Tailwind CSS v4 Installation & Setup using CLI

Tailwind CSS v4 is the latest version that comes with powerful utilities and enhanced performance.

✅ Prerequisites

Make sure you have the following installed:

- Node.js (v18 or higher recommended)
- npm (comes with Node.js)

🚀 1. Method 1: Using Tailwind CLI (Recommended for Quick Start / Prototyping)

🔧 Step-by-Step Setup

📁 1. Create Your Project Folder

```
mkdir my-tailwind-project

cd my-tailwind-project

npm init -y
```

📦 2. Install Tailwind CSS CLI

```
npm install -D tailwindcss

npx tailwindcss init
```

This creates tailwind.config.js.

📄 3. Create File Structure

```
touch src/input.css index.html
```

🖊️ 4. Configure input.css

```
/* src/input.css */

@tailwind base;

@tailwind components;

@tailwind utilities;
```

🧠 5. Configure tailwind.config.js (optional paths)

```
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: ["./index.html"],
  theme: {
    extend: {},
  },
  plugins: [],
}
```

⚙️ 6. Run Tailwind CLI Build Process

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

📄 7. Use Tailwind in index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Tailwind 4 CLI</title>
  <link href="dist/output.css" rel="stylesheet">
</head>
<body class="bg-gray-100 text-center p-10">
  <h1 class="text-3xl font-bold text-blue-600">Tailwind 4 CLI Setup Works!</h1>
</body>
</html>
```

🧪 2. Method 2: Using Tailwind Play CDN (For Quick Demos, No Build Step)

Use this **only** for demo/testing. You won't get purging or custom configs.

```html
<!DOCTYPE html>

<html lang="en">

<head>

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

  <title>Tailwind 4 CDN</title>

</head>

<body class="bg-emerald-100 p-10">

  <h1 class="text-2xl text-emerald-700">Hello from Tailwind 4 CDN!</h1>

</body>

</html>
```

✅ You can also customize using:

```html
<script>

tailwind.config = {

  theme: {

    extend: {

      colors: {

        skyblue: '#38bdf8',

      }

    }

  }

}

</script>
```

👷 **3. Method 3: Using PostCSS (For Custom Builds or Frameworks)**

🧱 **Step-by-Step**

1. **Install PostCSS + Tailwind**

```
npm install -D tailwindcss postcss autoprefixer
npx tailwindcss init -p
```

2. **Structure**

```
project/
├── src/
│   └── input.css
├── postcss.config.js
├── tailwind.config.js
├── index.html
└── dist/output.css
```

3. **postcss.config.js**

```
module.exports = {
 plugins: {
   tailwindcss: {},
   autoprefixer: {},
 }
}
```

4. **Build CSS**

```
npx postcss src/input.css -o dist/output.css --watch
```

💼 **4. Method 4: Install via Frameworks (Vite, Next.js, etc.)**

Example: Vite

```
npm create vite@latest my-app
```

```
cd my-app
```

```
npm install
```

```
npm install -D tailwindcss postcss autoprefixer
```

```
npx tailwindcss init -p
```

- Configure tailwind.config.js with:

```
content: ["./index.html", "./src/**/*.{js,ts,jsx,tsx}"]
```

```
@tailwind base;
```

```
@tailwind components;
```

```
@tailwind utilities;
```

- Import index.css in main.jsx.

Run:

```
npm run dev
```

❓ Interview Question Highlights

Question	Answer
What is Tailwind CLI?	It's a standalone tool to build and compile Tailwind CSS without additional build tools.
Can I use Tailwind without a build process?	Yes, using CDN, but it's not recommended for production.
What is the benefit of the CLI method?	Simplicity, no need for complex setup like Webpack or PostCSS.
Tailwind v4 new features?	Includes color palette updates, faster compiler, better plugin support, and support for CSS nesting.