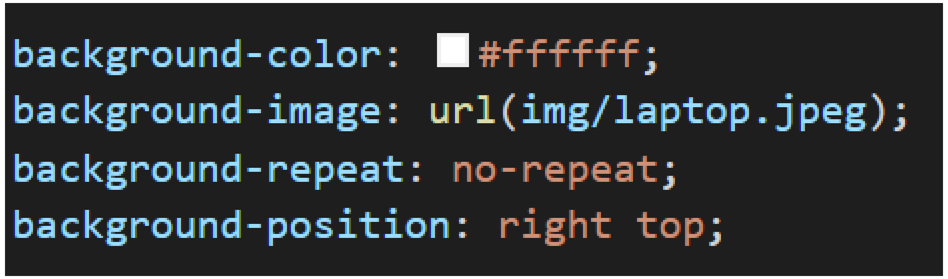
1.15.8 Background shorthand

When using the shorthand, the order of the property values is:

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position



```
background: □#ffffff url("img/laptop.jpeg") no-repeat right top;
```



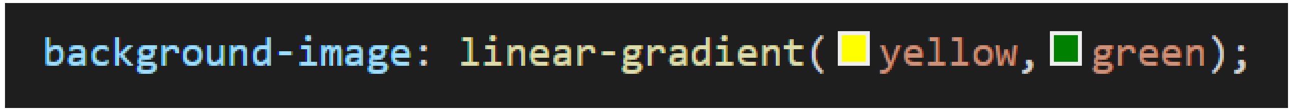
```
background-color: □#ffffff;
background-image: url(img/laptop.jpeg);
background-repeat: no-repeat;
background-position: right top;
```

1.15.9 CSS Summary

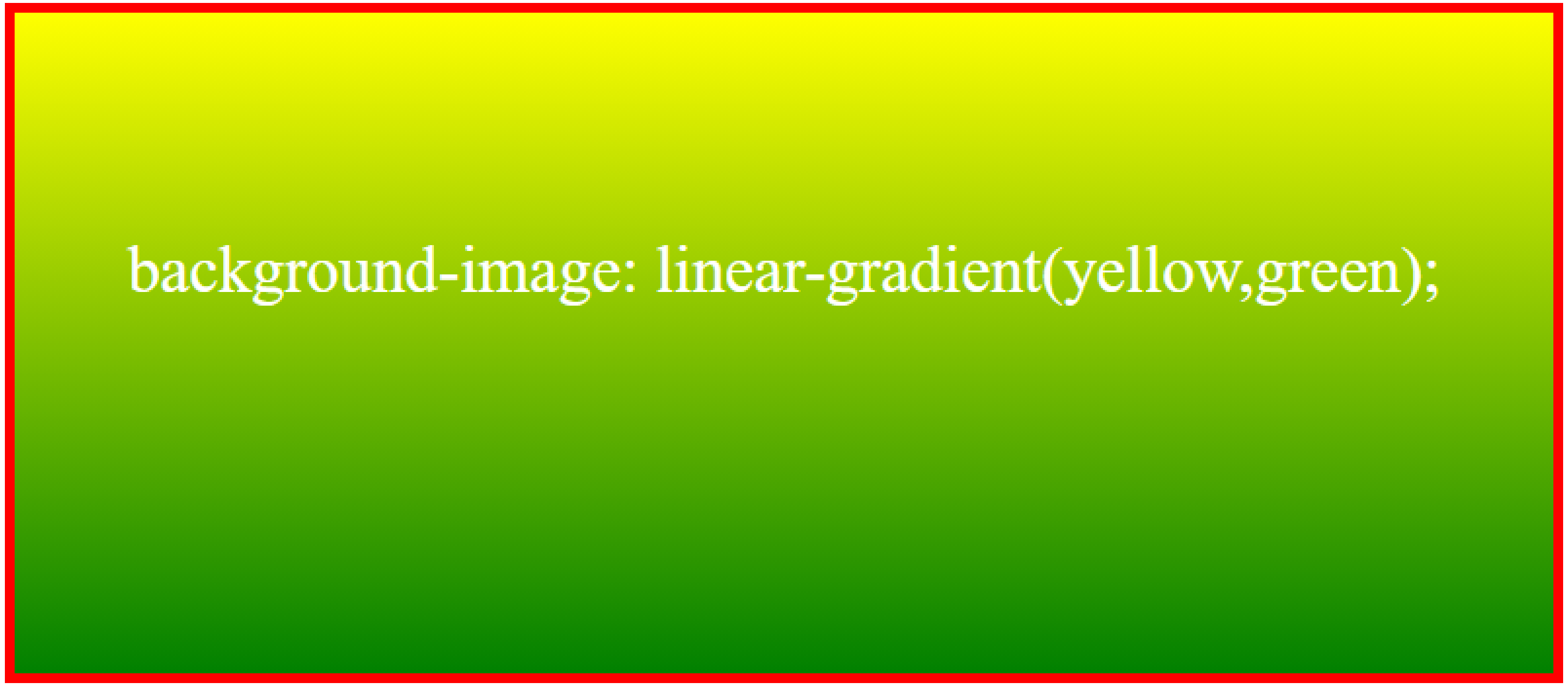
Property	Description
background	Sets all the background properties in one declaration
background-attachment	Sets whether a background image is fixed or scrolls with the rest of the page
background-clip	Specifies the painting area of the background
background-color	Sets the background color of an element
background-image	Sets the background image for an element
background-origin	Specifies where the background image(s) is/are positioned
background-position	Sets the starting position of a background image
background-repeat	Sets how a background image will be repeated
background-size	Specifies the size of the background image(s)

1.16 CSS Gradients

- CSS gradients let you display smooth transitions between two or more specified colors.
- To create a linear gradient, you must define at least two-color stops
- You can also set a starting point and a direction (or an angle) along with the gradient effect.



```
background-image: linear-gradient(🟨yellow,🟩green);
```



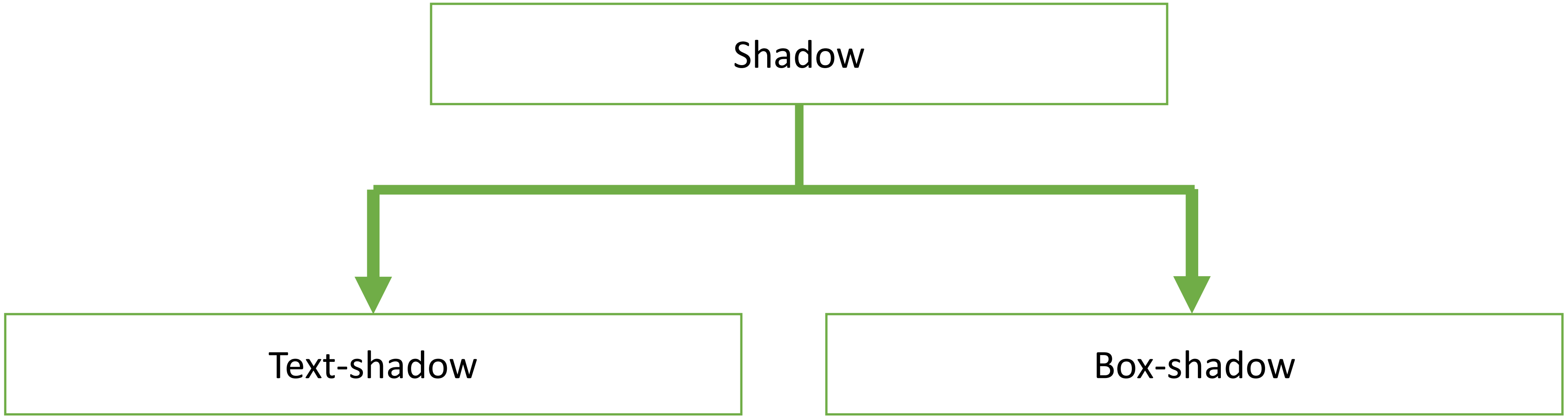
background-image: linear-gradient(yellow,green);

1.16 CSS Gradients

Property	Meaning
to right	Left-right
To left	Right-left
To bottom	Top-bottom Default
To top	Bottom-top
To left bottom	Left – Right bottom
Transparency	Same Color

1.17 CSS Shadows

You can add CSS to text and box elements

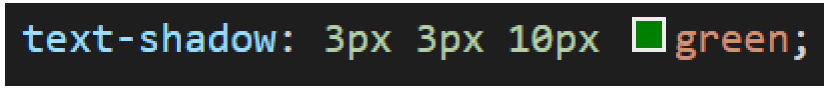


Shadow

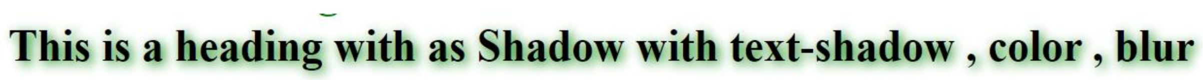
Text-shadow

Box-shadow

1.17 CSS Shadows

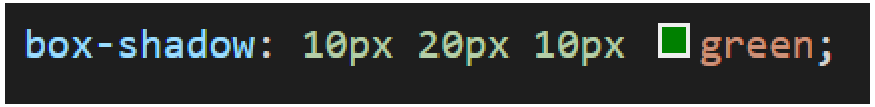


```
text-shadow: 3px 3px 10px 🟩green;
```

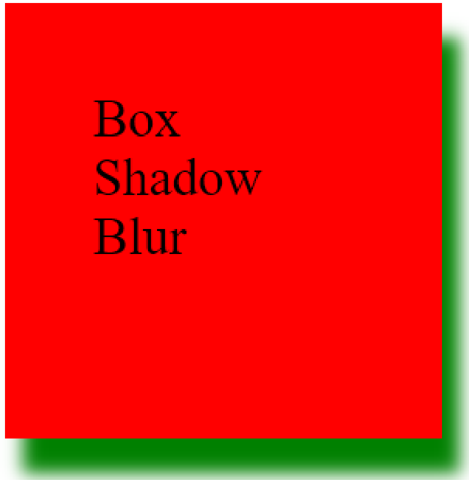


This is a heading with as Shadow with text-shadow , color , blur

Order:
Horizontal vertical blur color



```
box-shadow: 10px 20px 10px 🟩green;
```



Box
Shadow
Blur

1.18 CSS Transition

CSS transitions allows you to change property values smoothly, over a given duration.

To create a transition effect, you must specify two things:

- the CSS property you want to add an effect to
- the duration of the effect

Note: If the duration part is not specified, the transition will have no effect, because the default value is 0.

1.18.1 CSS Transition Delay

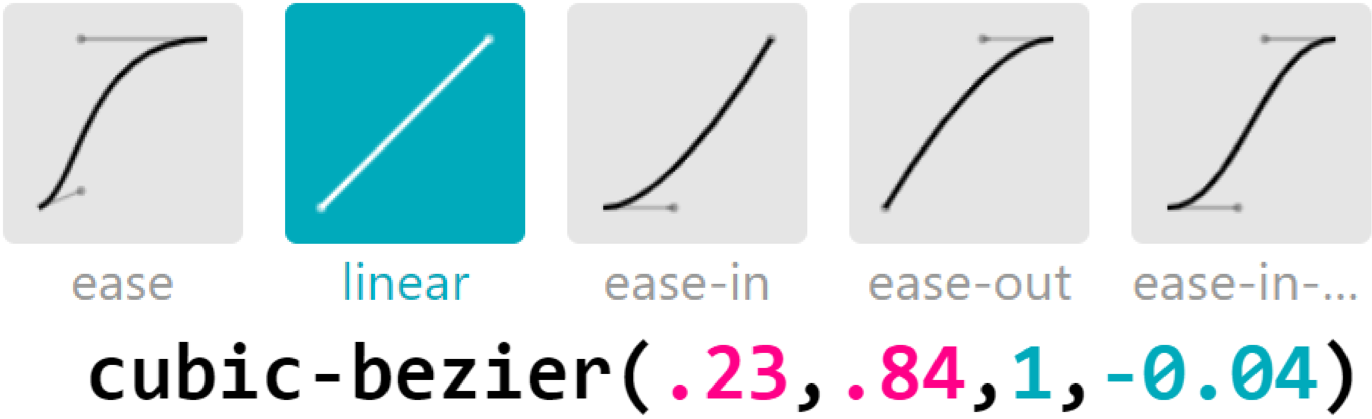
Specifies a delay (in seconds) for the transition effect

1.18 CSS Transition

1.18.2 CSS Transition Duration
Specifies a delay (in seconds) for the transition effect

1.18.3 CSS Transition Property
Specifies the name of the CSS property the transition effect is for

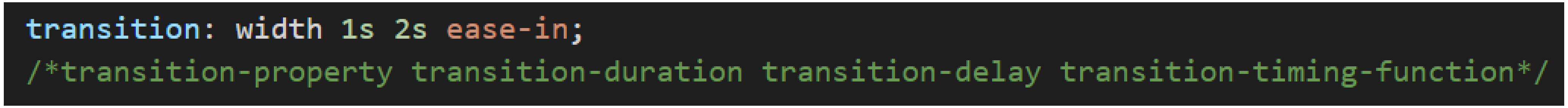
1.18.4 CSS Transition Timing Function
Specifies the speed curve of the transition effect



1.18 CSS Transition Timing Function

1.18.4 CSS Transition shorthand

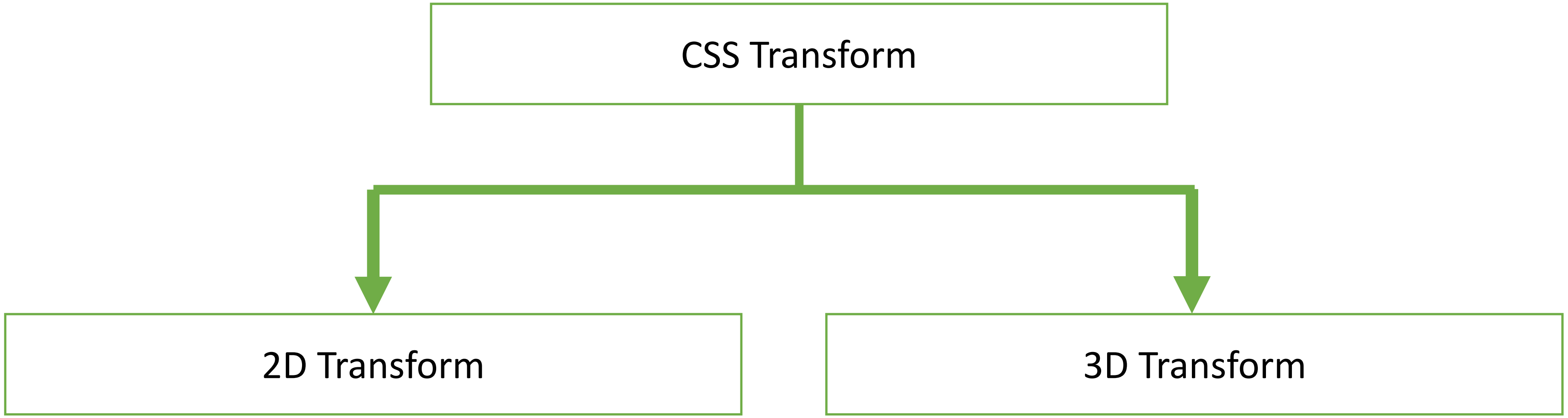
A shorthand property for setting the four transition properties into a single property



```
transition: width 1s 2s ease-in;
/*transition-property transition-duration transition-delay transition-timing-function*/
```

1.19 CSS Transform

CSS transforms allow you to move, rotate, scale, and skew elements.



CSS Transform

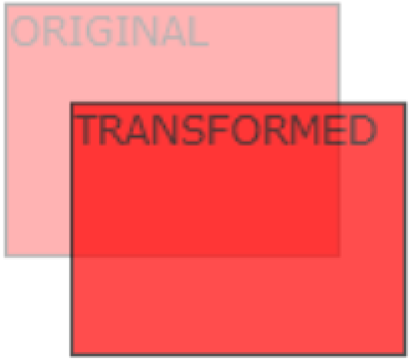
2D Transform

3D Transform

1.19.1 CSS 2D Transform

CSS transforms allow you to move, rotate, scale, and skew elements.

1.18.1 Translate
The translate() method moves an element from its current position (according to the parameters given for the X-axis and the Y-axis).



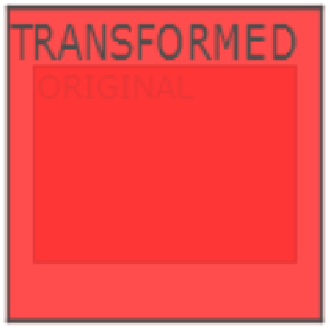
1.18.2 Rotate
The rotate() method rotates an element clockwise or counter-clockwise according to a given degree.

ORIGIN
TRANSFORMED

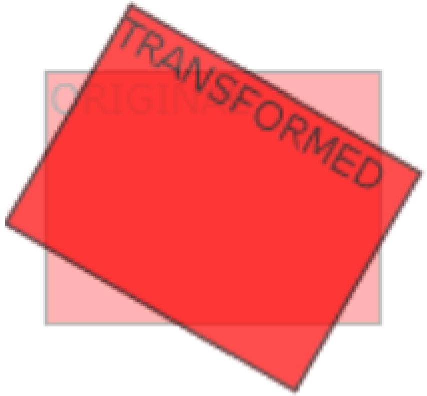
1.19.1 CSS 2D Transform

CSS transforms allow you to move, rotate, scale, and skew elements.

1.18.3 Scale
The scale() method increases or decreases the size of an element (according to the parameters given for the width and height).



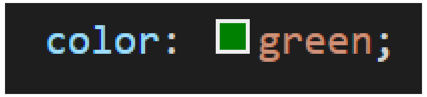
1.18.4 Skew
The skew() method skews an element along the X and Y-axis by the given angles.



1.20 Text Styling

1.19.1 Color

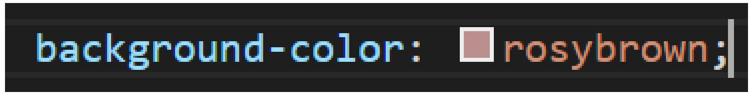
The color property is used to set the color of the text.



```
color: green;
```

1.19.2 Background Color

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



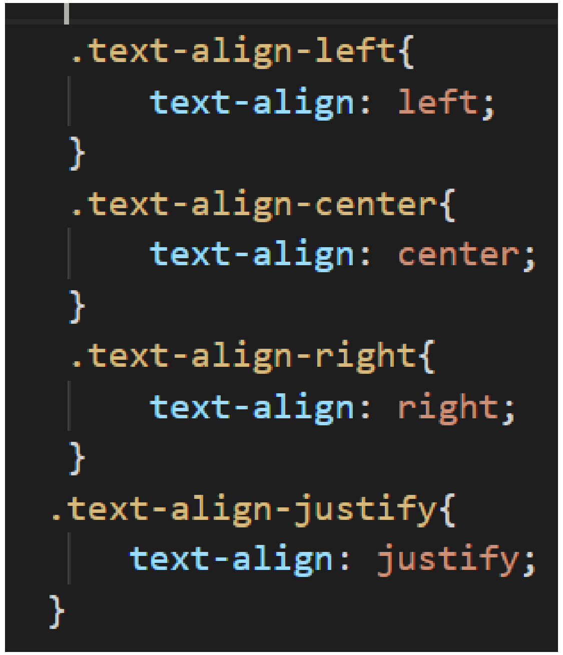
```
background-color: rosybrown;
```

1.20 Text Styling

1.19.3 Text Align
The text-align property is used to set the horizontal alignment of a text.

A text can be left or right aligned, centered, or justified.

1.19.4 Text Direction
The direction property can be used to change the text direction of an element.
1.19.5 Text Decoration
The text-decoration property is used to set or remove decorations from text.
1.19.6 Text Transform
The text-transform property is used to specify uppercase and lowercase letters or capitalize the first letter of each word.



```
 .text-align-left{
     text-align: left;
 }
 .text-align-center{
     text-align: center;
 }
 .text-align-right{
     text-align: right;
 }
.text-align-justify{
    text-align: justify;
}
```

1.20 Text Styling

1.19.7 Text Spacing
The text-indent property is used to specify the indentation of the first line of a text

1.19.8 Letter Spacing
The letter-spacing property is used to specify the
space between the characters in a text.

1.19.9 Line Height
The line-height property is used to specify the space between lines.

1.19.10 Word Spacing
The word-spacing property is used to specify the space between the words in a text.

1.19.11 White Space
The white-space property specifies how white-space inside an element is handled.

1.21.1 CSS Font Family -> collection of related fonts

Serif:

- Serif fonts are traditional typefaces using characters that have serifs which are small winged or flared tips extending off the tips of a letter.
- Serif fonts are typically used in printed books, newspapers, and magazines.
- Some popular serif fonts include Times New Roman, Garamond, Palatino, and Georgia.

Sans-Serif:

- Sans-serif fonts use characters without serifs and are more commonly seen in digital formats.
- A sans-serif font will typically be the default font in digital word processing programs.
- Sans-serif fonts include Arial, Helvetica, Verdana, Trebuchet MS, and Gill Sans.

Cursive:

- Cursive fonts use characters that have connective strokes which give the font a handwritten appearance.
- Cursive fonts include Comic Sans MS, Adobe Poetica, Sanvito, and Zapf-Chancery.

Fantasy:

- Fantasy fonts are stylized fonts that still maintain the characteristics of non-cursive, traditional alphabet glyphs.
- Examples include Cottonwood, Critter, and Alpha Geometrique.

Monospace:

- Fonts in the monospace font family have characters that are all the same width, giving text the appearance of a manual monospaced typewriter. Examples of monospaced fonts include Courier New, Monaco, Lucida Console, Consolas, and Everson Mono.

1.21 CSS Font Family

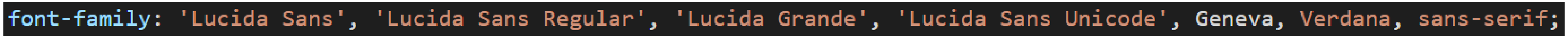
Font-family: <family-name> , <generic-name>;

<family-name>

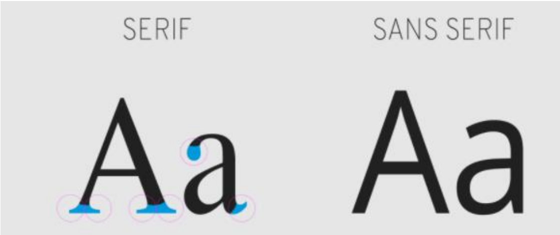
- The name of a font family. For example, "Times" and "Helvetica" are font families.
- **Font family names containing whitespace should be quoted**. For example: "Comic Sans MS".

<generic-name>

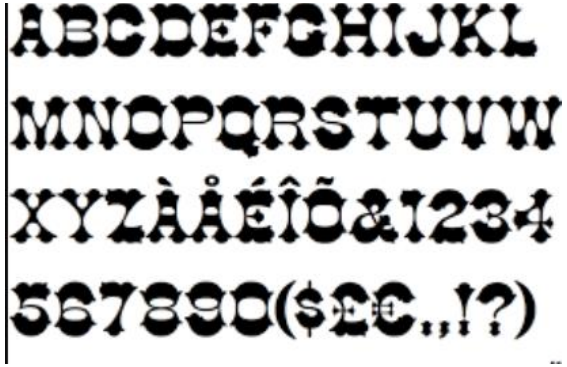
- Generic font families are a **fallback mechanism**, a means of preserving some of the style sheet author's intent when none of the specified fonts are available.
- Generic family names are keywords and must not be quoted.
- A generic font family should be the last item in the list of font family names.



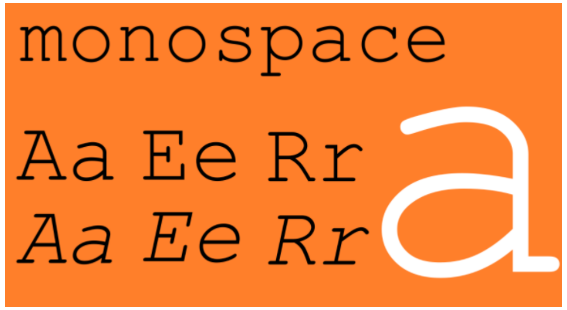
```
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
```



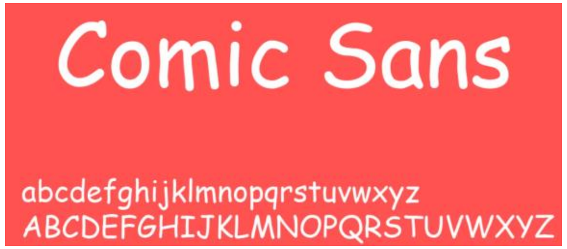
Serif / Serif sans



Cottonwood



monospace



Comic Sans

1.21.1 CSS Font Family

1.21.2 Font Style

The font-style property is mostly used to specify italic text.

- normal - The text is shown normally
- italic - The text is shown in italics

1.21.3 Font Weight

The font-weight property specifies the weight of a font

1.21.4 Font Variant

- The font-variant property specifies whether a text should be displayed in a small-caps font.
- In a small-caps font, all lowercase letters are converted to uppercase letters.
- However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

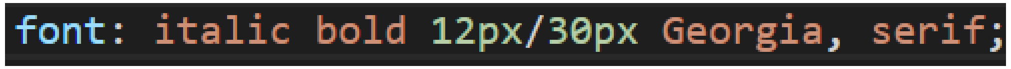
1.21.5 Font Size

- The font-size property sets the size of the text.
- Use h1-h6 for headers
- Use absolute/relative units

1.21.6 Font Shorthand

The font property is a shorthand property for:

- font-style
- font-variant
- font-weight
- font-size/line-height
- font-family



```
font: italic bold 12px/30px Georgia, serif;
```

Note: The font-size and font-family values are required. If one of the other values is missing, their default value are used.

1.22 CSS ICONS

Insert external icons library by adding necessary link stylesheet and script if necessary

1.23 CSS LIST

Property	Description
list-style	Sets all the properties for a list in one declaration
list-style-image	Specifies an image as the list-item marker
list-style-position	Specifies the position of the list-item markers (bullet points)
list-style-type	Specifies the type of list-item marker