What is Tailwind CSS?

- ✓ Tailwind CSS can be used to style websites in the fastest and easiest way.
- ✓ Tailwind CSS is basically a **utility-first CSS framework** for rapidly building custom user interfaces.
- ✓ It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke(specially made for a particular person, organization, or purpose) designs without any annoying opinionated styles you have to fight to override.
- ✓ In Tailwind, you simply bring tiny components together to construct a user interface that is unique.
- ✓ Tailwind CSS was created in 2017 by Adam Wathan and Steve Schoger.

"**Utility-first CSS framework packed with classes to build any design.**"

- ✓ Unlike Bootstrap, Tailwind CSS doesn't come with prebuilt classes like dropdown-menu and dropdown-item. (But there is an additional, paid UI kit offered by Tailwind's creators, since it's not part of the core offering)
- ✓ Instead, it's up to you to **design and build your own components and design**. They simply provide you with the tools to do it quickly and easily.
- ✓ Tailwind CSS comes with some extensions in VS Code (**Tailwind CSS IntelliSense**) that helps you autocomplete suggestions and provide class definitions without needing to leave your editor.
- ✓ Tailwind CSS also makes responsive designs easy as they handle all the media queries for you, you just need to use the breakpoint(**xs, sm, md, lg, xl**) you want to target in front of the class you're using.

Instead of writing CSS yourself and worrying about things like class naming conventions, Tailwind CSS handles it all for you, and you just need to use the classes directly on the elements you wish to style.

- ✓ The utility classes like p-4, text-center, and rounded-full. These are all abstractions of traditional CSS properties: padding, text-align, and border-radius.

Now let's take a look at a basic example of Tailwind CSS:

```
<div class="bg-gray-200 p-4">
  <h2 class="font-bold text-lg">Lizard</h2>
</div>
```

This shows an example application of different Tailwind CSS utility classes, such as:

- bg-gray-200 for the background color

Tailwind

- p-4 for padding
- font-bold and text-lg for text styling

Why Tailwind CSS?

- ✓ The faster UI building process.
- ✓ If having a unique design is vital to the project or your brand identity, then having the freedom to build completely custom might be the best choice, and that's what Tailwind CSS offers you.
- ✓ This means you need to design and build all your components from scratch.
- ✓ Tailwind's utility-first approach with highly customizable classes allows for greater styling flexibility suited to projects that need heavy customization and branding.

As we know there are many CSS frameworks but people always choose the fast and easy framework to learn and use in the project. Tailwind has come with inbuilt a lot of features and styles for users to choose from and is also used to reduce the tendency of writing CSS code and create a beautiful custom UI. It will help you to overcome the complicated task. Tailwind CSS creates small utilities with a defined set of options enabling easy integration of existing classes directly into the HTML code.

When to Use Tailwind CSS

- ✓ Tailwind CSS is best used to speed up the development process by writing less code.
- ✓ It comes with a design system that helps maintain consistency across various design requirements like padding, spacing, and so forth; with this, you do not have to worry about creating your design systems.
- ✓ You can also use Tailwind CSS if you are looking to use a framework that is easily configurable because it does not force you to use components (navigation bars, buttons, forms, and so forth) in the same way all the time; you get to choose what your components should look like.
- ✓ But you should never use Tailwind if you have not learned and practiced CSS.

Pros and Cons of Tailwind CSS

Here are some of the advantages of using Tailwind CSS:

- ✓ Faster development process
- ✓ Helps you practice your CSS more as the utilities are similar
- ✓ All utilities and components are easily customizable
- ✓ The overall file size for production is usually small
- ✓ Easy to learn if you already know CSS
- ✓ Good documentation for learning

Some of the disadvantages of using Tailwind CSS include:

- ✓ Your markup might look disorganized for large projects because all the styles are in the HTML files.
- ✓ It isn’t easy to learn if you don’t understand CSS well.
- ✓ You are forced to build everything from scratch, including your input elements. When you install Tailwind CSS, it removes all default CSS styles.
- ✓ Tailwind CSS is not the best option if you are looking to minimize time spent developing your website’s frontend and mainly focusing on the backend logic.

Understanding of Various Methods to use tailwind CSS (Installation)
> Tailwind CLI
> Framework Guides
> Play CDN

Using Tailwind via CDN

- ✓ Use the CDN to try Tailwind right in the browser without any build step.

Tailwind

- ✓ Add the CDN script tag to the <head> of your HTML file, and start using Tailwind's utility classes to style your content.

```
<script src="https://cdn.tailwindcss.com"></script>
```

Note: There are some limitations when CDN is used. Some of them are:

- Customize Tailwind's default theme can't be used
- Directives like @apply, @variants, etc can't be used
- Can't install third-party plugins

Find quickly all the class names and CSS properties with below cheat sheets.

https://nerdcave.com/tailwind-cheat-sheet

https://tailwindcomponents.com/cheatsheet/

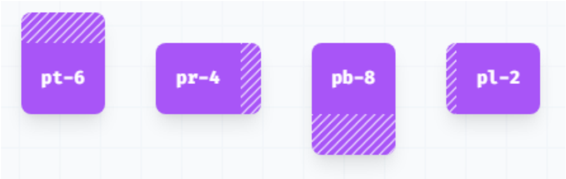
Spacing

Padding

Utilities for controlling an element's padding.

Add padding to a single side

- ✓ Control the padding on one side of an element using the **p{t|r|b|l}-{size}** utilities.
- ✓ For example,
  - ○ **pt-6** would add 1.5rem of padding to the top of an element,
  - ○ **pr-4** would add 1rem of padding to the right of an element,
  - ○ **pb-8** would add 2rem of padding to the bottom of an element,
  - ○ **pl-2** would add 0.5rem of padding to the left of an element.



Add horizontal padding

Control the horizontal padding of an element using the **px-{size}** utilities.

Add Vertical padding

Control the vertical padding of an element using the **py-{size}** utilities.

Add padding to all sides

Control the padding on all sides of an element using the **p-{size}** utilities.

Margin

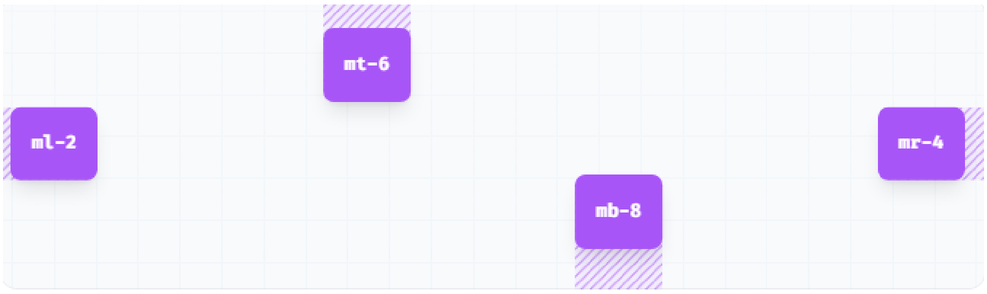
Utilities for controlling an element's margin.

Add margin to a single side

- ✓ Control the margin on one side of an element using the **m{t|r|b|l}-{size}** utilities.
  - ○ **mt-6** would add 1.5rem of margin to the top of an element,

Tailwind

- mr-4 would add 1rem of margin to the right of an element,
- mb-8 would add 2rem of margin to the bottom of an element,
- ml-2 would add 0.5rem of margin to the left of an element.



Add horizontal margin

Control the horizontal margin of an element using the **mx-{size}** utilities.

Add vertical margin

Control the vertical margin of an element using the **my-{size}** utilities.

Add margin to all sides

Control the margin on all sides of an element using the **m-{size}** utilities.

Class	CSS Property value	Pixel Value
.m-0	margin: 0px;	
.m-1	margin: 0.25rem;	4px
.m-2	margin: 0.5rem;	8px
.m-3	margin: 0.75rem;	12px
.m-4	margin: 1rem;	16px
.m-5	margin: 1.25rem;	20px
.m-6	margin: 1.5rem;	24px
.m-7	margin: 1.75rem;	28px
.m-8	margin: 2rem;	32px
.m-9	margin: 2.25rem;	36px
.m-10	margin: 2.5rem;	40px
.m-11	margin: 2.75rem;	44px

.m-12	margin: 3rem;	48px
.m-14	margin: 3.5rem;	56px
.m-16	margin: 4rem;	64px
.m-20	margin: 5rem;	80px
.m-24	margin: 6rem;	96px
.m-28	margin: 7rem;	112px
.m-32	margin: 8rem;	128px
.m-36	margin: 9rem;	144px
.m-40	margin: 10rem;	160px
.m-44	margin: 11rem;	176px
.m-48	margin: 12rem;	192px
.m-52	margin: 13rem;	208px
.m-56	margin: 14rem;	224px
.m-60	margin: 15rem;	240px
.m-64	margin: 16rem;	256px
.m-72	margin: 18rem;	288px
.m-80	margin: 20rem;	320px
.m-96	margin: 24rem;	384px
.m-auto	margin: auto;	
.m-px	margin: 1px;	
.m-0.5	margin: 0.125rem;	2px
.m-1.5	margin: 0.375rem;	6px
.m-2.5	margin: 0.625rem;	10px
.m-3.5	margin: 0.875rem;	14px

Use same values for the specific sides like right(mr), left(ml), top(mt), bottom(mb) and also same for the padding.

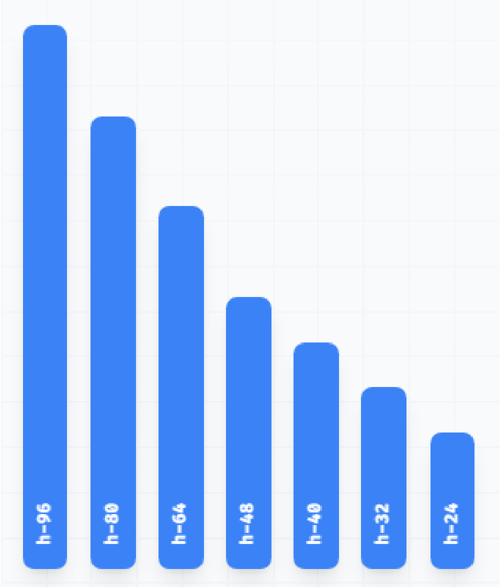
Sizing

In Tailwind, the height and width properties are denoted by the **.h-{number}** and **.w-{number}** classes.

Height

Fixed heights

Use **h-{number}** or h-px to set an element to a fixed height.



h-0	height: 0px;
h-px	height: 1px;
h-0.5	height: 0.125rem; /* 2px */
h-1	height: 0.25rem; /* 4px */
h-1.5	height: 0.375rem; /* 6px */
h-2	height: 0.5rem; /* 8px */
h-2.5	height: 0.625rem; /* 10px */
h-3	height: 0.75rem; /* 12px */
h-3.5	height: 0.875rem; /* 14px */
h-4	height: 1rem; /* 16px */
h-5	height: 1.25rem; /* 20px */
h-6	height: 1.5rem; /* 24px */
h-7	height: 1.75rem; /* 28px */
h-8	height: 2rem; /* 32px */
h-9	height: 2.25rem; /* 36px */
h-10	height: 2.5rem; /* 40px */
h-11	height: 2.75rem; /* 44px */
h-12	height: 3rem; /* 48px */
h-14	height: 3.5rem; /* 56px */
h-16	height: 4rem; /* 64px */
h-20	height: 5rem; /* 80px */

Tailwind

h-24	height: 6rem; /* 96px */
h-28	height: 7rem; /* 112px */
h-32	height: 8rem; /* 128px */
h-36	height: 9rem; /* 144px */
h-40	height: 10rem; /* 160px */
h-44	height: 11rem; /* 176px */
h-48	height: 12rem; /* 192px */
h-52	height: 13rem; /* 208px */
h-56	height: 14rem; /* 224px */
h-60	height: 15rem; /* 240px */
h-64	height: 16rem; /* 256px */
h-72	height: 18rem; /* 288px */
h-80	height: 20rem; /* 320px */
h-96	height: 24rem; /* 384px */
h-auto	height: auto;
h-1/2	height: 50%;
h-1/3	height: 33.333333%;
h-2/3	height: 66.666667%;
h-1/4	height: 25%;
h-2/4	height: 50%;
h-3/4	height: 75%;
h-1/5	height: 20%;
h-2/5	height: 40%;
h-3/5	height: 60%;
h-4/5	height: 80%;
h-1/6	height: 16.666667%;
h-2/6	height: 33.333333%;
h-3/6	height: 50%;
h-4/6	height: 66.666667%;
h-5/6	height: 83.333333%;
h-full	height: 100%;
h-screen	height: 100vh;

Full height

Use **h-full** to set an element’s height to 100% of its parent, as long as the parent has a defined height.

Viewport height

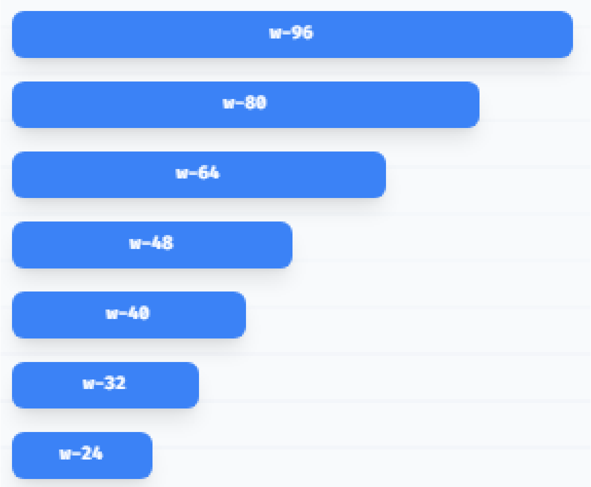
Use **h-screen** to make an element span the entire height of the viewport.

Width

Utilities for setting the width of an element.

Fixed widths

Use **w-{number}** or **w-px** to set an element to a fixed width.



w-0	width: 0px;
w-px	width: 1px;
w-0.5	width: 0.125rem; /* 2px */
w-1	width: 0.25rem; /* 4px */
w-1.5	width: 0.375rem; /* 6px */
w-2	width: 0.5rem; /* 8px */
w-2.5	width: 0.625rem; /* 10px */
w-3	width: 0.75rem; /* 12px */
w-3.5	width: 0.875rem; /* 14px */
w-4	width: 1rem; /* 16px */
w-5	width: 1.25rem; /* 20px */
w-6	width: 1.5rem; /* 24px */
w-7	width: 1.75rem; /* 28px */
w-8	width: 2rem; /* 32px */
w-9	width: 2.25rem; /* 36px */
w-10	width: 2.5rem; /* 40px */
w-11	width: 2.75rem; /* 44px */
w-12	width: 3rem; /* 48px */
w-14	width: 3.5rem; /* 56px */
w-16	width: 4rem; /* 64px */
w-20	width: 5rem; /* 80px */
w-24	width: 6rem; /* 96px */
w-28	width: 7rem; /* 112px */
w-32	width: 8rem; /* 128px */
w-36	width: 9rem; /* 144px */
w-40	width: 10rem; /* 160px */
w-44	width: 11rem; /* 176px */
w-48	width: 12rem; /* 192px */
w-52	width: 13rem; /* 208px */
w-56	width: 14rem; /* 224px */
w-60	width: 15rem; /* 240px */
w-64	width: 16rem; /* 256px */

w-72	width: 18rem; /* 288px */
w-80	width: 20rem; /* 320px */
w-96	width: 24rem; /* 384px */
w-auto	width: auto;
w-1/2	width: 50%;
w-1/3	width: 33.333333%;
w-2/3	width: 66.666667%;
w-1/4	width: 25%;
w-2/4	width: 50%;
w-3/4	width: 75%;
w-1/5	width: 20%;
w-2/5	width: 40%;
w-3/5	width: 60%;
w-4/5	width: 80%;
w-1/6	width: 16.666667%;
w-2/6	width: 33.333333%;
w-3/6	width: 50%;
w-4/6	width: 66.666667%;
w-5/6	width: 83.333333%;
w-1/12	width: 8.333333%;
w-2/12	width: 16.666667%;
w-3/12	width: 25%;
w-4/12	width: 33.333333%;
w-5/12	width: 41.666667%;
w-6/12	width: 50%;
w-7/12	width: 58.333333%;
w-8/12	width: 66.666667%;
w-9/12	width: 75%;
w-10/12	width: 83.333333%;
w-11/12	width: 91.666667%;
w-full	width: 100%;
w-screen	width: 100vw;

Full Width

Use **w-full** to set an element's width to 100%.

Viewport width

Use **w-screen** to make an element span the entire width of the viewport.

Setting the minimum width

Set the minimum width of an element using **min-w-*** utilities.

Setting the maximum width

Set the maximum width of an element using the **max-w-*** utilities.

Setting the minimum height

Set the minimum height of an element using **min-h-*** utilities.

Setting the maximum height

Set the maximum height of an element using **max-h-*** utilities.

* values can be 0, px, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 72, 80, 96 etc

Typography

Text color

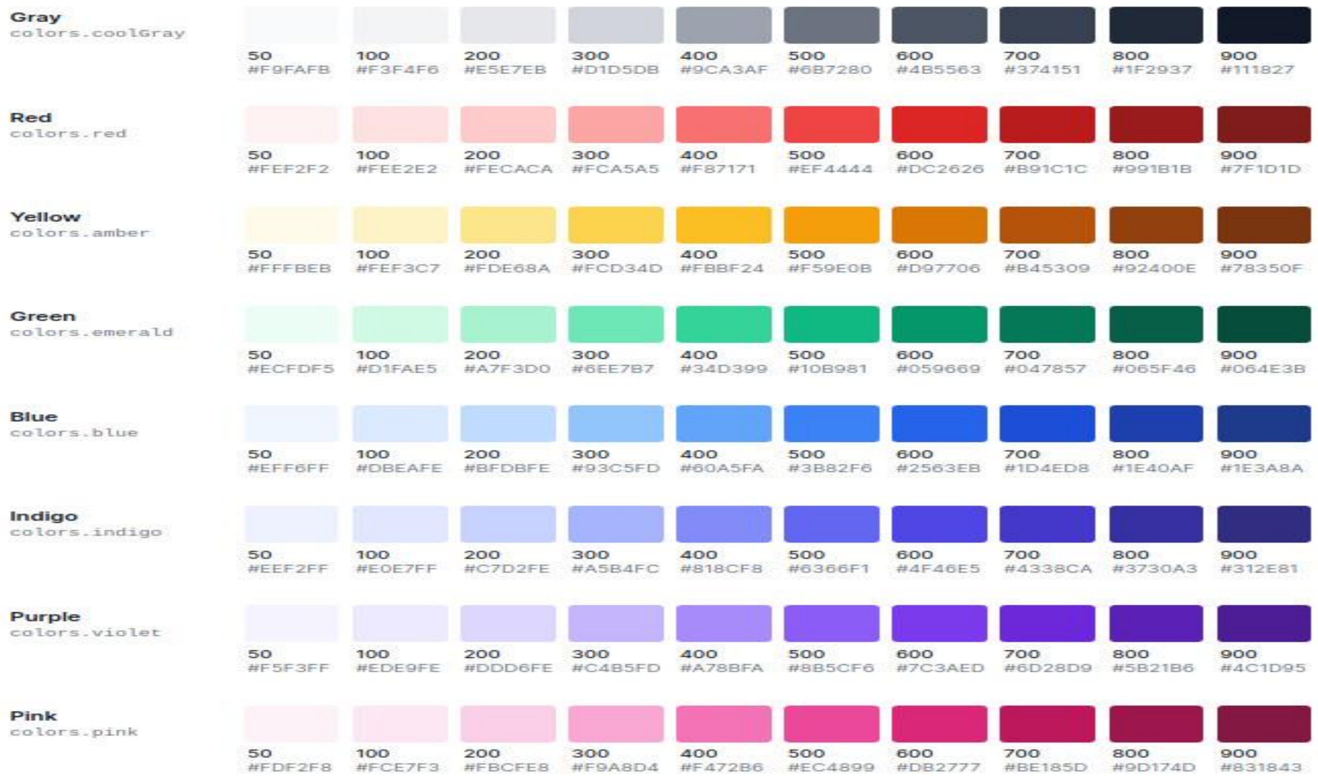
text-inherit	text-current	text-transparent	text-black	text-white	text-slate-50
text-slate-100	text-slate-200	text-slate-300	text-slate-400	text-slate-500	text-slate-600
text-slate-700	text-slate-800	text-slate-900	text-slate-950	text-gray-50	text-gray-100
text-gray-200	text-gray-300	text-gray-400	text-gray-500	text-gray-600	text-gray-700
text-gray-800	text-gray-900	text-gray-950	text-zinc-50	text-zinc-100	text-zinc-200
text-zinc-300	text-zinc-400	text-zinc-500	text-zinc-600	text-zinc-700	text-zinc-800
text-zinc-900	text-zinc-950	text-neutral-50	text-neutral-100	text-neutral-200	text-neutral-300
text-neutral-400	text-neutral-500	text-neutral-600	text-neutral-700	text-neutral-800	text-neutral-900
text-neutral-950	text-stone-50	text-stone-100	text-stone-200	text-stone-300	text-stone-400
text-stone-500	text-stone-600	text-stone-700	text-stone-800	text-stone-900	text-stone-950
text-red-50	text-red-100	text-red-200	text-red-300	text-red-400	text-red-500
text-red-600	text-red-700	text-red-800	text-red-900	text-red-950	text-orange-50
text-orange-100	text-orange-200	text-orange-300	text-orange-400	text-orange-500	text-orange-600
text-orange-700	text-orange-800	text-orange-900	text-orange-950	text-amber-50	text-amber-100
text-amber-200	text-amber-300	text-amber-400	text-amber-500	text-amber-600	text-amber-700
text-amber-800	text-amber-900	text-amber-950	text-yellow-50	text-yellow-100	text-yellow-200
text-yellow-300	text-yellow-400	text-yellow-500	text-yellow-600	text-yellow-700	text-yellow-800
text-yellow-900	text-yellow-950	text-lime-50	text-lime-100	text-lime-200	text-lime-300
text-lime-400	text-lime-500	text-lime-600	text-lime-700	text-lime-800	text-lime-900
text-lime-950	text-green-50	text-green-100	text-green-200	text-green-300	text-green-400

Tailwind

text-green-500	text-green-600	text-green-700	text-green-800	text-green-900	text-green-950
text-emerald-50	text-emerald-100	text-emerald-200	text-emerald-300	text-emerald-400	text-emerald-500
text-emerald-600	text-emerald-700	text-emerald-800	text-emerald-900	text-emerald-950	text-teal-50
text-teal-100	text-teal-200	text-teal-300	text-teal-400	text-teal-500	text-teal-600
text-teal-700	text-teal-800	text-teal-900	text-teal-950	text-cyan-50	text-cyan-100
text-cyan-200	text-cyan-300	text-cyan-400	text-cyan-500	text-cyan-600	text-cyan-700
text-cyan-800	text-cyan-900	text-cyan-950	text-sky-50	text-sky-100	text-sky-200
text-sky-300	text-sky-400	text-sky-500	text-sky-600	text-sky-700	text-sky-800
text-sky-900	text-sky-950	text-blue-50	text-blue-100	text-blue-200	text-blue-300
text-blue-400	text-blue-500	text-blue-600	text-blue-700	text-blue-800	text-blue-900
text-blue-950	text-indigo-50	text-indigo-100	text-indigo-200	text-indigo-300	text-indigo-400
text-indigo-500	text-indigo-600	text-indigo-700	text-indigo-800	text-indigo-900	text-indigo-950
text-violet-50	text-violet-100	text-violet-200	text-violet-300	text-violet-400	text-violet-500
text-violet-600	text-violet-700	text-violet-800	text-violet-900	text-violet-950	text-purple-50
text-purple-100	text-purple-200	text-purple-300	text-purple-400	text-purple-500	text-purple-600
text-purple-700	text-purple-800	text-purple-900	text-purple-950	text-fuchsia-50	text-fuchsia-100
text-fuchsia-200	text-fuchsia-300	text-fuchsia-400	text-fuchsia-500	text-fuchsia-600	text-fuchsia-700
text-fuchsia-800	text-fuchsia-900	text-fuchsia-950	text-pink-50	text-pink-100	text-pink-200
text-pink-300	text-pink-400	text-pink-500	text-pink-600	text-pink-700	text-pink-800
text-pink-900	text-pink-950	text-rose-50	text-rose-100	text-rose-200	text-rose-300
text-rose-400	text-rose-500	text-rose-600	text-rose-700	text-rose-800	text-rose-900
text-rose-950					

Note: The color's values can be changeable according to your need from 50-950.

Sample example of colors



Gray colors.coolGray	50 #F9FAFB	100 #F3F4F6	200 #E5E7EB	300 #D1D5DB	400 #9CA3AF	500 #6B7280	600 #4B5563	700 #374151	800 #1F2937	900 #111827
Red colors.red	50 #FEF2F2	100 #FEE2E2	200 #FECACA	300 #FCA5A5	400 #F87171	500 #EF4444	600 #DC2626	700 #B91C1C	800 #991B1B	900 #7F1D1D
Yellow colors.amber	50 #FFFBEB	100 #FEF3C7	200 #FDE68A	300 #FCD34D	400 #FBBF24	500 #F59E0B	600 #D97706	700 #B45309	800 #92400E	900 #78350F
Green colors.emerald	50 #ECFDF5	100 #D1FAE5	200 #A7F3D0	300 #6EE7B7	400 #34D399	500 #10B981	600 #059669	700 #047857	800 #065F46	900 #064E3B
Blue colors.blue	50 #EFF6FF	100 #DBEAFE	200 #BFDBFE	300 #93C5FD	400 #60A5FA	500 #3B82F6	600 #2563EB	700 #1D4ED8	800 #1E40AF	900 #1E3A8A
Indigo colors.indigo	50 #EEF2FF	100 #E0E7FF	200 #C7D2FE	300 #A5B4FC	400 #818CF8	500 #6366F1	600 #4F46E5	700 #4338CA	800 #3730A3	900 #312E81
Purple colors.violet	50 #F5F3FF	100 #EDE9FE	200 #DDD6FE	300 #C4B5FD	400 #A78BFA	500 #8B5CF6	600 #7C3AED	700 #6D28D9	800 #5B21B6	900 #4C1D95
Pink colors.pink	50 #FDF2F8	100 #FCE7F3	200 #FBCFE8	300 #F9A8D4	400 #F472B6	500 #EC4899	600 #DB2777	700 #BE185D	800 #9D174D	900 #831843

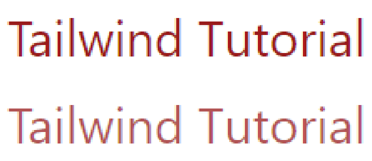
Text opacity

Control the opacity of an element's placeholder color using the **text-opacity-{amount}** utilities.

Note: The number of the opacity can be changeable from 0 to 100 with the span of 5.

Example:

```
<p class="text-red-800 p-2 text-3xl">Tailwind Tutorial</p>
<p class="text-red-800 text-opacity-75 p-2 text-3xl">Tailwind Tutorial</p>
```



Tailwind Tutorial

Tailwind Tutorial

Font family

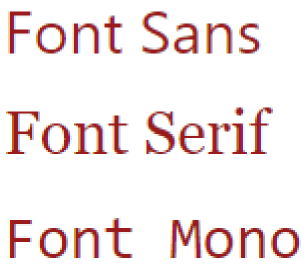
Font family classes:

- ✓ **font-sans**
- ✓ **font-serif**
- ✓ **font-mono**

font-sans	font-family: system-ui, -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, "Helvetica Neue", Arial, "Noto Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";
font-serif	font-family: Georgia, Cambria, "Times New Roman", Times, serif;
font-mono	font-family: Menlo, Monaco, Consolas, "Liberation Mono", "Courier New", monospace;

Example:

```
      <p class="text-red-800 p-2 text-3xl font-sans">Font Sans</p>
       <p class="text-red-800 p-2 text-3xl font-serif">Font Serif</p>
       <p class="text-red-800 p-2 text-3xl font-mono">Font Mono</p>
```


Font Sans

Font Serif

Font Mono

Font size

Font size classes:

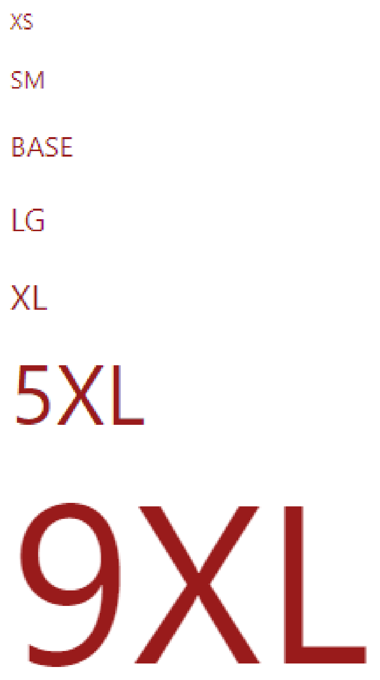
text-xs	This class defines the text size as extra small.
text-sm	This class defines the text size as small.
text-base	This class defines the text size as base size.
text-lg	This class defines the text size as large.
text-xl	This class defines the text size as extra-large.
text-2xl	This class defines the text size as 2 times extra-large.
text-3xl	This class defines the text size as 3 times extra-large.
text-4xl	This class defines the text size as 4 times extra-large.
text-5xl	This class defines the text size as 5 times extra-large.
text-6xl	This class defines the text size as 6 times extra-large.
text-7xl	This class defines the text size as 7 times extra-large.
text-8xl	This class defines the text size as 8 times extra-large.
text-9xl	This class defines the text size as 9 times extra-large.

Note: Change the size in the component with the required size mentioned above.

Example:

```
<p class="text-red-800 p-2 text-xs">XS</p>
       <p class="text-red-800 p-2 text-sm">SM</p>
       <p class="text-red-800 p-2 text-base">BASE</p>
```

```
        <p class="text-red-800 p-2 text-lg">LG</p>
        <p class="text-red-800 p-2 text-xl">XL</p>
        <p class="text-red-800 p-2 text-5xl">5XL</p>
        <p class="text-red-800 p-2 text-9xl">9XL</p>
```



XS

SM

BASE

LG

XL

5XL

9XL

Font style

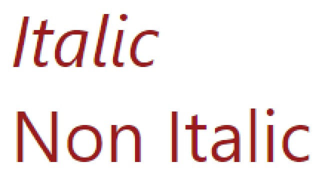
Font Style classes:

- ✓ italic
- ✓ non-italic

italic	This class is used to set the font style as *italic*.
non-italic	This is basically the default font, which is used to reset italic text at different breakpoints.

Example:

```
  <p class="text-red-800 p-2 text-5xl italic">Italic</p>
  <p class="text-red-800 p-2 text-5xl non-italic">Non Italic</p>
```



Italic

Non Italic

Font weight

Font weight classes:

font-thin	This class sets the *font-weight* to 100.
font-extralight	This class sets the *font-weight* to 200.
font-light	This class sets the *font-weight* to 300.
font-normal	This class sets the *font-weight* to 400.
font-medium	This class sets the *font-weight* to 500.

font-semibold	This class sets the *font-weight* to 600.
font-bold	This class sets the *font-weight* to 700.
font-extrabold	This class sets the *font-weight* to 800.
font-black	This class sets the *font-weight* to 900.

List style type

Utilities for controlling the bullet/number style of a list.

List Style Type classes:

list-none	No marker is shown in this mode.
list-disc	This is the default value. The marker is a filled circle.
list-decimal	A marker is a decimal number, beginning with 1.

Syntax:

```
<element class="List Style Type">...</element>
```

Text align

This class is used to specify the horizontal alignment of text in an element.

Text Alignment classes:

text-left
text-center
text-right
text-justify

Text decoration

This class is used to “decorate” the content of the text. It is essentially decorating the text with different kinds of lines.

Text Decoration classes:

underline
line-through
no-underline

Text transform

This class is used to control the capitalization of the text.

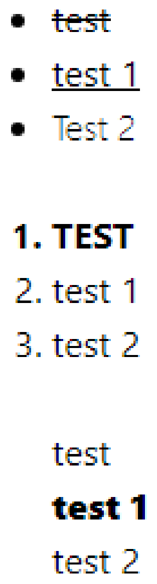
Text Transform classes:

uppercase
lowercase
capitalize
normal-case

Example:

Tailwind

```
<div class="m-24">
  <ul class="list-disc">
      <li class="line-through">test</li>
      <li class="underline">test 1</li>
      <li class="capitalize font-extralight">test 2</li>
  </ul>
  <br>
  <ol class="list-decimal">
      <li class="uppercase font-bold">test</li>
      <li>test 1</li>
      <li>test 2</li>
  </ol>
  <br>
  <ul class="list-none">
      <li>test</li>
      <li class="font-extrabold">test 1</li>
      <li>test 2</li>
  </ul>
  </div>
```



- ~~test~~
- test 1
- Test 2

1. **TEST**
2. test 1
3. test 2

test
test 1
test 2

Arbitrary values

If you need to use a one-off value that doesn’t make sense to include in your theme, use square brackets to generate a property on the fly using any arbitrary value.

```
<p class="text-[#50d71e]">
  <!-- ... -->
</p>
```

Background

Background color

- ✓ Control the background color of an element using the **bg-{color}** utilities.
- ✓ This class is used to specify the background color of an element. The background covers the total size of the element with padding and border but excluding margin. It makes the text so easy to read for the user.

Background Color classes:

bg-inherit	bg-current	bg-transparent	bg-black	bg-white	bg-slate-50
bg-slate-100	bg-slate-200	bg-slate-300	bg-slate-400	bg-slate-500	bg-slate-600
bg-slate-700	bg-slate-800	bg-slate-900	bg-slate-950	bg-gray-50	bg-gray-100
bg-gray-200	bg-gray-300	bg-gray-400	bg-gray-500	bg-gray-600	bg-gray-700
bg-gray-800	bg-gray-900	bg-gray-950	bg-zinc-50	bg-zinc-100	bg-zinc-200
bg-zinc-300	bg-zinc-400	bg-zinc-500	bg-zinc-600	bg-zinc-700	bg-zinc-800
bg-zinc-900	bg-zinc-950	bg-neutral-50	bg-neutral-100	bg-neutral-200	bg-neutral-300
bg-neutral-400	bg-neutral-500	bg-neutral-600	bg-neutral-700	bg-neutral-800	bg-neutral-900
bg-neutral-950	bg-stone-50	bg-stone-100	bg-stone-200	bg-stone-300	bg-stone-400
bg-stone-500	bg-stone-600	bg-stone-700	bg-stone-800	bg-stone-900	bg-stone-950
bg-red-50	bg-red-100	bg-red-200	bg-red-300	bg-red-400	bg-red-500
bg-red-600	bg-red-700	bg-red-800	bg-red-900	bg-red-950	bg-orange-50
bg-orange-100	bg-orange-200	bg-orange-300	bg-orange-400	bg-orange-500	bg-orange-600
bg-orange-700	bg-orange-800	bg-orange-900	bg-orange-950	bg-amber-50	bg-amber-100
bg-amber-200	bg-amber-300	bg-amber-400	bg-amber-500	bg-amber-600	bg-amber-700
bg-amber-800	bg-amber-900	bg-amber-950	bg-yellow-50	bg-yellow-100	bg-yellow-200
bg-yellow-300	bg-yellow-400	bg-yellow-500	bg-yellow-600	bg-yellow-700	bg-yellow-800
bg-yellow-900	bg-yellow-950	bg-lime-50	bg-lime-100	bg-lime-200	bg-lime-300
bg-lime-400	bg-lime-500	bg-lime-600	bg-lime-700	bg-lime-800	bg-lime-900
bg-lime-950	bg-green-50	bg-green-100	bg-green-200	bg-green-300	bg-green-400

Tailwind

bg-green-500	bg-green-600	bg-green-700	bg-green-800	bg-green-900	bg-green-950
bg-emerald-50	bg-emerald-100	bg-emerald-200	bg-emerald-300	bg-emerald-400	bg-emerald-500
bg-emerald-600	bg-emerald-700	bg-emerald-800	bg-emerald-900	bg-emerald-950	bg-teal-50
bg-teal-100	bg-teal-200	bg-teal-300	bg-teal-400	bg-teal-500	bg-teal-600
bg-teal-700	bg-teal-800	bg-teal-900	bg-teal-950	bg-cyan-50	bg-cyan-100
bg-cyan-200	bg-cyan-300	bg-cyan-400	bg-cyan-500	bg-cyan-600	bg-cyan-700
bg-cyan-800	bg-cyan-900	bg-cyan-950	bg-sky-50	bg-sky-100	bg-sky-200
bg-sky-300	bg-sky-400	bg-sky-500	bg-sky-600	bg-sky-700	bg-sky-800
bg-sky-900	bg-sky-950	bg-blue-50	bg-blue-100	bg-blue-200	bg-blue-300
bg-blue-400	bg-blue-500	bg-blue-600	bg-blue-700	bg-blue-800	bg-blue-900
bg-blue-950	bg-indigo-50	bg-indigo-100	bg-indigo-200	bg-indigo-300	bg-indigo-400
bg-indigo-500	bg-indigo-600	bg-indigo-700	bg-indigo-800	bg-indigo-900	bg-indigo-950
bg-violet-50	bg-violet-100	bg-violet-200	bg-violet-300	bg-violet-400	bg-violet-500
bg-violet-600	bg-violet-700	bg-violet-800	bg-violet-900	bg-violet-950	bg-purple-50
bg-purple-100	bg-purple-200	bg-purple-300	bg-purple-400	bg-purple-500	bg-purple-600
bg-purple-700	bg-purple-800	bg-purple-900	bg-purple-950	bg-fuchsia-50	bg-fuchsia-100
bg-fuchsia-200	bg-fuchsia-300	bg-fuchsia-400	bg-fuchsia-500	bg-fuchsia-600	bg-fuchsia-700
bg-fuchsia-800	bg-fuchsia-900	bg-fuchsia-950	bg-pink-50	bg-pink-100	bg-pink-200
bg-pink-300	bg-pink-400	bg-pink-500	bg-pink-600	bg-pink-700	bg-pink-800
bg-pink-900	bg-pink-950	bg-rose-50	bg-rose-100	bg-rose-200	bg-rose-300
bg-rose-400	bg-rose-500	bg-rose-600	bg-rose-700	bg-rose-800	bg-rose-900
bg-rose-950					

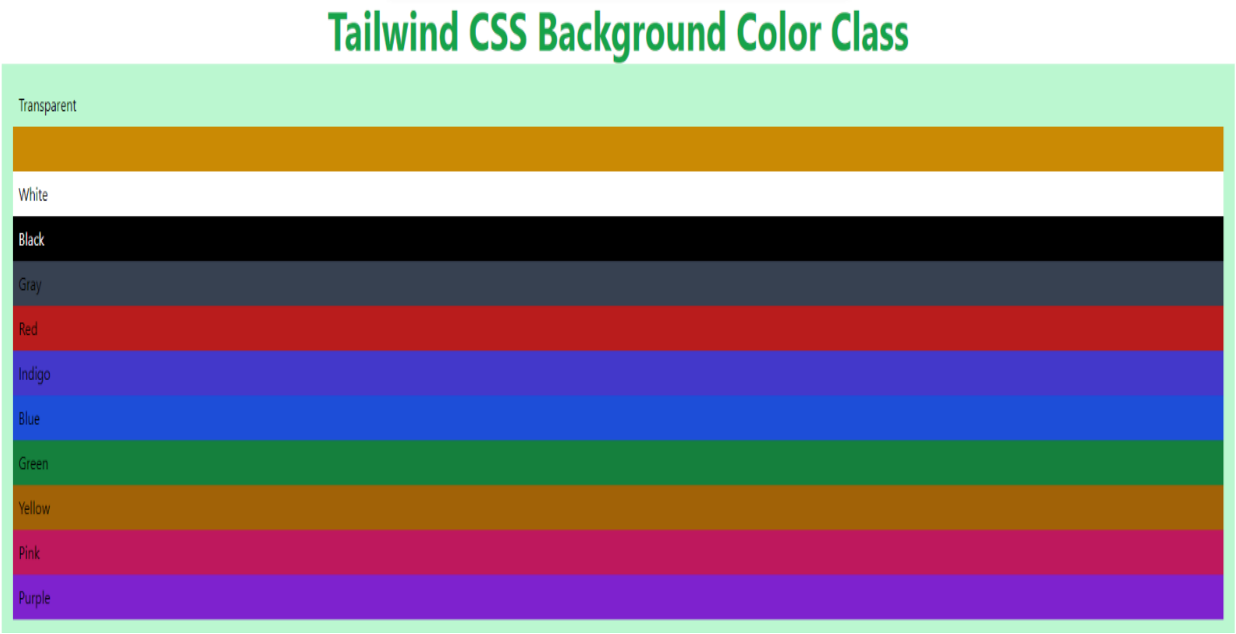
Note: The color’s values can be changeable according to your need from 50-950.

Example:

```
<body class="text-center mx-4">
   <h1 class="text-green-600 text-5xl font-bold">
      Tailwind CSS Background Color Class
   </h1>
```

Tailwind

```
  <div class="mx-14 bg-green-200 text-justify p-4">
    <p class="bg-transparent p-2">Transparent</p>
    <p class="bg-current text-yellow-600 p-2">Current</p>
    <p class="bg-white p-2"> White </p>
    <p class="bg-black text-white p-2">Black</p>
    <p class="bg-gray-700 p-2"> Gray</p>
    <p class="bg-red-700 p-2"> Red</p>
    <p class="bg-indigo-700 p-2">Indigo</p>
    <p class="bg-blue-700 p-2"> Blue</p>
    <p class="bg-green-700 p-2">Green</p>
    <p class="bg-yellow-700 p-2"> Yellow</p>
    <p class="bg-pink-700 p-2">Pink</p>
    <p class="bg-purple-700 p-2"> Purple</p>
  </div>
</body>
```



Tailwind CSS Background Color Class

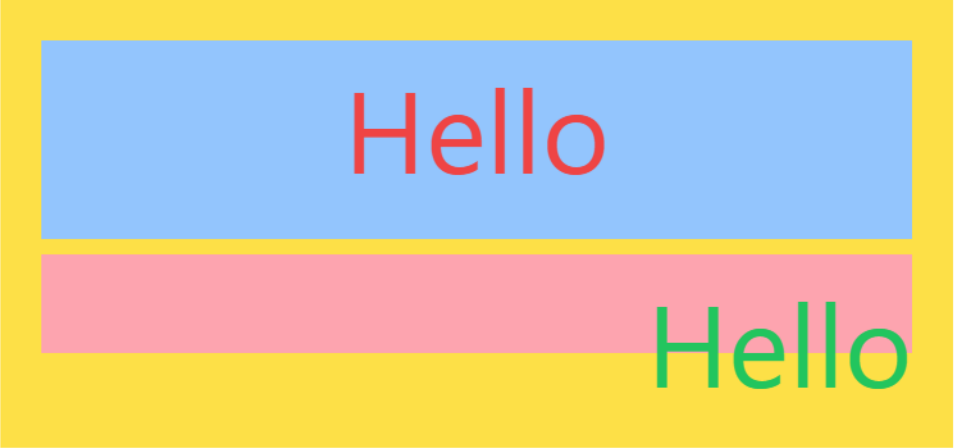
```
Transparent

White
Black
Gray
Red
Indigo
Blue
Green
Yellow
Pink
Purple
```

Example:

Design a **layout** using **Tailwind CSS** that consists of a **container** with a height of 350px, a yellow background, and padding. Inside this container, create two text blocks:

1. The **first block** should occupy **half the height** of the container, have a **blue background**, display "Hello" in **90px red text** and **center-aligned**.
2. The **second block** should occupy **one-fourth the height** of the container, have a **rose-colored background**, display "Hello" in **90px green text** and **right-aligned**

```
<div class="h-[350px] bg-yellow-300 p-5 m-3">
   <div class="h-1/2 bg-blue-300 text-[90px] text-red-500 text-center m-3">Hello</div>
   <div class="h-1/6 bg-rose-300 text-[90px] text-green-500 text-right m-3">Hello</div>
</div>
```



Background Attachment

Utilities for controlling how a background image behaves when scrolling.

bg-fixed **(background-attachment: fixed;)**

The background is fixed relative to the viewport. Even if an element has a scrolling mechanism, the background doesn't move with the element.

bg-local **(background-attachment: local;)**

The background is fixed relative to the element's contents. If the element has a scrolling mechanism, the background scrolls with the element's contents, and the background painting area and background positioning area are relative to the scrollable area of the element rather than to the border framing them.

bg-scroll **(background-attachment: scroll;)**

The background is fixed relative to the element itself and does not scroll with its contents.

Reference Link: https://tailwindcss.com/docs/background-attachment

Background Position

Utilities for controlling the position of an element's background image.

Use the **bg-{side}** utilities to control the position of an element's background image.

bg-bottom	background-position: bottom;
bg-center	background-position: center;
bg-left	background-position: left;
bg-left-bottom	background-position: left bottom;
bg-left-top	background-position: left top;
bg-right	background-position: right;
bg-right-bottom	background-position: right bottom;
bg-right-top	background-position: right top;
bg-top	background-position: top;

Background Repeat

Utilities for controlling the repetition of an element's background image.

bg-repeat	background-repeat: repeat;
bg-no-repeat	background-repeat: no-repeat;
bg-repeat-x	background-repeat: repeat-x;
bg-repeat-y	background-repeat: repeat-y;

Repeat

Use **bg-repeat** to repeat the background image both vertically and horizontally.

```
<div class="bg-repeat" style="background-image: url(...)"></div>
```

No Repeat

Use **bg-no-repeat** when you don’t want to repeat the background image.

```
<div class="bg-no-repeat" style="background-image: url(...)"> </div>
```

Repeat Horizontally

Use **bg-repeat-x** to repeat the background image only horizontally.

```
<div class="bg-repeat-x" style="background-image: url(...)"></div>
```

Repeat Vertically

Use **bg-repeat-y** to repeat the background image only vertically.

```
<div class="bg-repeat-y" style="background-image: url(...)"></div>
```

Background Size

Utilities for controlling the background size of an element's background image.

bg-auto	background-size: auto;
bg-cover	background-size: cover;
bg-contain	background-size: contain;

Auto

Use **bg-auto** to display the background image at its default size.

```
<div class="bg-auto bg-no-repeat" style="background-image: url(...)"></div>
```

Cover

Use **bg-cover** to scale the background image until it fills the background layer.

```
<div class="bg-cover" style="background-image: url(...)"></div>
```

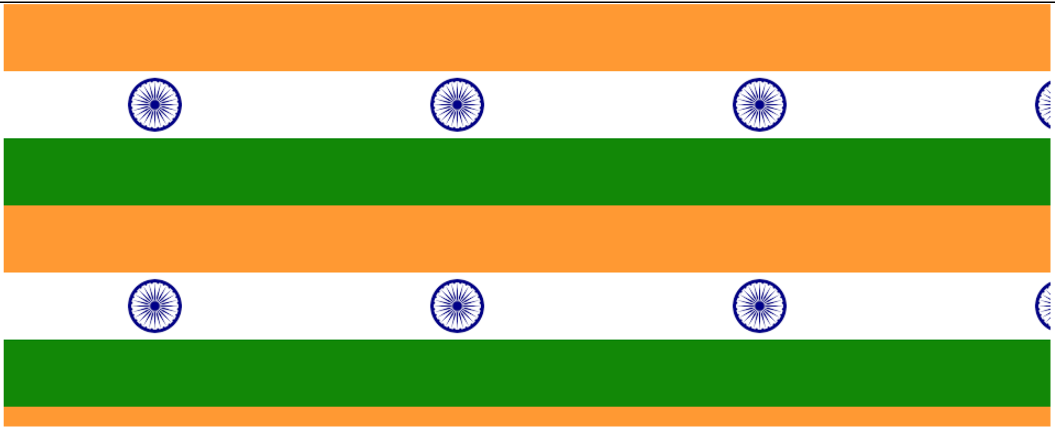
Contain

Use **bg-contain** to scale the background image to the outer edges without cropping or stretching.

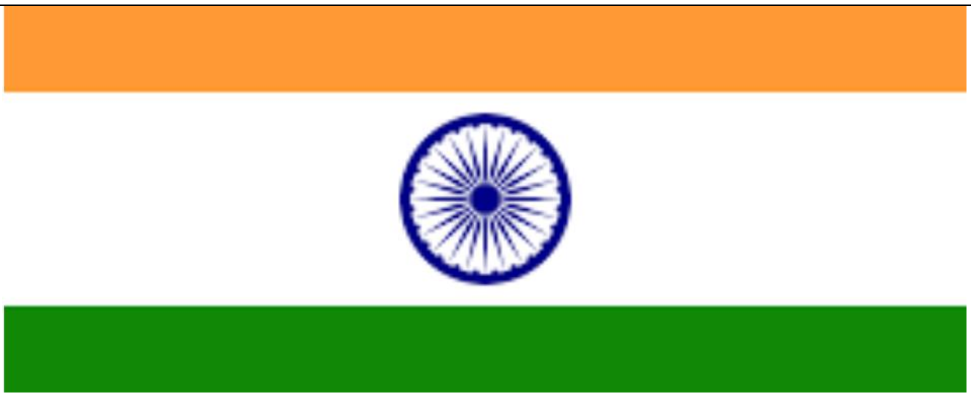
```
<div class="bg-contain" style="background-image: url(...)"></div>
```

Examples

```
<div class="m-4 h-96 w-1/2 bg-[url('2.png')]"></div>
```



```
<div class="m-4 h-96 w-1/2 bg-[url('2.png')] bg-no-repeat bg-cover bg-center"></div>
```



For understanding purpose

Via the style attribute:

If you prefer to skip the config then you can just add it using the style attribute, like this:

```
<div style="background-image: url('../public/images/hero.jpg');"></div>
```

Via using an arbitrary value:

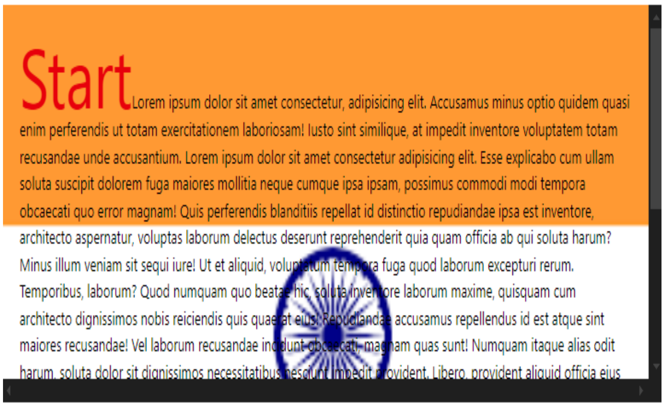
If you prefer not to use inline styles and have a single-use image then this is the best approach.

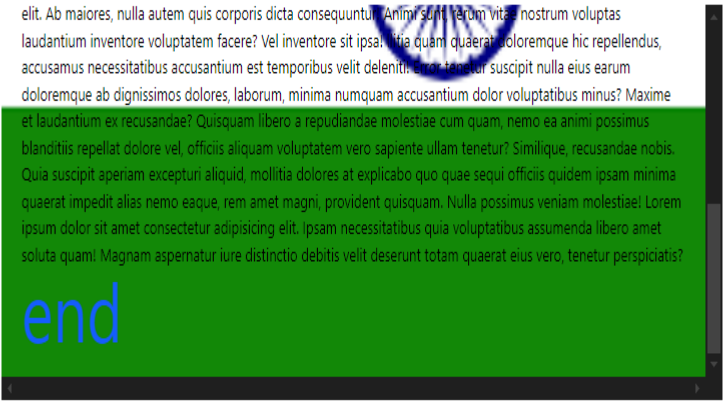
```
<div class="bg-[url('../public/images/hero.jpg')]"></div>
```

Tailwind

The background will move as you scroll

```
<div class="bg-[url(2.png)] h-72 p-5 m-6 overflow-scroll bg-local bg-cover">Kindly note take the content larger. (lorem*10)
</div>
```

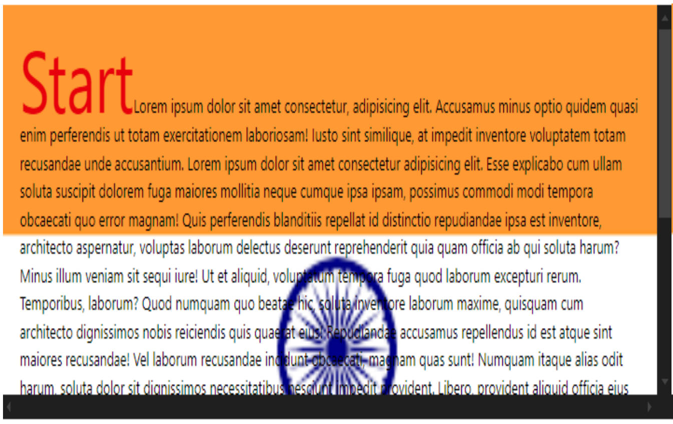

Start Lorem ipsum dolor sit amet consectetur, adipisicing elit. Accusamus minus optio quidem quasi enim perferendis ut totam exercitationem laboriosam! Iusto sint similique, at impedit inventore voluptatem totam recusandae unde accusantium. Lorem ipsum dolor sit amet consectetur adipisicing elit. Esse explicabo cum ullam soluta suscipit dolorem fuga maiores mollitia neque cumque ipsa ipsam, possimus commodi modi tempora obcaecati quo error magnam! Quis perferendis blanditiis repellat id distinctio repudiandae ipsa est inventore, architecto aspernatur, voluptas laborum delectus deserunt reprehenderit quia quam officia ab qui soluta harum? Minus illum veniam sit sequi iure! Ut et aliquid, voluptatum tempora fuga quod laborum excepturi rerum. Temporibus, laborum? Quod numquam quo beatae hic, soluta inventore laborum maxime, quisquam cum architecto dignissimos nobis reiciendis quis quaerat eius! Repudiandae accusamus repellendus id est atque sint maiores recusandae! Vel laborum recusandae incidunt obcaecati, magnam quas sunt! Numquam itaque alias odit harum, soluta dolor sit dignissimos necessitatibus nesciunt impedit provident. Libero, provident aliquid officia eius

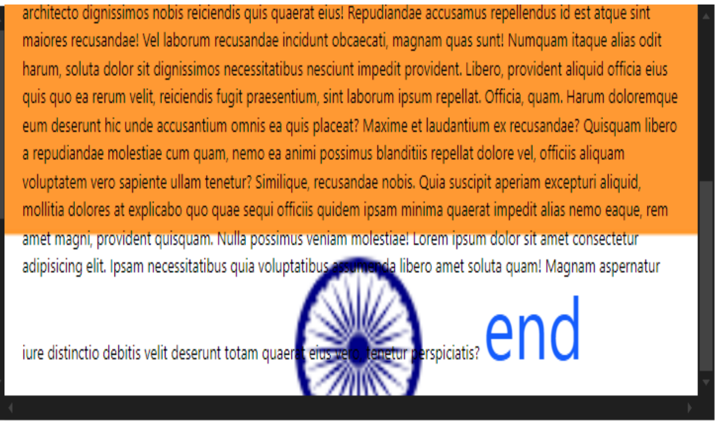

elit. Ab maiores, nulla autem quis corporis dicta consequuntur. Animi sunt, rerum vitae nostrum voluptas laudantium inventore voluptatem facere? Vel inventore sit ipsa! Mollitia quam quaerat doloremque hic repellendus, accusamus necessitatibus accusantium est temporibus velit deleniti! Error tenetur suscipit nulla eius earum doloremque ab dignissimos dolores, laborum, minima numquam accusantium dolor voluptatibus minus? Maxime et laudantium ex recusandae? Quisquam libero a repudiandae molestiae cum quam, nemo ea animi possimus blanditiis repellat dolore vel, officiis aliquam voluptatem vero sapiente ullam tenetur? Similique, recusandae nobis. Quia suscipit aperiam excepturi aliquid, mollitia dolores at explicabo quo quae sequi officiis quidem ipsam minima quaerat impedit alias nemo eaque, rem amet magni, provident quisquam. Nulla possimus veniam molestiae! Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsam necessitatibus quia voluptatibus assumenda libero amet soluta quam! Magnam aspernatur iure distinctio debitis velit deserunt totam quaerat eius vero, tenetur perspiciatis?

end

To set the background image to scroll with the content

```
<div class="bg-[url(2.png)] p-5 h-72 m-6 overflow-scroll bg-scroll bg-cover">Kindly note take the content larger. (lorem*10)
</div>
```


Start Lorem ipsum dolor sit amet consectetur, adipisicing elit. Accusamus minus optio quidem quasi enim perferendis ut totam exercitationem laboriosam! Iusto sint similique, at impedit inventore voluptatem totam recusandae unde accusantium. Lorem ipsum dolor sit amet consectetur adipisicing elit. Esse explicabo cum ullam soluta suscipit dolorem fuga maiores mollitia neque cumque ipsa ipsam, possimus commodi modi tempora obcaecati quo error magnam! Quis perferendis blanditiis repellat id distinctio repudiandae ipsa est inventore, architecto aspernatur, voluptas laborum delectus deserunt reprehenderit quia quam officia ab qui soluta harum? Minus illum veniam sit sequi iure! Ut et aliquid, voluptatum tempora fuga quod laborum excepturi rerum. Temporibus, laborum? Quod numquam quo beatae hic, soluta inventore laborum maxime, quisquam cum architecto dignissimos nobis reiciendis quis quaerat eius! Repudiandae accusamus repellendus id est atque sint maiores recusandae! Vel laborum recusandae incidunt obcaecati, magnam quas sunt! Numquam itaque alias odit harum, soluta dolor sit dignissimos necessitatibus nesciunt impedit provident. Libero, provident aliquid officia eius


architecto dignissimos nobis reiciendis quis quaerat eius! Repudiandae accusamus repellendus id est atque sint maiores recusandae! Vel laborum recusandae incidunt obcaecati, magnam quas sunt! Numquam itaque alias odit harum, soluta dolor sit dignissimos necessitatibus nesciunt impedit provident. Libero, provident aliquid officia eius quis quo ea rerum velit, reiciendis fugit praesentium, sint laborum ipsum repellat. Officia, quam. Harum doloremque eum deserunt hic unde accusantium omnis ea quis placeat? Maxime et laudantium ex recusandae? Quisquam libero a repudiandae molestiae cum quam, nemo ea animi possimus blanditiis repellat dolore vel, officiis aliquam voluptatem vero sapiente ullam tenetur? Similique, recusandae nobis. Quia suscipit aperiam excepturi aliquid, mollitia dolores at explicabo quo quae sequi officiis quidem ipsam minima quaerat impedit alias nemo eaque, rem amet magni, provident quisquam. Nulla possimus veniam molestiae! Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsam necessitatibus quia voluptatibus assumenda libero amet soluta quam! Magnam aspernatur

iure distinctio debitis velit deserunt totam quaerat eius vero, tenetur perspiciatis? end

overflow-scroll:

Ensures that the content inside the div is scrollable when it overflows the defined height. This creates a vertical or horizontal scrollbar when necessary.

Background Image

Utilities for controlling an element's background image.

To give an element a linear gradient background, use one of the bg-gradient-{direction} utilities, in combination with the gradient color stop utilities.

Classes

bg-none	This class is used **not to set any *linear-gradient***.
bg-gradient-to-t	This class is used to set the ***linear-gradient* to the top**.
bg-gradient-to-tr	This class is used to set the ***linear-gradient* to the top right**.
bg-gradient-to-r	This class is used to set the ***linear-gradient* to right**.
bg-gradient-to-br	This class is used to set the ***linear-gradient* to the bottom right**.
bg-gradient-to-b	This class is used to set the ***linear-gradient* to the bottom**.
bg-gradient-to-bl	This class is used to set the ***linear-gradient* to the bottom left**.
bg-gradient-to-l	This class is used to set the ***linear-gradient* to left**.
bg-gradient-to-tl	This class is used to set the ***linear-gradient* to the top left**.

Starting color

Set the starting color of a gradient using the **from-{color}** utilities.

```
<div class="bg-gradient-to-r from-indigo-500"></div>
```

Ending color

Set the ending color of a gradient using the **to-{color}** utilities.

```
<div class="bg-gradient-to-r from-cyan-500 to-blue-500"></div>
```

Gradients to do not fade in from transparent by default. To fade in from transparent, reverse the gradient direction and use a from-{color} utility.

Middle color

Add a middle color to a gradient using the **via-{color}** utilities.

```
<div class="bg-gradient-to-r from-indigo-500 via-purple-500 to-pink-500"></div>
```

Gradients with a from-{color} and via-{color} will fade out to transparent by default if no to-{color} is present.

Specifying stop positions

For more control over the gradient color stop positions, combine the **from-{position}, via-{position} and to-{position}** utilities with the gradient color utilities.

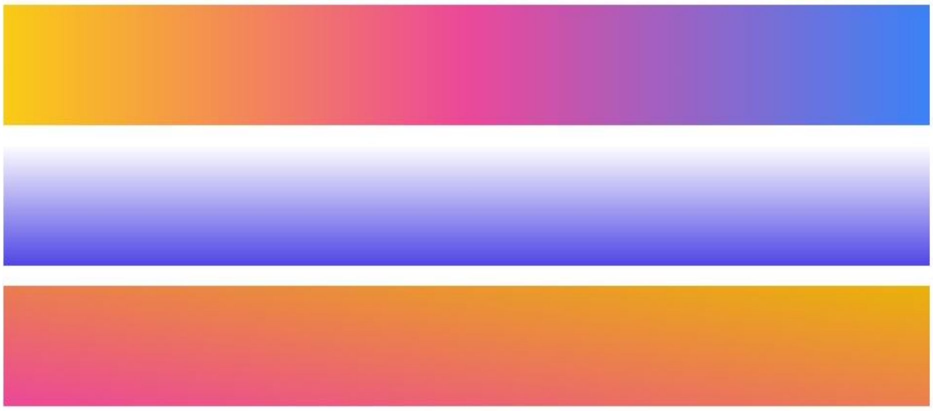
Example:

```
<div class="m-4 h-24 w-1/2 bg-gradient-to-r from-yellow-400 via-pink-500 to-blue-500"></div>
   <div class="m-4 h-24 w-1/2 bg-gradient-to-t from-indigo-600"></div>
```

Tailwind

```
<div class="m-4 h-24 w-1/2 bg-gradient-to-tr from-pink-500 to-yellow-500"></div>
```

Output



Add gradient to text

```
<h1 class=" text-9xl bg-gradient-to-r from-blue-600 via-green-500 to-indigo-400">Hello World</h1>
```

This will not add gradient to text. Follow below steps to add gradient to the text.

Step 1: Add the inline-block class to our element.

```
<h1 class=" text-9xl bg-gradient-to-r from-blue-600 via-green-500 to-indigo-400 inline-block">hello world</h1>
```

Now, the gradient takes up only the width of the text, instead of the width of the parent element.

Step 2: Make your text transparent

Make your text transparent using the text-transparent class.

```
<h1 class=" text-9xl bg-gradient-to-r from-blue-600 via-green-500 to-indigo-400 inline-block text-transparent">hello world</h1>
```

The text will disappear.

Step 3: Clip the text to the background

This is the most important thing for making gradient text, it uses the background-clip CSS property, which has multiple values and one of them is text, to use that specific property and value in Tailwind CSS we just need to add the **bg-clip-text** class.

```
<h1 class="text-9xl bg-gradient-to-r from-blue-600 via-green-500 to-indigo-400 inline-block text-transparent bg-clip-text ">hello world</h1>
```

hello world

- ✓ bg-gradient-to-r creates the gradient and makes it from left to right
- ✓ from-blue-600 sets our starting color, which will go on the left
- ✓ via-green-500 sets our middle color, which is optional
- ✓ to-indigo-400 sets our ending color, which will go on the right and can be optional if you want the end to be transparent

Background Opacity

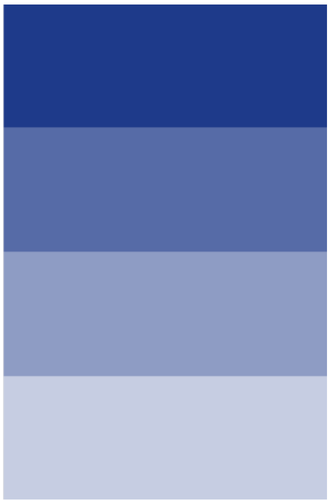
Utilities for controlling the opacity of an element's background color.

Control the opacity of an element’s background color using the **bg-opacity-{amount}** utilities.

bg-opacity-0	--tw-bg-opacity: 0;
bg-opacity-5	--tw-bg-opacity: 0.05;
bg-opacity-10	--tw-bg-opacity: 0.1;
bg-opacity-20	--tw-bg-opacity: 0.2;
bg-opacity-25	--tw-bg-opacity: 0.25;
bg-opacity-30	--tw-bg-opacity: 0.3;
bg-opacity-40	--tw-bg-opacity: 0.4;
bg-opacity-50	--tw-bg-opacity: 0.5;
bg-opacity-60	--tw-bg-opacity: 0.6;
bg-opacity-70	--tw-bg-opacity: 0.7;
bg-opacity-75	--tw-bg-opacity: 0.75;
bg-opacity-80	--tw-bg-opacity: 0.8;
bg-opacity-90	--tw-bg-opacity: 0.9;
bg-opacity-95	--tw-bg-opacity: 0.95;
bg-opacity-100	--tw-bg-opacity: 1;

Example:

```
<div class="h-48 w-1/4 bg-blue-900"></div>
   <div class="h-48 w-1/4 bg-blue-900 bg-opacity-75"></div>
   <div class="h-48 w-1/4 bg-blue-900 bg-opacity-50"></div>
   <div class="h-48 w-1/4 bg-blue-900 bg-opacity-25"></div>
```



Borders

border-width

This class is used to set the border width of all four sides of an element.
Tailwind CSS follows a consistent naming convention for border width classes:
border-{side}-{width}.
side can be **t** (top), **b** (bottom), **l** (left), or **r**(right)

Border Width Classes:

border	border-0	border-2	border-4
border-8	border-t	border-t-0	border-t-2
border-t-4	border-t-8	border-r	border-r-0
border-r-2	border-r-4	border-r-8	border-b
border-b-0	border-b-2	border-b-4	border-b-8
border-l	border-l-0	border-l-2	border-l-4
border-l-8			

border-color

This class is used to specify the border color of an element.

Border Color classes:

border-transparent	The border color will be transparent.
border-current	The border will color depend on the parent element color.
border-black	The border color will be black.
border-white	The border color will be white.
border-gray-50	The border color will be gray.
border-red-50	The border color will be red.
border-blue-50	The border color will be blue.
border-indigo-50	The border color will be indigo.
border-purple-50	The border color will be purple.
border-green-50	The border color will be green.
border-yellow-50	The border color will be yellow.
border-pink-50	The border color will be pink.

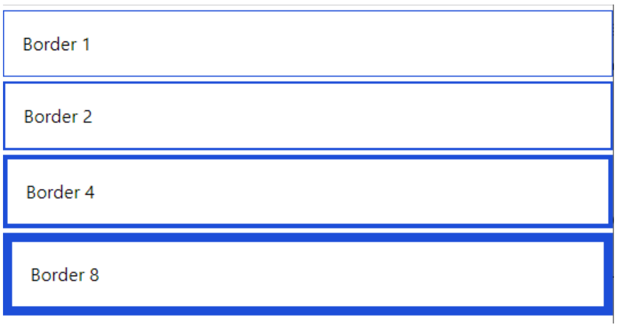
Note: The color's values can be changeable according to your need from 50-900, the span should be 100, after the 100.

Example:

```
<div>
    <div class="p-4 mt-1 border border-blue-700">Border 1</div>
    <div class="p-4 mt-1 border-2 border-blue-700">Border 2</div>
    <div class="p-4 mt-1 border-4 border-blue-700">Border 4</div>
```

```
        <div class="p-4 mt-1 border-8 border-blue-700">Border 8</div>
    </div>
```

Output:



Border 1

Border 2

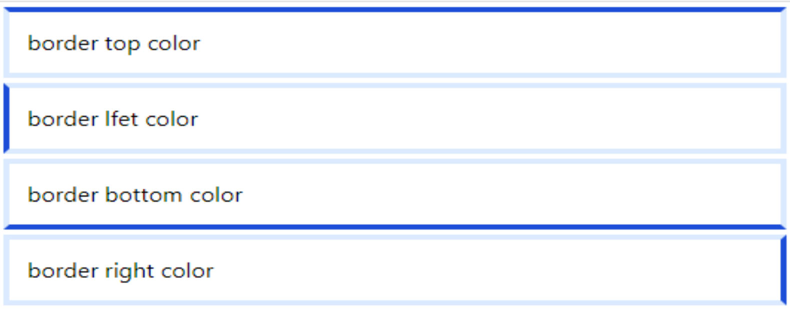
Border 4

Border 8

Example: Applying Border Color to Individual Sides

```
<div>
        <div class="border-4 m-1 border-blue-100 border-t-blue-700 p-3">border top
        color</div>
        <div class="border-4 m-1 border-blue-100 border-l-blue-700 p-3">border lfet
        color</div>
        <div class="border-4 m-1 border-blue-100 border-b-blue-700 p-3">border
        bottom color</div>
        <div class="border-4 m-1 border-blue-100 border-r-blue-700 p-3">border right
        color</div>
    </div>
```

Output:



border top color

border lfet color

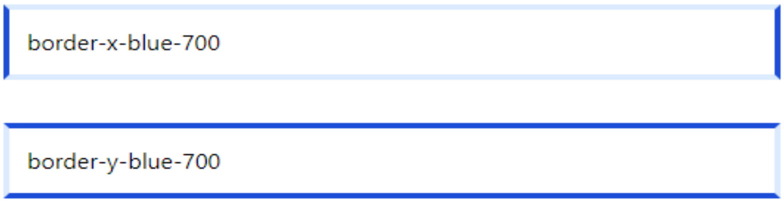
border bottom color

border right color

Example: Border color horizontal and vertical sides using border-{x|y}-{color}

```
<div>
        <div class="border-4 m-1 border-blue-100 border-x-blue-700 p-3">border-x-blue-
700</div>
        <br>
        <div class="border-4 m-1 border-blue-100 border-y-blue-700 p-3">border-y-blue-
700</div>
 </div>
```

Output:



border-x-blue-700

border-y-blue-700

border-style

This class is used for controlling the style of an element’s borders.

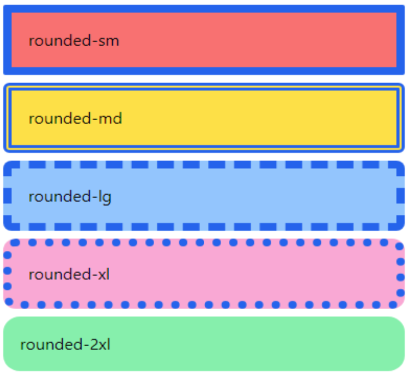
Border Style Classes:

border-solid
border-dashed
border-dotted
border-double
border-none
border-hidden

Example: Border-style

```
   <div class="w-96 rounded-sm border-8 border-blue-600 bg-red-400 border-solid p-4 m-2">rounded-sm</div>
   <div class="w-96 rounded-md border-8 border-blue-600 bg-yellow-300 border-double p-4 m-2">rounded-md</div>
   <div class="w-96 rounded-lg border-8 border-blue-600 bg-blue-300 border-dashed p-4 m-2">rounded-lg</div>
   <div class="w-96 rounded-xl border-8 border-blue-600 bg-pink-300 border-dotted p-4 m-2">rounded-xl</div>
   <div class="w-96 rounded-2xl border-8 border-blue-600 bg-green-300 border-none p-4 m-2">rounded-2xl</div>
```

Output:



rounded-sm

rounded-md

rounded-lg

rounded-xl

rounded-2xl

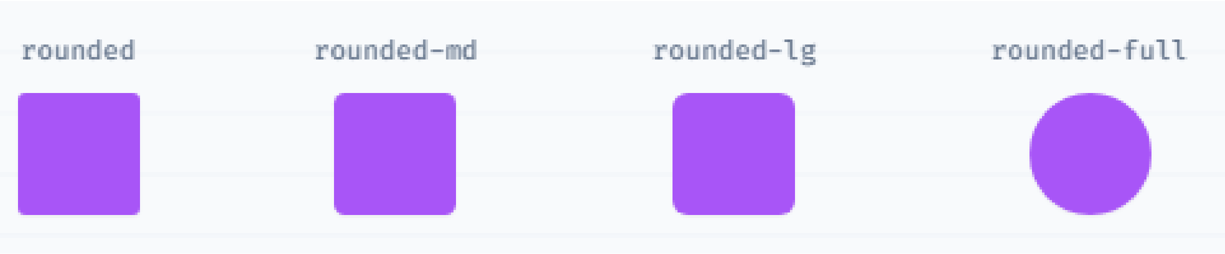
border-radius

Utilities for controlling the border radius of an element.

Use utilities like rounded-sm, rounded, or rounded-lg to apply different border radius sizes to an element.

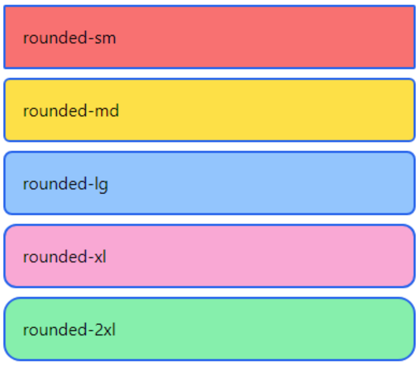
Tailwind

rounded-none	border-radius: 0px;
rounded-sm	border-radius: 0.125rem; /* 2px */
rounded	border-radius: 0.25rem; /* 4px */
rounded-md	border-radius: 0.375rem; /* 6px */
rounded-lg	border-radius: 0.5rem; /* 8px */
rounded-xl	border-radius: 0.75rem; /* 12px */
rounded-2xl	border-radius: 1rem; /* 16px */
rounded-3xl	border-radius: 1.5rem; /* 24px */
rounded-full	border-radius: 9999px;



```
<div class="w-96 rounded-sm border-2 border-blue-600 bg-red-400 p-4 m-2">rounded-sm</div>
    <div class="w-96 rounded-md border-2 border-blue-600 bg-yellow-300 p-4 m-2">rounded-md</div>
    <div class="w-96 rounded-lg border-2 border-blue-600 bg-blue-300 p-4 m-2">rounded-lg</div>
    <div class="w-96 rounded-xl border-2 border-blue-600 bg-pink-300 p-4 m-2">rounded-xl</div>
    <div class="w-96 rounded-2xl border-2 border-blue-600 bg-green-300 p-4 m-2">rounded-2xl</div>
```

Output:



rounded-sm

rounded-md

rounded-lg

rounded-xl

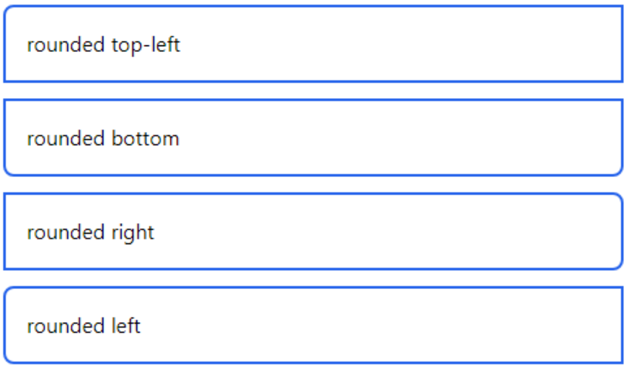
rounded-2xl

Example: Rounded Corners by Specific Sides

```
<div class="rounded-tl-lg border-blue-600 p-4 border-[2px] m-3">
rounded top-left</div>
<div class="rounded-b-lg border-blue-600 p-4 border-[2px] m-3">
rounded bottom</div>
<div class="rounded-r-lg border-blue-600 p-4 border-[2px] m-3">
```

```
rounded right</div>
<div class="rounded-l-lg border-blue-600 p-4 border-[2px] m-3">
rounded left</div>
```

Output:



rounded top-left

rounded bottom

rounded right

rounded left

Layout

Container

- ✓ In Tailwind CSS, a container is used to fix the max-width of an element to match the min-width of the breakpoint. It comes very handy when content has to be displayed in a responsive manner to every breakpoint.

Breakpoints in tailwind CSS are as follows.

Breakpoint	min-width
sm	640px
md	768px
lg	1024px
xl	1280px
2xl	1536px

Tailwind CSS does not center itself automatically and also does not contain any pre-defined padding.

The following are some utility classes that make the container class stand out.

mx-auto: To center the container, we use mx-auto utility class. It adjusts the margin of the container automatically so that the container appears to be in center.

Syntax:

```
<element class=" container mx-auto">...</element>
```

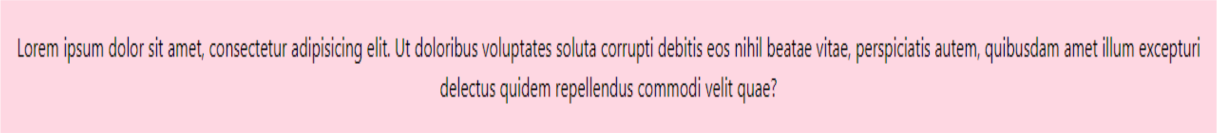
px-{size}: To add padding the container, we use px-{size} utility class. It adds horizontal padding to the container which is equal to the size mentioned.

Syntax:

```
<element class="container px-20">...</element>
```

Example:

```
<div class="container mx-auto p-4 bg-pink-200">
      <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Ut doloribus voluptates soluta corrupti debitis eos nihil beatae vitae, perspiciatis autem, quibusdam amet illum excepturi delectus quidem repellendus commodi velit quae?</p></div>
```



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Ut doloribus voluptates soluta corrupti debitis eos nihil beatae vitae, perspiciatis autem, quibusdam amet illum excepturi delectus quidem repellendus commodi velit quae?

Display

Utilities for controlling the display box type of an element.

Use inline, inline-block, and block to control the flow of text and elements.

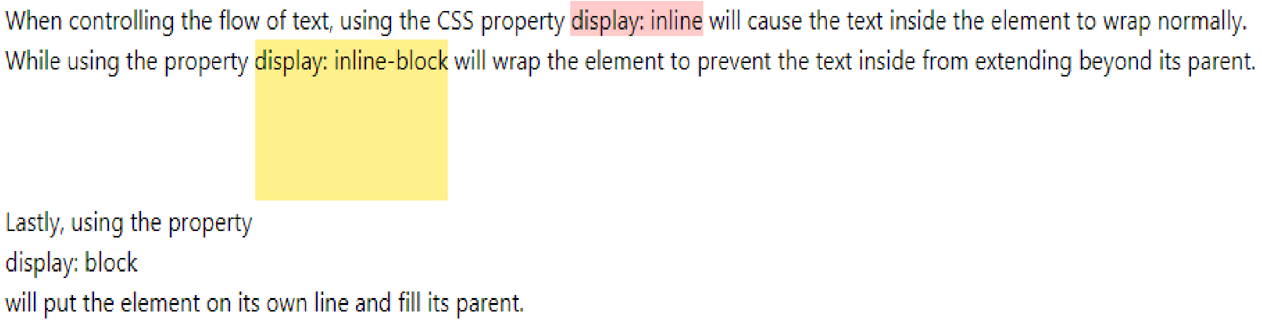
block	display: block;

Tailwind

inline-block	display: inline-block;
inline	display: inline;
hidden	display: none;

Example

```
<div>
When controlling the flow of text, using the CSS property
<p class="inline h-24 bg-red-200">display: inline</p>
will cause the text inside the element to wrap normally.</div>

<div>
While using the property
<p class="inline-block h-24 bg-yellow-200">display: inline-block</p>
will wrap the element to prevent the text inside from extending beyond its parent.
</div>

<div>
Lastly, using the property
<span class="block">display: block</span>
will put the element on its own line and fill its parent.
</div>
```



When controlling the flow of text, using the CSS property display: inline will cause the text inside the element to wrap normally.
While using the property display: inline-block will wrap the element to prevent the text inside from extending beyond its parent.

Lastly, using the property
display: block
will put the element on its own line and fill its parent.

Floats

Utilities for controlling the wrapping of content around an element.

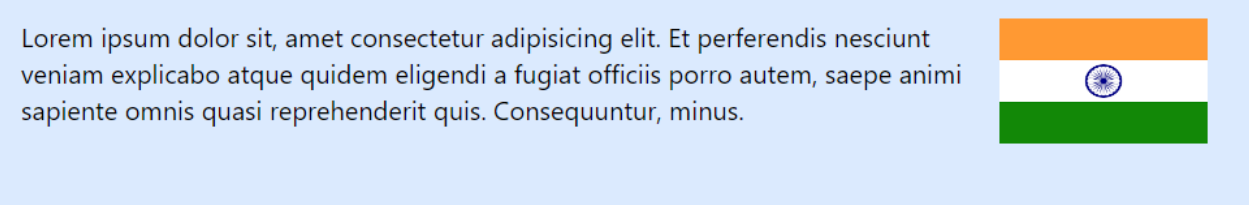
float-right	float: right;
float-left	float: left;
float-none	float: none;

Floating elements to the right

Use float-right to float an element to the right of its container.

```
<div class="m-4 p-4 bg-blue-100 h-40 w-1/2">
<img src="2.png" class="float-right px-4 w-48 h-24">
```

```
<p class="text-xl">Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.</p>
</div>
```

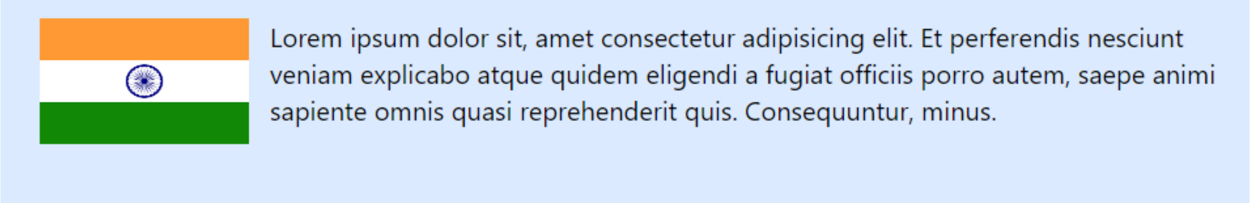


Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.

Floating elements to the left

Use float-left to float an element to the left of its container.

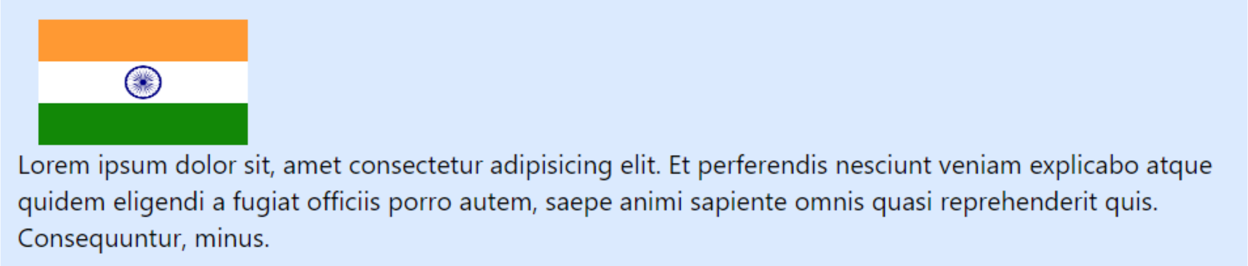
```
<div class="m-4 p-4 bg-blue-100 h-40 w-1/2">
  <img src="2.png" class="float-left px-4 w-48 h-24">
  <p class="text-xl ">Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.</p>
  </div>
```



Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.

Float None

```
<div class="m-4 p-4 bg-blue-100 h-52 w-1/2">
    <img src="2.png" class="float-none px-4 w-48 h-24">
    <p class="text-xl ">Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.</p>
    </div>
```



Lorem ipsum dolor sit, amet consectetur adipisicing elit. Et perferendis nesciunt veniam explicabo atque quidem eligendi a fugiat officiis porro autem, saepe animi sapiente omnis quasi reprehenderit quis. Consequuntur, minus.

Effects

Box Shadow

Utilities for controlling the box shadow of an element.

Use the **shadow-sm, shadow, shadow-md, shadow-lg, shadow-xl, or shadow-2xl** utilities to apply different sized outer box shadows to an element.

shadow-sm
shadow
shadow-md
shadow-lg
shadow-xl
shadow-2xl

Example:

```
<div class="h-24 w-24 bg-black "></div>
```

```
<div class="h-24 w-24 bg-black shadow-2xl bg-yellow-200"></div>
```

Box Shadow Color

Utilities for controlling the color of a box shadow.

Use the **shadow-{color}** utilities to change the color of an existing box shadow. By default colored shadows have an opacity of 100%, but you can adjust this using the opacity modifier.

shadow-inherit	shadow-current	shadow-transparent	shadow-black	shadow-white
shadow-slate-50	shadow-slate-100	shadow-slate-200	shadow-slate-300	shadow-slate-400
shadow-slate-500	shadow-slate-600	shadow-slate-700	shadow-slate-800	shadow-slate-900

Tailwind

shadow-slate-950	shadow-gray-50	shadow-gray-100	shadow-gray-200	shadow-gray-300
shadow-gray-400	shadow-gray-500	shadow-gray-600	shadow-gray-700	shadow-gray-800
shadow-gray-900	shadow-gray-950	shadow-zinc-50	shadow-zinc-100	shadow-zinc-200
shadow-zinc-300	shadow-zinc-400	shadow-zinc-500	shadow-zinc-600	shadow-zinc-700
shadow-zinc-800	shadow-zinc-900	shadow-zinc-950	shadow-neutral-50	shadow-neutral-100
shadow-neutral-200	shadow-neutral-300	shadow-neutral-400	shadow-neutral-500	shadow-neutral-600
shadow-neutral-700	shadow-neutral-800	shadow-neutral-900	shadow-neutral-950	shadow-stone-50
shadow-stone-100	shadow-stone-200	shadow-stone-300	shadow-stone-400	shadow-stone-500
shadow-stone-600	shadow-stone-700	shadow-stone-800	shadow-stone-900	shadow-stone-950
shadow-red-50	shadow-red-100	shadow-red-200	shadow-red-300	shadow-red-400
shadow-red-500	shadow-red-600	shadow-red-700	shadow-red-800	shadow-red-900
shadow-red-950	shadow-orange-50	shadow-orange-100	shadow-orange-200	shadow-orange-300
shadow-orange-400	shadow-orange-500	shadow-orange-600	shadow-orange-700	shadow-orange-800
shadow-orange-900	shadow-orange-950	shadow-amber-50	shadow-amber-100	shadow-amber-200
shadow-amber-300	shadow-amber-400	shadow-amber-500	shadow-amber-600	shadow-amber-700
shadow-amber-800	shadow-amber-900	shadow-amber-950	shadow-yellow-50	shadow-yellow-100
shadow-yellow-200	shadow-yellow-300	shadow-yellow-400	shadow-yellow-500	shadow-yellow-600
shadow-yellow-700	shadow-yellow-800	shadow-yellow-900	shadow-yellow-950	shadow-lime-50
shadow-lime-100	shadow-lime-200	shadow-lime-300	shadow-lime-400	shadow-lime-500
shadow-lime-600	shadow-lime-700	shadow-lime-800	shadow-lime-900	shadow-lime-950
shadow-green-50	shadow-green-100	shadow-green-200	shadow-green-300	shadow-green-400

Tailwind

shadow-green-500	shadow-green-600	shadow-green-700	shadow-green-800	shadow-green-900
shadow-green-950	shadow-emerald-50	shadow-emerald-100	shadow-emerald-200	shadow-emerald-300
shadow-emerald-400	shadow-emerald-500	shadow-emerald-600	shadow-emerald-700	shadow-emerald-800
shadow-emerald-900	shadow-emerald-950	shadow-teal-50	shadow-teal-100	shadow-teal-200
shadow-teal-300	shadow-teal-400	shadow-teal-500	shadow-teal-600	shadow-teal-700
shadow-teal-800	shadow-teal-900	shadow-teal-950	shadow-cyan-50	shadow-cyan-100
shadow-cyan-200	shadow-cyan-300	shadow-cyan-400	shadow-cyan-500	shadow-cyan-600
shadow-cyan-700	shadow-cyan-800	shadow-cyan-900	shadow-cyan-950	shadow-sky-50
shadow-sky-100	shadow-sky-200	shadow-sky-300	shadow-sky-400	shadow-sky-500
shadow-sky-600	shadow-sky-700	shadow-sky-800	shadow-sky-900	shadow-sky-950
shadow-blue-50	shadow-blue-100	shadow-blue-200	shadow-blue-300	shadow-blue-400
shadow-blue-500	shadow-blue-600	shadow-blue-700	shadow-blue-800	shadow-blue-900
shadow-blue-950	shadow-indigo-50	shadow-indigo-100	shadow-indigo-200	shadow-indigo-300
shadow-indigo-400	shadow-indigo-500	shadow-indigo-600	shadow-indigo-700	shadow-indigo-800
shadow-indigo-900	shadow-indigo-950	shadow-violet-50	shadow-violet-100	shadow-violet-200
shadow-violet-300	shadow-violet-400	shadow-violet-500	shadow-violet-600	shadow-violet-700
shadow-violet-800	shadow-violet-900	shadow-violet-950	shadow-purple-50	shadow-purple-100
shadow-purple-200	shadow-purple-300	shadow-purple-400	shadow-purple-500	shadow-purple-600
shadow-purple-700	shadow-purple-800	shadow-purple-900	shadow-purple-950	shadow-fuchsia-50
shadow-fuchsia-100	shadow-fuchsia-200	shadow-fuchsia-300	shadow-fuchsia-400	shadow-fuchsia-500
shadow-fuchsia-600	shadow-fuchsia-700	shadow-fuchsia-800	shadow-fuchsia-900	shadow-fuchsia-950

Tailwind

shadow-pink-50	shadow-pink-100	shadow-pink-200	shadow-pink-300	shadow-pink-400
shadow-pink-500	shadow-pink-600	shadow-pink-700	shadow-pink-800	shadow-pink-900
shadow-pink-950	shadow-rose-50	shadow-rose-100	shadow-rose-200	shadow-rose-300
shadow-rose-400	shadow-rose-500	shadow-rose-600	shadow-rose-700	shadow-rose-800
shadow-rose-900	shadow-rose-950			

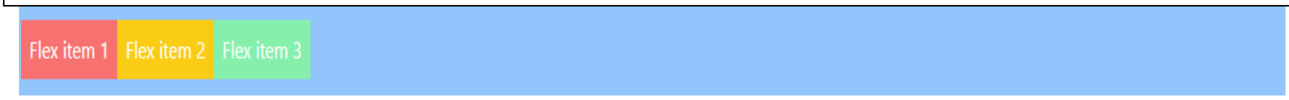
Flexbox

- ✓ The CSS flexbox is a vital feature to develop the frontend.
- ✓ It is the alternative of CSS flex Property for fast development of front-end. It is used to set the length of flexible items. The flex class is much responsive and mobile-friendly.

Display

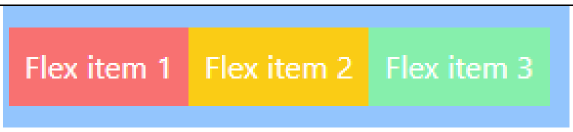
flex: It is used to display an element as a **block-level** flex container.

```
<div class="flex p-3 bg-blue-300 text-white">
   <div class="p-2 bg-red-400">Flex item 1</div>
   <div class="p-2 bg-yellow-400">Flex item 2</div>
   <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```


Flex item 1 Flex item 2 Flex item 3

inline-flex: It is used to display an element as an **inline-level** flex container.

```
<div class="inline-flex p-3 bg-blue-300 text-white">
   <div class="p-2 bg-red-400">Flex item 1</div>
   <div class="p-2 bg-yellow-400">Flex item 2</div>
   <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```


Flex item 1 Flex item 2 Flex item 3

Flex Direction

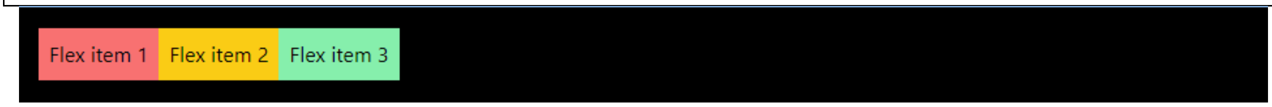
Utilities for controlling the direction of flex items.

flex-row	flex-direction: row;
flex-row-reverse	flex-direction: row-reverse;
flex-col	flex-direction: column;
flex-col-reverse	flex-direction: column-reverse;

Row

Use **flex-row** to position flex items horizontally in the same direction as text:

```
<div class="w-1/2 flex flex-row bg-black p-4">
   <div class="p-2 bg-red-400">Flex item 1</div>
   <div class="p-2 bg-yellow-400">Flex item 2</div>
   <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```

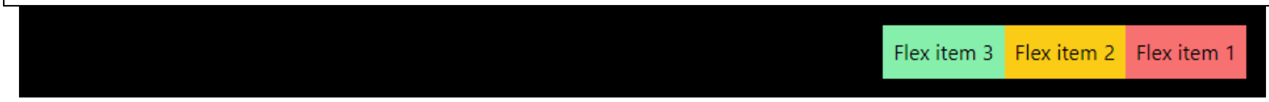

Flex item 1 Flex item 2 Flex item 3

Row reversed

Tailwind

Use **flex-row-reverse** to position flex items horizontally in the opposite direction:

```
 <div class="w-1/2 flex flex-row-reverse bg-black p-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```

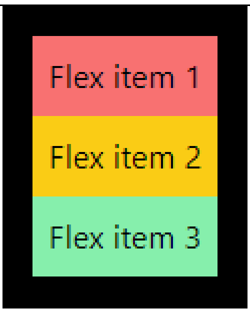


Flex item 3 Flex item 2 Flex item 1

Column

Use **flex-col** to position flex items vertically:

```
 <div class="m-4 inline-flex flex-col bg-black p-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```

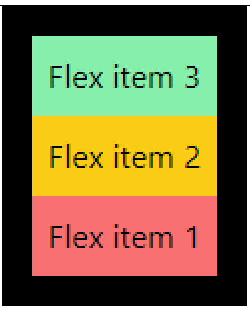


Flex item 1
Flex item 2
Flex item 3

Column reversed

Use **flex-col-reverse** to position flex items vertically in the opposite direction:

```
<div class="inline-flex flex-col-reverse bg-black p-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
 </div>
```



Flex item 3
Flex item 2
Flex item 1

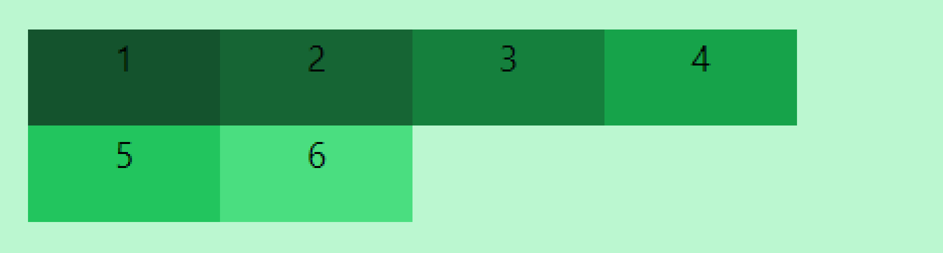
Flex wrap

Wrap

Use **.flex-wrap** to allow flex items to wrap:

Tailwind

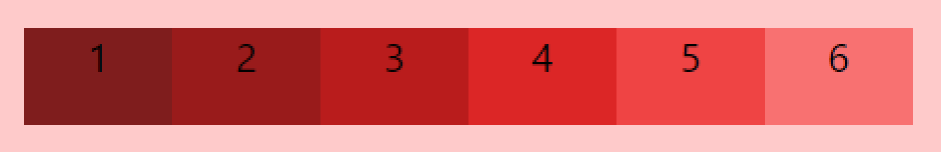
```
<div class="m-24 w-1/4 flex flex-wrap bg-green-200 p-4 text-center text-lg">
<div class="bg-green-900 w-24 h-12">1</div>
<div class="bg-green-800 w-24 h-12">2</div>
<div class="bg-green-700 w-24 h-12">3</div>
<div class="bg-green-600 w-24 h-12">4</div>
<div class="bg-green-500 w-24 h-12">5</div>
<div class="bg-green-400 w-24 h-12">6</div>
</div>
```



No Wrap

Use **.flex-no-wrap** to prevent flex items from wrapping, causing inflexible items to overflow the container if necessary:

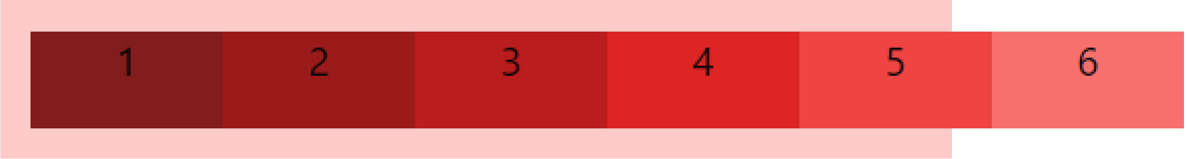
```
<div class="m-24 w-1/4 flex flex-no-wrap bg-red-200 p-4 text-center text-xl">
   <div class="bg-red-900 w-24 h-12">1</div>
   <div class="bg-red-800 w-24 h-12">2</div>
   <div class="bg-red-700 w-24 h-12">3</div>
   <div class="bg-red-600 w-24 h-12">4</div>
   <div class="bg-red-500 w-24 h-12">5</div>
   <div class="bg-red-400 w-24 h-12">6</div>
   </div>
```



```
<div class="m-24 w-1/4 flex flex-no-wrap bg-red-200 p-4 text-center text-xl">
   <div class="flex-none bg-red-900 w-24 h-12">1</div>
   <div class="flex-none bg-red-800 w-24 h-12">2</div>
   <div class="flex-none bg-red-700 w-24 h-12">3</div>
   <div class="flex-none bg-red-600 w-24 h-12">4</div>
   <div class="flex-none bg-red-500 w-24 h-12">5</div>
   <div class="flex-none bg-red-400 w-24 h-12">6</div>
```

```
  </div>
```

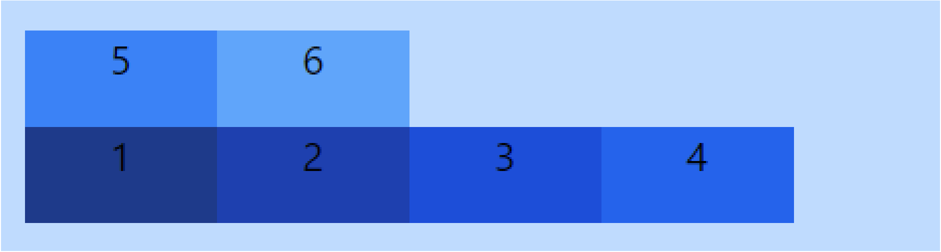
Note: Use **flex-none** to prevent a flex item from growing or shrinking



Wrap reversed

Use **.flex-wrap-reverse** to wrap flex items in the reverse direction:

```
<div class="m-24 w-1/4 flex flex-wrap-reverse bg-blue-200 p-4 text-center text-xl">
  <div class="bg-blue-900 w-24 h-12">1</div>
  <div class="bg-blue-800 w-24 h-12">2</div>
  <div class="bg-blue-700 w-24 h-12">3</div>
  <div class="bg-blue-600 w-24 h-12">4</div>
  <div class="bg-blue-500 w-24 h-12">5</div>
  <div class="bg-blue-400 w-24 h-12">6</div>
  </div>
```



Box Alignment

Justify Content

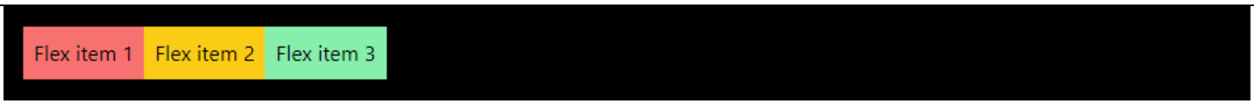
Utilities for controlling how flex and grid items are positioned along a container's main axis.

justify-start	justify-content: flex-start;
justify-end	justify-content: flex-end;
justify-center	justify-content: center;
justify-between	justify-content: space-between;
justify-around	justify-content: space-around;
justify-evenly	justify-content: space-evenly;

Start

Use **justify-start** to justify items against the start of the container's main axis:

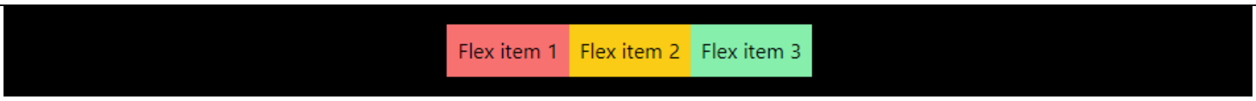
```
<div class="w-1/2 flex justify-start bg-black p-4 m-4">
        <div class="p-2 bg-red-400">Flex item 1</div>
        <div class="p-2 bg-yellow-400">Flex item 2</div>
        <div class="p-2 bg-green-300">Flex item 3</div>
      </div>
```



Center

Use **justify-center** to justify items along the center of the container's main axis:

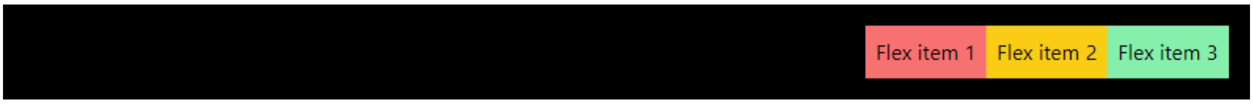
```
      <div class="w-1/2 flex justify-center bg-black p-4 m-4">
        <div class="p-2 bg-red-400">Flex item 1</div>
        <div class="p-2 bg-yellow-400">Flex item 2</div>
        <div class="p-2 bg-green-300">Flex item 3</div>
      </div>
```



End

Use **justify-end** to justify items against the end of the container's main axis:

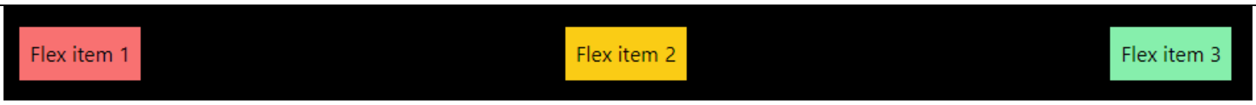
```
<div class="w-1/2 flex justify-end bg-black p-4 m-4">
        <div class="p-2 bg-red-400">Flex item 1</div>
        <div class="p-2 bg-yellow-400">Flex item 2</div>
        <div class="p-2 bg-green-300">Flex item 3</div>
      </div>
```



Space between

Use **justify-between** to justify items along the container's main axis such that there is an equal amount of space between each item:

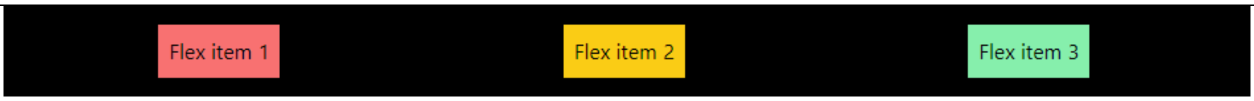
```
<div class="w-1/2 flex justify-between bg-black p-4 m-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
</div>
```



Space around

Use **justify-around** to justify items along the container's main axis such that there is an equal amount of space on each side of each item:

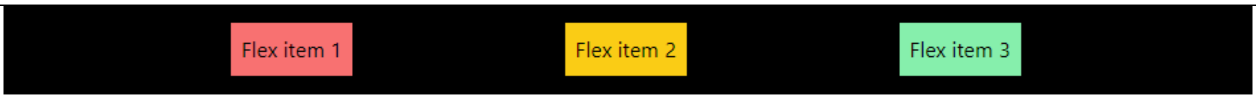
```
<div class="w-1/2 flex justify-around bg-black p-4 m-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
</div>
```



Space evenly

Use **justify-evenly** to justify items along the container's main axis such that there is an equal amount of space around each item, but also accounting for the doubling of space you would normally see between each item when using justify-around:

```
<div class="w-1/2 flex justify-evenly bg-black p-4 m-4">
  <div class="p-2 bg-red-400">Flex item 1</div>
  <div class="p-2 bg-yellow-400">Flex item 2</div>
  <div class="p-2 bg-green-300">Flex item 3</div>
</div>
```



Justify Items

This class accepts two values in tailwind CSS. It is the alternative to the CSS justify-items property. This class is used for controlling how grid items are aligned along their inline axis.

Justify Items:

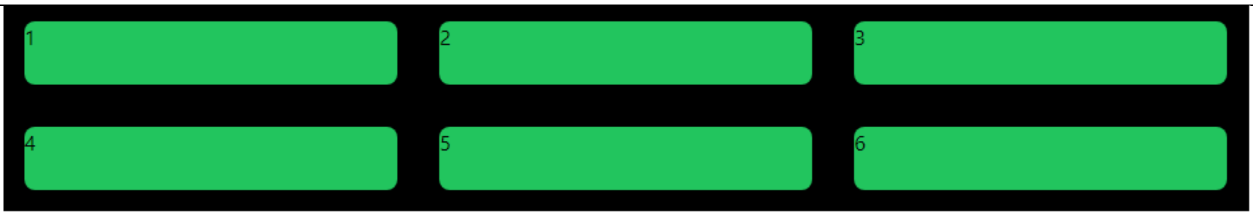
justify-items-auto
justify-items-start
justify-items-end
justify-items-center
justify-items-stretch

justify-items-auto: This class is used to justify grid items automatically on their inline axis.

```
<element class="justify-items-auto">...</element>
```

Example:

```
<div id="main" class="w-1/2 grid justify-items-auto grid-cols-3 bg-black m-4">
        <div class="bg-green-500 rounded-lg m-4 h-12">1</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">2</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">3</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">4</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">5</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">6</div>
     </div>
```

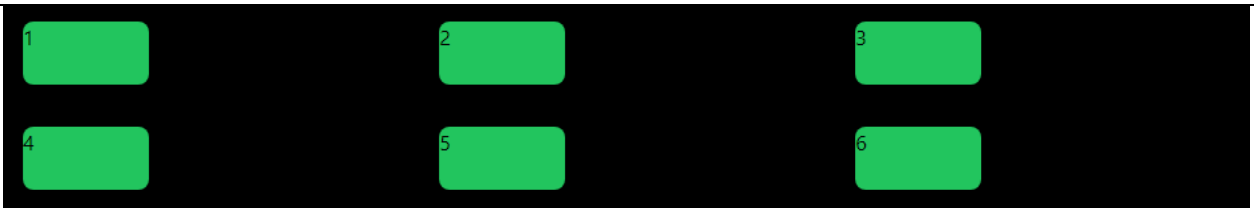


justify-items-start: This class is used to justify grid items against the start of their inline axis.

```
<element class="justify-items-start">...</element>
```

Example:

```
 <div id="main" class="w-1/2 grid justify-items-start grid-cols-3 bg-black m-4">
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">1</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">2</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">3</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">4</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">5</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">6</div>
    </div>
```

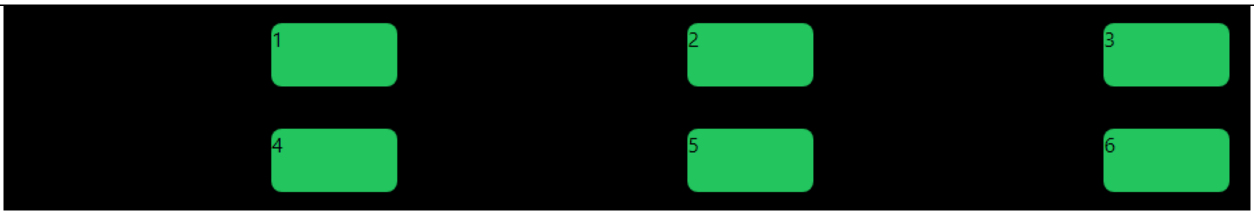


justify-items-end: This class is used to justify grid items against the end of their inline axis.

```
<element class="justify-items-end">...</element>
```

Example:

```
 <div id="main" class="w-1/2 grid justify-items-end grid-cols-3 bg-black m-4">
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">1</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">2</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">3</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">4</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">5</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">6</div>
    </div>
```

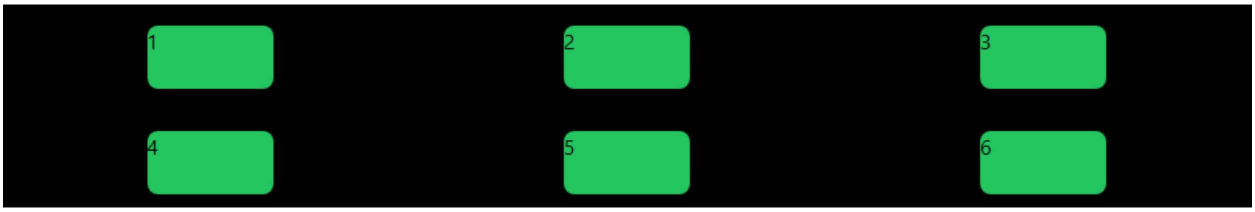


justify-items-center: This class is used to justify grid items along their inline axis.

```
<element class="jjustify-items-center">...</element>
```

Example:

```
<div id="main" class="w-1/2 grid justify-items-center grid-cols-3 bg-black m-4">
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">1</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">2</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">3</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">4</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">5</div>
        <div class="bg-green-500 rounded-lg m-4 h-12 w-24">6</div>
    </div>
```

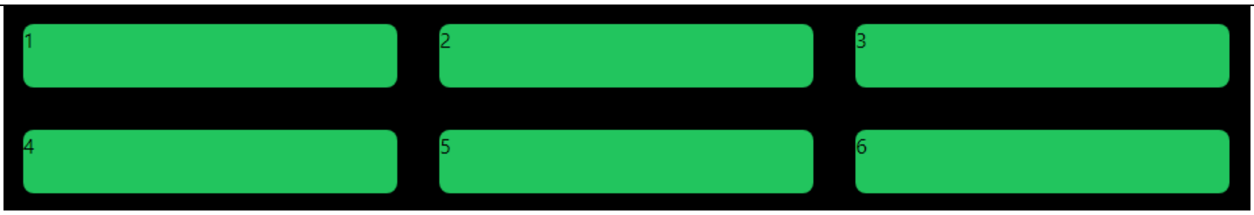


justify-items-stretch: This class is used to stretch items along their inline axis.

```
<element class="justify-items-stretch">...</element>
```

Example:

```
  <div id="main" class="w-1/2 grid justify-items-stretch grid-cols-3 bg-black m-4">
        <div class="bg-green-500 rounded-lg m-4 h-12">1</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">2</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">3</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">4</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">5</div>
        <div class="bg-green-500 rounded-lg m-4 h-12">6</div>
    </div>
```



Grid

Grid Template Columns

It is used to set the number of columns and size of the columns of the grid, here we will do the same but for fast development of front-end. The number of columns is set by the number of values given to this class.

grid-cols-1	Each row concedes only one column.
grid-cols-2	Each row concedes only two columns.
grid-cols-3	Each row concedes only three columns.
grid-cols-4	Each row concedes only four columns.
grid-cols-5	Each row concedes only five columns.
grid-cols-6	Each row concedes six columns.
grid-cols-7	Each row concedes seven-columns.
grid-cols-8	Each row concedes eight columns.
grid-cols-9	Each row concedes nine columns.
grid-cols-10	Each row concedes ten columns.
grid-cols-11	Each row concedes eleven columns.
grid-cols-12	Each row concedes twelve columns.
grid-cols-none	Does not follow the grid-column property.

Syntax:

```
<element class="grid grid-cols-number"> Contents... </element>
```

Gap

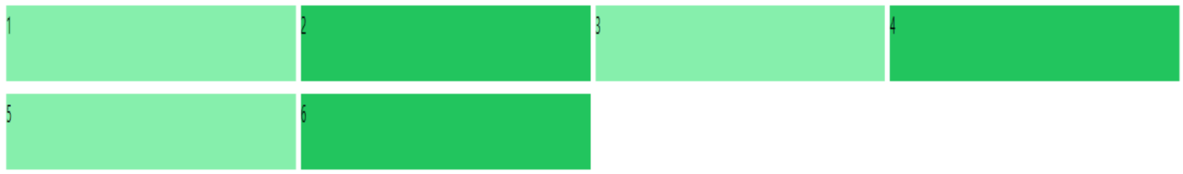
Utilities for controlling gutters between grid and flexbox items.

gap-0	gap-x-0	gap-y-0	gap-px	gap-x-px
gap-y-px	gap-0.5	gap-x-0.5	gap-y-0.5	gap-1
gap-x-1	gap-y-1	gap-1.5	gap-x-1.5	gap-y-1.5
gap-2	gap-x-2	gap-y-2	gap-2.5	gap-x-2.5
gap-y-2.5	gap-3	gap-x-3	gap-y-3	gap-3.5
gap-x-3.5	gap-y-3.5	gap-4	gap-x-4	gap-y-4
gap-5	gap-x-5	gap-y-5	gap-6	gap-x-6
gap-y-6	gap-7	gap-x-7	gap-y-7	gap-8
gap-x-8	gap-y-8	gap-9	gap-x-9	gap-y-9
gap-10	gap-x-10	gap-y-10	gap-11	gap-x-11
gap-y-11	gap-12	gap-x-12	gap-y-12	gap-14
gap-x-14	gap-y-14	gap-16	gap-x-16	gap-y-16
gap-20	gap-x-20	gap-y-20	gap-24	gap-x-24
gap-y-24	gap-28	gap-x-28	gap-y-28	gap-32
gap-x-32	gap-y-32	gap-36	gap-x-36	gap-y-36
gap-40	gap-x-40	gap-y-40	gap-44	gap-x-44
gap-y-44	gap-48	gap-x-48	gap-y-48	gap-52
gap-x-52	gap-y-52	gap-56	gap-x-56	gap-y-56

gap-60	gap-x-60	gap-y-60	gap-64	gap-x-64
gap-y-64	gap-72	gap-x-72	gap-y-72	gap-80
gap-x-80	gap-y-80	gap-96	gap-x-96	gap-y-96

Example:

```
        <div id="main" class="grid grid-cols-4 gap-2 p-4">
                <div class="bg-green-300  h-12">1</div>
                <div class="bg-green-500  h-12">2</div>
                <div class="bg-green-300  h-12">3</div>
                <div class="bg-green-500 h-12">4</div>
                <div class="bg-green-300 h-12">5</div>
                <div class="bg-green-500  h-12">6</div>
          </div>
```



Tailwind modifiers

Pseudo-classes like :hover, :focus, :active, :visited

- ✓ Tailwind CSS uses the hover, focus, active, visited variants to style an element when the user mouse move over it, focuses it, actively clicks/tapped it and visited it.
- ✓ These variants allow you to create interactive and dynamic user interfaces without writing custom CSS.

Below are examples of several pseudo-class states supported in Tailwind CSS.

1) **Hover:** The hover variant is used to style an element when the user hovers the mouse pointer over it. The modifier used is **hover:**
   a. For example, you can use the **hover:bg-red-500** class to change the background color of an element to red when the user hovers the mouse over it.

2) **Focus:** Applied when an element has the focus. It is also enabled by default in Tailwind CSS. The modifier used is **focus:**
   a. For example, you can use the **focus:bg-red-500** class to change the background color of an element to red when an element has the focus.

3) **Active:** Applied when an element is being activated by the user. The modifier used is **active:**
   a. For example, you can use the **active:bg-red-500** class to change the background color of an element to red when an element is being activated by the user.

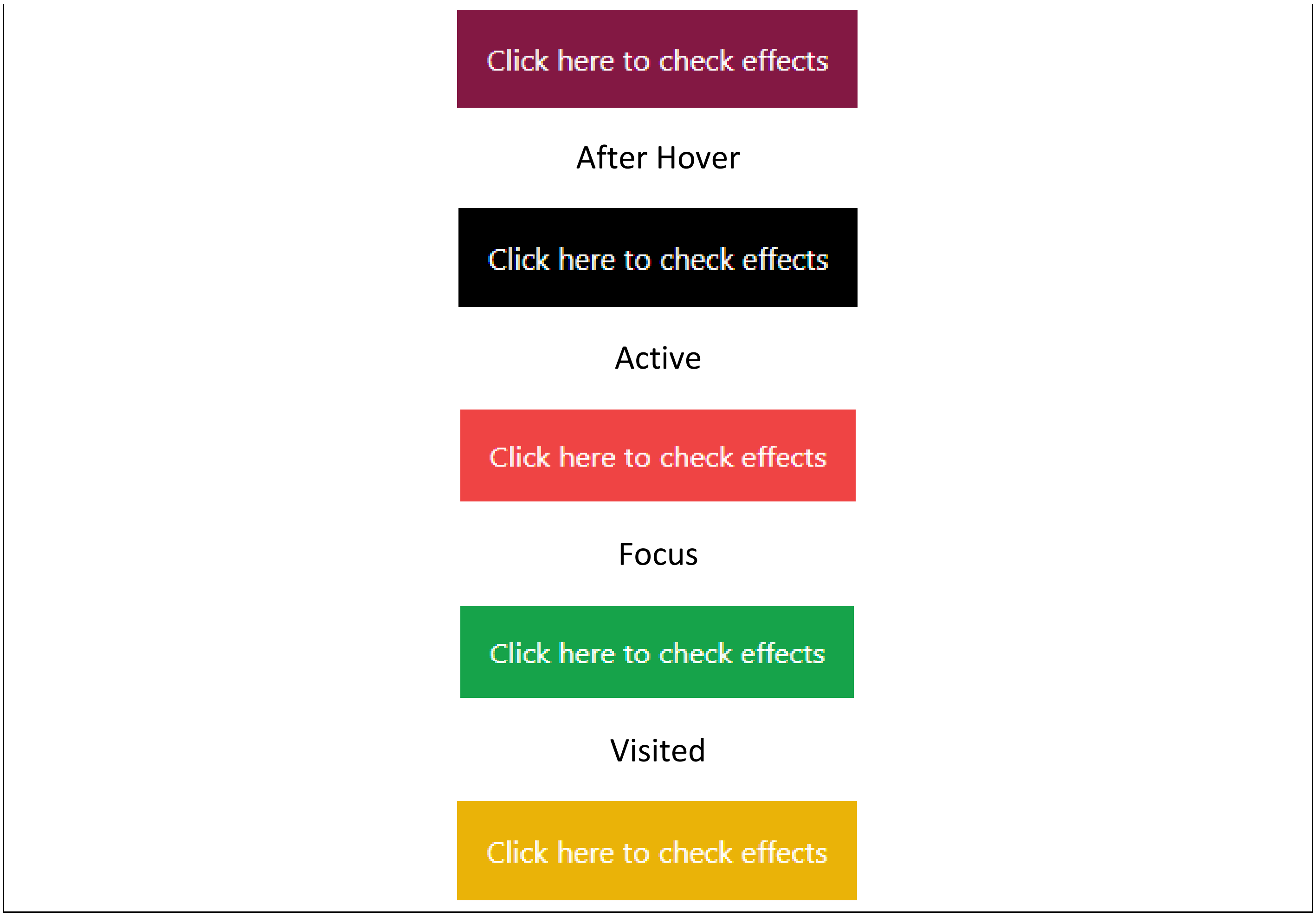
4) **Visited:** Applied once a user has visited a link. The modifier used is **visited:**
   a. For example, you can use the **visited:bg-red-500** class to change the background color of an element to red when a user has visited a link.

Example to understand pseudo class

```
<head><script src="https://cdn.tailwindcss.com"></script></head>

<body>

   <a href="#" class="bg-pink-900 p-4 px-4 text-white m-4 hover:bg-black active:bg-red-500 visited:bg-yellow-500 focus:bg-green-600">Click here to check effects</a>

</body>
```

Output

Initially



Click here to check effects

After Hover

Click here to check effects

Active

Click here to check effects

Focus

Click here to check effects

Visited

Click here to check effects

Pseudo-elements like ::before , ::after, ::first-line, ::first-letter

Use the **content-{value}** utilities along with the before and after variant modifiers to set the contents of the ::before and ::after pseudo-elements.

Before ::before

- ✓ This pseudo-element ::before allows us to add content before an element. It's like having a virtual container right before the actual content of an element.
- ✓ By using the ::before pseudo-element, we can insert content before the actual content of an element.

After ::after

- ✓ Similar to ::before, Pseudo-element ::after allows us to add content after an element. It's like having a virtual container right after the actual content of an element.
- ✓ By using the ::after pseudo-element, we can insert content after the actual content of an element.

This way, we can add extra visual elements or important information to our web page without modifying the original HTML structure.

Here's a simple example of Tailwind CSS code for the ::after and ::before utilities:

Tailwind

```
<p class="text-7xl m-4 before:content-['before'] before:text-green-500 after:content-['Hello'] after:text-red-500 after:text-5xl"> Lorem ipsum dolor sit amet </p>
```

Output:

```
before Lorem ipsum dolor sit amet Hello
```

First Letter ::first-letter

- ✓ The ::first-letter pseudo-element allows you to style the first letter of a block of text or a paragraph in a unique way.
- ✓ It can be used to make the first letter bigger, apply a different font, change its color, or add decorative elements to enhance the visual appeal.
- ✓ This way, you can easily customize the initial letter without altering the rest of the text.

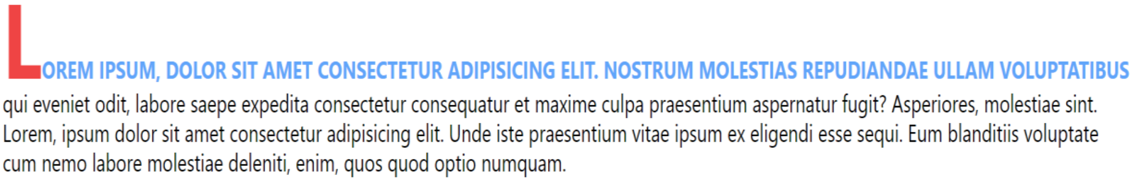
First Line ::first-line

- ✓ The ::first-line pseudo-element allows you to style the first line of a block of text or a paragraph separately.
- ✓ It's like having a special focus on the first line of your content.

Here's a simple example of Tailwind CSS code for the ::first-letter and ::first-line utilities:

```
<p class="p-4 first-line:uppercase first-line:font-bold  first-letter:text-9xl first-letter:text-red-500 first-line:text-blue-400 text-3xl">

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Nostrum molestias repudiandae ullam voluptatibus qui eveniet odit, labore saepe expedita consectetur consequatur et maxime culpa praesentium aspernatur fugit? Asperiores, molestiae sint. Lorem, ipsum dolor sit amet consectetur adipisicing elit. Unde iste praesentium vitae ipsum ex eligendi esse sequi. Eum blanditiis voluptate cum nemo labore molestiae deleniti, enim, quos quod optio numquam.

</p>
```

Output:



```
LOREM IPSUM, DOLOR SIT AMET CONSECTETUR ADIPISICING ELIT. NOSTRUM MOLESTIAS REPUDIANDAE ULLAM VOLUPTATIBUS
qui eveniet odit, labore saepe expedita consectetur consequatur et maxime culpa praesentium aspernatur fugit? Asperiores, molestiae sint.
Lorem, ipsum dolor sit amet consectetur adipisicing elit. Unde iste praesentium vitae ipsum ex eligendi esse sequi. Eum blanditiis voluptate
cum nemo labore molestiae deleniti, enim, quos quod optio numquam.
```

Referencing an attribute value

These content utilities even support CSS features like the attr() function, which you can use to reference a value stored in an attribute:

Example:

<div before="Hello World" class="before:content-[attr(before)]"></div>

Output: Hello World

Using spaces and underscores

Since whitespace denotes the end of a class in HTML, replace any spaces in an arbitrary value with an underscore:

Example:

<div class="before:content-['Hello_World']"></div>

Output: Hello World

If you need to include an actual underscore, you can do this by escaping it with a backslash:

Example:

<div class="before:content-['Hello_World']"></div>

Output: Hello_World

Transitions and Animation

transition-property

- ✓ Utilities for controlling which CSS properties transition.
- ✓ **Controlling transitioned properties**
- ✓ Use the **transition-{properties}** utilities to specify which properties should transition when they change.

Tailwind class	CSS Property
transition-none	transition-property: none;
transition-all	transition-property: all;
transition	transition-property: color, background-color, border-color, text-decoration-color, fill, stroke, opacity, box-shadow, transform, filter, backdrop-filter;
transition-colors	transition-property: color, background-color, border-color, text-decoration-color, fill, stroke;
transition-shadow	transition-property: box-shadow;

transition-duration

- ✓ Utilities for controlling the duration of CSS transitions.
- ✓ **Changing transition duration**
- ✓ Use the **duration-{amount}** utilities to control an element's transition-duration.

Tailwind class	CSS Property
duration-0	transition-duration: 0s;
duration-75	transition-duration: 75ms;
duration-100	transition-duration: 100ms;
duration-150	transition-duration: 150ms;
duration-200	transition-duration: 200ms;
duration-300	transition-duration: 300ms;
duration-500	transition-duration: 500ms;
duration-700	transition-duration: 700ms;
duration-1000	transition-duration: 1000ms;

transition-delay

- ✓ Utilities for controlling the delay of CSS transitions.
- ✓ Delaying transitions
- ✓ Use the **delay-{amount}** utilities to control an element's transition-delay.

Tailwind class	CSS Property
delay-0	transition- delay: 0s;
delay-75	transition- delay: 75ms;
delay-100	transition- delay: 100ms;
delay-150	transition- delay: 150ms;
delay-200	transition- delay: 200ms;
delay-300	transition- delay: 300ms;
delay-500	transition- delay: 500ms;
delay-700	transition- delay: 700ms;
delay-1000	transition- delay: 1000ms;

Example:

```
<button class="px-10 py-4 bg-black text-white my-11 mx-4 text-2xl hover:px-11 hover:py-5 hover:text-black hover:bg-white hover:border-2 hover:border-black rounded-md delay-1000 transition-all duration-1000">Click Here</button>
```

animation

Utilities for animating elements with CSS animations.

Spin

Add the animate-spin utility to add a linear spin animation to elements **like loading indicators**.

Ping

Add the animate-ping utility to make an element scale and fade like a **radar ping or ripple of water** — useful for things like **notification badges**.

Pulse

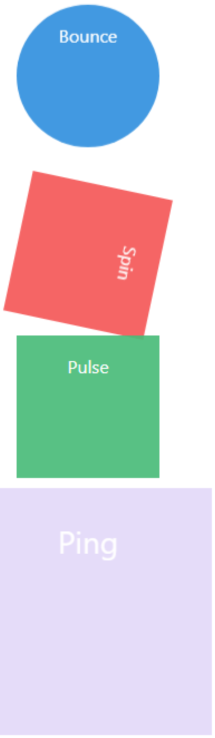
Add the animate-pulse utility to make an element gently fade in and out — useful for things like skeleton loaders.

Bounce

Add the animate-bounce utility to make an element bounce up and down — useful for things like “**scroll down**” indicators.

Example:

```
<div class="p-4 m-4"></div>
<div class="w-32 m-auto h-32 rounded-full bg-blue-500 animate-bounce text-white p-4">Bounce</div>
<div class="w-32 m-auto h-32 bg-red-500 animate-spin text-white p-4">Spin</div>
<div class="w-32 h-32 m-auto bg-green-500 animate-pulse text-white p-4">Pulse</div>
<div class="p-5 m-5"></div>
<div class="w-32 m-auto h-32 bg-purple-500 animate-ping text-white p-4">Ping</div>
```



Responsive Design

Using responsive utility variants to build adaptive user interfaces.

- ✓ Every utility class in Tailwind can be applied conditionally at different breakpoints, which makes it a piece of cake to build complex responsive interfaces without ever leaving your HTML.
- ✓ Without below line, a website might not display correctly on mobile devices, often causing the content to be too large or too small to read easily.

```
<head><meta name="viewport" content="width=device-width, initial-scale=1.0"></head>
```

There are five breakpoints by default, inspired by common device resolutions:

Breakpoint prefix	Minimum width	CSS
sm	640px	@media(min-width: 640px){ ... }
md	768px	@media(min-width: 768px){ ... }
lg	1024px	@media (min-width: 1024px){ ... }
xl	1280px	@media (min-width: 1280px){ ... }
2xl	1536px	@media (min-width: 1536px){ ... }

To add a utility but only have it take effect at a certain breakpoint, all you need to do is prefix the utility with the breakpoint name, followed by the : character

```
<!-- Width of 16 by default, 32 on medium screens, and 48 on large screens -->
<head><meta name="viewport" content="width=device-width, initial-scale=1.0"></head>
<img class="w-16 md:w-32 lg:w-48" src="2.png">
```

Customizing Default Tailwind CSS classes

Often the biggest challenge when working with a framework is figuring out what you’re supposed to do when there’s something you need that the framework doesn’t handle for you.

Customizing your theme

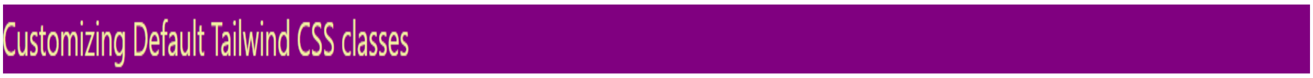
If you want to change things like your color palette, spacing scale, typography scale, or breakpoints, add your customizations to the theme section of your tailwind.config.js file

Using arbitrary values

When you find yourself really needing something which is not already available in tailwind, use Tailwind’s square bracket notation to generate a class on the fly with any arbitrary value:

Example

```
<head>
   <script src="https://cdn.tailwindcss.com"></script>
   <script>
      tailwind.config = {
        theme: {
          colors: {
             ljuPurple: 'purple',
          },
          fontSize:{
             ljusize:'30px',
          }
        }
      }
     </script></head>
<p class="bg-ljuPurple text-[#f3f3a2] text-ljusize">Customizing Default Tailwind CSS
classes</p>
```

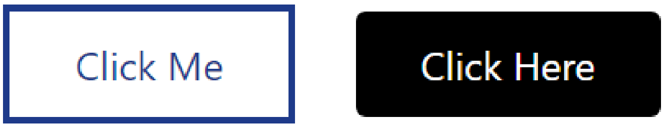


Customizing Default Tailwind CSS classes

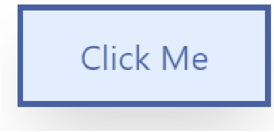
Kindly note that if you add default CSS classes along with custom classes, the default class styles may not apply correctly. Create a new file and add the script as shown in the code above.

Example

Design a link and a button using tailwind css as shown in below image.

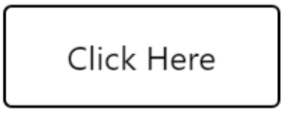


Add link "Click Me" and design it as shown in above image. Add hover effect and link will be displayed as shown in below image. Also, add animation of your choice.



After hover

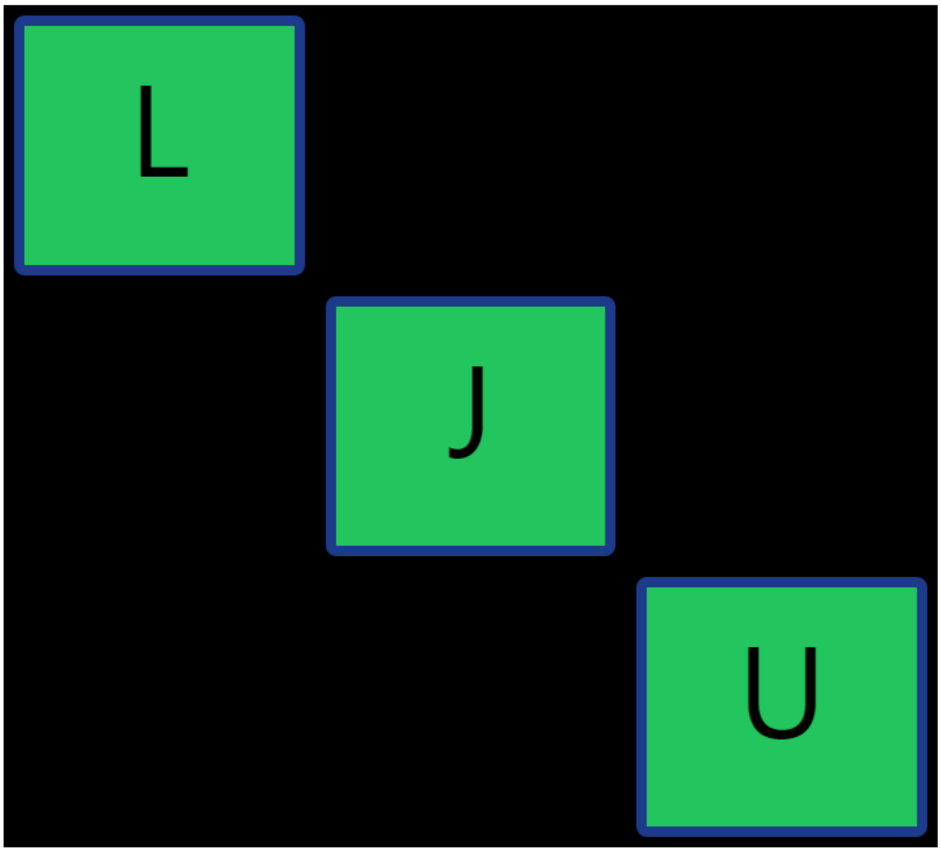
Add button "Click Here" and design it as shown in image. Add hover effect and button will be displayed as shown in below image. Also, add transition and add animation of your choice.



After hover

```
<head>
   <script src="https://cdn.tailwindcss.com"></script>
</head>
<body>
   <a class="hover:animate-pulse  my-11 mx-4 inline-block border-4 border-blue-900 text-blue-900 px-10 py-4 hover:bg-blue-100 text-2xl hover:shadow-2xl " href="#" >Click Me</a>

   <button class="hover:animate-bounce px-10 py-4 bg-black text-white my-11 mx-4 text-2xl hover:text-black hover:bg-white hover:border-2 hover:border-black rounded-md transition-all duration-1000 hover:px-11 hover:py-5">Click Here</button>

</body>
```

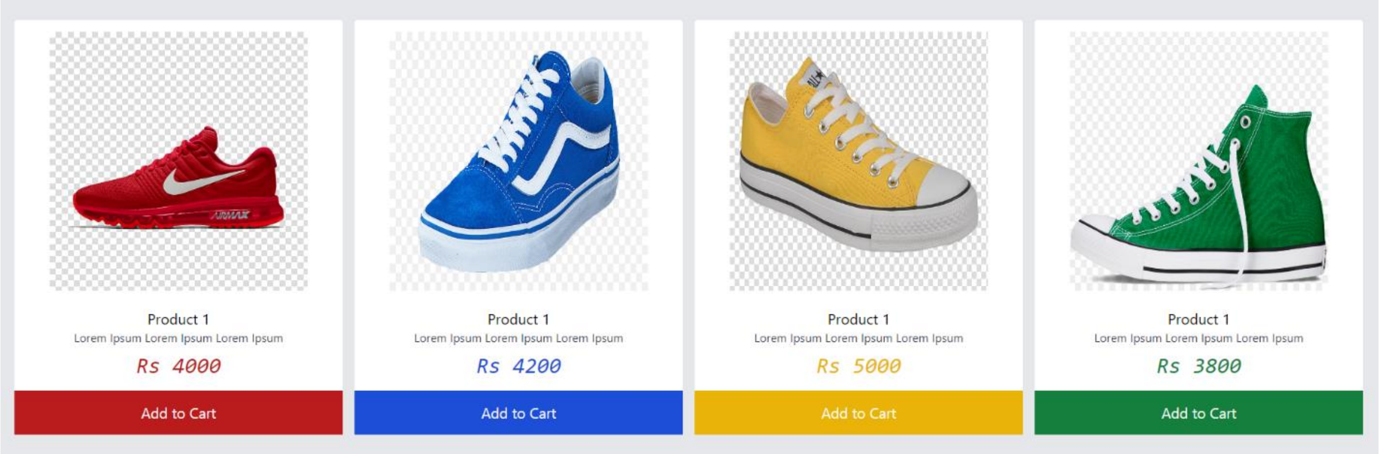
Example



```
<div class="grid justify-items-auto grid-cols-3 bg-black m-4">
   <div class="m-2 flex bg-green-500 rounded-lg border-8 border-blue-900 h-[200px] text-[100px] justify-center">L</div>
   <div></div>
   <div ></div>
   <div ></div>
   <div class="m-2 flex bg-green-500 border-8 border-blue-900 rounded-lg h-[200px] text-[100px] justify-center">J</div>
   <div></div>
   <div ></div>
   <div ></div>
   <div class="m-2 flex bg-green-500 border-8 border-blue-900 rounded-lg h-[200px] text-[100px] justify-center">U</div>
</div>
```

Example

Design product grids as shown in below image.

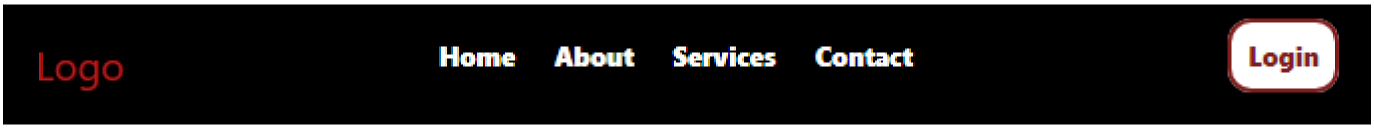

Product 1
Lorem Ipsum Lorem Ipsum Lorem Ipsum
Rs 4000
Add to Cart
Product 1
Lorem Ipsum Lorem Ipsum Lorem Ipsum
Rs 4200
Add to Cart
Product 1
Lorem Ipsum Lorem Ipsum Lorem Ipsum
Rs 5000
Add to Cart
Product 1
Lorem Ipsum Lorem Ipsum Lorem Ipsum
Rs 3800
Add to Cart

```
<head>
   <script src="https://cdn.tailwindcss.com"></script>
</head>
<body>
<div class="grid grid-cols-4 gap-4 m-2 bg-gray-200 p-8">
   <div class="rounded hover:shadow-2xl hover:shadow-black bg-white">
      <img src="t2.jpg" class="w-96 h-96 p-4 mx-auto">
      <div class="p-2">
         <h2 class="text-xl text-center">Product 1</h2>
         <p class="text-center text-gray-700">Lorem Ipsum Lorem Ipsum Lorem Ipsum</p>
         <p class="text-red-700 text-center text-3xl italic font-mono p-2">Rs 4000</p>
      </div>
      <div class="block bg-red-700 p-4"><a href="#" class="text-white text-xl block text-center">Add to Cart</a></div>
   </div>

   <div class="rounded hover:shadow-2xl hover:shadow-black bg-white">
      <img src="t3.png" class="w-96 h-96 p-4 mx-auto">
      <div class="p-2">
         <h2 class="text-xl text-center">Product 1</h2>
         <p class="text-center text-gray-700">Lorem Ipsum Lorem Ipsum Lorem Ipsum</p>
         <p class="text-blue-700 text-center text-3xl italic font-mono p-2">Rs 4200</p>
      </div>
      <div class="block bg-blue-700 p-4"><a href="#" class="text-white text-xl block text-center">Add to Cart</a></div>
   </div>
```

```
  <div class="rounded hover:shadow-2xl hover:shadow-black bg-white">
    <img src="t1.jpg" class="w-96 h-96 p-4 mx-auto">
    <div class="p-2">
      <h2 class="text-xl text-center">Product 1</h2>
      <p class="text-center text-gray-700">Lorem Ipsum Lorem Ipsum Lorem Ipsum</p>
      <p class="text-yellow-500 text-center text-3xl italic font-mono p-2">Rs 5000</p>
    </div>
    <div class="block bg-yellow-500 p-4"><a href="#" class="text-white text-xl block text-center">Add to Cart</a></div>
  </div>

  <div class="rounded hover:shadow-2xl hover:shadow-black bg-white">
    <img src="t4.png" class="w-96 h-96 p-4 mx-auto">
    <div class="p-2">
      <h2 class="text-xl text-center">Product 1</h2>
      <p class="text-center text-gray-700">Lorem Ipsum Lorem Ipsum Lorem Ipsum</p>
      <p class="text-green-700 text-center text-3xl italic font-mono p-2">Rs 3800</p>
    </div>
    <div class="block bg-green-700 p-4"><a href="#" class="text-white text-xl block text-center">Add to Cart</a></div>
  </div>
</div>
</body>
```

Example

Design navbar as shown in below image using tailwind classes. Also add animated utility to make an element gently fade in and out



```
<nav class="bg-black p-4 m-8">
  <div class="mx-auto flex justify-between">
   <!-- Logo on the left -->
   <div class="text-red-700 text-xl">
    <a href="#">Logo</a>
   </div>
   <!-- Links in the middle -->
   <div>
    <a href="#" class="font-extrabold text-white p-2 hover:text-red-800">Home</a>
```

```
        <a href="#" class="font-extrabold text-white p-2 hover:text-red-800">About</a>
        <a href="#" class="font-extrabold text-white p-2 hover:text-red-800">Services</a>
        <a href="#" class="font-extrabold text-white p-2 hover:text-red-800">Contact</a>
      </div>
      <!-- Login link on the right -->
      <div>
        <a href="#" class="animate-pulse text-red-900 font-extrabold hover:text-red-900 border-2 p-2 bg-white rounded-xl border-red-900">Login</a>

      </div>
    </div>
  </nav>
```