/M mdn web docs _

clear

The `clear` [CSS](#) property sets whether an element must be moved below (cleared) [floating](#) elements that precede it. The `clear` property applies to floating and non-floating elements.

Try it

CSS Demo: clear   RESET

```
clear: none;
```

```
clear: left;
```

```
clear: right;
```

```
clear: both;
```

When applied to non-floating blocks, it moves the [border edge](#) of the element down until it is below the [margin edge](#) of all relevant floats. The non-floated block's top margin collapses.

Vertical margins between two floated elements on the other hand will not collapse. When applied to floating elements, the margin edge of the bottom element is moved below the margin edge of all relevant floats. This affects the position of later floats, since later floats cannot be positioned higher than earlier ones.

The floats that are relevant to be cleared are the earlier floats within the same [block formatting context](#).

> **Note:** If an element contains only floated elements, its height collapses to nothing.
> If you want it to always be able to resize, so that it contains floating elements
> inside it, set the value of the element's `display` property to `flow-root` .
>
>   CSS
>
> ```css
> #container {
>   display: flow-root;
> }
> ```

Syntax

CSS

```css
/* Keyword values */
clear: none;
clear: left;
clear: right;
clear: both;
clear: inline-start;
clear: inline-end;

/* Global values */
clear: inherit;
clear: initial;
clear: revert;
clear: revert-layer;
clear: unset;
```

Values

`none`

Is a keyword indicating that the element is *not* moved down to clear past floating
elements.

`left`

Is a keyword indicating that the element is moved down to clear past *left* floats.

`right`

Is a keyword indicating that the element is moved down to clear past *right* floats.

`both`

Is a keyword indicating that the element is moved down to clear past *both* left and right floats.

`inline-start`

Is a keyword indicating that the element is moved down to clear floats on *start side of its containing block*, that is the *left* floats on ltr scripts and the *right* floats on rtl scripts.

`inline-end`

Is a keyword indicating that the element is moved down to clear floats on *end side of its containing block*, that is the *right* floats on ltr scripts and the *left* floats on rtl scripts.

Formal definition

Initial value	none
Applies to	block-level elements
Inherited	no
Computed value	as specified
Animation type	discrete

Formal syntax

```
clear =
  inline-start   |
  inline-end     |
  block-start    |
  block-end      |
  left           |
  right          |
  top            |
  bottom         |
  both-inline    |
  both-block     |
  both           |
  none
```

Examples

clear: left

HTML

HTML	Play

```html
<div class="wrapper">
  <p class="black">
    Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet
    diam. Duis mattis varius dui. Suspendisse eget dolor.
  </p>
  <p class="red">Lorem ipsum dolor sit amet, consectetuer adipiscing elit.</p>
  <p class="left">This paragraph clears left.</p>
</div>
```

CSS

CSS	Play

```css
.wrapper {
  border: 1px solid black;
  padding: 10px;
}
.left {
  border: 1px solid black;
  clear: left;
}
.black {
  float: left;
  margin: 0;
  background-color: black;
  color: #fff;
  width: 20%;
}
.red {
  float: left;
  margin: 0;
  background-color: pink;
  width: 20%;
}
p {
  width: 50%;
}
```

Play

Lorem ipsum dolor
sit amet,
consectetuer
adipiscing elit.
Phasellus sit amet
diam. Duis mattis
varius dui.
Suspendisse eget
dolor.

Lorem ipsum dolor
sit amet,
consectetuer
adipiscing elit.

This paragraph clears left.

clear: right

HTML

HTML	Play

```html
<div class="wrapper">
  <p class="black">
    Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet
    diam. Duis mattis varius dui. Suspendisse eget dolor.
  </p>
  <p class="red">Lorem ipsum dolor sit amet, consectetuer adipiscing elit.</p>
  <p class="right">This paragraph clears right.</p>
</div>
```
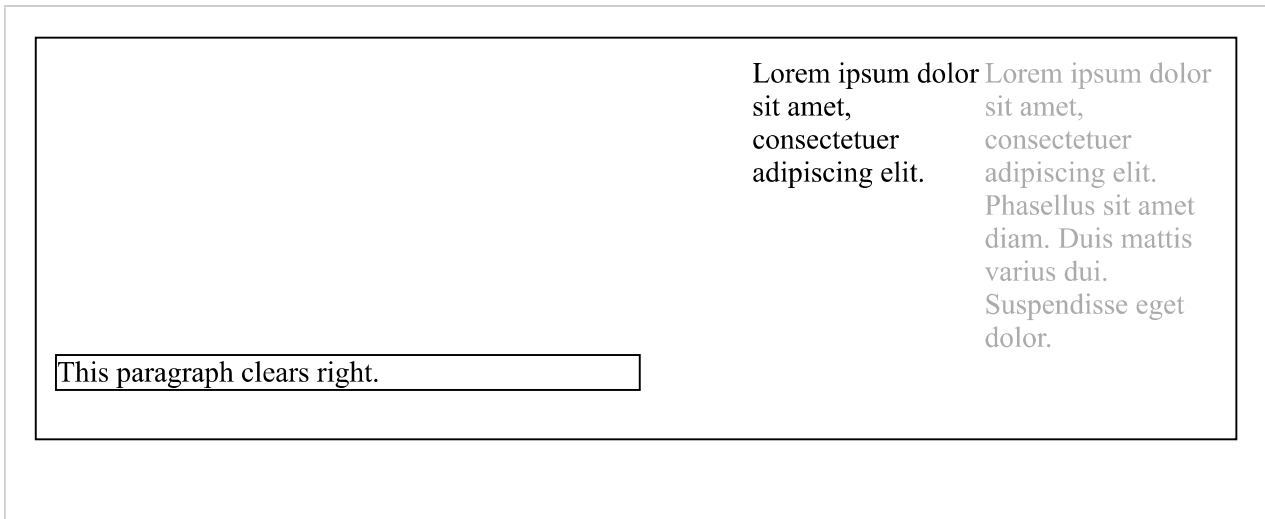
CSS

CSS	Play

```css
.wrapper {
  border: 1px solid black;
  padding: 10px;
}
.right {
  border: 1px solid black;
  clear: right;
}
.black {
  float: right;
  margin: 0;
  background-color: black;
  color: #fff;
  width: 20%;
}
.red {
  float: right;
```

```
    margin: 0;
    background-color: pink;
    width: 20%;
  }
  p {
    width: 50%;
  }
```

Play

This paragraph clears right.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet diam. Duis mattis varius dui. Suspendisse eget dolor.

clear: both

HTML

HTML   Play

```
<div class="wrapper">
  <p class="black">
    Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet
    diam. Duis mattis varius dui. Suspendisse eget dolor. Fusce pulvinar lacus
    ac dui.
  </p>
  <p class="red">
    Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet
    diam. Duis mattis varius dui. Suspendisse eget dolor.
  </p>
  <p class="both">This paragraph clears both.</p>
</div>
```

CSS

CSS   Play

```
.wrapper {
  border: 1px solid black;
  padding: 10px;
```

```
  }
  .both {
    border: 1px solid black;
    clear: both;
  }
  .black {
    float: left;
    margin: 0;
    background-color: black;
    color: #fff;
    width: 20%;
  }
  .red {
    float: right;
    margin: 0;
    background-color: pink;
    width: 20%;
  }
  p {
    width: 45%;
  }
```

Play

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet diam. Duis mattis varius dui. Suspendisse eget dolor. Fusce pulvinar lacus ac dui.

This paragraph clears both.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet diam. Duis mattis varius dui. Suspendisse eget dolor.

Specifications

Specification
[Cascading Style Sheets Level 2 Revision 2 (CSS 2.2) Specification](#) [# propdef-clear](#)

Specification
CSS Logical Properties and Values Level 1 # float-clear

Browser compatibility

Report problems with this compatibility data on GitHub

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
clear	1	12	1	3.5	1	18	4	10.1	1	1.0	4.4
both	1	12	1	15	1	18	4	14	1	1.0	4.4
inline-end	118	118	55	104	15	118	55	79	15	25.0	118
inline-start	118	118	55	104	15	118	55	79	15	25.0	118
left	1	12	1	15	1	18	4	14	1	1.0	4.4
right	1	12	1	15	1	18	4	14	1	1.0	4.4

Tip: you can click/tap on a cell for more information.

Full support   No support

See also

- **CSS basic box model**

Help improve MDN

Was this page helpful to you?

Yes   No

Learn how to contribute.

This page was last modified on Jan 30, 2024 by MDN contributors.