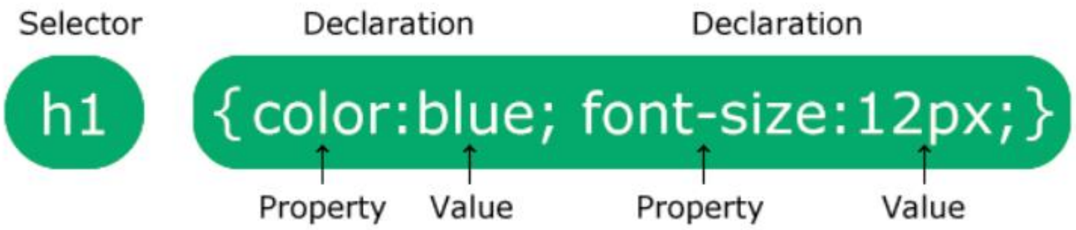
CSS NOTES W3SCHOOLS

What is CSS?

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

CSS Syntax



The CSS element Selector

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

The CSS id Selector

```
#para1 {
  text-align: center;
  color: red;
}
```

The CSS class Selector

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

The CSS Universal Selector

```
* {
  text-align: center;
  color: blue;
}
```

The CSS Grouping Selecto

```
h1, h2, p {
  text-align: center;
  color: red;
}
```

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- External CSS
- Internal CSS
- Inline CSS

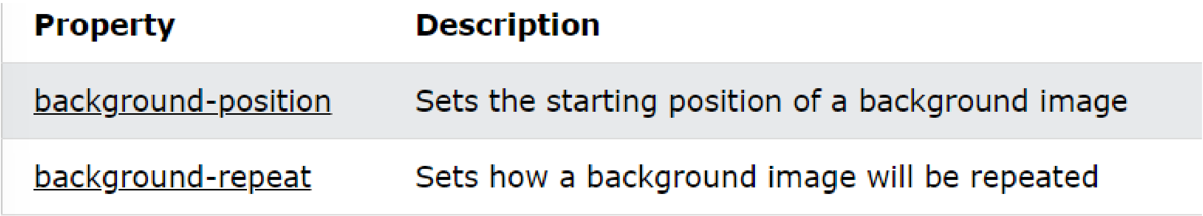
Comments

```
/* This is a single-line comment */
p {
  color: red;
}
```

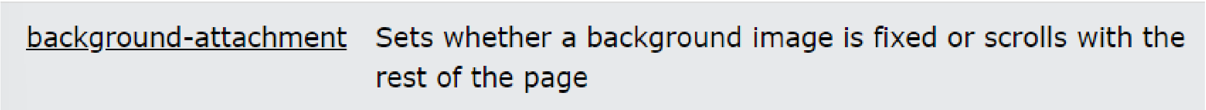
CSS Backgrounds

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

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
  background-attachment: scroll;
}
```



Property	Description
background-position	Sets the starting position of a background image
background-repeat	Sets how a background image will be repeated



background-attachment	Sets whether a background image is fixed or scrolls with the rest of the page

Shorthand

```
body {
  background: #ffffff url("img_tree.png") no-repeat right top;
}
```

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)

CSS Borders

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

The following values are allowed:

- `dotted` - Defines a dotted border
- `dashed` - Defines a dashed border
- `solid` - Defines a solid border
- `double` - Defines a double border
- `groove` - Defines a 3D grooved border. The effect depends on the border-color value
- `ridge` - Defines a 3D ridged border. The effect depends on the border-color value
- `inset` - Defines a 3D inset border. The effect depends on the border-color value
- `outset` - Defines a 3D outset border. The effect depends on the border-color value
- `none` - Defines no border
- `hidden` - Defines a hidden border

```
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
```

```
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
```

CSS Border Width

```
p.one {
  border-style: solid;
  border-width: 5px;

border-width: 25px 10px 4px 35px; /* 25px top, 10px right, 4px bottom and
35px left */
}
```

CSS Border Color

```
p.one {
  border-style: solid;

border-color: blue;
  border-color: red green blue yellow; /* red top, green right, blue
bottom and yellow left */
}
```

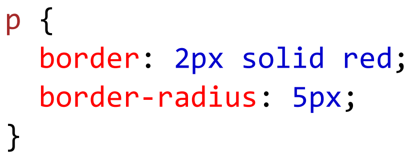
CSS Border - Individual Sides

```
p {
  border-top-style: dotted;
  border-right-style: solid;
  border-bottom-style: dotted;
  border-left-style: solid;
}
```

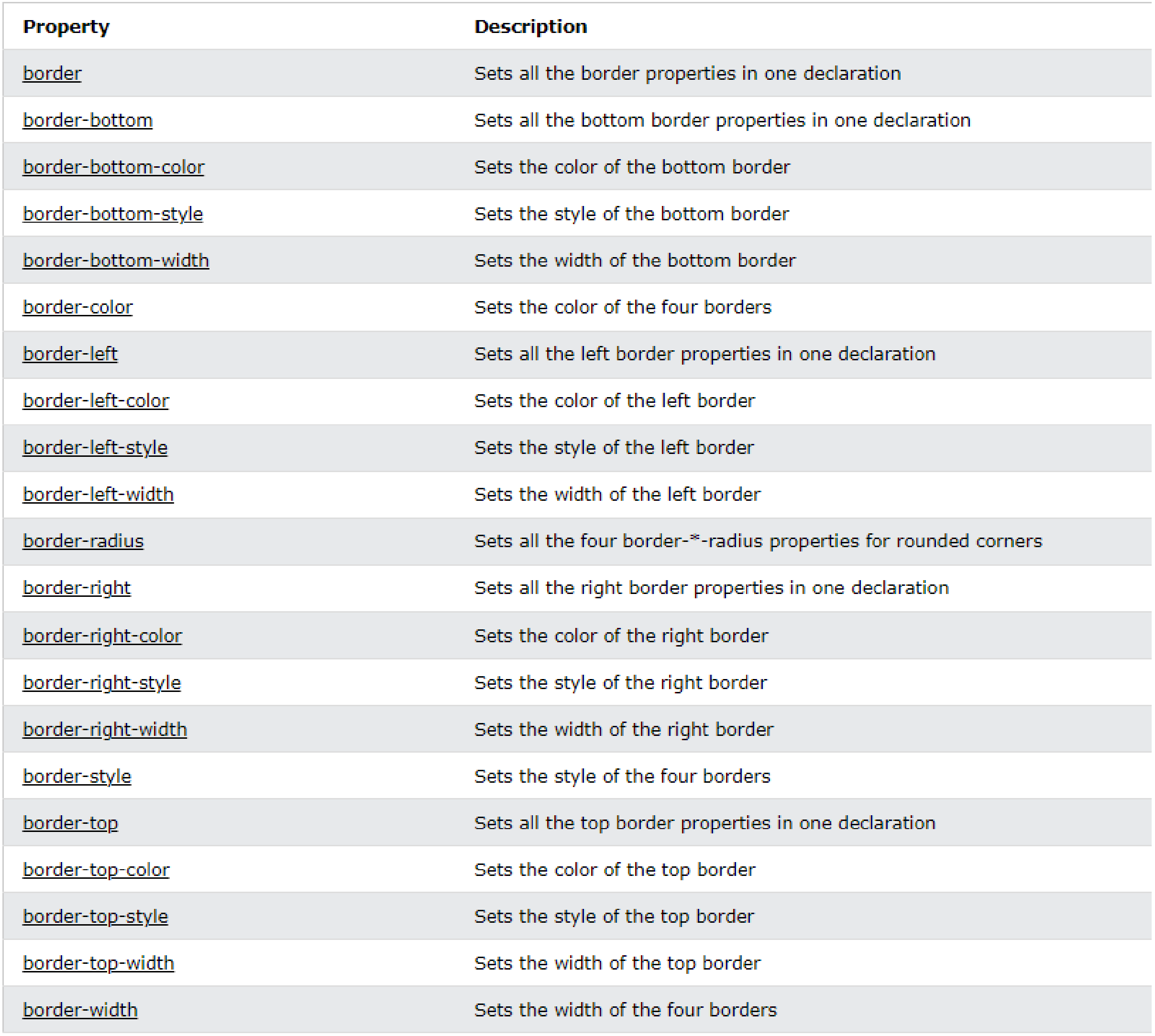
CSS Shorthand Border Property

```
p {
  border: 5px solid red;
}

p {
  border-bottom: 6px solid red;
}
```

CSS Rounded Borders

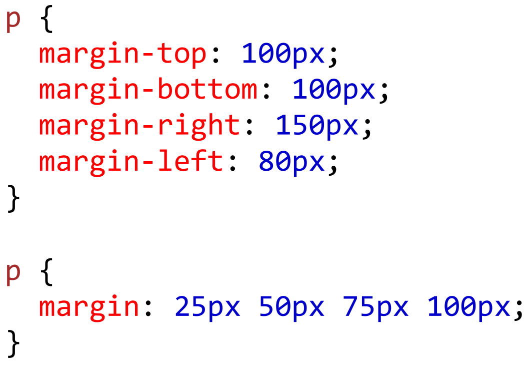


```
p {
  border: 2px solid red;
  border-radius: 5px;
}
```



Property	Description
border	Sets all the border properties in one declaration
border-bottom	Sets all the bottom border properties in one declaration
border-bottom-color	Sets the color of the bottom border
border-bottom-style	Sets the style of the bottom border
border-bottom-width	Sets the width of the bottom border
border-color	Sets the color of the four borders
border-left	Sets all the left border properties in one declaration
border-left-color	Sets the color of the left border
border-left-style	Sets the style of the left border
border-left-width	Sets the width of the left border
border-radius	Sets all the four border-*-radius properties for rounded corners
border-right	Sets all the right border properties in one declaration
border-right-color	Sets the color of the right border
border-right-style	Sets the style of the right border
border-right-width	Sets the width of the right border
border-style	Sets the style of the four borders
border-top	Sets all the top border properties in one declaration
border-top-color	Sets the color of the top border
border-top-style	Sets the style of the top border
border-top-width	Sets the width of the top border
border-width	Sets the width of the four borders

CSS Margins



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

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

- **margin: 25px;**

- all four margins are 25px

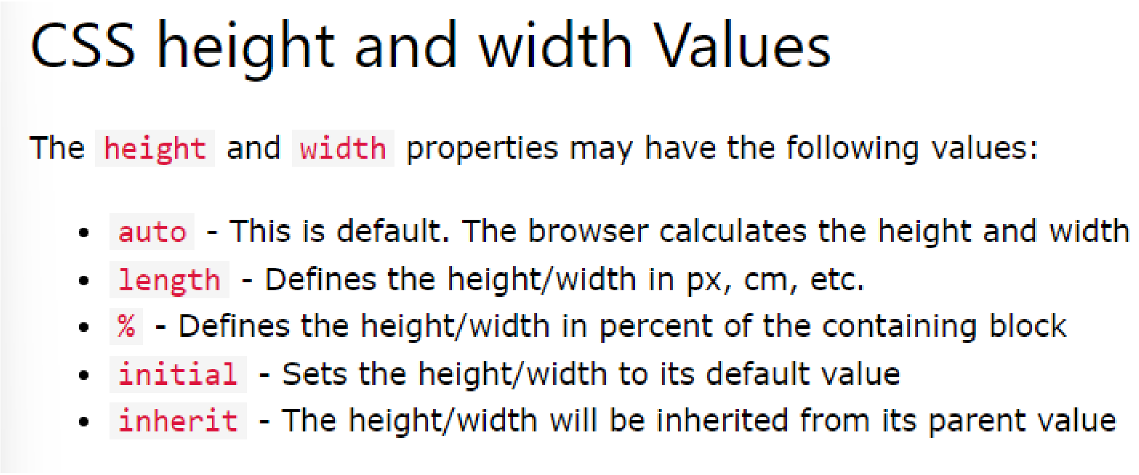
You can set the margin property to auto to horizontally center the element within its container.

```
div {
  width: 300px;
  margin: auto;
  border: 1px solid red;
}
```

CSS Padding

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

CSS Height, Width and Max-width



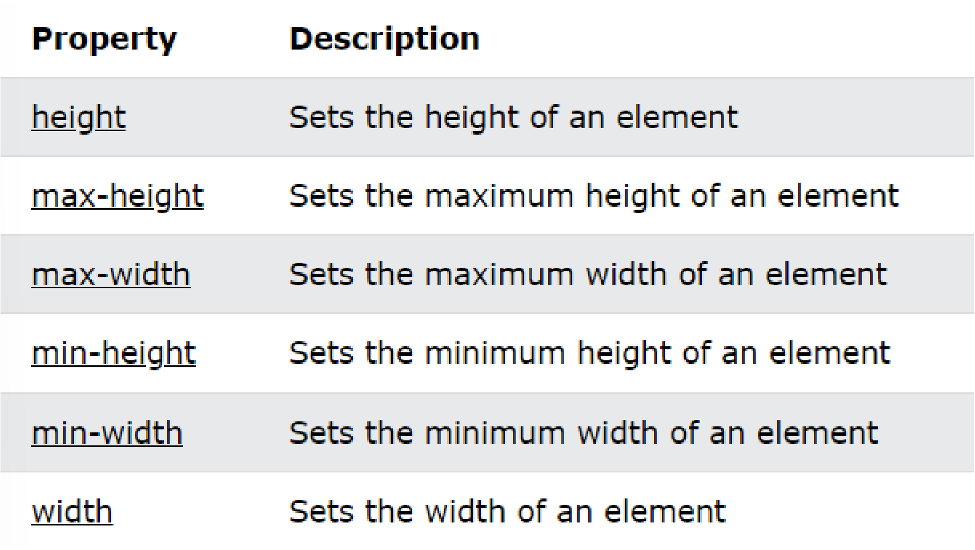
CSS height and width Values

The `height` and `width` properties may have the following values:

- `auto` - This is default. The browser calculates the height and width
- `length` - Defines the height/width in px, cm, etc.
- `%` - Defines the height/width in percent of the containing block
- `initial` - Sets the height/width to its default value
- `inherit` - The height/width will be inherited from its parent value

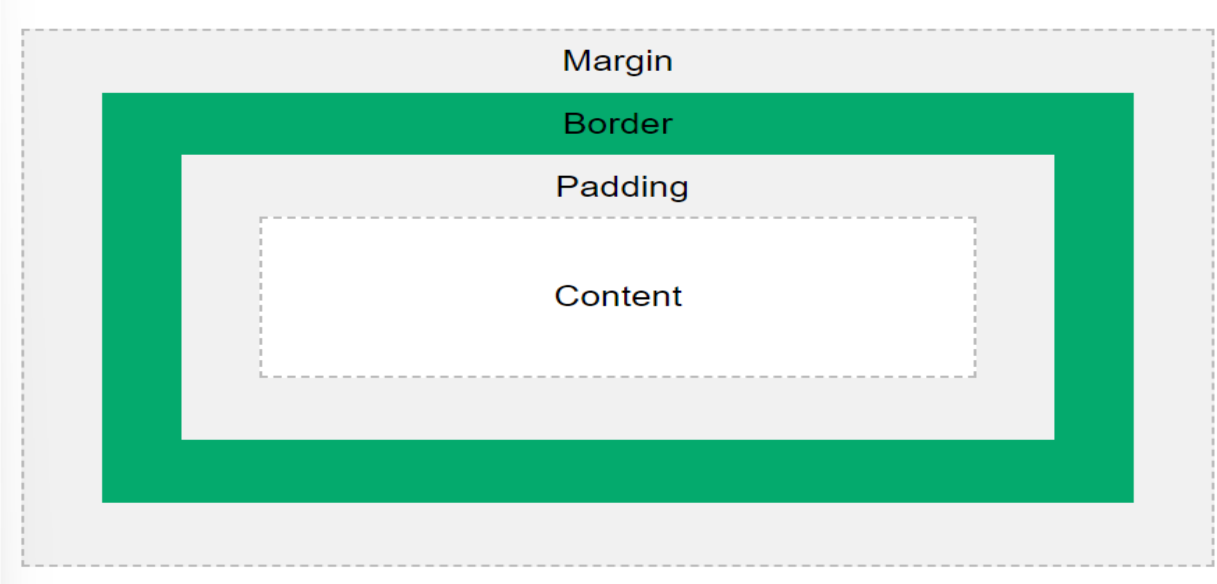
Setting max-width

```
div {
  max-width: 500px;
  height: 100px;
  background-color: powderblue;
}
```

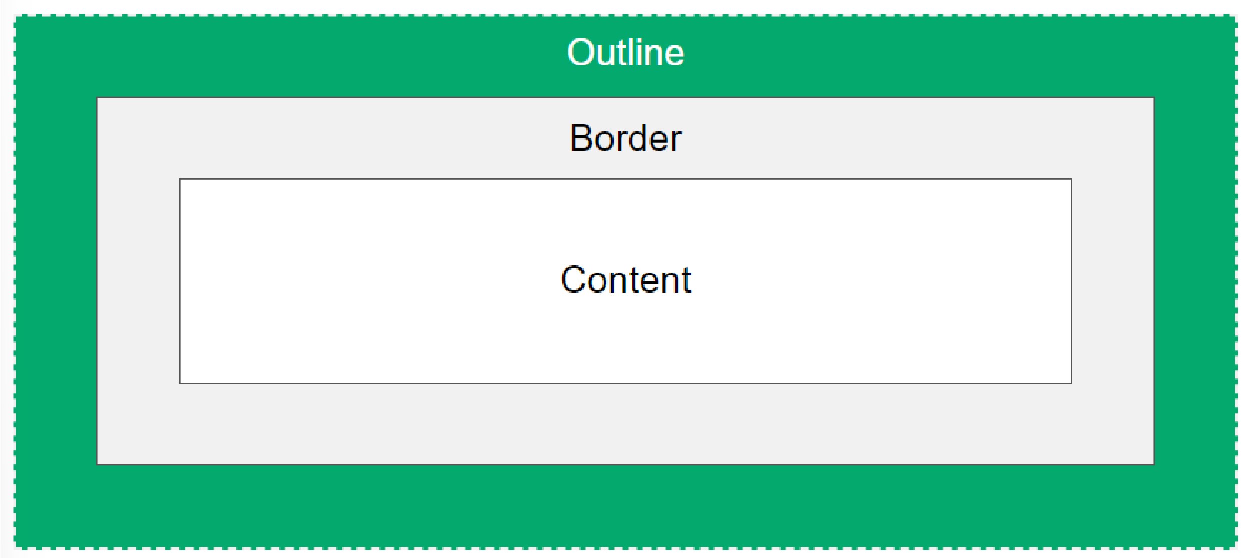


Property	Description
height	Sets the height of an element
max-height	Sets the maximum height of an element
max-width	Sets the maximum width of an element
min-height	Sets the minimum height of an element
min-width	Sets the minimum width of an element
width	Sets the width of an element

The CSS Box Model



CSS Outline



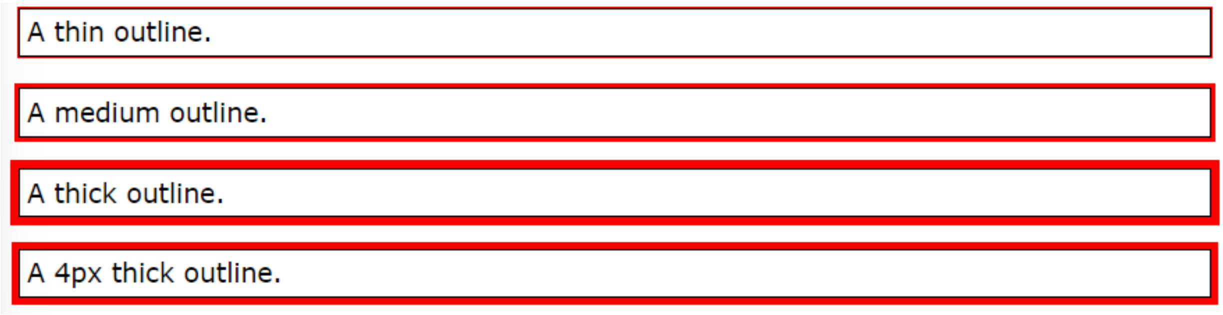
- outline-style
- outline-color
- outline-width
- outline-offset
- outline

```
p.dotted {outline-style: dotted;}
p.dashed {outline-style: dashed;}
p.solid {outline-style: solid;}
p.double {outline-style: double;}
p.groove {outline-style: groove;}
p.ridge {outline-style: ridge;}
p.inset {outline-style: inset;}
p.outset {outline-style: outset;}
```

CSS Outline Width

The `outline-width` property specifies the width of the outline, and can have one of the following values:

- thin (typically 1px)
- medium (typically 3px)
- thick (typically 5px)
- A specific size (in px, pt, cm, em, etc)



A thin outline.

A medium outline.

A thick outline.

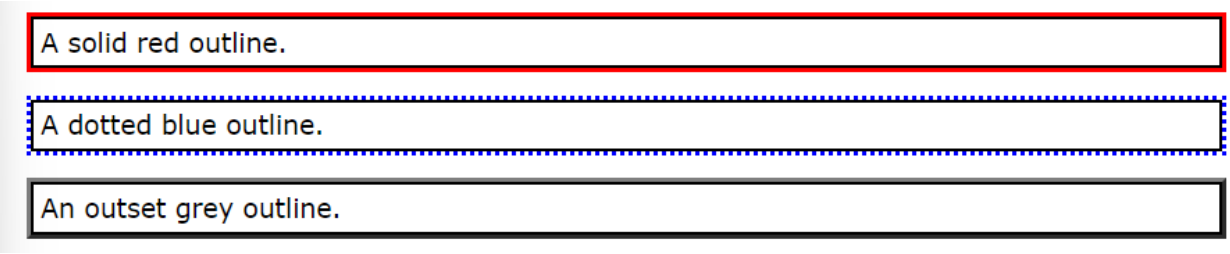
A 4px thick outline.

CSS Outline Color

The `outline-color` property is used to set the color of the outline.

The color can be set by:

- name - specify a color name, like "red"
- HEX - specify a hex value, like "#ff0000"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- HSL - specify a HSL value, like "hsl(0, 100%, 50%)"
- invert - performs a color inversion (which ensures that the outline is visible, regardless of color background)
- `outline-style: solid;`


A solid red outline.

A dotted blue outline.

An outset grey outline.

CSS Outline Shorthand

```
p.ex1 {outline: dashed;}
p.ex2 {outline: dotted red;}
p.ex3 {outline: 5px solid yellow;}
p.ex4 {outline: thick ridge pink;}
```

CSS Text

```
div {
  background-color: blue;
  color: white;
}
```

Property	Description
color	Specifies the color of text

Text Alignment and Text Direction

Property	Description
direction	Specifies the text direction/writing direction
text-align	Specifies the horizontal alignment of text
text-align-last	Specifies how to align the last line of a text
unicode-bidi	Used together with the direction property to set or return whether the text should be overridden to support multiple languages in the same document
vertical-align	Sets the vertical alignment of an element

```
p.a {
  text-align-last: right;
}
```

The direction and unicode-bidi properties can be used to change the text direction of an element:

```
p {
  direction: rtl;
  unicode-bidi: bidi-override;
}
```

The vertical-align property sets the vertical alignment of an element.

```
img.a {
  vertical-align: baseline;
}

img.b {
  vertical-align: text-top;
}

img.c {
  vertical-align: text-bottom;
}

img.d {
  vertical-align: sub;
}

img.e {
  vertical-align: super;
}
```

Text Decoration

```
h1 {
  text-decoration-line: overline;
}

h2 {
  text-decoration-line: line-through;
}

h3 {
  text-decoration-line: underline;
}

p {
  text-decoration-line: overline underline;
}
```

The text-decoration-color property is used to set the color of the decoration line.

```
text-decoration-color: blue;

 text-decoration-style: solid;
```

```
  text-decoration-thickness: auto;
```

shorthand

```
p {
  text-decoration: underline red double 5px;
}

a {
  text-decoration: none;
}
```

Property	Description
text-decoration	Sets all the text-decoration properties in one declaration
text-decoration-color	Specifies the color of the text-decoration
text-decoration-line	Specifies the kind of text decoration to be used (underline, overline, etc.)
text-decoration-style	Specifies the style of the text decoration (solid, dotted, etc.)
text-decoration-thickness	Specifies the thickness of the text decoration line

CSS Text Transformation

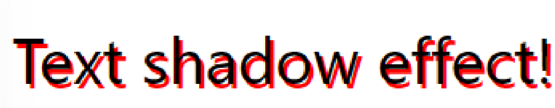
```
p.uppercase {
  text-transform: uppercase;
}

p.lowercase {
  text-transform: lowercase;
}

p.capitalize {
  text-transform: capitalize;
}
```

Text Spacing

Property	Description
letter-spacing	Specifies the space between characters in a text
line-height	Specifies the line height
text-indent	Specifies the indentation of the first line in a text-block
white-space	Specifies how to handle white-space inside an element
word-spacing	Specifies the space between words in a text

Text Shadow

```
h1 {
  text-shadow: 2px 2px;
}
```



Text shadow effect!

CSS Fonts

```
.p1 {
  font-family: "Times New Roman", Times, serif;
}
```

Font Style

```
p.normal {
  font-style: normal;
}
```

Font Weight

```
p.normal {
  font-weight: normal;
}
```

Font Size

```
h1 {
  font-size: 40px;
}
```

Property	Description
font	Sets all the font properties in one declaration
font-family	Specifies the font family for text
font-size	Specifies the font size of text
font-style	Specifies the font style for text
font-variant	Specifies whether or not a text should be displayed in a small-caps font
font-weight	Specifies the weight of a font

CSS Icons

```
<!DOCTYPE html>
<html>
<head>
<script src="https://kit.fontawesome.com/a076d05399.js" crossorigin="anonymo
us"></script>
</head>
<body>

<i class="fas fa-cloud"></i>
<i class="fas fa-heart"></i>
<i class="fas fa-car"></i>
<i class="fas fa-file"></i>
<i class="fas fa-bars"></i>

</body>
</html>
```

CSS Links

- `a:link` - a normal, unvisited link
- `a:visited` - a link the user has visited
- `a:hover` - a link when the user mouses over it
- `a:active` - a link the moment it is clicked
-

-
  ```
  /* unvisited link */
  a:link {
    color: red;
  }

  /* visited link */
  a:visited {
    color: green;
  }

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

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

HTML Lists and CSS List Properties

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

```
ul.a {
  list-style-type: circle;
}
```

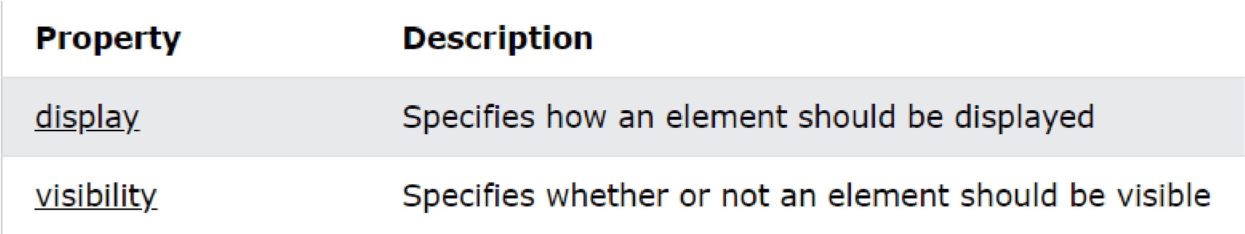
CSS Tables

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

Property	Description
border	Sets all the border properties in one declaration
border-collapse	Specifies whether or not table borders should be collapsed
border-spacing	Specifies the distance between the borders of adjacent cells
caption-side	Specifies the placement of a table caption
empty-cells	Specifies whether or not to display borders and background on empty cells in a table
table-layout	Sets the layout algorithm to be used for a table

CSS Layout - The display Property

```
h1.hidden {
  display: none;

display: block;
display: inline;
}
```



Property	Description
display	Specifies how an element should be displayed
visibility	Specifies whether or not an element should be visible

```
h1.hidden {
  visibility: hidden;
}
```

However, the element will still take up the same space as before. The element will be hidden, but still affect the layout:

CSS Layout - width and max-width

```
div.ex2 {
  max-width: 500px;
  margin: auto;
  border: 3px solid #73AD21;
}
```

CSS Layout - The position Property

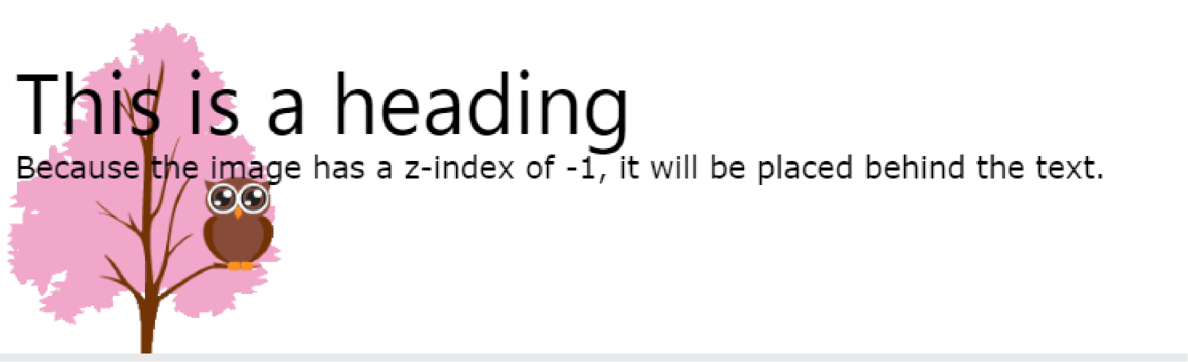
The `position` property specifies the type of positioning method used for an element.

There are five different position values:

- `static`
- `relative`
- `fixed`
- `absolute`
- `sticky`

Property	Description
bottom	Sets the bottom margin edge for a positioned box
clip	Clips an absolutely positioned element
left	Sets the left margin edge for a positioned box
position	Specifies the type of positioning for an element
right	Sets the right margin edge for a positioned box
top	Sets the top margin edge for a positioned box

The z-index Property



This is a heading

Because the image has a z-index of -1, it will be placed behind the text.

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

Property	Description
z-index	Sets the stack order of an element

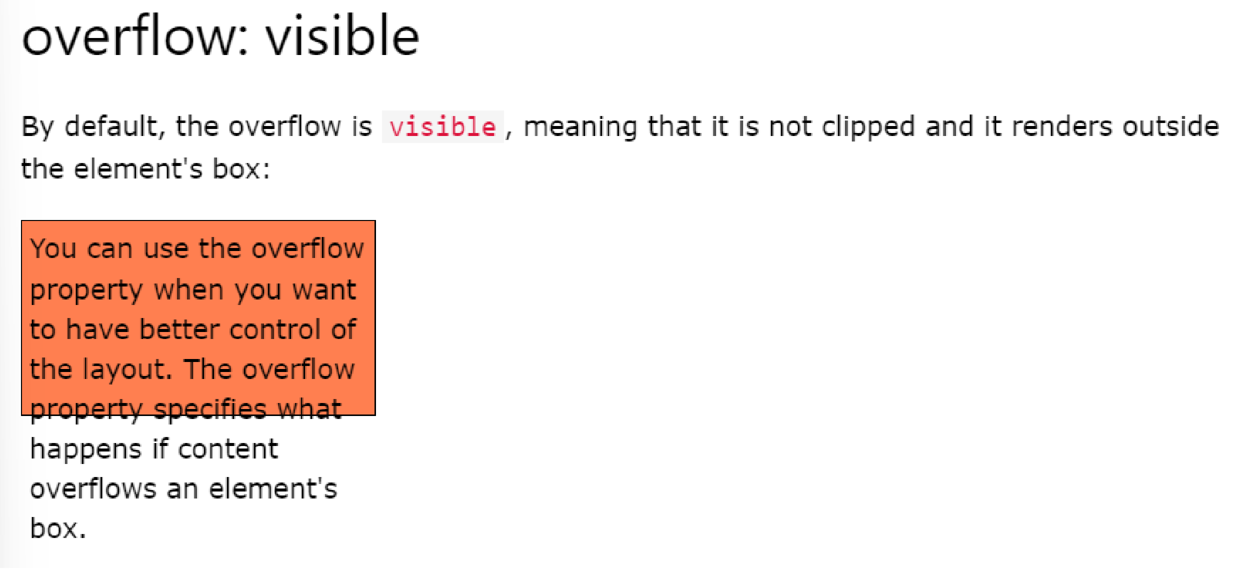
CSS Layout - Overflow

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

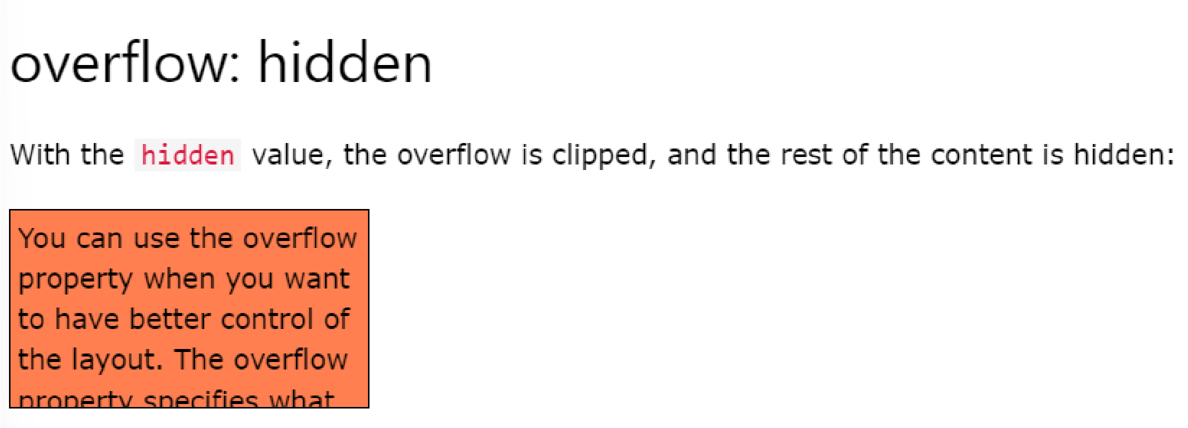
- `scroll` - The overflow is clipped, and a scrollbar is added to see the rest of the content
- `auto` - Similar to `scroll`, but it adds scrollbars only when necessary



overflow: visible

By default, the overflow is `visible`, meaning that it is not clipped and it renders outside the element's box:

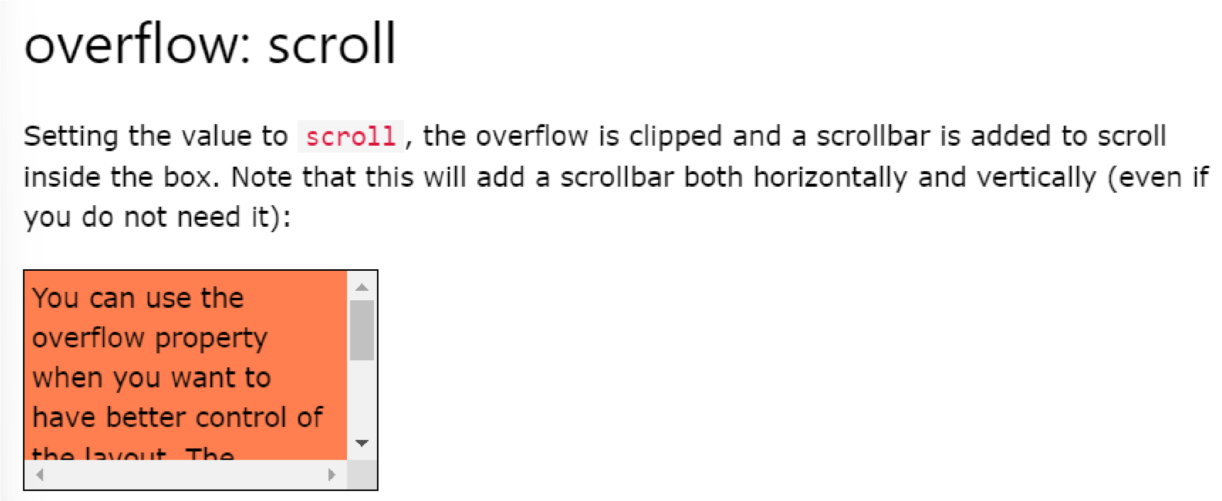
You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.



overflow: hidden

With the `hidden` value, the overflow is clipped, and the rest of the content is hidden:

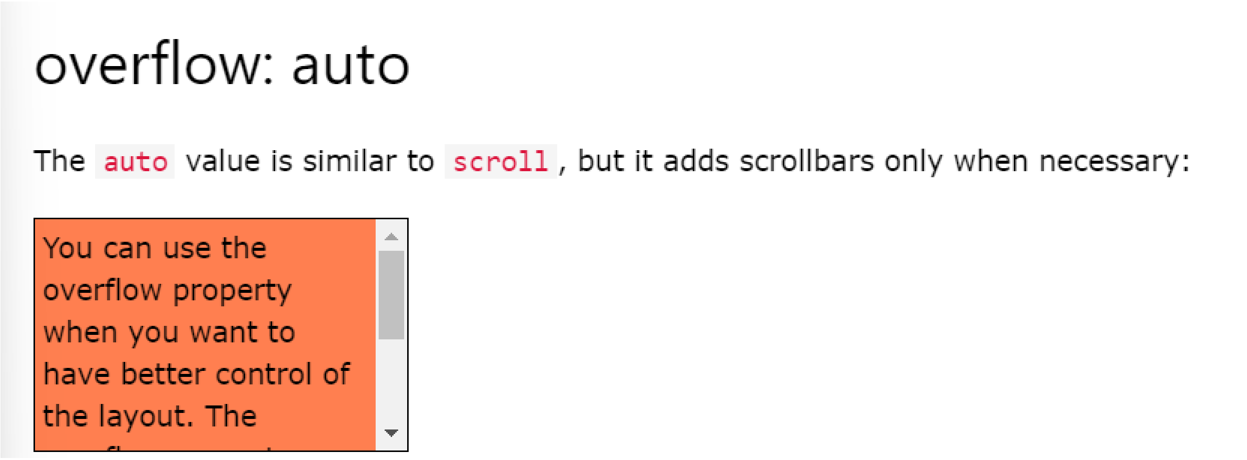
You can use the overflow property when you want to have better control of the layout. The overflow property specifies what



overflow: scroll

Setting the value to `scroll`, the overflow is clipped and a scrollbar is added to scroll inside the box. Note that this will add a scrollbar both horizontally and vertically (even if you do not need it):

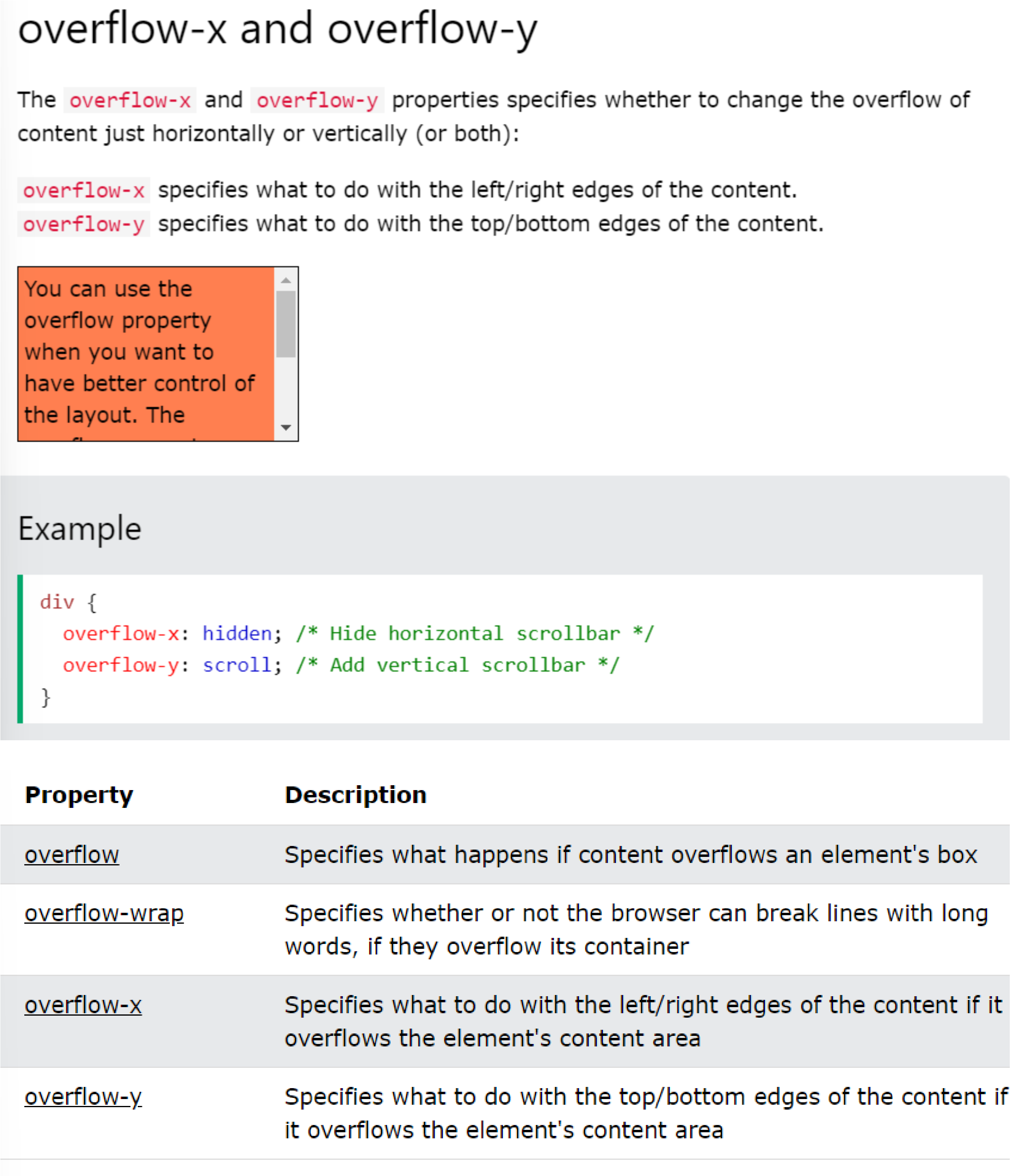
You can use the overflow property when you want to have better control of the layout. The



overflow: auto

The `auto` value is similar to `scroll`, but it adds scrollbars only when necessary:

You can use the overflow property when you want to have better control of the layout. The



overflow-x and overflow-y

The `overflow-x` and `overflow-y` properties specifies whether to change the overflow of content just horizontally or vertically (or both):

`overflow-x` specifies what to do with the left/right edges of the content.
`overflow-y` specifies what to do with the top/bottom edges of the content.

You can use the overflow property when you want to have better control of the layout. The

Example

```
div {
  overflow-x: hidden; /* Hide horizontal scrollbar */
  overflow-y: scroll; /* Add vertical scrollbar */
}
```

Property	Description
overflow	Specifies what happens if content overflows an element's box
overflow-wrap	Specifies whether or not the browser can break lines with long words, if they overflow its container
overflow-x	Specifies what to do with the left/right edges of the content if it overflows the element's content area
overflow-y	Specifies what to do with the top/bottom edges of the content if it overflows the element's content area

CSS Layout - float and clear

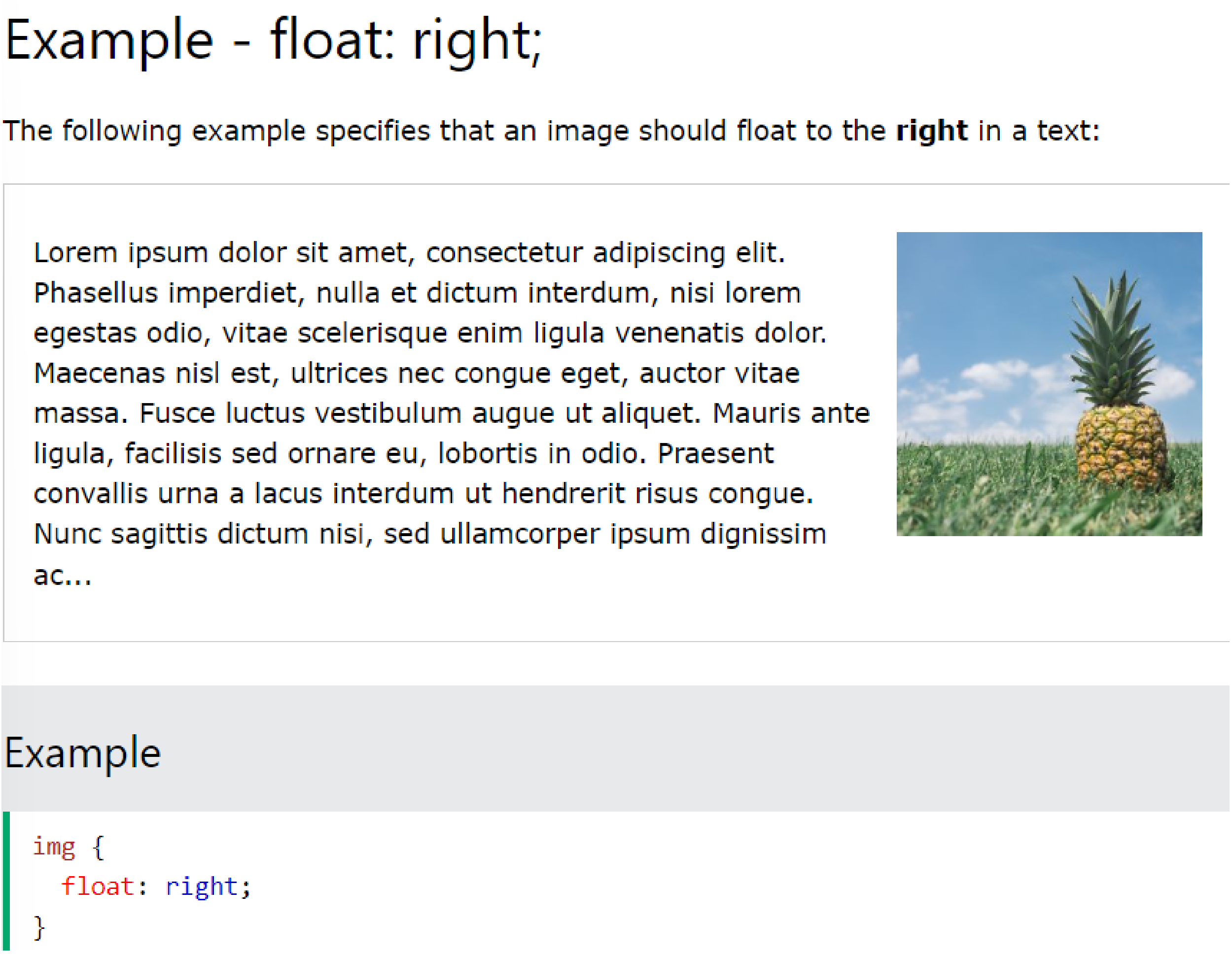
The CSS `float` property specifies how an element should float.

The CSS `clear` property specifies what elements can float beside the cleared element and on which side.

The `float` property is used for positioning and formatting content e.g. let an image float left to the text in a container.

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



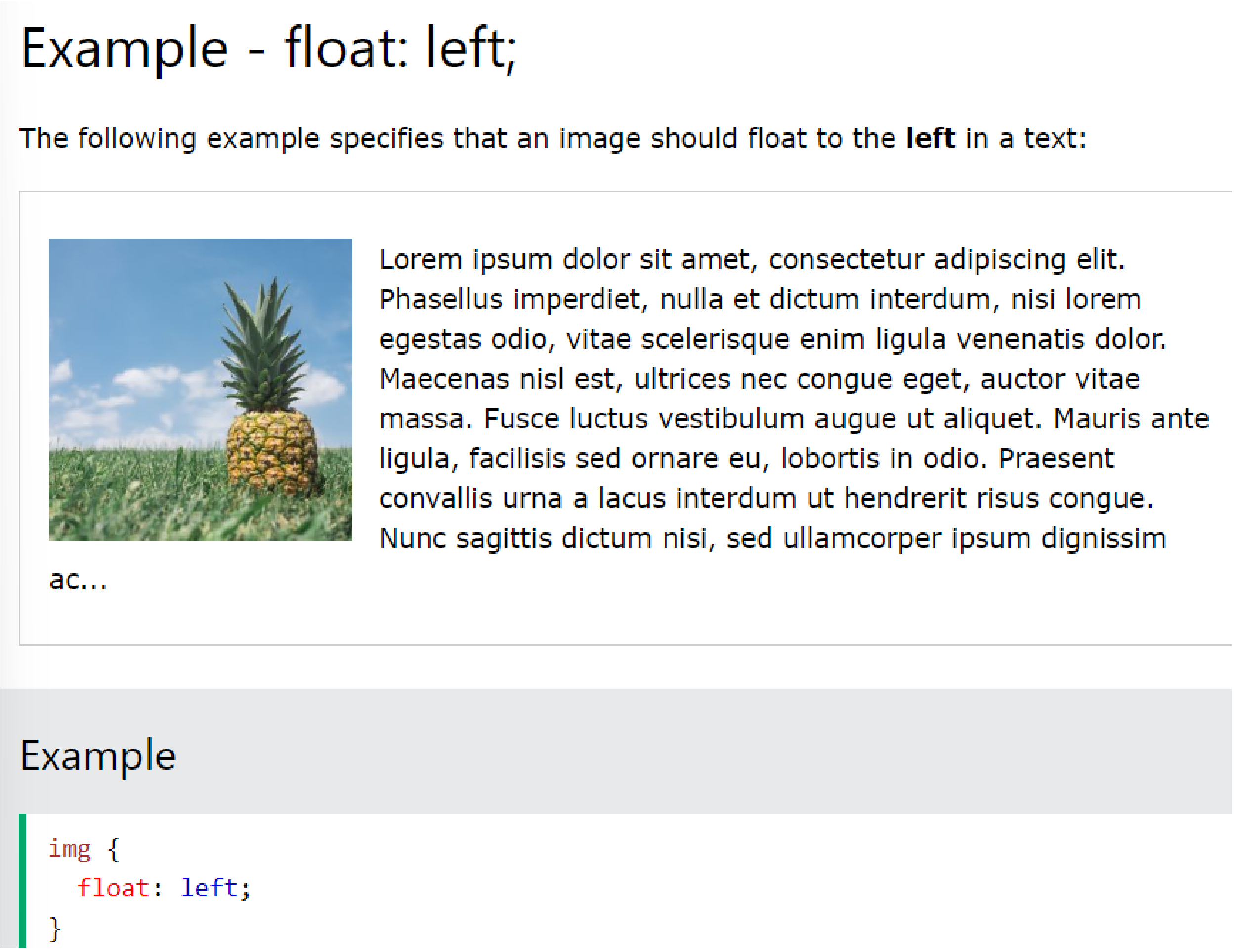
Example - float: right;

The following example specifies that an image should float to the **right** in a text:

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...

Example

```
img {
  float: right;
}
```



Example - float: left;

The following example specifies that an image should float to the **left** in a text:

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...

Example

```
img {
  float: left;
}
```

CSS Layout - clear and clearfix

When we use the `float` property, and we want the next element below (not on right or left), we will have to use the `clear` property.

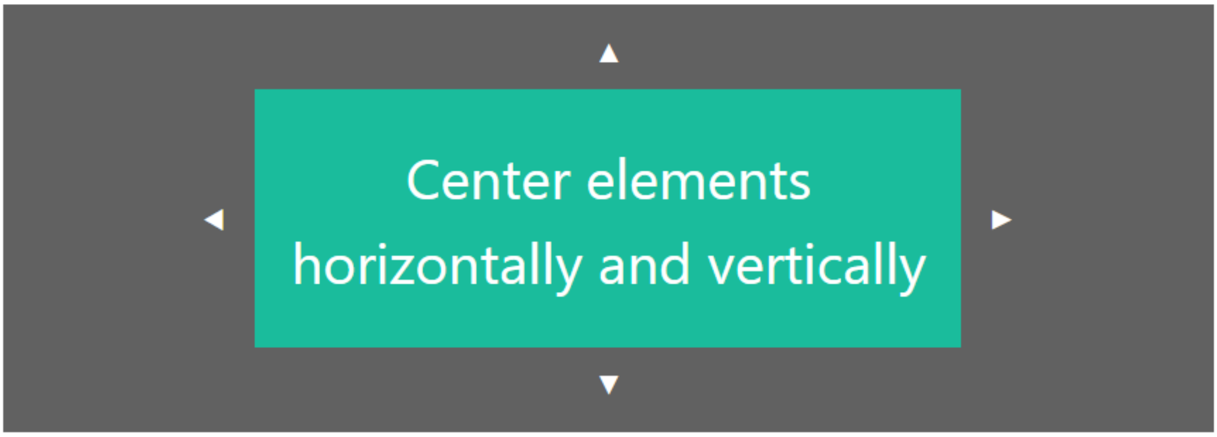
The `clear` property specifies what should happen with the element that is next to a floating element.

The `clear` property can have one of the following values:

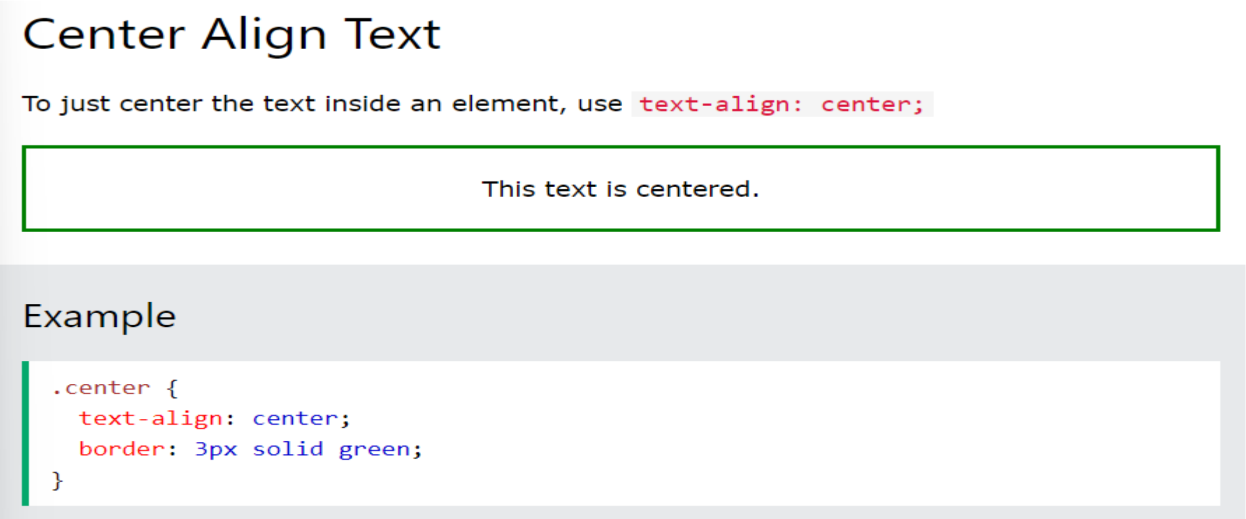
- `none` - The element is not pushed below left or right floated elements. This is default
- `left` - The element is pushed below left floated elements
- `right` - The element is pushed below right floated elements
- `both` - The element is pushed below both left and right floated elements
- `inherit` - The element inherits the clear value from its parent

The display: inline-block Value

```
span.a {
  display: inline; /* the default for span */
  width: 100px;
  height: 100px;
  padding: 5px;
  border: 1px solid blue;
  background-color: yellow;
}

span.b {
  display: inline-block;
  width: 100px;
  height: 100px;
  padding: 5px;
  border: 1px solid blue;
  background-color: yellow;
}

span.c {
  display: block;
  width: 100px;
  height: 100px;
  padding: 5px;
  border: 1px solid blue;
  background-color: yellow;
}
```



```
.center {
  margin: auto;
  width: 50%;
  border: 3px solid green;
  padding: 10px;
}
```



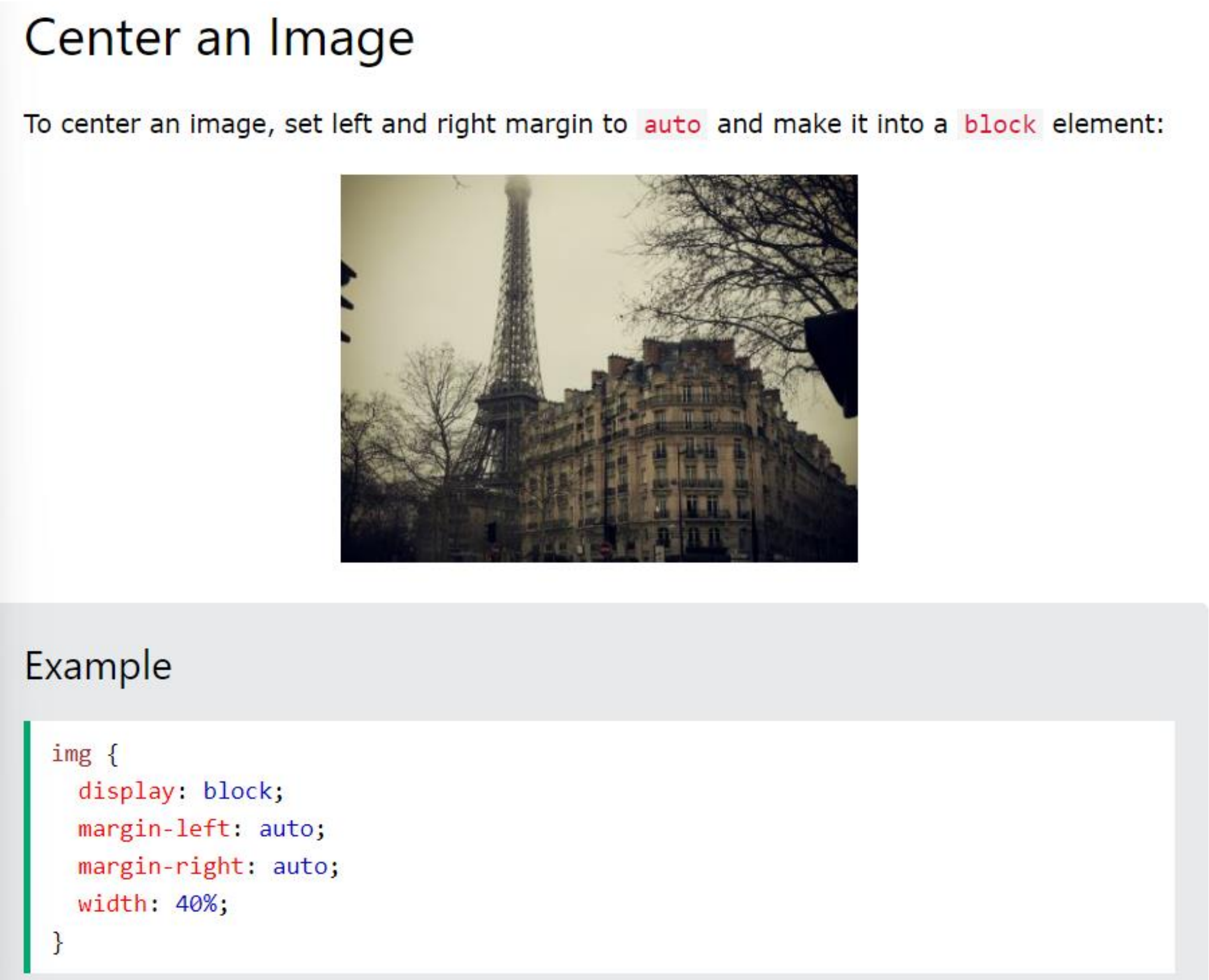
Center Align Text

To just center the text inside an element, use `text-align: center;`

This text is centered.

Example

```
.center {
  text-align: center;
  border: 3px solid green;
}
```



Center an Image

To center an image, set left and right margin to `auto` and make it into a `block` element:

Example

```
img {
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 40%;
}
```

CSS Combinators

There are four different combinators in CSS:

- descendant selector (space)

  ```
  div p {
    background-color: yellow;
  }
  ```

- child selector (>)

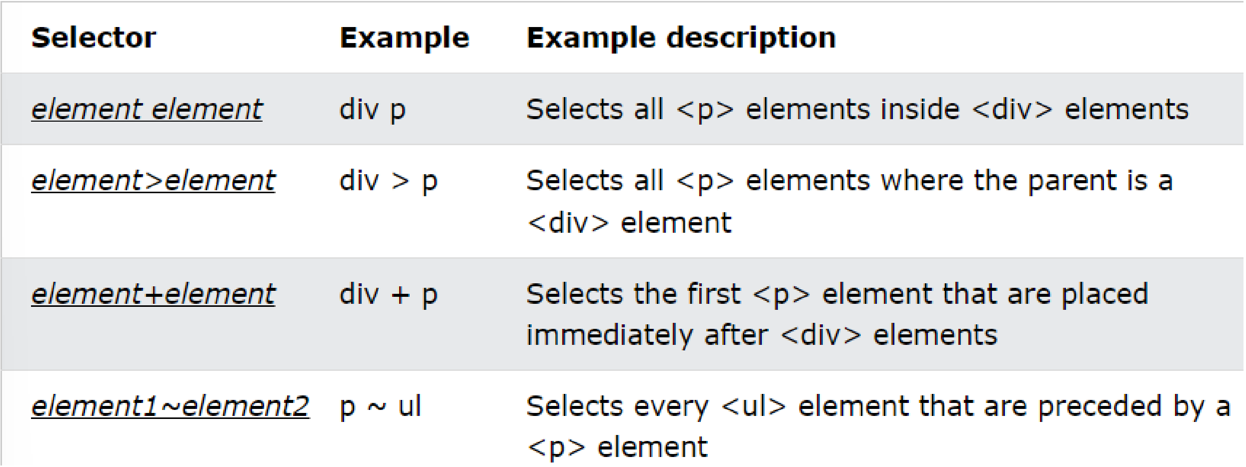
  ```
  div > p {
    background-color: yellow;
  }
  ```

- adjacent sibling selector (+)

  ```
  div + p {
    background-color: yellow;
  }
  ```

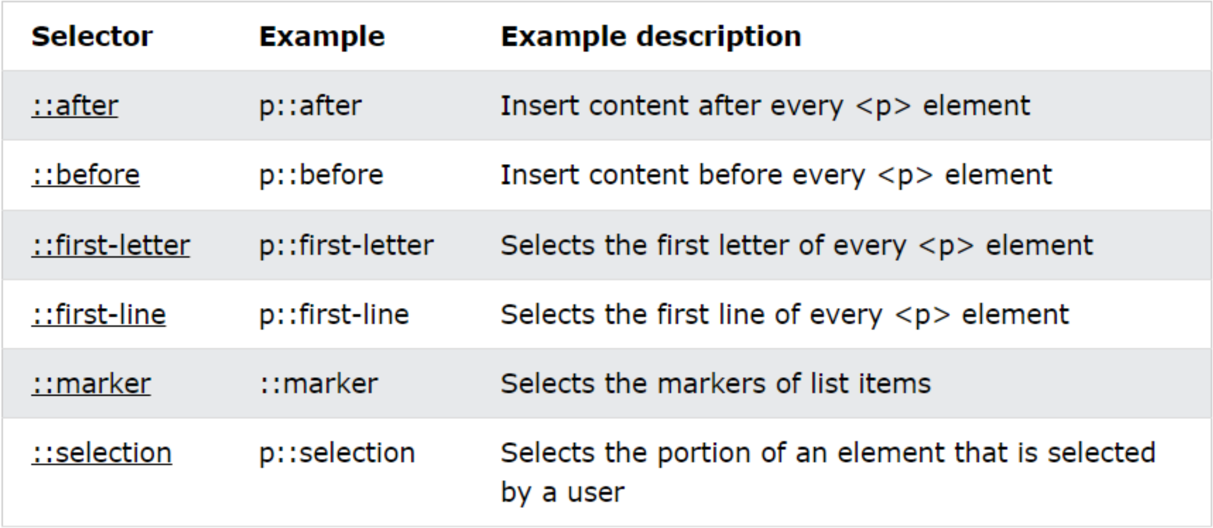
- general sibling selector (~)

```
div ~ p {
  background-color: yellow;
}
```



Selector	Example	Example description
element element	div p	Selects all <p> elements inside <div> elements
element>element	div > p	Selects all <p> elements where the parent is a <div> element
element+element	div + p	Selects the first <p> element that are placed immediately after <div> elements
element1~element2	p ~ ul	Selects every <ul> element that are preceded by a <p> element

All CSS Pseudo Elements

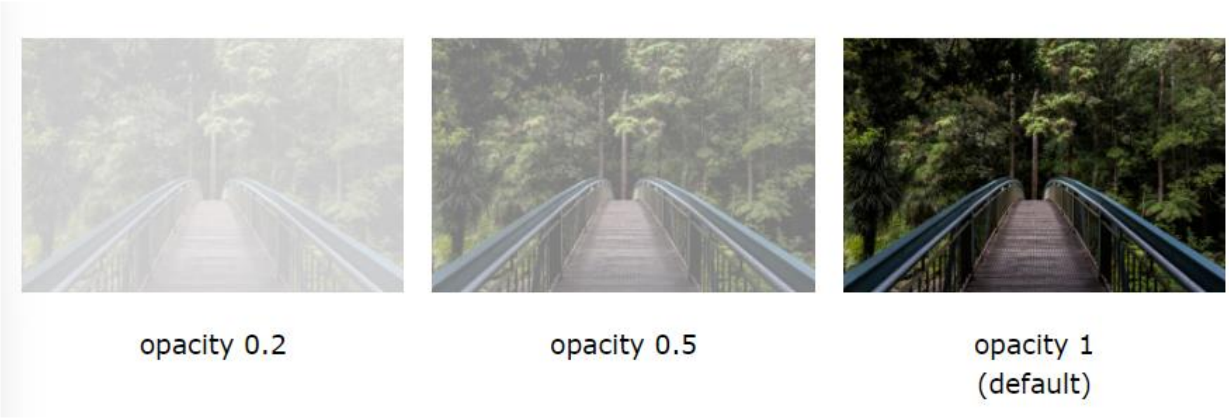


Selector	Example	Example description
::after	p::after	Insert content after every <p> element
::before	p::before	Insert content before every <p> element
::first-letter	p::first-letter	Selects the first letter of every <p> element
::first-line	p::first-line	Selects the first line of every <p> element
::marker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

CSS Opacity / Transparency

Transparent Image

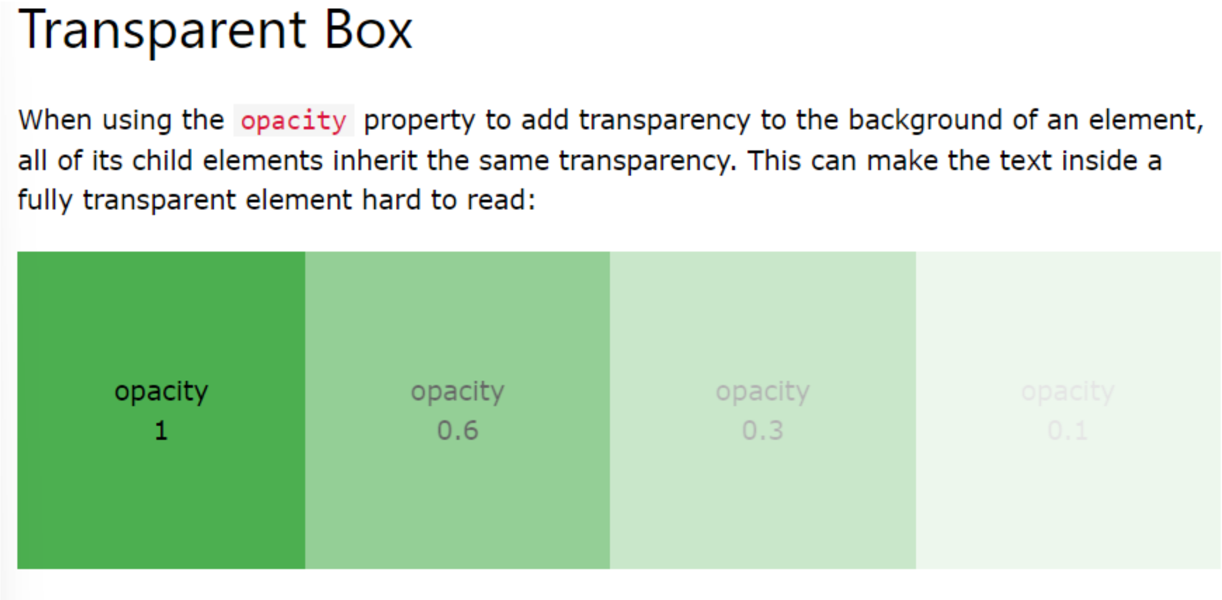
```
img {
  opacity: 0.5;
}
```



opacity 0.2

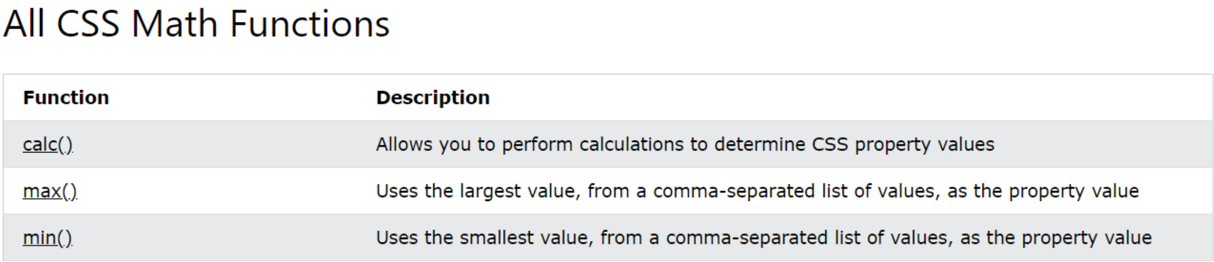
opacity 0.5

opacity 1 (default)



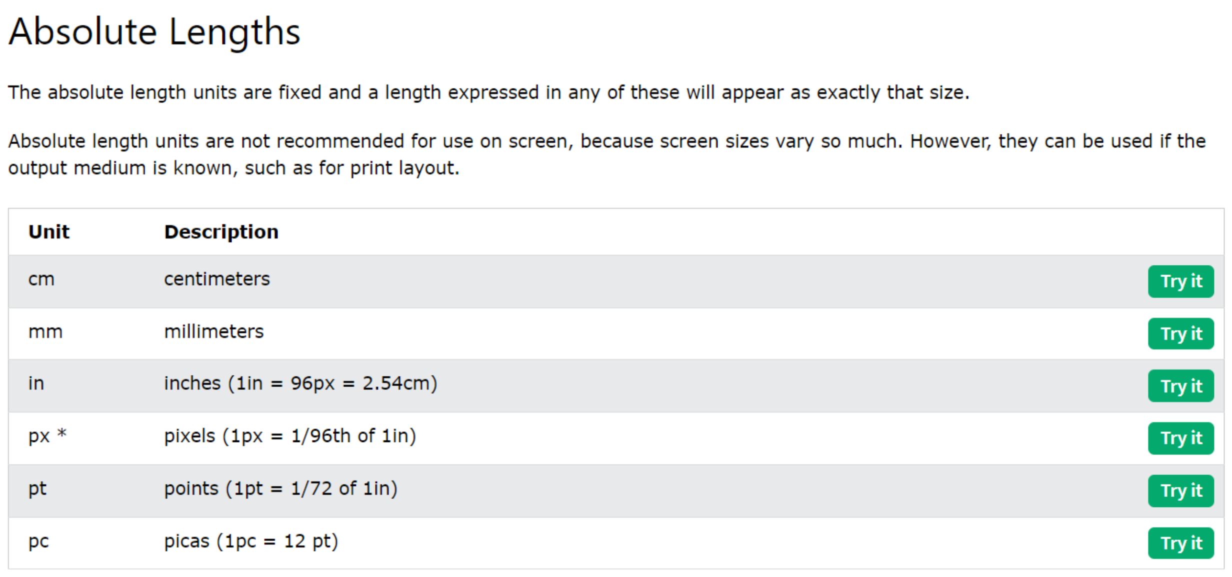
Transparent Box

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



All CSS Math Functions

Function	Description
calc()	Allows you to perform calculations to determine CSS property values
max()	Uses the largest value, from a comma-separated list of values, as the property value
min()	Uses the smallest value, from a comma-separated list of values, as the property value

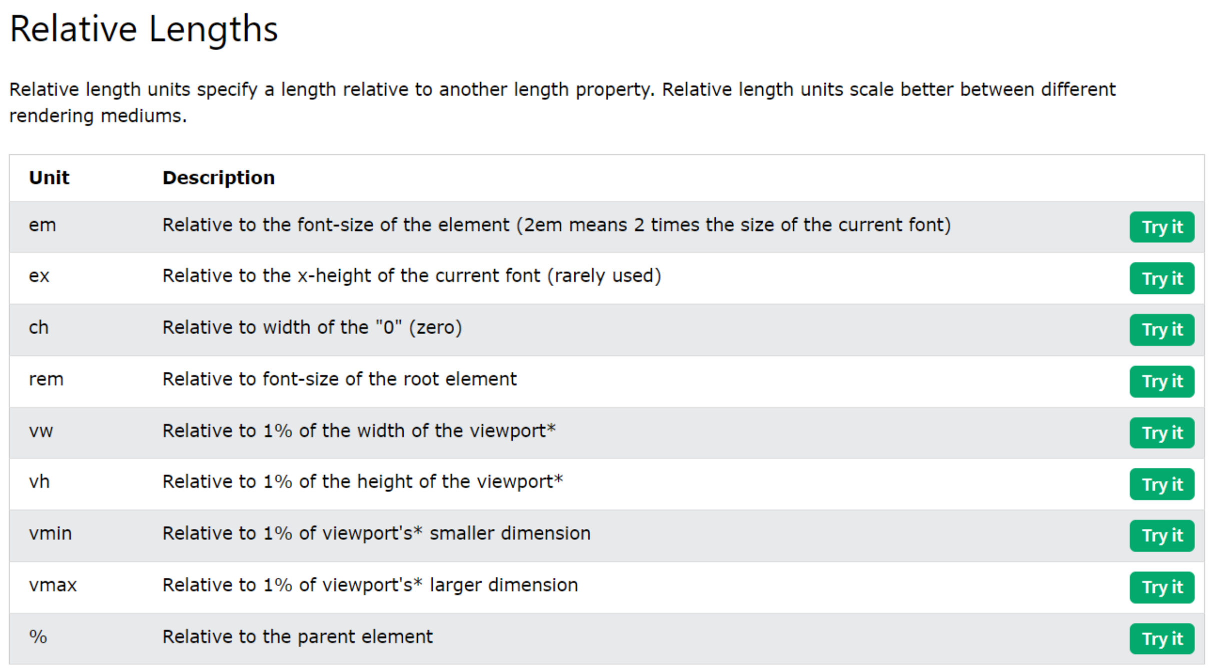


Absolute Lengths

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

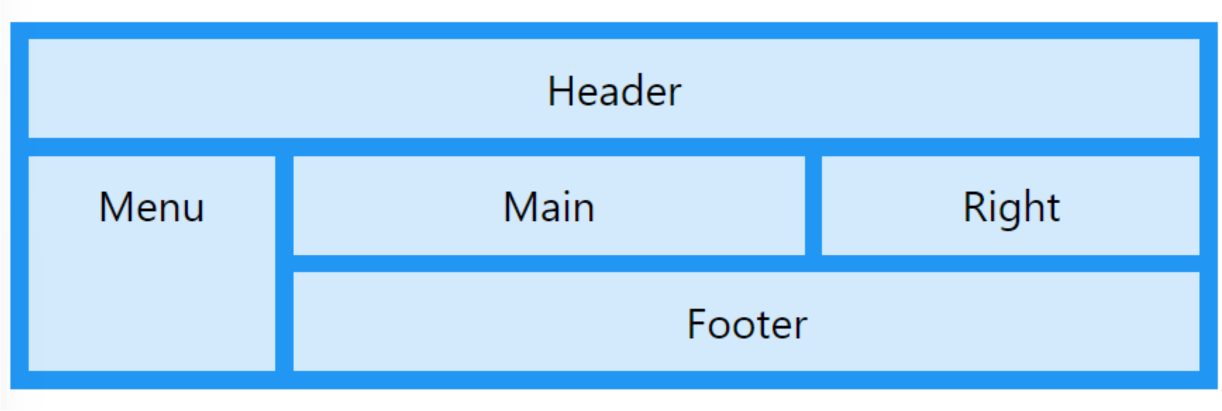


Relative Lengths

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

Unit	Description	
em	Relative to the font-size of the element (2em means 2 times the size of the current font)	Try it
ex	Relative to the x-height of the current font (rarely used)	Try it
ch	Relative to width of the "0" (zero)	Try it
rem	Relative to font-size of the root element	Try it
vw	Relative to 1% of the width of the viewport*	Try it
vh	Relative to 1% of the height of the viewport*	Try it
vmin	Relative to 1% of viewport's* smaller dimension	Try it
vmax	Relative to 1% of viewport's* larger dimension	Try it
%	Relative to the parent element	Try it

CSS Grid Layout Module


Header
Menu
Main
Right
Footer