

Vue Coding Rule

Front 정의

- Props은 보는 사람을 위해 자세히 정의한다. (type : Object 사용 X)

```
props: {  
  id: { type: Number },  
  badcomment: { type: Number },  
  opinion_type: { type: Boolean },  
  content: { type: String },  
  created_at: { type: String },  
  updated_at: { type: String },  
  user: { type: Number },  
  article: { type: Number },  
  emotion: { type: String },  
  like_users_count: { type: Number },  
  like_users: { type: Array },  
  username: { type: String },  
  board_type: { type: Boolean },  
},
```

1. 컴포넌트 이름

- 컴포넌트 파일 이름은 PascalCase를 이용 한다.
- Abbreviation (축약어)를 사용하지 않는다.
- 예시: MyComponent.vue

2. 베이스 컴포넌트 이름

- 한 페이지의 특정 기능/부분을 이용할 경우 Prefix (접두사)를 통일한다.
- 예시: VButton, VTable, VIcon

3. 강한 연관성을 가진 컴포넌트 이름

- 부모 컴포넌트와 연결되어 있는 자식 컴포넌트들의 이름의 경우 접두사를 통일한다.
- 예시: TodoList, TodoListItem, TodoListItemButton, SearchSideBar, SearchSidebarNavigation

Vuex Convention

1. Module은 기능 단위로 자른다.

2. state은 중첩해서 사용하지 않는다 (should not be nested)

- state은 언더바를 사용해 다음과 같이 지정한다: products_name, this_is_convention

3. Getters

- 참인지 거짓인지 판단할 경우 `is` 를 사용한다. (예시: `isUserLoggedIn`, `isUser`)
- Boolean이 아니면 `get` 으로 시작한다.

4. Actions

- 모듈 밖에서 상태 변화가 일어날 경우, action에서 일어나야 한다.
- 여러개를 불러올 때는: `fetchProducts`
- 한 개를 불러올 때는: `findProduct`
- 생성 시에는: `createProduct`
- 수정 시에는: `updateProduct`
- 삭제 시에는: `deleteProduct`

5. Mutations

- Mutation만 module의 state를 변경할 수 있다.
- 대문자를 사용한다. (예시. `ADD_PRODUCT`, `SET_CURRENT_PRODUCT_ID`)
- `SET`: 비어있는 state에 데이터를 배정한다.
- `ADD`: 이미 배정되어 있는 state에 새로운 데이터를 추가한다.
- `REMOVE`: `ADD`의 반대이다.