You might recall lecture 3 ("Different Ways of Using Vue"): You can use Vue.js to control **parts**of (possibly multiple HTML) pages OR you use it to build so-called "**Single Page Applications**" (**SPA**s).

If you control multiple, independent parts of HTML pages, you will often work with **multiple Vue apps**(i.e. you create multiple apps by calling createApp() more than once).

On the other hand, if you're building a SPA, you typically work with just **one "root app"** (i.e. createApp() is only used once in your entire codebase) and you instead **build up a user interface with multiple components**.

You absolutely are allowed to also use components in cases where you have multiple Vue apps but you typically won't use multiple Vue apps if you build one big connected user interface.

Why?

Because **Vue apps are independent from each other** - they **can't really communicate** with each other. You might find "hacks" to make it work but there's no great "official" way of sharing data between apps, updating something in app A in case something happens in app B etc.

**Components** on the other hand - as you will learn soon - **DO offer certain communication mechanisms** that allow you to exchange data between them. Hence you can build one connected UI if you work with one root app that holds multiple components.