

一、Vuex映射：

① 数据既可以在main.js中定义，也可以在文件夹store中的index.js文件中定义。（下面的例子在main.js中定义）

```
import Vue from 'vue'
import App from './App'
import router from './router'
import Vuex from "vuex";
Vue.use(Vuex);

Vue.config.productionTip = false

const store = new Vuex.Store({
  state: {
    list: [
      {name: "张", age: 10},
      {name: "高", age: 11},
      {name: "李", age: 12},
      {name: "王", age: 13},
    ],
  },
  mutations: {
    changes(state, key) {
      state.list[key].name = "Hello World"
    }
  },
})
```

定义好之后，同样在下面别忘了引入：

```
/* eslint-disable no-new */
new Vue({
  el: '#app',
  router,
  store,
  components: { App },
  template: '<App/>'
})
```

② 定义好之后，在app.vue中引入map...等映射口：

```

<script>
import {mapState,mapGetters,mapMutations,mapActions} from "vuex";
export default {
  name: 'App',
  computed: {
    ...mapState([
      "list",
    ])
  },
  methods: {
    ...mapMutations([
      "changes"
    ])
  },
}
</script>

```

注意：state（数据）、getters（计算属性）需要映射在computed实例中，而mutations（同步操作放），actions（异步操作方法）等需要放在methods实例中。

③ 使用：

```

<template>
  <div id="app">
    <li v-for="(item,key) in list" :key="key">
      <p>{{item.name}}</p>
      <button @click="changes(key)">点击改变</button>
    </li>
  </div>
</template>

```

遍历：直接就可以遍computed历映射出来的list。

事件：（上面是点击按钮改变对应下标的内容）直接调用methods里面映射出来的方法即可，只需要在 @click="事件(参数)" 加上参数即可。

张

点击改变

高

点击改变

李

点击改变

王

点击改变