

CSS layout cheat sheet

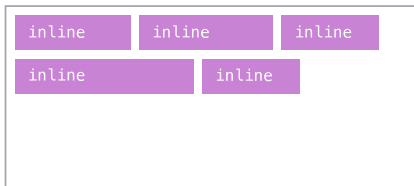
[Layout mechanics](#) · [Centering elements](#) · [Common code](#)

Layout mechanics

Display

Controls how an element is represented within the flow.

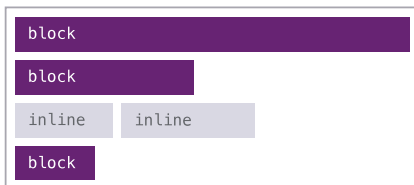
`display: inline`



✗ padding ✗ margin ✗ width

Allows other elements beside; margin, padding & width don't work.

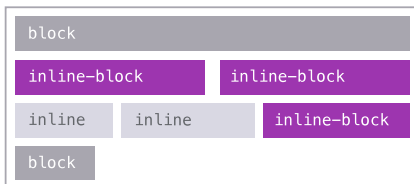
`display: block`



✓ padding ✓ margin ✓ width

Takes up an entire line; margin, padding & width work.

`display: inline-block`



✓ padding ✓ margin ✓ width

Allows other elements beside; margin, padding & width work. Can create columns, but will force a space between boxes.

Float

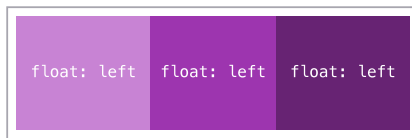
Controls whether text is wrapped around the element.

`float: left|right|none`



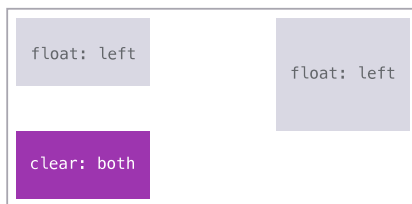
Allows other elements to wrap around the element.

Multiple floats



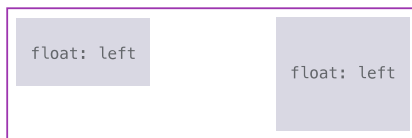
Can create columns with boxes touching sides.

`clear: left|right|both`



Force the element below floated elements.

`overflow: hidden`



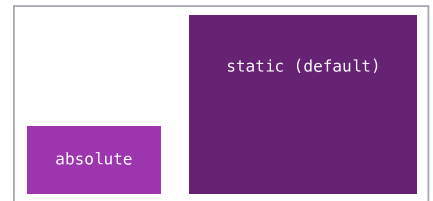
overflow: hidden

Use on a parent element to force it to wrap around the floated children—a clearfix.

Position

Gives strict, coordinate-based control over layout.

`position: absolute`



Move an element around based on coordinates.

`position: relative`



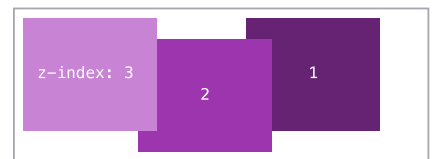
Added to a parent element to reset absolute child's coordinates.

`position: fixed`



Forces an element to not move when the page is scrolled.

`z-index`



Control the stacking order of elements—higher number is closer.

Centering elements

text-align: center

Works only on `display: inline` & `inline-block` elements.

Must be applied to the parent element.

```
<figure class="img-box">
  The mighty Argentinosauros</figcaption>
</figure>

.img-box {
  text-align: center;
}
```

margin: 0 auto

Works only on `display: block` elements.

The element must have a `width`

```
<div class="box">Stegosaurus</div>

.box {
  width: 24em; /* Without a width `auto` won't work */
  margin-left: auto;
  margin-right: auto;
}
```

You can also specify just `margin-left: auto` and `margin-right: auto` if you want margins on the top or bottom.

vertical-align: middle

Works only on `display: inline` & `inline-block` elements.

```
<ul>
  <li>Pteranodon</li>
  <li>Quetzalcoatlus</li>
</ul>

ul li {
  display: inline-block;
  vertical-align: middle;
}
```

Centering absolute

Use `transform` & `50%` coordinates to center an absolutely positioned element.

```
<div class="banner">
  <div class="content">
    <h1>Micropachycephalosaurus</h1>
    <p>Longest dinosaur name ever!</p>
  </div>
</div>

.banner {
  position: relative;
}

.content {
  position: absolute;
  left: 50%;
  transform: translateX(-50%);
}
```

Or vertical centering too...

```
.content {
  position: absolute;
  left: 50%;
  top: 50%;
  transform: translate(-50%, -50%);
}
```

Centering with float

There's no `float: center`

You cannot center floated elements.

Centering with flexbox

Flex box has a bunch of different alignment classes—that are always applied to the parent.

```
<div class="card">
  <h2>Edmontosaurus</h2>
  <a href="#">See the bones!</a>
</div>

.card {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-content: center;
  align-items: center;
}
```

This will be completely centered within the box.

[See the flexbox cheat sheet for more details.](#)

Common code

Border box

Used to change layout math for width & padding.

Put at the top of every CSS file.

```
html {
  box-sizing: border-box;
}

*, *::before, *::after {
  box-sizing: inherit;
}
```

Clearfix for float

Add to the parent elements of floats to force the parent to surround the floated element.

Can be used instead of `overflow: hidden`

```
.clearfix::after {
  content: " ";
  display: block;
  clear: both;
}
```

Flexible images

Use `width` & `display` to make images flex to their parent's size.

```
img {
  display: block;
  width: 100%;
}
```

[← CSS grid cheat sheet](#)

[≡ Topics](#)

[Navigation →](#)

Cheat sheets & checklists

Want a quick review with just a list of all the cheat sheets, checklists & flowcharts on the website? Here it is!

[See all the cheat sheets & checklists](#)

Help Learn the Web be the best it can! [Send anonymous feedback.](#)

© Thomas J Bradley—[learntheweb.courses](#).

All code for this website licensed under the [MIT License](#).

All tutorial code licensed under the [CCo Public Domain Dedication](#).

All text, imagery, design, tutorials & videos licensed under the [CC BY-NC-SA License](#).

Unless otherwise noted.



[Privacy & cookies policy](#)

Learn the Web, Ottawa, ON, Canada

 [Get help](#)

 [Solutions for teachers](#)

 [Funding](#)

 [GitHub](#)  [Bugs](#)

Last updated — [2018-10-22T13:58:46-04:00](#)