
 Vue.js

Learn ▾

Special Sponsor

 Standard Library

Style Guide BETA

Rule Categories

- Priority A: Essential
- Priority B: Strongly Recommended
- Priority C: Recommended
- Priority D: Use with Caution

Priority A Rules: Essential

- Multi-word component names
- Component data
- Prop definitions
- Keyed v-for
- Avoid v-if with v-for
- Component style scoping
- Private property names

Priority B Rules: Strongly Recommended

- Component files
- Single-file component filename casing
- Base component names
- Single-instance component names
- Tightly coupled component names
- Order of words in component names
- Self-closing components
- Component name casing in templates
- Component name casing in JS/Jsx

Style Guide

This is the official style guide for Vue-specific code. If you use Vue in a project, it's a great reference to avoid errors, bikeshedding, and anti-patterns. However, we don't believe that any style guide is ideal for all teams or projects, so mindful deviations are encouraged based on past experience, the surrounding tech stack, and personal values.

For the most part, we also avoid suggestions about JavaScript or HTML in general. We don't mind whether you use semicolons or trailing commas. We don't mind whether your HTML uses single-quotes or double-quotes for attribute values. Some exceptions will exist however, where we've found that a particular pattern is helpful in the context of Vue.

Soon, we'll also provide tips for enforcement. Sometimes you'll simply have to be disciplined, but wherever possible, we'll try to show you how to use ESLint and other automated processes to make enforcement simpler.

Finally, we've split rules into four categories:

Rule Categories

Priority A: Essential

These rules help prevent errors, so learn and abide by them at all costs. Exceptions may exist, but should be very rare and only be made by those with expert knowledge of both JavaScript and Vue.



„This is the official style guide for Vue-specific code. If you use Vue in a project, it's a great reference to avoid errors, bikeshedding, and anti-patterns. **However, we don't believe that any style guide is ideal for all teams or projects**, so mindful deviations are encouraged based on past experience, the surrounding tech stack, and personal values.“