

Getters

NEXT: MUTATIONS ➔

If you needed a portion of `products` , you could do this:

```
computed() {  
  firstProduct() {  
    return this.$store.products[0]  
  }  
}
```

We are doing this right in our component. What if more than one component needs `firstProduct` ? We will just have to write the method in all the component that needs it. We can do better by using a getter in the store and exposing this getter to all the components:

```
// ./src/store/getters  
export const productGetters = {  
  // All products  
  allProducts: (state, getters) => {  
    return state.products  
  },  
  // Get Product by ID  
  productById: (state, getters) => id => {  
    if (getters.allProducts.length > 0) {  
      return getters.allProducts.filter(p => p._id === id)[0]  
    } else {  
      return state.product  
    }  
  },  
}  
  
export const manufacturerGetters = {  
  // All manufacturers  
  allManufacturers: state => state.manufacturers  
}
```

Rather than the trivial `firstProduct` example, we have created getters for our shop. Don't bother about where the actual data is or how it's fetched. We will discuss that in **Actions**.

The name of getters already tells you what they do, and the code is straight to the point. Let's import the methods and use them our store:

```
// ./src/store/getters  
  
// ...  
  
// Import getters  
import { productGetters, manufacturerGetters } from './getters'  
  
export default new Vuex.Store({  
  strict: true,  
  state: {  
    cart: [],  
    showLoader: false,  
    product: {},  
    products: [],  
    manufacturers: []  
  },  
  // GETTERS  
  getters: Object.assign({}, productGetters, manufacturerGetters)  
})
```

We will see how these data are passed down to components in the **Passing State Data to Components** title

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