

# State Management

NEXT: [GETTERS](#) ➔

Vuex is a centralized state management Vue dependent library. What Vuex does for you is, rather than coordinate state's data in components, you can coordinate them in a huge state object and pass them down a state portion to whatever component needs a portion of the state.

The term state here, as we mentioned already is just an object. It can get giant depending on the size of the project. Giant or small, it's basically:

```
{}JS
```

Before we start creating and managing state, let's install and configure Vuex:

Table of Contents

1

State

```
npm install --save vuexBASH
```

Just like every other plugin, you need to configure Vue with Vuex:

```
// ./src/store/index
import Vue from 'vue'
import Vuex from 'vuex'

Vue.use(Vuex)JS
```

## #State

Let's create our app's state with the initial data:

```
// ./src/store/index
export default new Vuex.Store({
  strict: true,
  state: {
    // bought items
    cart: [],
    // ajax loader
    showLoader: false,
    // selected product
    product: {},
    // all products
    products: [],
    // all manufacturers
    manufacturers: []
  }
})JS
```

Remember, a state is just an object. This time it contains all the initial data that our app is going to need.

You need to be able to access the store state and methods from a component instance. To do so, let's attach the store to the Vue instance

```
// ./src/main
import Vue from 'vue'
import store from './store'
//...
new Vue({
  el: '#app',
  store,
  // ...
})JS
```

This way you can access `cart` from the component, thus:

```
this.$store.state.cartJS
```

You can also retrieve the value of `cart` using `getter` store methods. Let's see how.

Getting Started		
1	Introduction	Free
2	Tools	Free
3	Our Task	Free
4	Hello World	Free
Vue Basics		
5	Vue CLI	Free
6	Components and the Vue Instance	Free
7	Template Syntax	Free
8	Conditional & List Rendering	Free
9	Styles & Classes	Free
Routing		
10	Single Page Apps	Free
11	Creating Routes	Free
12	Route Outlets & Links	Free
13	Nested & Dynamic Routes	Free
Forms		
14	Two Way Binding (Reactivity)	Free
15	Form Validation	Free
16	Handling Form Submissions	Free
API Backend		
17	Prelude	Free
18	Setup	Free
19	Provisioning a MongoDB Database	Free
20	Enabling CORS	Free
21	Schemas & Models	Free
22	Routes and Controllers	Free
23	Testing with POSTman	Free
Vuex		
24	The Current Problem	Free
25	State Management	Free
26	Getters	Free
27	Mutations	Free
28	Actions	Free
Using Store In Components		
29	State and Actions on Components	Free
30	LAB: Product List	Free
31	LAB: Product Details	Free
32	LAB: Admin Features	Free
33	LAB: Spinner, Cart & Store Subscription	Free