

Tailwind [v4.1]

作者： -- 天气预报

日期： -- 2025-09-20

01. 官网

≡ <https://tailwindcss.com/>

- ui

The screenshot shows the TailwindCSS website homepage. At the top, there's a navigation bar with a search icon, 'Ctrl K', 'Docs', 'Blog', 'Showcase', 'Sponsor', and a 'Plus' button. Below the navigation is a large hero section with the text 'Rapidly build modern websites without ever leaving your HTML.' and a subtext about utility-first CSS. A 'Get started' button is visible. On the right side, there's a small image of a book titled 'The Anti-Patterns No. 4' with the subtitle 'Close Warfare'. A code editor window is overlaid on the bottom left, displaying a snippet of Tailwind CSS code:

```
1 <div class="flex flex-col items-center p-7 rounded-2xl">
2   <div>
3     
4   </div>
5   <div class="flex">
6     <span class="text-2xl font-medium">Class Warfare</span>
7     <span class="font-medif>The Anti-Patterns</span>
8     <span class="flex">
9       <span>No. 4</span>
10      <span>Close Warfare</span>
```

ⓘ Note

✓ Tailwind (v4.1) 推荐新项目直接使用即可

1.1 官方文档

≡ <https://tailwindcss.com/docs/installation/using-vite>

- ui

The screenshot shows the Tailwind CSS documentation page for "Using Vite". The left sidebar has sections like Documentation, Components, Templates, UI Kit, Playground, Course (NEW), and Community. Under "GETTING STARTED", "Installation" is selected, showing sub-links for Editor setup, Compatibility, and Upgrade guide. The main content area starts with "INSTALLATION" and "Get started with Tailwind CSS". It explains how Tailwind CSS works by scanning HTML files and generating static CSS. Below this is the "Installation" section, which links to "Using Vite", "Using PostCSS", "Tailwind CLI", "Framework Guides", and "Play CDN". The "Using Vite" link is underlined. The "Using Vite" section details the process of creating a new Vite project, installing Tailwind CSS via npm, and configuring the Vite plugin. It includes three terminal screenshots showing the command `npm create vite@latest my-project` and its execution.

- ui

1.2 在线运行

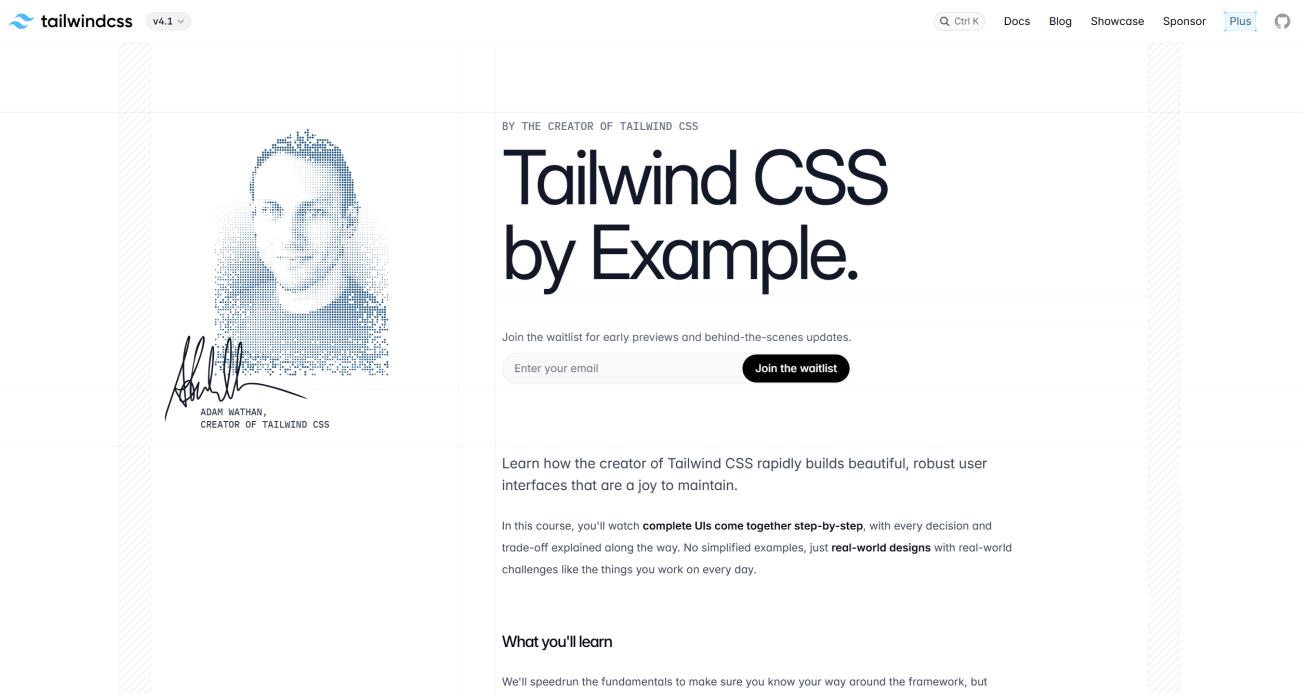
≡ <https://play.tailwindcss.com/>

● ui

1.3 福利课程

≡ <https://tailwindcss.com/course>

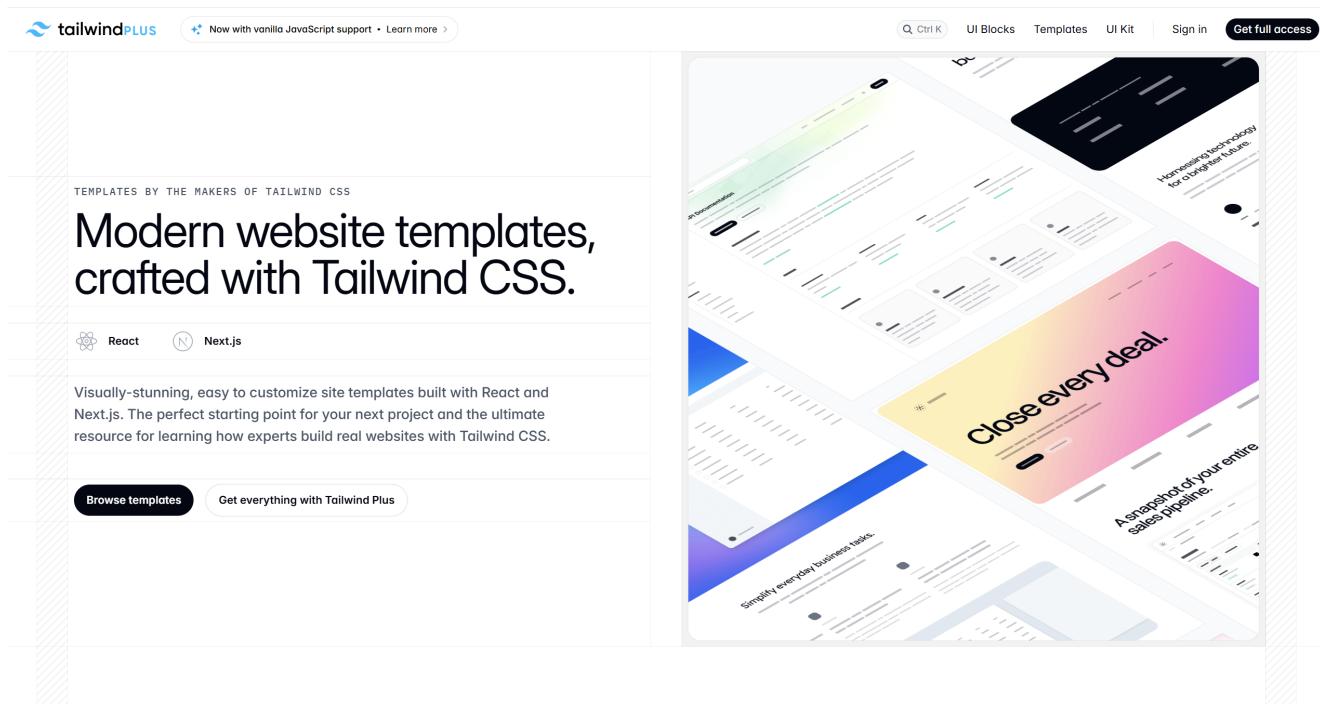
- ✓ 邮件报名、邮件推送油管视频 [科学上网] (* 小课程、分享特殊场景案例实现)



1.4 付费服务

≡ <https://tailwindcss.com/plus/templates?ref=sidebar>

- ui



1.5 官方组件

≡ <https://tailwindcss.com/plus/ui-blocks?ref=sidebar>

✓ 免费

● ui

[tailwind PLUS](#) Now with vanilla JavaScript support • Learn more >

PAGE SECTIONS MARKETING APPLICATION UI ECOMMERCE PRICING

Marketing

Heroes, feature sections, newsletter sign up forms — everything you need to build beautiful marketing websites.

Hero Sections 12 components

Feature Sections 15 components

CTA Sections 11 components

Bento Grids 3 components

Pricing Grids 8 components

About us 1 component

Sign up newsletter 1 component

Trusted by developers 1 component

● ui

Marketing Application UI Ecommerce Documentation

PAGE SECTIONS Hero Sections Feature Sections CTA Sections Bento Grids Pricing Sections Header Sections Newsletter Sections Stats Testimonials

UI BLOCKS / MARKETING / PAGE SECTIONS Hero Sections

Use these Tailwind CSS hero section examples to add important messaging, product photos, and call-to-actions to the top of your website. These hero examples are designed and built by the Tailwind CSS team, and include a variety of different styles and layouts.

Simple centered

Preview Code

5-DAY MINI-COURSE
Build UIs that don't suck.
Short, tactical video lessons from the creator of Tailwind CSS, delivered directly to your inbox every day for a week.
[Get the free course](#)

Announcing our next round of funding. [Read more →](#)

Data to enrich your online business

1.6 博客新闻

≡ <https://tailwindcss.com/blog>

● ui

The screenshot shows the Tailwind CSS website's "Latest Updates" page. At the top left is the Tailwind CSS logo and version v4.1. The top right features navigation links: Ctrl K (Search), Docs, Blog, Showcase, Sponsor, and Plus. A large, stylized illustration of a keyboard is positioned in the background. The main content area has two news items. The first item, dated July 25, 2025, is titled "Vanilla JavaScript support for Tailwind Plus" and discusses UI blocks requiring JavaScript. The second item, dated May 14, 2025, is titled "Compass: A starter kit for online courses" and describes a Next.js starter kit for publishing online courses. Each news item includes a "Read more" link.

1.7 社区插件

≡ <https://tailscan.com/>

- ui

The screenshot shows the Tailscan website, described as "The best devtool for Tailwind CSS". The header includes the Tailscan logo, navigation links for Home, Features, Pricing, Blog, Resources, and Support, and a "My Account" button. The main headline reads "The best devtool for Tailwind CSS". Below it, a sub-headline says "Build, design and debug your Tailwind website visually with Tailscan, right within the browser." Two buttons are present: "Try on this page" and "Get Tailscan now". A testimonial banner below states "Used by + 1000 other developers, designers and product managers!" followed by a grid of small user profile icons. A large screenshot of the Tailscan interface shows a browser window with developer tools open, displaying Tailwind CSS code and a preview of the UI. A prominent message in the screenshot says "Never miss the cache again!"

☰ Tailscan Core | Tailscan Core + AI

● ui

The screenshot shows the Tailscan Core pricing page. At the top, it says "Pricing" and "Level up your development experience". It notes compatibility with Chrome and Chromium-based browsers (Edge, Brave, and Arc). Below this, two plans are listed:

- Tailscan Core**: \$79/one-time. Includes Pay once, use Core forever; Includes all Core updates, forever; License for 3 devices; Permanent access to the Tailscan community discord.
- Tailscan Core + Tailscan AI**: \$15/mo (Billed \$180 annually). Includes Tailscan AI (unlimited requests*); Pay once, use Core forever; Includes all Core updates, forever; License for 3 devices; Permanent access to the Tailscan community discord.

Both plans come with a 7-day refund policy and dedicated support. A note says they are used by over 1000 developers, designers, and product managers, accompanied by small profile icons.

● ui

The screenshot shows the Tailwind CSS documentation page for the v4.1 version. The left sidebar includes links for Documentation, Components, Templates, UI Kit, Playground, Course (NEW), and Community. Under GETTING STARTED, the Installation section is currently selected, showing sub-links for Editor setup, Compatibility, and Upgrade guide. The main content area is titled "Get started with Tailwind CSS". It explains how Tailwind CSS works by scanning HTML files and generating static CSS. It's fast, flexible, and reliable. Below this is the "Installation" section, which provides steps for Using Vite, Using PostCSS, Tailwind CLI, Framework Guides, and Play CDN. Step 01: Create your project. Step 02: Install Tailwind CSS. Step 03: Configure the Vite plugin. A modal window on the right shows a dark-themed interface for selecting CSS classes like mt-2, text-3xl, font-medium, tracking-tight, text-gray-950, and dark:text-white. Below the modal are three terminal windows showing the command to create a Vite project, install Tailwind CSS via npm, and add the plugin to vite.config.ts.

● ui

The screenshot shows the TailwindCSS website's documentation page. On the left, there's a sidebar with sections like Documentation, Components, Templates, UI Kit, Playground, Course (NEW), and Community. Under GETTING STARTED, there are links for Installation, Core Concepts, Colors, and Functions and directives. A central modal window contains a tip about setting up Tailwind with Vite, followed by a link to explore framework guides. To the right, there's a dark-themed interface showing a code editor with Tailwind CSS utility classes and a terminal window.

02. 环境

[≡ https://tailwindcss.com/docs/installation/using-vite](https://tailwindcss.com/docs/installation/using-vite)

● ui

The screenshot shows the 'Using Vite' documentation page. It includes a sidebar with sections like Installation, Core Concepts, Colors, and Functions and directives. The main content area is a step-by-step guide for integrating Tailwind CSS with Vite. Step 1: 'Create your project' shows a terminal command to create a new Vite project. Step 2: 'Install Tailwind CSS' shows a terminal command to install 'tailwindcss' and '@tailwindcss/vite' via npm. Step 3: 'Configure the Vite plugin' shows a code snippet for adding the plugin to vite.config.ts. Step 4: 'Import Tailwind CSS' shows a code snippet for adding an '@import' rule to a CSS file. Step 5: 'Start your build process' shows a terminal command to run the build process with 'npm run dev'.

2.1 CDN

≡ <https://tailwindcss.com/docs/installation/play-cdn>

- `index.html`

```
● ● ●

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Tailwind</title>
    <script src="https://cdn.jsdelivr.net/npm/@tailwindcss/browser@4">
</script>
    <style type="text/tailwindcss">
      @theme {
        --color-primary: #000;
      }
    </style>
  </head>
  <body>
    <div class="p-10 bg-primary">
      <h1 class="text-2xl text-center text-white">tailwind</h1>
    </div>
  </body>
</html>
```

2.2 Tailwind CLI

≡ <https://tailwindcss.com/docs/installation/tailwind-cli>

- `npm`



```
npm init --yes  
npm install tailwindcss @tailwindcss/cli
```

- **src/input.css**



```
@import "tailwindcss";
```

- **src/index.html**



```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <meta charset="UTF-8">  
    <title>Tailwind</title>  
    <link rel="stylesheet" href=".//output.css">  
  </head>  
  <body>  
    <div class="p-10 bg-rose-500">  
      <h1 class="text-2xl text-center text-white">tailwind</h1>  
    </div>  
  </body>  
</html>
```

- **npx**



```
npx @tailwindcss/cli -i ./src/input.css -o ./src/output.css --watch
```

2.3 Next

≡ <https://nextjs.org/>

- ✓ 内置安装引导支持、仅需创建项目时选中即可

- npm



```
npx create-next-app@latest
```

- console



```
# 项目名称
1. What is your project named? [name]

# 是否安装 TypeScript
2. Would you like to use TypeScript? [No / Yes]

# 是否安装 ESLint 代码检查工具
3. Which linter would you like to use? [ESLint / Biome / None]

# 是否安装 Tailwind
4. Would you like to use Tailwind CSS? No / Yes

# 是否需要 src 作为项目源代码存放目录
5. Would you like your code inside a `src/` directory? [No / Yes]

# 是否安装 文件路由系统
6. Would you like to use App Router? (recommended) [No / Yes]

# 是否安装 Turbopack 打包构建工具
7. Would you like to use Turbopack? (recommended) [No / Yes]

# 是否配置 @/* 别名 模块导入方式
8. Would you like to customize the import alias (`@/*` by default)? [No / Yes]

# 配置别名 是否使用 @/*
9. Would you like to use the @/* import alias? [Yes / No]
```

9. What import alias would you like configured? @/*

- **page.tsx**



```
export default function Page() {
    return (
        <div className="p-10 bg-rose-500">
            <h1 className="text-2xl text-white text-center">tailwind</h1>
        </div>
    )
}
```

2.4 Vite React

≡ 请认真学习本节示例过程、后续示例基于此环境、本环境无须掌握 React

2.4.1 init

≡ npm init --yes

- **package.json**



```
{
  "name": "tailwind",
  "version": "1.0.0",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "type": "module"
}
```

2.4.2 react

≡ npm install react react-dom

- package.json



```
{
  "name": "tailwind",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "type": "module",
  "dependencies": {
    "react": "^19.1.1",
    "react-dom": "^19.1.1"
  }
}
```

2.4.3 vite

≡ npm install vite -D

- package.json



```
{
  "name": "tailwind",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "type": "module",
  "dependencies": {
    "react": "^19.1.1",
    "react-dom": "^19.1.1"
  },
  "devDependencies": {
    "vite": "^7.1.6"
  }
}
```

2.4.4 plugin-react

≡ <https://cn.vite.dev/plugins/>

✓ npm install @vitejs/plugin-react -D
 ✓ npm install @vitejs/plugin-react-swc -D [<https://swc.rs/>]

- package.json



```
{  
  "name": "tailwind",  
  "version": "1.0.0",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\"$Error: no test specified\\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC",  
  "description": "",  
  "type": "module",  
  "dependencies": {  
    "react": "^19.1.1",  
    "react-dom": "^19.1.1"  
  },  
  "devDependencies": {  
    "@vitejs/plugin-react-swc": "^4.1.0",  
    "vite": "^7.1.6"  
  }  
}
```

2.4.5 vite.config.js

≡ 项目根目录创建 vite.config.js 文件

- **vite.config.js**



```
import { defineConfig } from "vite";  
import react from "@vitejs/plugin-react-swc";  
  
export default defineConfig({  
  plugins: [  
    react(),  
  ]  
})
```

2.4.6 index.html

☰ 项目根目录创建文件 index.html



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>React</title>
  </head>
  <body>
    <div id="root"></div>
    <script type="module" src=".src/main.jsx"></script>
  </body>
</html>
```

2.4.7 src/main.jsx

☰ 项目根目录创建目录及文件 src/app.jsx | src/main.jsx

● src/app.jsx



```
export function App() {
  return (
    <div>react</div>
  )
}
```

● src/main.jsx



```
import { createRoot } from "react-dom/client";
import { App } from "./app.jsx";

const root = document.getElementById("root");

createRoot(root).render(<App/>);
```

● package.json



```
{
  "name": "tailwind",
  "version": "1.0.0",
  "scripts": {
    "dev": "vite dev"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "type": "module",
  "dependencies": {
    "react": "^19.1.1",
    "react-dom": "^19.1.1"
  },
  "devDependencies": {
    "@vitejs/plugin-react-swc": "^4.1.0",
    "vite": "^7.1.6"
  }
}
```

● npm



```
npm run dev # http://localhost:5173/
```

2.4.8 tailwind

```
≡ npm install tailwindcss @tailwindcss/vite -D
```

- package.json



```
{
  "name": "tailwind",
  "version": "1.0.0",
  "scripts": {
    "dev": "vite dev"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "type": "module",
  "dependencies": {
    "react": "^19.1.1",
    "react-dom": "^19.1.1"
  },
  "devDependencies": {
    "@tailwindcss/vite": "^4.1.13",
    "@vitejs/plugin-react-swc": "^4.1.0",
    "tailwindcss": "^4.1.13",
    "vite": "^7.1.6"
  }
}
```

- vite.config.js



```
import { defineConfig } from "vite";
import react from "@vitejs/plugin-react-swc";
import tailwindcss from "@tailwindcss/vite";

export default defineConfig({
  plugins: [
    react(),
    tailwindcss(),
  ]
})
```

● src/globals.css



```
@import "tailwindcss";
```

● src/main.jsx



```
import { createRoot } from "react-dom/client";
import { App } from "./app.jsx";
import "./globals.css";

const root = document.getElementById("root");
createRoot(root).render(<App/>)
```

● src/app.jsx



```
export function App() {
  return (
    <div className="p-32 bg-rose-500">
      <h1 className="font-mono text-center text-white">
        <span className="text-4xl md:text-6xl lg:text-7xl duration-300">
          tailwind
        </span>
      </h1>
    </div>
  )
}
```

03. 开发工具

≡ <https://tailwindcss.com/docs/editor-setup>

3.1 IDE

- ui

[tailwindcss](https://tailwindcss.com) v4.1

GETTING STARTED

Editor setup

Tooling to improve the developer experience when working with Tailwind CSS.

Syntax support

Tailwind CSS uses custom CSS syntax like `@theme`, `@variant`, and `@source`, and in some editors this can trigger warnings or errors where these rules aren't recognized.

If you're using VS Code, our official [Tailwind CSS Intellisense](#) plugin includes a dedicated Tailwind CSS language mode that has support for all of the custom at-rules and functions Tailwind uses.

In some cases, you may need to disable native CSS linting/validation if your editor is very strict about the syntax it expects in your CSS files.

IntelliSense for VS Code

The official [Tailwind CSS Intellisense](#) extension for Visual Studio Code enhances the Tailwind development experience by providing users with advanced features such as autocomplete, syntax highlighting, and linting.

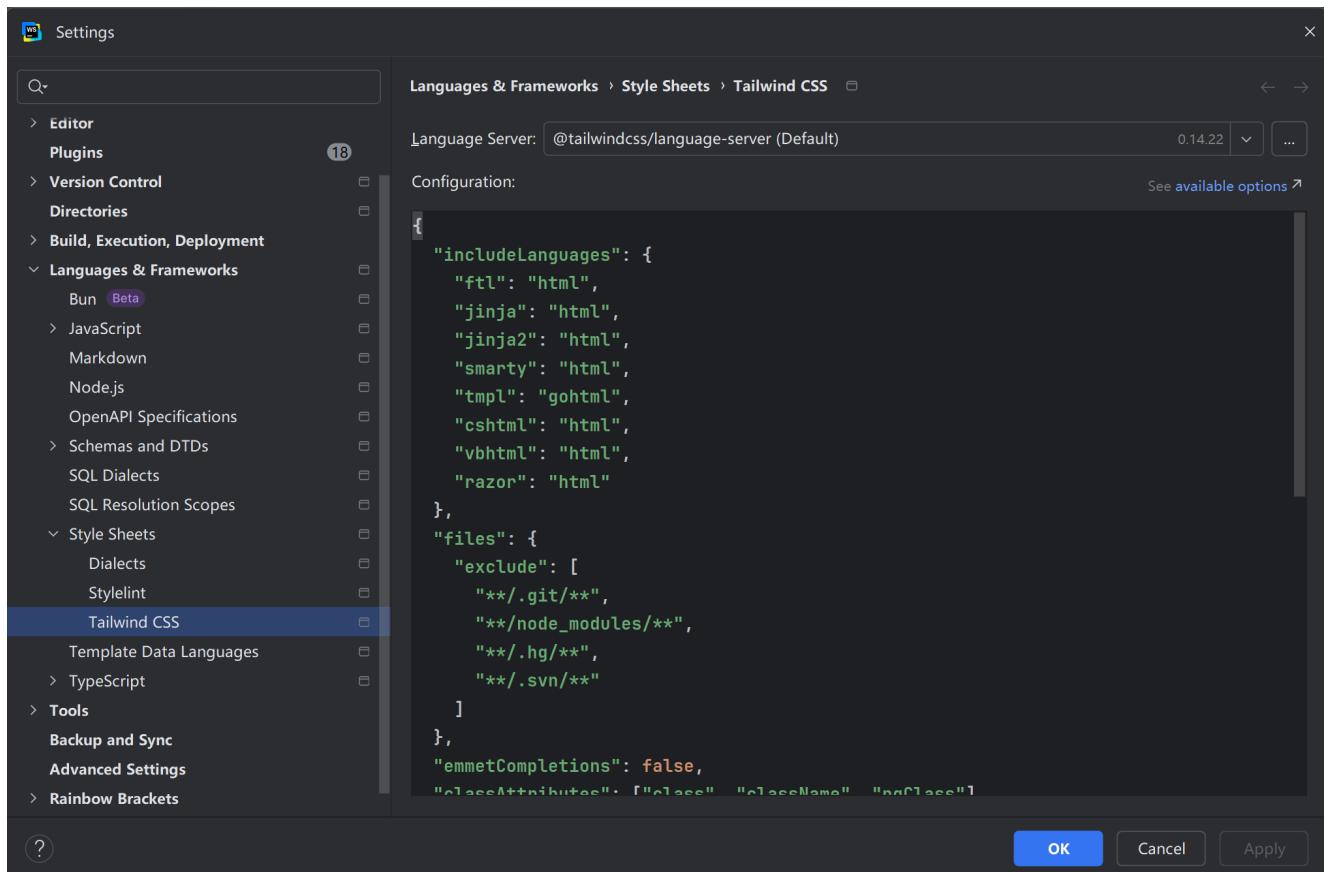
From the creators of Tailwind CSS
Make your ideas look awesome, without relying on a designer.

"This is the survival kit I wish I had when I started building apps."
Derrick Reimer, SavvyCat

≡ WebStorm 内置 Tailwind Language Server 语言服务引擎

✓ 引擎自动开启、无需其它配置、可自动提示 Tailwind、可检测类选择器

● ui



● ui

app.jsx

```

1  export function App() { Show usages
2    return (
3      <div className="bg-rose-500 p-32">
4        <h1 className="text-center font-mono text-white">
5          <span class="tailwind">
6            .text-white {
7              color: var(--color-white) /* #fff = #ffffff */;
8            }
9          </span>
10         </h1>
11       </div>
12     )
13   }

```

A tooltip is displayed over the CSS class definition for ".text-white". The tooltip contains the following content:

```

.tailwind {
  color: var(--color-white) /* #fff = #ffffff */;
}

```

≡ 列编辑: ALT + SHIFT + INSERT [二次按下为取消列编辑状态]

● ui

app.jsx

```

1  export function App() { Show usages
2    return (
3      <div className="p-32 bg-rose-500">
4        <div className="flex gap-5">
5          <button className="px-5 py-3 bg-purple-700 text-white hover:bg-purple-300">tailwind</button>
6          <button className="px-5 py-3 bg-purple-700 text-white hover:bg-purple-300">react</button>
7          <button className="px-5 py-3 bg-purple-700 text-white hover:bg-purple-300">next</button>
8          <button className="px-5 py-3 bg-purple-700 text-white hover:bg-purple-300">motion</button>
9          <button className="px-5 py-3 bg-purple-700 text-white hover:bg-purple-300">zustand</button>
10         </div>
11       </div>
12     )
13   }
14 
```

A red rectangular box highlights the entire list of buttons within the `<div>` element at line 4.

3.2 格式化

≡ <https://github.com/tailwindlabs/prettier-plugin-tailwindcss>

● ui

A screenshot of a browser window showing the GitHub repository page for 'tailwindlabs/prettier-plugin-tailwindcss'. The page includes the README file, which contains code snippets for Tailwind CSS classes like 'bg-sky-700' and 'bg-sky-800'. It also shows the contributor list with 43 contributors and a languages section indicating 94.0% TypeScript and 6.0% JavaScript.

● npm

A screenshot of a terminal window showing the command 'npm install prettier prettier-plugin-tailwindcss -D' being run. The command is highlighted in blue.

● prettier.config.js

A screenshot of a terminal window showing the configuration of prettier.config.js. The file content is shown as:

```
export default {
  plugins: ["prettier-plugin-tailwindcss"],
}
```

● npx



```
npx prettier . --write
```

≡ 推荐人为控制书写顺序 [位置 | 盒子模型 | 排版 | 视觉 | 其它]

- `src/app.jsx`



```
export function App() {
  return (
    <div className="p-32 bg-slate-200">
      <div className="relative size-48 bg-rose-500">
        <div className="absolute left-1/2 top-1/2 -translate-1/2
          size-1/2
          bg-purple-700
          animate-spin" />
      </div>
    </div>
  )
}
```

ⓘ Note

✓ <https://codeguide.bootcss.com/>

- ui

声明顺序

相关的属性声明应当归为一组，并按照下面的顺序排列：

1. Positioning
2. Box model
3. Typographic
4. Visual
5. Misc

由于定位（positioning）可以从正常的文档流中移除元素，并且还能覆盖盒模型（box model）相关的样式，因此排在首位。盒模型排在第二位，因为它决定了组件的尺寸和位置。

其他属性只是影响组件的 内部 或者是不影响前两组属性，因此排在后面。

完整的属性列表及其排列顺序请参考 [Bootstrap property order for Stylelint](#)。

```
.declaration-order {
  /* Positioning */
  position: absolute;
  top: 0;
  right: 0;
  bottom: 0;
  left: 0;
  z-index: 100;

  /* Box-model */
  display: block;
  float: right;
  width: 100px;
  height: 100px;

  /* Typography */
  font: normal 13px "Helvetica Neue", sans-serif;
  line-height: 1.5;
  color: #333;
  text-align: center;

  /* Visual */
  background-color: #f5f5f5;
  border: 1px solid #e5e5e5;
  border-radius: 3px;

  /* Misc */
  opacity: 1;
}
```

3.3 兼容性

≡ <https://tailwindcss.com/docs/compatibility>

● ui

GETTING STARTED

Compatibility

Learn about browser support and compatibility with other tooling.

Browser support

Tailwind CSS v4.0 is designed for and tested on modern browsers, and the core functionality of the framework specifically depends on these browser versions:

- = **Chrome 111** (released March 2023)
- = **Safari 16.4** (released March 2023)
- = **Firefox 128** (released July 2024)

Tailwind also includes support for many bleeding-edge platform features like `field-sizing: content`, `@starting-style`, and `text-wrap: balance` that have limited browser support. It's up to you if you want to use these modern features in your projects — if the browsers you're targeting don't support them, simply don't use those utilities and variants.

If you're unsure about the support for a modern platform feature, the [Can I use](#) database is a great resource.

Sass, Less, and Stylus

Tailwind CSS v4.0 is a full-featured CSS build tool designed for a specific workflow, and is not designed to be used with CSS preprocessors like Sass, Less, or Stylus.

ON THIS PAGE

Browser support

- Sass, Less, and Stylus
- Build-time imports
- Variables
- Nesting
- Loops
- Color and math functions
- CSS modules
- Scoping concerns
- Performance
- Explicit context sharing
- Vue, Svelte, and Astro



5-day mini-course

Build UIs that don't suck.

Short, tactical video lessons from the creator of Tailwind CSS, delivered directly to your inbox every day for a week.

[Get the free course →](#)

04. 样式初始化

≡ <https://tailwindcss.com/docs/preflight>