

VsCode 插件

```
Vue VSCode Snippets
Vue Official
Eslint
Prettier
stylelint
```

Node 版本 18+, 安装pnpm

注意：接下来所有安装都要通过pnpm 安装！！

```
pnpm create vue@latest
```

```
S:\lint>pnpm create vue@latest
.../18f96477a3d-bf0 | +1 +
.../18f96477a3d-bf0 | Progress: resolved 1, reused 0, downloaded 1, added 1, done

Vue.js - The Progressive JavaScript Framework

✓ 请输入项目名称: ... vue-ts-demo
✓ 是否使用 TypeScript 语法? ... 否 / 是
✓ 是否启用 JSX 支持? ... 否 / 是
✓ 是否引入 Vue Router 进行单页面应用开发? ... 否 / 是
✓ 是否引入 Pinia 用于状态管理? ... 否 / 是
✓ 是否引入 Vitest 用于单元测试? ... 否 / 是
✓ 是否要引入一款端到端 (End to End) 测试工具? » 不需要
✓ 是否引入 ESLint 用于代码质量检测? ... 否 / 是
✓ 是否引入 Prettier 用于代码格式化? ... 否 / 是
✓ 是否引入 Vue DevTools 7 扩展用于调试? (试验阶段) ... 否 / 是

正在初始化项目 S:\lint\vue-ts-demo...
项目初始化完成, 可执行以下命令:

  cd vue-ts-demo
  pnpm install
  pnpm format
  pnpm dev
```

```
pnpm i eslint-config-airbnb-base -D
pnpm i eslint-import-resolver-typescript -D
```

.eslintrc.cjs

```
/* eslint-env node */
require('@rushstack/eslint-patch/modern-module-resolution')

module.exports = {
  root: true,
  extends: [
    'plugin:vue/vue3-essential',
    'eslint:recommended',
    'airbnb-base',
    '@vue/eslint-config-typescript',
    '@vue/eslint-config-prettier/skip-formatting'
  ]
}
```

```

],
parserOptions: {
  ecmaVersion: 'latest'
},
settings: {
  'import/resolver': {
    typescript: {
      project: './tsconfig.*.json'
    }
  }
},
rules: {
  'import/no-extraneous-dependencies': 'off',
  'import/extensions': [
    'error',
    'ignorePackages',
    { js: 'never', jsx: 'never', ts: 'never', tsx: 'never' }
  ],
  'import/prefer-default-export': 'off',
  'no-plusplus': 'off'
}
}

```

.eslintignore

```

dist
node_modules
public
.husky
.vscode
.idea
*.sh
*.md
index.html

.eslintrc.cjs
.prettierrc.json
.stylelintrc.cjs

```

env.d.ts

```

/// <reference types="vite/client" />
declare module '*.vue' {
  import type { DefineComponent } from 'vue'

  const component: DefineComponent<{}, {}, any>
  export default component
}

```

执行npm run lint 可以整体检查效果了

stylelint 配置

```
pnpm i stylelint stylelint-config-standard stylelint-config-recommended-scss stylelint-config-recommended-vue postcss postcss-html postcss-scss stylelint-config-recess-order stylelint-config-html -D
```

依赖	说明	备注
stylelint	stylelint 核心库	stylelint
stylelint-config-standard	Stylelint 标准共享配置	stylelint-config-standard 文档
stylelint-config-recommended-scss	扩展 stylelint-config-recommended 共享配置并为 SCSS 配置其规则	stylelint-config-recommended-scss 文档
stylelint-config-recommended-vue	扩展 stylelint-config-recommended 共享配置并为 Vue 配置其规则	stylelint-config-recommended-vue 文档
stylelint-config-recess-order	提供优化样式顺序的配置	CSS 书写顺序规范
stylelint-config-html	共享 HTML (类似 HTML) 配置, 捆绑 postcss-html 并对其进行配置	stylelint-config-html 文档
postcss-html	解析 HTML (类似 HTML) 的 PostCSS 语法	postcss-html 文档
postcss-scss	PostCSS 的 SCSS 解析器	postcss-scss 文档, 支持 CSS 行类注释

.vscode配置

```
{
  "editor.codeActionsOnSave": {
    "source.fixAll": "explicit"
  },
  "editor.formatOnSave": true,
  "editor.defaultFormatter": "esbenp.prettier-vscode",
  "stylelint.validate": ["css", "scss", "less", "vue"]
}
```

.stylelintignore

```
dist
node_modules
public
.husky
.vscode
.idea
*.sh
*.md

stats.html
```

.stylelintrc

```
module.exports = {
  // 继承推荐规范配置
  extends: [
    'stylelint-config-standard',
    'stylelint-config-recommended-scss',
```

```

    'stylelint-config-recommended-vue/scss',
    'stylelint-config-html/vue',
    'stylelint-config-recess-order'
  ],
  // 指定不同文件对应的解析器
  overrides: [
    {
      files: ['**/*.vue,html'],
      customSyntax: 'postcss-html'
    },
    {
      files: ['**/*.css,scss'],
      customSyntax: 'postcss-scss'
    }
  ],
  // 自定义规则
  rules: {
    // 允许 global 、 export 、 v-deep等伪类
    'selector-pseudo-class-no-unknown': [
      true,
      {
        ignorePseudoClasses: ['global', 'export', 'v-deep', 'deep']
      }
    ],
    'selector-class-pattern': null,
    // 'selector-no-vendor-prefix': null,
    // 'value-no-vendor-prefix': null,
    // 'alpha-value-notation': null,
    'color-function-notation': null,
    // 'rule-empty-line-before': null,
    'no-descending-specificity': null,
    // 'number-leading-zero': null,
    // 'declaration-block-no-redundant-longhand-properties': null,
    'font-family-no-missing-generic-family-keyword': null
  }
}

```

husky

先搞一个git 仓库，或者git init

```
npx husky-init
```

```
pnpm install
```

Lint-staged 增量检测提交代码

```
pnpm i lint-staged -D
```

修改package.json

```

"scripts": {
  "prepare": "husky install",
  "lint:lint-staged": "lint-staged"
},
"lint-staged": {
  "*.{js,ts}": [
    "eslint --fix",
    "prettier --write"
  ],
  "*.{cjs,json}": [
    "prettier --write"
  ],
  "*.{vue,html}": [
    "eslint --fix",
    "prettier --write",
    "stylelint --fix --allow-empty-input"
  ],
  "*.{scss,css}": [
    "stylelint --fix --allow-empty-input",
    "prettier --write"
  ]
}

```

去如下路径 `.husky > _ > pre-commit` 修改提交hooks

```

# npm test
npm run lint:lint-staged --allow-empty

```

commitlint

```
pnpm install @commitlint/cli @commitlint/config-conventional -D
```

commitlint.config.cjs

```

module.exports = {
  // 继承的规则
  extends: ['@commitlint/config-conventional'],
  // @see: https://commitlint.js.org/#/reference-rules
  rules: {
    'subject-case': [0], // subject大小写不做校验

    // 类型枚举, git提交type必须是以下类型
    'type-enum': [
      2,
      'always',
      [
        'feat', // 新增功能
        'fix', // 修复缺陷
        'docs', // 文档变更
        'style', // 代码格式 (不影响功能, 例如空格、分号等格式修正)
        'refactor', // 代码重构 (不包括 bug 修复、功能新增)
        'perf', // 性能优化
        'test', // 添加疏漏测试或已有测试改动
        'build', // 构建流程、外部依赖变更 (如升级 npm 包、修改 webpack 配置等)
        'ci', // 修改 CI 配置、脚本
        'revert', // 回滚 commit

```

```
    'chore' // 对构建过程或辅助工具和库的更改（不影响源文件、测试用例）
  ]
}
}
```

生成commit 提交hook

```
npx husky add .husky/commit-msg "npx --no -- commitlint --edit $1"
```

```
git add .
git commit -m"fix: 我干了什么"
```

完结撒花!