

Create React App with Vite — Quick Commands & Explanation

This is a one-page cheatsheet to create a modern React app using **Vite**, avoid interactive prompts, install Tailwind, and handle daily workflows (start, stop, reopen). Follow commands exactly in PowerShell (Windows) or a POSIX shell (macOS / Linux).

1 — Quick non-interactive project creation (recommended)

Create a new React project (JavaScript) without interactive prompts:

```
# creates a new project named `w4day23` using the React (JS) template
npm create vite@latest w4day23 -- --template react
```

Why this works: the `-- --template react` portion is forwarded to `create-vite` and tells it exactly which template to use so it doesn't prompt for the framework/variant.

2 — Alternative: interactive (if you want choices)

If you prefer the guided prompts use:

```
npm create vite@latest
```

Then choose `React` → `JavaScript` → `No` for experimental options, etc.

3 — Enter project & install dependencies

After project scaffolding finishes, run:

```
cd w4day23
npm install
```

`npm install` will install the dev server and other dependencies created by Vite.

4 — Start dev server

To start the development server:

```
npm run dev
```

Open the URL shown in your terminal (usually `http://localhost:5173`).

5 — Stop dev server and reopen later

If you close the terminal or stop the server (Ctrl+C):

To reopen the project later (from the project folder):

```
# if node_modules still exists
npm run dev

# if node_modules was removed (e.g., you deleted it or pulled repo on a new
machine)
npm install
npm run dev
```

6 — Build for production and preview

```
# create optimized production build
npm run build

# preview the production build locally
npm run preview
```

7 — Installing Tailwind CSS (step-by-step)

From your project root run exactly:

```
# 1) install Tailwind and PostCSS tooling
npm install -D tailwindcss postcss autoprefixer

# 2) generate config files (creates tailwind.config.js and postcss.config.js)
npx tailwindcss init -p
```

If `npx tailwindcss init -p` errors with "'tailwind' is not recognized..." then run the `npx` command explicitly via npm's `create-vite` project bin: you can also try the direct binary path (POSIX):

```
# POSIX (macOS / Linux)
./node_modules/.bin/tailwindcss init -p
```

On Windows PowerShell, if that fails, try:

```
npx.cmd tailwindcss init -p
```

If the CLI still isn't present in `node_modules/.bin`, reinstall the package:

```
npm uninstall tailwindcss  
npm install -D tailwindcss
```

If PowerShell prevents scripts from running, run as admin or allow execution for current user:

```
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
```

8 — Tailwind configuration (what to edit)

Open `tailwind.config.js` and set `content` to scan your source files:

```
// tailwind.config.js  
module.exports = {  
  content: ['./index.html', './src/**/*.js,jsx,ts,tsx'],  
  theme: { extend: {} },  
  plugins: [],  
}
```

Add Tailwind directives to your CSS (for Vite default: `src/index.css` or `src/main.css`):

```
@tailwind base;  
@tailwind components;  
@tailwind utilities;
```

Then import that CSS from `src/main.jsx` or `src/main.js` (Vite default):

```
import './index.css' // or ./main.css depending on template
```

Restart `npm run dev` if necessary.

9 — One-liners & shortcuts

Create + install + start (single-line):

```
# non-interactive create + install + start  
npm create vite@latest w4day23 -- --template react && cd w4day23 && npm  
install && npm run dev
```

Create + Tailwind + start (single-line):

```
npm create vite@latest w4day23 -- --template react && cd w4day23 && npm
install && npm install -D tailwindcss postcss autoprefixer && npx
tailwindcss init -p && npm run dev
```

10 — If you must stay with Create React App (CRA)

CRA still works but can cause headaches with some CLI tool installs on Windows. Quick CRA commands:

```
npx create-react-app my-app
cd my-app
npm install
npm start
```

CRA doesn't require Vite knowledge, but Vite is faster and increasingly the industry standard. If you're comfortable with CRA, you can keep using it; however, when tooling like Tailwind CLI misbehaves, Vite tends to avoid those problems.

11 — Troubleshooting (common Windows issues)

- `tailwind` is not recognized: check `node_modules/.bin` for `tailwindcss.cmd`. If missing, reinstall `tailwindcss`.
- PowerShell execution policy: run `Set-ExecutionPolicy RemoteSigned -Scope CurrentUser`.
- If `npm create vite` still prompts, pass `-- --template react` to skip framework selection.
- If `npm` warns about unknown flags when you append `-y`, remove `-y`; `create-vite` does not support `-y` as some other tools do.

12 — Summary / Recommendation

- Use `npm create vite@latest my-app -- --template react` for a fast, predictable, non-interactive setup.
- Use `npm run dev` to start the dev server and `npm run build` to create production bundles.
- For Tailwind, install the three packages and run `npx tailwindcss init -p`. If PowerShell blocks you, use the `.cmd` or adjust policy.

If you want, I can also generate a **short bash/PowerShell script** that runs all the commands for you (choose platform), or convert this into a printable checklist. Tell me which you prefer.