



Tailwind CSS

Version 3.4 · verified 2026-07-27

A compact guide to Tailwind CSS —
utility classes, layout, and configuration.

Contents

Mental model	2
Spacing (Padding & Margin)	2
Sizing (Width & Height)	3
Flexbox	4
Grid Layout	6
Typography	7
Colors & Backgrounds	8
Responsive Design	9
States & Interactions	10
Further reading	11

Mental model

Tailwind provides low-level utility classes that map directly to CSS properties, allowing you to build designs directly in your markup. Instead of creating custom CSS classes, you combine utilities to style elements, enforcing consistency and reducing context switching.

Spacing (Padding & Margin)

Utility	CSS equivalent	Meaning
<code>p-4</code>	<code>padding: 1rem;</code>	Padding on all sides
<code>px-4</code>	<code>padding-left: 1rem;</code> <code>padding-right: 1rem;</code>	Padding on x-axis (left/right)
<code>mt-2</code>	<code>margin-top: 0.5rem;</code>	Margin top

Tailwind's default spacing scale is proportional: 1 equals `0.25rem` (4px). Utilities follow a predictable pattern: property (`m` , `p`) + optional side (`t` , `r` , `b` , `l` , `x` , `y`) + scale value.

```
<div class="pt-6 pb-8 px-4 mt-2">  
  Content here  
</div>
```

Tip: You can use negative values for margins by prefixing with a dash, like `-mt-4`.

Sizing (Width & Height)

Utility	CSS equivalent	Meaning
<code>w-64</code>	<code>width: 16rem;</code>	Fixed width
<code>w-full</code>	<code>width: 100%;</code>	Full width of parent
<code>h-screen</code>	<code>height: 100vh;</code>	Full viewport height

Sizing uses the same proportional scale as spacing. Fractional values (`w-1/2` , `w-2/3`) and special values (`full` , `screen` , `auto`) provide flexible dimensions.

```
<div class="w-1/2 h-48 max-w-sm">  
  Fixed height, fluid width  
</div>
```

Gotcha: `w-screen` will cause a horizontal scrollbar if the page has a vertical scrollbar. Prefer `w-full` for block elements.

Flexbox

Utility	CSS equivalent	Meaning
<code>flex</code>	<code>display: flex;</code>	Defines a flex container
<code>flex-col</code>	<code>flex-direction: column;</code>	Stack items vertically
<code>justify-between</code>	<code>justify-content: space-between;</code>	Space items evenly

Flexbox utilities control layout, alignment, and distribution of space. Apply `flex` to a parent container, then use modifiers on the parent or children to dictate the layout.

```
<div class="flex flex-row justify-between">
  <div>Logo</div>
  <div class="flex gap-4">
    <a href="#">Nav 1</a>
    <a href="#">Nav 2</a>
  </div>
</div>
```

Note: The `gap-{size}` utility is supported in both flex and grid layouts.

Grid Layout

Utility	CSS equivalent	Meaning
<code>grid</code>	<code>display: grid;</code>	Defines a grid container
<code>grid-cols-3</code>	<code>grid-template-columns: repeat(3, minmax(0, 1fr));</code>	3 equal columns
<code>col-span-2</code>	<code>grid-column: span 2 / span 2;</code>	Span 2 columns

Grid utilities provide a 12-column grid system out of the box. Use `grid-cols-{n}` to define columns, and `col-span-{n}` on children to span multiple tracks.

```
<div class="grid grid-cols-1 md:grid-cols-3 gap-6">  
  <div class="col-span-2">Main Content</div>  
  <div>Sidebar</div>  
</div>
```

Typography

Utility	CSS equivalent	Meaning
text-lg	font-size: 1.125rem; line-height: 1.75rem;	Large text
font-bold	font-weight: 700;	Bold text
text-center	text-align: center;	Center align

Typography utilities bundle sensible defaults for `font-size` and `line-height`. You can control color, weight, tracking, and leading with separate utilities.

```
<h1 class="text-2xl font-bold text-gray-900">  
  Headline  
</h1>  
<p class="text-sm leading-relaxed text-gray-600">  
  Body text goes here.  
</p>
```

Tip: The `truncate` utility handles overflow, text-overflow, and white-space in one class.

Colors & Backgrounds

Utility	CSS equivalent	Meaning
text-blue-500	color: rgb(59 130 246);	Text color
bg-red-100	background-color: rgb(254 226 226);	Background color
border-gray-200	border-color: rgb(229 231 235);	Border color

Tailwind includes an extensive color palette. Colors range from 50 (lightest) to 950 (darkest). Apply colors to text, backgrounds, borders, rings, and more.

```
<button class="bg-blue-600 text-white rounded-md">  
  Click Me  
</button>
```

Responsive Design

Utility	CSS equivalent	Meaning
sm:	@media (min-width: 640px)	Small screens and up
md:	@media (min-width: 768px)	Medium screens and up
lg:	@media (min-width: 1024px)	Large screens and up

Tailwind uses a mobile-first approach. Unprefixed utilities apply to all screen sizes. Prefix utilities with breakpoint names (sm: , md: , lg: , xl: , 2xl:) to apply them at specific widths and above.

```
<div class="w-full md:w-1/2 lg:w-1/3 p-4">
```

Responsive card

```
</div>
```

Warning: Never use a breakpoint prefix without a base (unprefixed) class to define the default mobile state.

States & Interactions

Utility	Meaning
<code>hover:</code>	Applies when element is hovered
<code>focus:</code>	Applies when element has focus
<code>active:</code>	Applies when element is active (clicked)

State modifiers function exactly like responsive prefixes. They allow you to style elements under specific interactive conditions without writing custom CSS selectors.

```
<button class="bg-blue-500 hover:bg-blue-600">  
  Interactive Button  
</button>
```

Further reading

- [Tailwind CSS Configuration](#)
- [Arbitrary Values in Tailwind](#)