

도구

온라인으로 사용해보기

Vue SFC를 사용해 보기 위해서 컴퓨터에 무엇을 설치할 필요가 없습니다. 브라우저에서 바로 실행할 수 있는 온라인 연습장이 있습니다:

Vue SFC 온라인 연습장

항상 최신 버전 상태

컴포넌트 컴파일 결과를 확인할 수 있도록 설계

StackBlitz(Vue + Vite)

브라우저에서 실제 Vite 개발 서버를 실행하는 IDE와 유사한 환경

로컬 설정과 유사한 환경

또한 버그 신고 시 이러한 온라인 연습장을 사용하여 재현 코드를 제공하는 것이 좋습니다.

프로젝트의 스케폴딩

Vite

Vite는 Vue SFC를 지원하는 가볍고 빠른 최신 빌드 도구입니다. Vue의 저자이기도 한 Evan You가 만들었습니다!

Vite + Vue를 시작하려면 다음을 실행하기만 하면 됩니다:

```
npm      pnpm      yarn      bun
```

```
$ npm create vue@latest
```

sh

이 명령은 공식 Vue 프로젝트 스케폴딩 도구인 **create-vue**를 설치하고 실행합니다.

Vite에 대한 자세한 내용은 **Vite** 문서를 확인하세요.

Vite 프로젝트에서 Vue 관련 동작을 구성하려면(예: Vue 컴파일러에 옵션 전달) **@vitejs/plugin-vue** 문서를 확인하세요.

위에서 언급한 두 온라인 연습장은 Vite 프로젝트 파일로 다운로드를 지원합니다.

Vue CLI

Vue CLI는 웹팩 기반의 Vue 공식 툴체인입니다. 특정 웹팩 전용 기능에 의존하는 유지보수 상태가 아니라면, Vite로 새 프로젝트를 시작하는 것이 좋습니다. Vite는 대부분의 경우 우수한 개발자 경험을 제공합니다.

Vue CLI에서 Vite로 마이그레이션하는 방법에 대한 정보:

VueSchool.io에서 **Vue CLI -> Vite** 마이그레이션 가이드에 대해 알아보기(영문)

자동 마이그레이션에 도움이 되는 도구/플러그인

브라우저 내 템플릿 편집에 대한 참고 사항

빌드 과정 없이 Vue를 사용할 때, 컴포넌트 템플릿은 페이지의 HTML에 직접 작성되거나 인라인된 JavaScript 문자열로 작성됩니다. 이러한 경우 Vue는 즉시 템플릿 컴파일을 수행하기 위해 템플릿 컴파일러를 브라우저에 제공해야 합니다. 반면에 빌드 과정으로 템플릿을 미리 컴파일하면 컴파일러가 필요하지 않습니다. 클라이언트 번들 크기를 줄이기 위해 Vue는 다양한 사용 사례에 최적화된 다른 "빌드"를 제공합니다.

`vue.runtime.*` 으로 시작하는 빌드 파일은 **런타임 전용 빌드**입니다. 컴파일러를 포함하지 않습니다. 이러한 빌드를 사용할 때 모든 템플릿은 빌드 과정을 통해 사전 컴파일되어야 합니다.

.runtime 을 포함하지 않는 빌드 파일은 **전체 빌드**입니다. 컴파일러를 포함하고 브라우저에서 직접 템플릿 컴파일을 지원합니다. 그러나 페이로드가 ~14kb 증가합니다.

SFC의 모든 템플릿이 미리 컴파일되어 있기 때문에 기본 도구 설정은 런타임 전용 빌드를 사용합니다. 어떤 이유로 빌드 과정에서도 브라우저 내 템플릿 컴파일이 필요한 경우, vue 의 대신 vue/dist/vue.esm-bundler.js 로 지정하여 빌드 도구를 구성하면 됩니다.

빌드 과정이 없는 사용을 위한 더 가벼운 대안을 찾고 있다면 **petite-vue**를 확인하십시오.

IDE 지원

권장 IDE 설정은 **VS Code + Vue - Official extension**(이전의 Volar) 입니다. Vue - Official extension은 문법 강조 표시, TypeScript 지원, 템플릿 표현식 및 컴포넌트 props에 대한 인텔리센스를 제공합니다.

TIP

Vue - Official extension은 Vue 2 전용 공식 VS Code 확장인 **Vetur**를 대체합니다. 현재 **Vetur**가 설치되어 있는 경우 Vue 3 프로젝트에서 비활성화해야 합니다.

WebStorm은 Vue SFC에 대한 지원을 기본적으로 제공하며 매우 훌륭합니다.

Language Service Protocol(LSP)을 지원하는 다른 IDE도 LSP를 통해 Volar의 핵심 기능을 활용할 수 있습니다.

서브라임 텍스트(Sublime Text): **LSP-Volar**

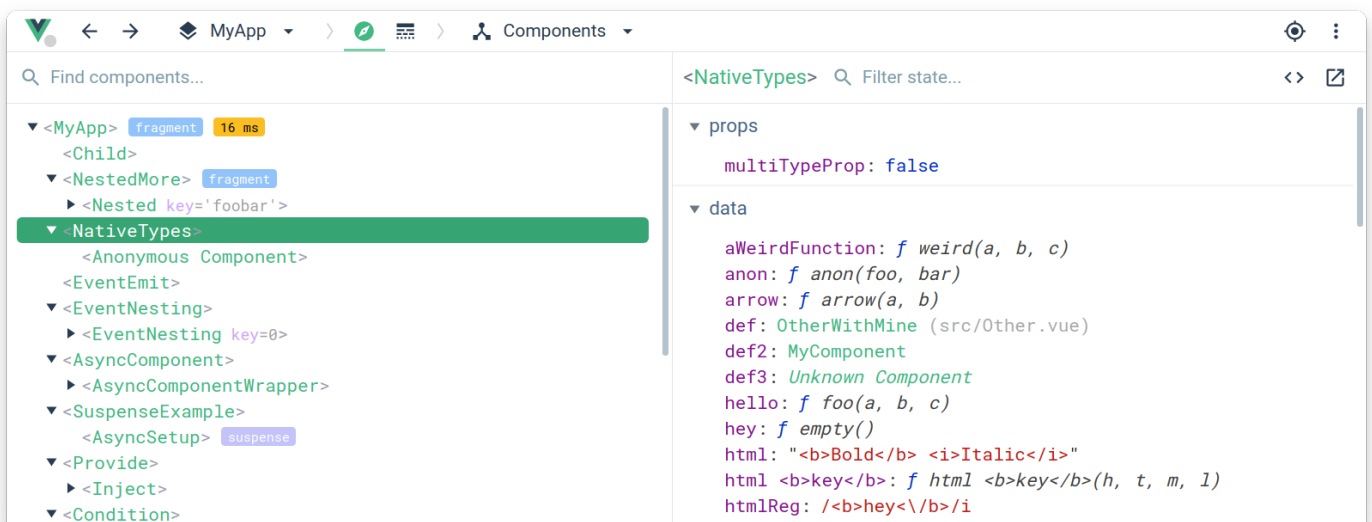
vim / Neovim: **coc-volar**

emacs: **lsp-mode**

브라우저 개발자 도구

Vue School의 무료 동영상 강의 보기

Vue 브라우저 개발자 도구 확장 프로그램을 사용하면, Vue 앱의 컴포넌트 트리를 탐색하고 개별 컴포넌트의 상태를 검사하고 상태 관리 이벤트 및 프로필 성능을 추적할 수 있습니다.



Vue 브라우저 개발자 도구 문서
Chrome 확장 프로그램
Firefox 애드온
Edge 확장 프로그램
독립형 Electron 앱

타입스크립트

주요 내용: TypeScript로 Vue 사용하기.

Volar는 템플릿 표현식 및 교차 컴포넌트 props 유효성 검사를 포함하여 `<script lang="ts">` 블록을 사용하여 SFC에 대한 유형 검사를 제공합니다.

명령줄에서 동일한 유형 검사를 수행하거나 SFC용 `d.ts` 파일을 생성하려면 **vue-tsc** 를 사용합니다.

테스팅

주요 내용: 테스트 가이드.

Cypress는 E2E 테스트에 권장됩니다. **Cypress** 컴포넌트 테스트 러너를 통해 Vue SFC에 대한 컴포넌트 테스트에도 사용할 수 있습니다.

Vitest는 속도를 중시하는 Vue/Vite 팀원들이 만든 테스트 러너입니다. Vite 기반 앱이 유닛/컴포넌트 테스트를 위해 동일한 즉각적인 피드백 루프를 제공하도록 특별히 설계되었습니다.

Jest는 **vite-jest**를 통해 Vite와 함께 작동하도록 만들 수 있습니다. 그러나 이는 Vitest가 훨씬 더 효율적인 통합으로 유사한 기능을 제공하므로, Vite 기반 설정으로 마이그레이션해야 하는 기존 Jest 기반 테스트 제품군이 있는 경우에만 권장됩니다.

Linting(코드 유효성 검사)

Vue 팀은 SFC별 린팅 규칙을 지원하는 **ESLint** 플러그인인 **eslint-plugin-vue**를 유지 관리합니다.

이전에 Vue CLI를 사용하던 사용자는 웹팩 로더를 통해 린터를 구성하는 데 익숙할 수 있습니다. 그러나 Vite 기반 빌드 설정을 사용할 때 일반적인 권장 사항은 다음과 같습니다.

`npm install -D eslint eslint-plugin-vue` 를 실행한 다음 **eslint-plugin-vue** 의 설정 가이드를 따릅니다.

VS Code용 **ESLint**와 같은 IDE 확장 **ESLint**를 설정하면, 개발 중에 편집기에서 바로 린터 피드백을 얻을 수 있습니다. 이것은 개발 서버를 시작할 때 불필요한 Linting 비용을 방지합니다.

프로덕션 빌드 명령의 일부로 ESLint를 실행하여 프로덕션으로 빌드 실행하기 전에 전체 린터 피드백을 받습니다.

(선택 사항) git commit 시 수정된 파일을 자동으로 린트하도록 **lint-staged**와 같은 도구를 설정합니다.

Formatting

VS Code 확장 **Volar**는 Vue SFC에 대한 포매팅을 즉시 제공합니다.

또는 **Prettier**는 빌트인 Vue SFC 형식 지원을 제공합니다.

SFC의 커스텀 블록 통합하기

커스텀 블록은 다른 리퀘스트 쿼리로 가져온 동일한 Vue 파일로 컴파일 됩니다. 이러한 가져오기 요청을 처리하는 것은 기본 빌드 도구에 달려 있습니다.

Vite를 사용하는 경우, 일치하는 커스텀 블록을 실행 가능한 JavaScript로 변환하려면, 커스텀 Vite 플러그인을 사용해야 합니다. 예제

Vue CLI 또는 일반 웹팩을 사용하는 경우, 일치하는 블록을 변환하도록 웹팩 로더를 구성해야 합니다. 예제

저수준 패키지

@vue/compiler-sfc

Docs

이 패키지는 Vue 핵심 모노레포의 일부이며 항상 기본 vue 패키지와 동일한 버전으로 게시됩니다. 기본 vue 패키지의 의존성으로 포함되며 vue/compiler-sfc 아래에 프록시되므로 개별적으로 설치할 필요가 없습니다.

패키지 자체는 Vue SFC 처리를 위한 저수준 유틸리티를 제공하며, 커스텀 툴에서 Vue SFC를 지원해야 하는 툴 개발자를 위한 것입니다.

TIP

이 패키지의 버전이 Vue 런타임과 동기화되도록 하려면 항상 vue/compiler-sfc deep import를 통해 이 패키지를 사용하는 것이 좋습니다.

@vitejs/plugin-vue

Docs

Vite에서 Vue SFC 지원을 제공하는 공식 플러그인입니다.

vue-loader

Docs

webpack에서 Vue SFC 지원을 제공하는 공식 로더입니다. Vue CLI를 사용하는 경우, Vue CLI에서 vue-loader 옵션 수정에 대한 문서를 참조하세요.

기타 온라인 연습장

VueUse 연습장

Repl.it(Vue + Vite)

CodeSandbox

Codepen

Components.studio

WebComponents.dev