

## 2.Vue 개념

# 1-1. Vue개발환경 설정하기

## 1. vue.js 주요개념

- 대화형 Web Interface 를 개발하는데 사용하는 Java Script framework
- 점진적으로 채택할 수 있게 설계된 프레임워크
- 학습 곡선이 낮아 기반 지식을 가지면 쉽게 시작할 수 있음
- **MVVM** Pattern View Model  
Model-View-View Model로 구조화하여 개발

## 2. vue.js 특징

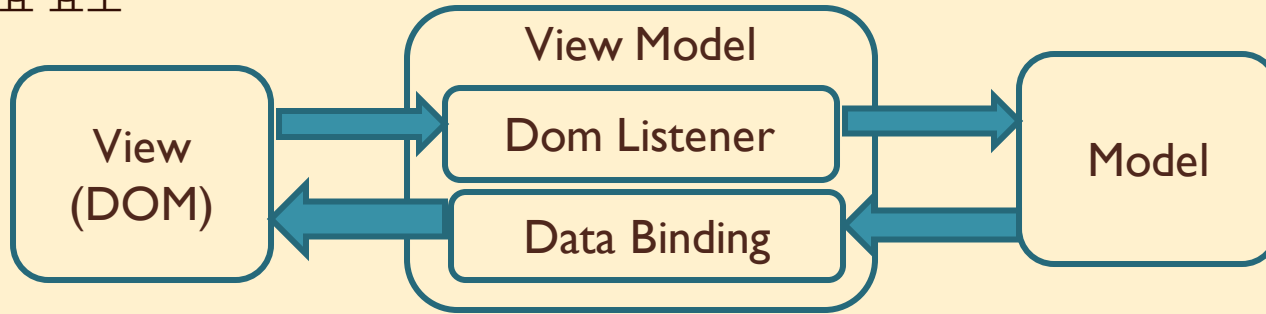
- 양방향 데이터 바인딩과 Virtual DOM을 사용해 Angular와 React.js의 장점을 동시에 가짐
- 뷰와 상태 관리, SPA 라우터 등 핵심 요소에만 집중하며 높은 성능을 가짐.
- HTML과 CSS코드를 재사용가능하게 함.

## 3. 개발 환경(대표적 유형 2가지)

CLI를 사용하는 개발 환경(실무 많이 사용)	CDN 방식으로 개발 (통합 배포 유리)
node.js	STS
npm	HTML + CDN 선언
visual studio	

## 1-2. Vue 특징 및 실행환경 설정하기

### 2. vue.js 주요 요소



항목	주요기능
View	사용자에게 보이는 화면
Dom	HTML 문서에 들어가는 요소(Tag , Class,속성) 정보를 담고있는 Data Tree
Dom Listener	Dom 변경 내역에 즉각 반응 , 특정 Logic 수행 장치
Model	Server에서 가져온 Data를 Java Script 객체 형태로 저장
Data Binding	View로 표시되는 내용과 Model의 Data를 동기화
View Model	View와 model의 중간 영역 Dom Listener과 Data Binding을 제공하는 영역

## 1-3. Vue개발환경 설정하기(node.js 설치 확인)

1) cmd 선택 (명령어 입력) → 각각의 version 확인

- node --version
- npm --version

# 1-3. Vue개발환경 설정하기(CDN 방식)

## 1. Vue 설치방법

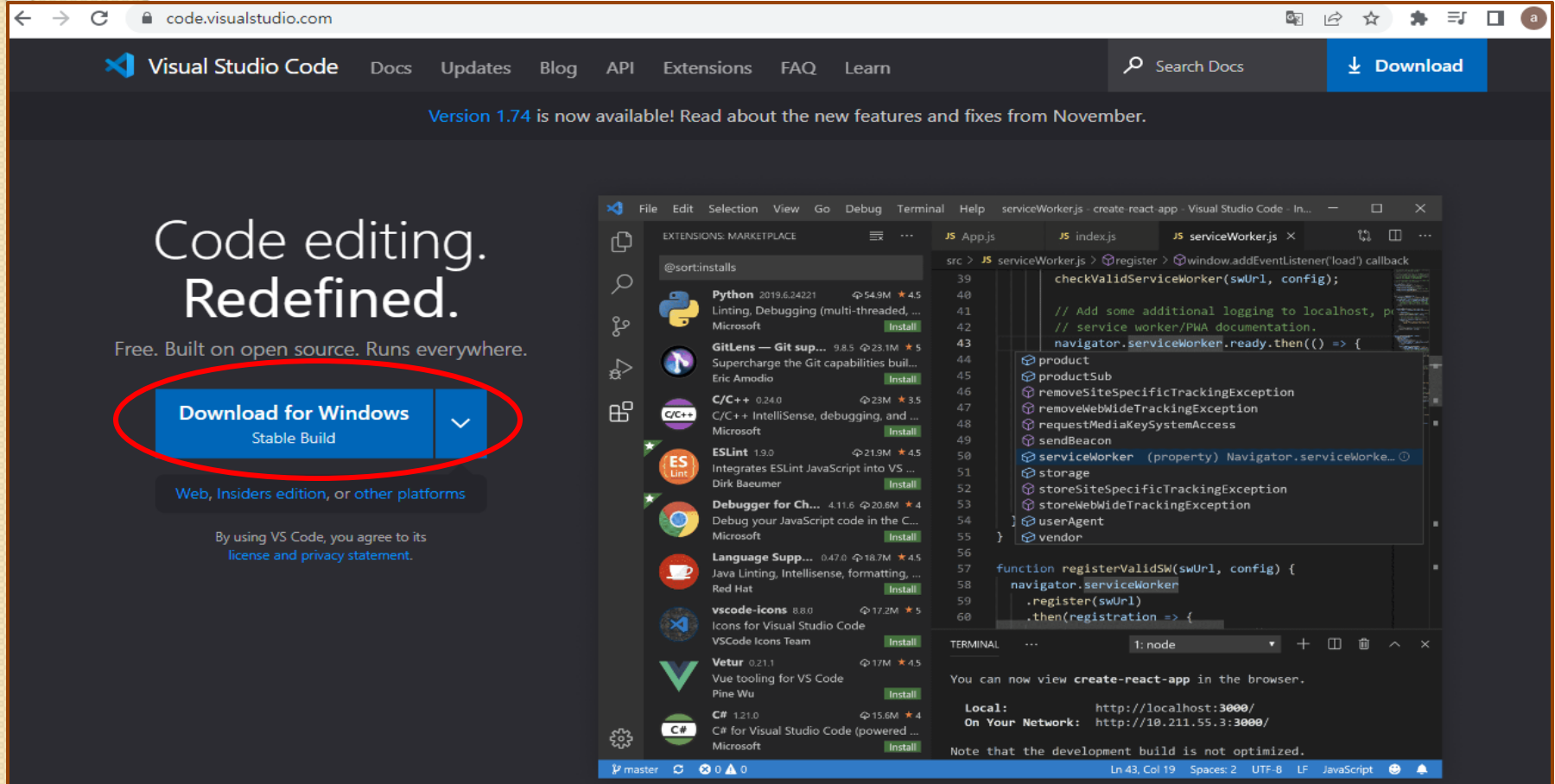
- 일반적으로 **Vue CLI** 방식을 많이 활용
- Vue CLI 방식 적용시 프로젝트를 **scaffold**하고, 최신 프론트엔드 워크플로우를 위한 **batteries-included** build를 제공
- **scaffold**는 **개발을 용이하기 시작할 수 있는 발판**을 제공.
- **batteries-included**는 외부 라이브러리를 더하지 않아도 기본적으로 제공하는 표준 라이브러리만으로도 시작하는데 가능
- Vite는 공식문서에서 개발 권장

방식	방법	내용
CDN	<pre>&lt;script src="https://unpkg.com/vue@next"&gt;&lt;/script&gt; &lt;!-- 또는 --&gt; &lt;script src="https://unpkg.com/vue@3.2.31"&gt;&lt;/script&gt;</pre>	프로토타이핑 또는 학습 목적이라면, 아래 코드로 최신 버전을 사용
npm	<pre>npm install vue@next</pre>	Vue를 사용하여 대규모 애플리케이션을 구축할 때 NPM를 이용한 설치를 권장
<b>Vue CLI</b>	<pre>npm install -g @vue/cli</pre>	Vue CLI는 <b>웹팩 기반 빌드 도구</b> 특정 웹팩 기능에 의존하지 않는한 <b>Vite</b> 로 새 프로젝트를 시작 Vue CLI를 사용하기 위해서는 @vue/cli v4.5 이상의 버전을 설치
<b>Vite</b> + Vue	<pre>\$ npm init vue@latest  cd {product name} npm install npm run dev</pre>	Vite는 <b>개발 서버를 구동할 때 매우 빠름.</b> 소스 코드의 변경이 일어났을 때 전체 모듈을 번들링 하는게 아 니라 변경된 모듈만 교체하기 때문에 개발을 더욱 빠르게 진행  공식 Vue 프로젝트 스캐폴딩 도구인 create-vue 를 설치하고 실행

## 2-1. VS Code 설치하기

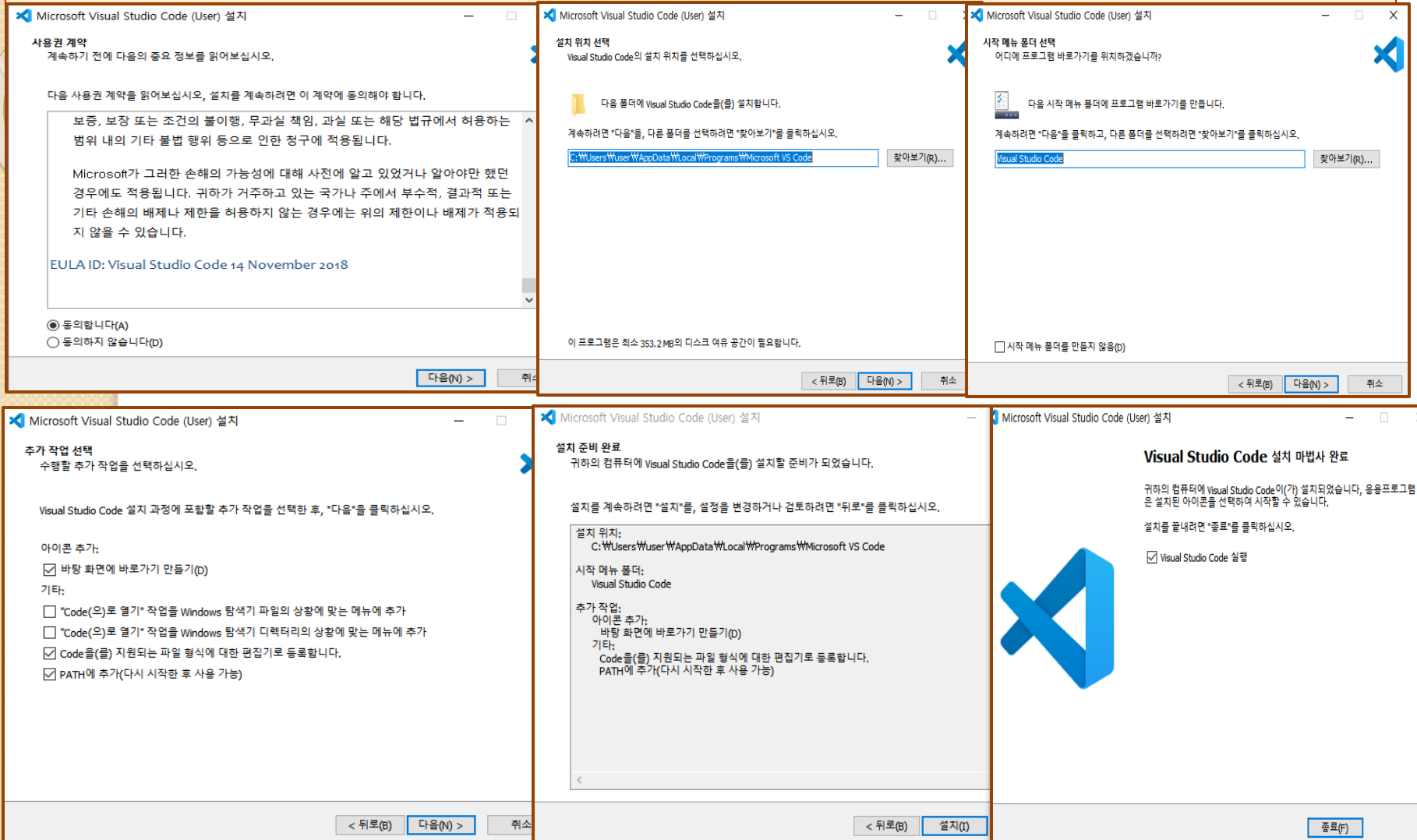
- 1] VS Code 설치하기 site  
- <https://code.visualstudio.com/>

### 2] 설치



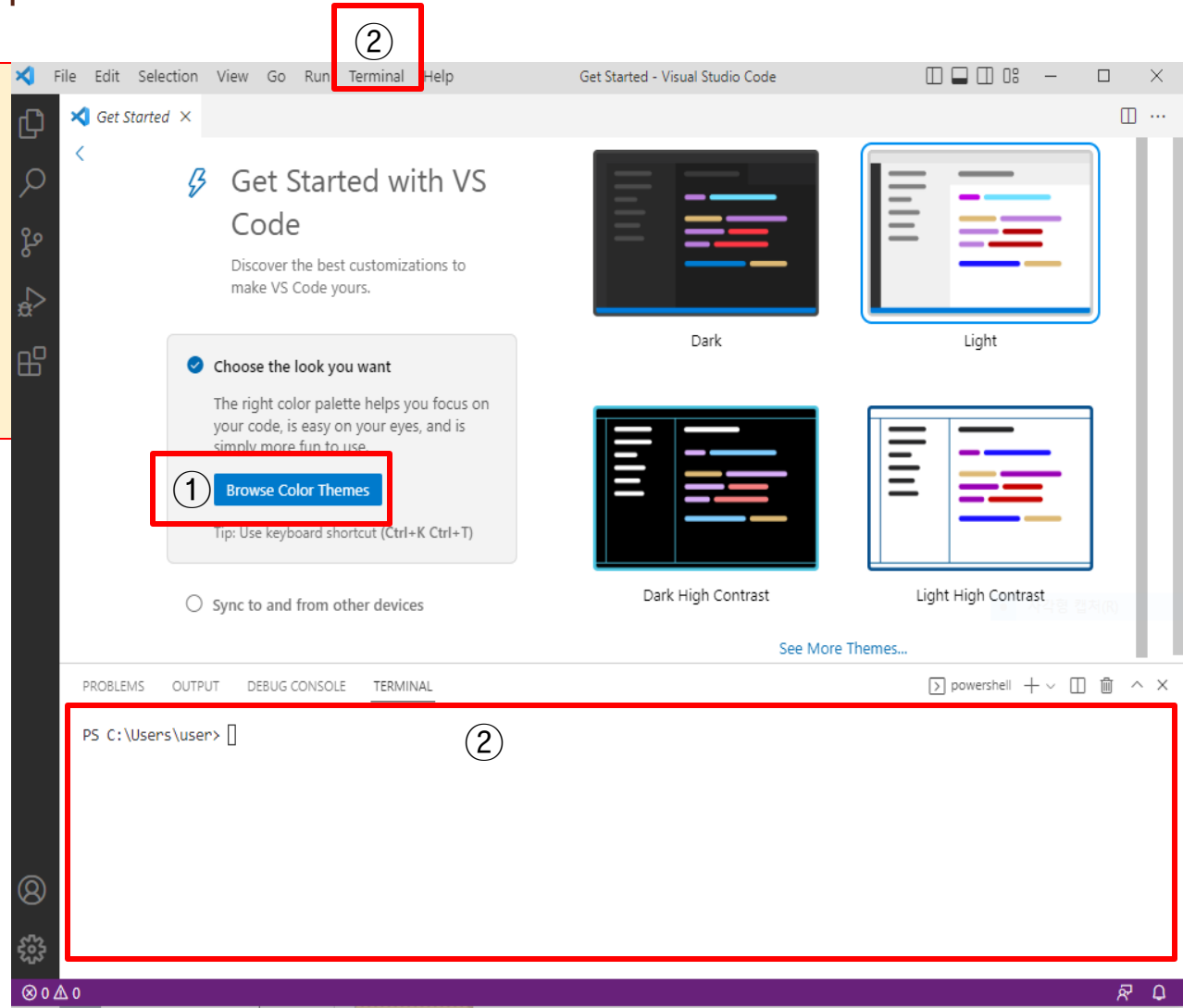
## 2-2. VS Code 설치하기

### 1] VSCodeUserSetup-x64-1.74.2.exe 더블클릭



## 2-3. VS Code 실행하기

- ① 전체 배경화면 설정
- ② Terminal → New Terminal  
선택하면  
아래와 같은 Terminal 환경에서  
각 종 명령어 입력 가능





## 3-1. Vue개념

### 1) Vue 개념

- ① 사용자 인터페이스를 만들기 위한 **동적 JavaScript 프레임워크**
- ② HTML, CSS 및 JavaScript에 대한 기본적인 이해만으로 Vue.js에서 웹 앱 구성을 시작
- ③ 짧은 학습 곡선은 이 점진적 프레임워크의 특징,
- ④ 유연하며 웹 개발에서 유틸리티 또는 포괄적인 프레임워크로 사용

### 2) Vue 특징

#### [1] 선언형(Declarative)

- 컴포넌트(원하는 결과, 뷰)를 얻기 위해 <tag></tag> jsx 문법을 통해 구현

#### [2] Component 단위로 개발(Component Based Development)

- 작은 레고 블록을 조립하듯 Component 들을 모아서 개발

#### [3] **Virtual DOM(가상돔)**기반

- 어딘가에 데이터로만 존재하는 것(눈에 보이지 않는다.) 하는 역할이 뭐냐 하면 가상의 트리 구조를 만들고 이를 실제로 브라우저에 그림

#### [4] **JSX**

- JSX는 자바스크립트를 확장한 문법이다. = 자바스크립트

#### [5] **SPA(Single Page Application)**

- 사용자와의 상호 작용이 많아짐에 따라, 데이터 정보 전송 과부화로 인한 속도 저하 등 문제점
- 라이브러리 또는 프레임워크는 **뷰 렌더링**을 서버가 아닌 **웹 브라우저**가 담당
- SPA의 경우 서버에서 제공하는 페이지가 한 개
- 다른 주소에 다른 화면을 보여주는것을 Routing(라우팅)
- 앱의 규모가 커지면 자바스크립트 파일이 너무 커짐

<https://github.com/stylehosting/example-spring>

## 3-2-1. component

### 1] component 개념

- ① 자바스크립트에서 재사용할 수 있도록 코드를 분리한 파일을 **모듈**이라고 정의
- ② Vue에서도 마찬가지로 **UI(HTML, CSS, JS)**를 재사용할 수 있도록 정의한 것을 **컴포넌트**
- ③ 컴포넌트로 캡슐화 한 후 필요한 곳에서 사용
- ④ 컴포넌트로 독립적으로 분리함으로써 코드를 클린하게 하여 유지보수

### 2] component 정의 방법

#### ① 문자열 템플릿 (string template)

```
// BookComponent.js 파일
export default {
  data() {
    return {
      subtitle: '도서명'
    }
  },
  template: `
    <article class="book">
      <div class="book__subtitle">{{ subtitle }}</div>
      <div class="book__title">HTML 강좌</div>
    </article>
  `
}
```

문자열 템플릿\*\*은 가독성이 떨어지고 예쁘지 않습니다.

- `.js` 는 HTML과 JavaScript는 모듈화 할 수있지만 CSS는 누락
- 모든 문제들을 해결하기 위해 Vue.js는 Webpack, Browserify, Vite와 같은 빌드 도구를 활용하여 .vue 확장자를 가진 Single File Component(SFC)를 사용

#### ② Single File Component (SFC)

- SFC는 template, script, style 크게 세 가지로 구성

## 3-2-2. component

### ② Single File Component (SFC)

```
// BookComponent.vue 파일
<template>
  <article class="book">
    <div class="book_subtitle">{{ subtitle }}</div>
    <div class="book_title">HTML 강좌</div>
  </article>
</template>

<script>
export default {
  data() {
    return {
      subtitle: '도서명'
    }
  },
}
</script>

<style scoped>
</style>
```

- 실무에서는 문자열 템플릿 대신 Single File Component를 사용

### ③ template / script / style 설명

- [1] template
  - HML Tag , vue 제공 Directive, Template, 다른 component
- [2] script
  - Java Script가 들어가 동작을 정의
  - 단일 파일 컴포넌트에서는 어떠한 관련된 vue 파일은 반드시 export default가 필요
- [3] style
  - CSS등의 style

## 3-2-3. component

### 2] component 등록 방법

#### [1] 전역 등록 (Global Registration)

- **app.component**를 이용해서 컴포넌트를 등록하면, 컴포넌트는 애플리케이션 **전역 등록**이 되어 모든 컴포넌트 인스턴스의 템플릿 내부에서 사용

```
const app = createApp({ ... })  
app.component('BookComponent', BookComponent)
```

#### [2] 지역 등록 (Local Registration)

- 전역 등록 하게 되면 컴포넌트를 사용하지 않더라도 계속해서 최종 빌드에 해당 컴포넌트가 포함되는 것을 의미
- 이는 사용자가 D/L하는 자바스크립트 파일의 크기를 불필요하게 증가

```
const app = createApp({  
  components: {  
    BookComponent: BookComponent  
  }  
})
```

### 3] component 사용

#### [1] 컴포넌트는 template에서 사용

- <BookComponent></BookComponent> → 첫 단어를 대문자로 시작하는 표기법
- <book-component></book-component> → 모두 소문자로 표현하며, 단어와 단어 사이에는 하이픈(-)를 사용

#### [2] component 네이밍 룰

- 컴포넌트를 사용할 때 PascalCased를 권장
- SFC, string template에서 PascalCase를 사용 가능하며 DOM Template에서는 kebab-case를 사용

## 4-1-1. Create vueJs

### 1] vueJs new App site 개발 ( Project 생성[Vite적용] )

① C:\vueJs\Sources 폴더 생성

② **command** → C:\vueJs\Sources>npm init vue

- 이 명령은 Vue 프로젝트 스캐폴딩 도구인 create-vue를 설치하고 실행  
Vue.js - The Progressive JavaScript Framework

✓ Project name: ... **vue3Basic01**

✓ Add TypeScript? ... **No** / Yes

✓ Add JSX Support? ... **No** / Yes

✓ Add Vue Router for Single Page Application development? ... **No** / Yes

✓ Add Pinia for state management? ... **No** / Yes

✓ Add Vitest for Unit Testing? ... **No** / Yes

✓ Add an End-to-End Testing Solution? » **No**

✓ Add ESLint for code quality? ... No / **Yes**

✓ Add Prettier for code formatting? ... No / **Yes**

Scaffolding project in C:\vueJs\Sources\learn-vue3...

Done. Now run:

cd **vue3Basic01**

npm install

npm run lint

npm run dev

C:\vueJs\Sources>

③ visual studio에서 파일 → 폴더 열기 → C:\vueJs\Sources\ **vue3Basic01** 선택

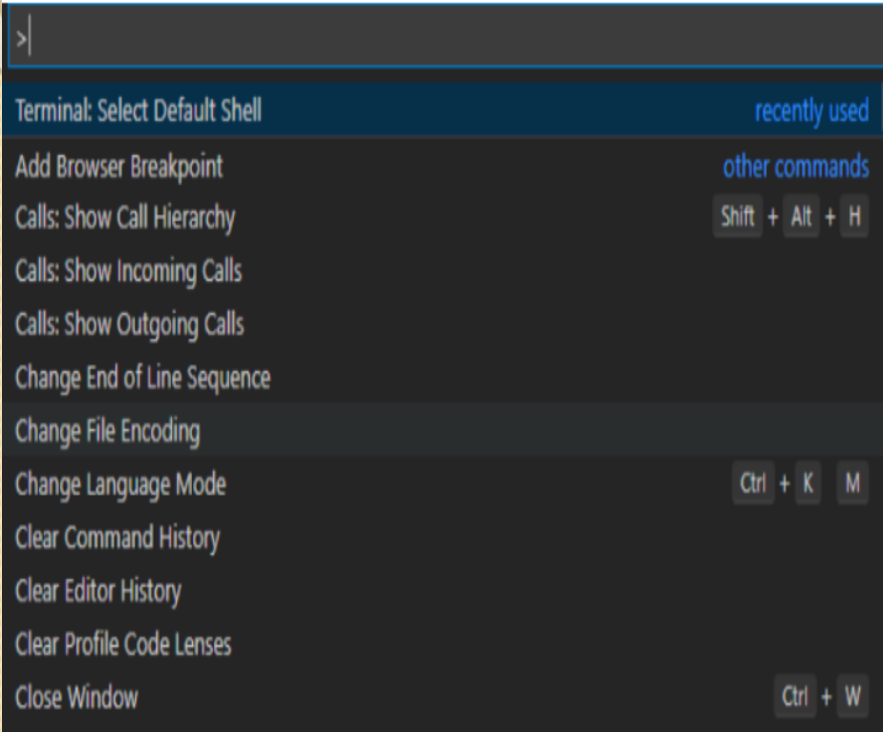
④ visual studio Terminal에서 **의존된 Library**설치하기 위해

PS C:\vueJs\Sources\learn-vue3> **npm install**

## 4-2. Create vueJs

명령어가 실행되지 않으면 해결책 → PS C:\vueJs\Sources\vue3Basic01 > npm install

1. VSCode의 기본 터미널의 종류를 바꿔주면 해결 가능  
우선 ctrl + Shift + P를 눌러 아래와 같은 창을 띄워준다.



2. terminal: select default profile을 클릭

3. Command Prompt 클릭

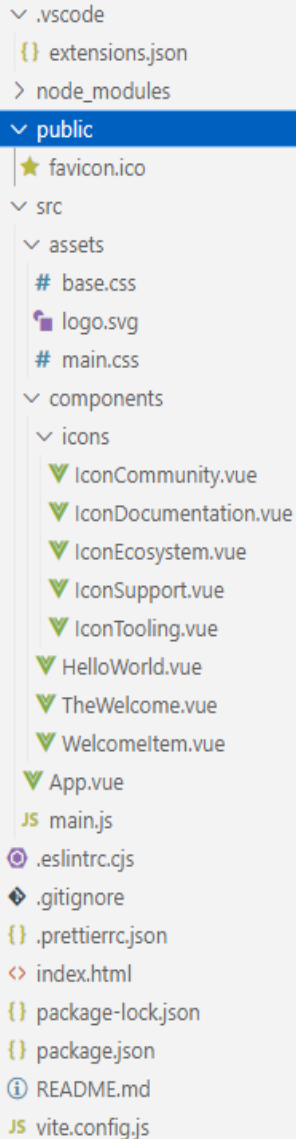
4. 터미널 닫았다가 다시 실행  
(안 되면 visual studio 종료후 다시 실행)



5. visual studio Terminal에서 실행  
C:\vueJs\Sources\vue3Basic01>npm run dev

## 4-3. vueJs Project 구조

### 1. Project 구조



#### [1] vite.config.js

- Vite 명령어로 dev 서버를 실행할 때 **프로젝트 루트의 vite.config.js 파일 확인을 시도**
- alias : 파일 시스템의 경로에 별칭을 만들 때 사용합니다. 미리 설정된 '@' 기호를 통하여 './src' 디렉토리에 절대경로로 쉽게 접근

```
import { fileURLToPath, URL } from 'url'
```

```
import { defineConfig } from 'vite'  
import vue from '@vitejs/plugin-vue'
```

```
// https://vitejs.dev/config/  
export default defineConfig({  
  plugins: [vue()],  
  resolve: {  
    alias: {  
      // @ -> 현재 src 폴더를 가르침  
      '@': fileURLToPath(new URL('./src', import.meta.url))  
    }  
  }  
})
```

#### [2] package.json

- npm 으로 관리하기 위한 프로젝트 정보를 갖고 있는 파일

#### [3] ESLint, Prettiser

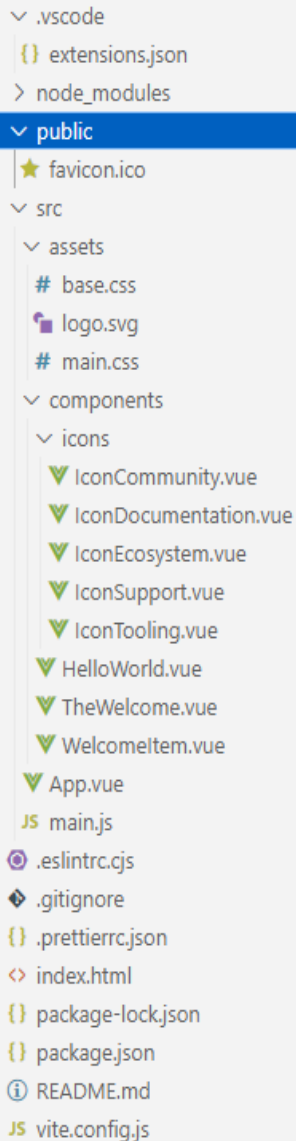
- ①ESLint : <https://eslint.org/>는 코드 검사기로 코드에 에러가 있는지 검사해주 도구.
- ②Prettier :(<https://prettier.io/>)는 코드 포맷터로 코드를 일관성있고 예쁘게 정렬해 주는 도구.

#### [4] 기타

- ①App.vue : Root Component.
- ②index.html : 맨 처음 실행하는 html.

## 4-4. vueJs Project 구조

### 2. VSCode 에서 ESLint 기반으로 Format On Save 설정



```
[1] settings.json 코드
// settings.json
{
  "eslint.validate": [
    "javascript",
    "javascriptreact",
    "typescript",
    "typescriptreact",
    // "html", // 삭제
    "vue",
    "markdown"
  ],
  "editor.codeActionsOnSave": {
    "source.fixAll.eslint": true
  },
  "editor.tabSize": 2,
}
```

#### [2] ESLint 기반요소

- ①eslint.validate: 검사해야 하는 언어를 ESLint 확장에 공지.
- ②editor.codeActionsOnSave: VS Code의 저장 이벤트에 대한 후
- ③source.fixAll.eslint : 저장 중인 파일의 문제를 수정하라는 메시지가 표시

#### [3] Project 실행원리

- ① index.html 에서 main.js 포함.
- ② main.js에서 createApp(App).mount('#app') 수행   ./App.vue mount  
createApp은 Vue Instance를 생성하는 메소드 로 #app를 index.html에 mount함

#### [4] visual studio Terminal에서 실행 lint 적용

C:\WvueJs\WSources\vue3Basic01>npm run lint