TODO

Using Vue3, typescript, router, vuex

**Reactive**

Component data is reactive meaning that if we change it during the lifecycle of the component, Vue will automatically re-render the component to reflect the change

**Options API**

We use Options API in a Vue application to write and define different components. With this API, we can use options such as data, methods, and mounted. To state it simply, Options API is an old way to structure a Vue.JS application. Due to some limitations in this API, Composition API was introduced in Vue 3.

simple to get started with – for a developer coming with no experience at all

**Composition API Post**

Learning #Vue?

Vue 3 introduced the Composition API a new optional syntax for writing "composition functions", functions that can easily be reused between components.

Composition API doesn’t affect how we define our component templates or CSS it just affects how we define our components in JavaScript. The first thing to note is that the composition API doesn’t use component options like data, methods, and computed, instead all the logic and declarations go into the new setup function.

The new setup method is executed before the component is created and once the props are resolved.

To create reactive data with the composition API we can use the ref (reactive reference) method. The function setup returns an object, and all its properties will be available for use in the render context (can be used in the template)

The main benefit of the composition API is that it allows complex multi-feature components to be composed more easily, and it makes the reuse of logic across components easier as well.

To sum it up:

Advantages:

- Multi-feature components

- Logic reuse

Disadvantages:

- Need to follow a certain pattern

It's good to know that we should avoid using this inside setup as it won’t refer to the component instance. setup is called before data properties, computed properties, or methods are resolved so that they won’t be available within setup.

Link to learn Vue like pros: <https://betterprogramming.pub/12-vuejs-best-practices-for-pro-developers-28d1f629018c>

Link to my last post of my Vue project: <https://www.linkedin.com/posts/shani-kupiec-28a064228_vue3-trello-fullstack-activity-6922913961998766081-U8An?utm_source=linkedin_share&utm_medium=member_desktop_web>