

# Vuex

—任科

## Vuex是什么？

**官方解释：** Vuex 是一个专为 Vue.js 应用程序开发的**状态管理模式**。它采用集中式存储管理应用的所有组件的状态，并以相应的规则保证状态以一种可预测的方式发生变化。

**个人理解：** Vuex是用来管理组件之间通信的一个插件。

# 为什么要用VueX?

兄弟组件、跨级嵌套组件之间想要实现通信呢？

试想一下，当做中大型项目时，面对一大堆组件之间的通信，还有一大堆的逻辑代码，会不会很抓狂？那为何不把组件之间共享的数据给“拎”出来，在一定的规则下管理这些数据呢？这就是Vuex的基本思想了。

## 什么情况下应该使用 Vuex?

如果应用够简单，最好不要使用 Vuex。

构建一个中大型单页应用，Vuex可以更好地在组件外部管理状态。

。

# 核心概念

**State** :存放状态（组件间公用数据）

**Getter** :就是从state中派生出状态，比如将state中的某个状态进行过滤然后获取新的状态。

**Mutation** : 用来更改状态。

**Action** :就是mutation的加强版，主要用来进行异步操作。

**Module**: 对于大型项目，当用同一个容器来装这些状态，容器就会变得很大，很难维护，所以我们就可以把容器分成几块，把状态分类来装。

# 开始使用

安装vuex: `npm install vuex --save-dev`

```
import Vue from 'vue';
import Vuex from 'vuex'
Vue.use(Vuex); //在创建Vue实例之前
const myStore = new Vuex.Store({
  state: {
    //存放组件之间共享的数据
  },
  mutations: {
    //显式的更改state里的数据
  },
  getters: {
    //获取数据的方法
  },
  actions: {
    // 异步操作
  }
});
```

```
new Vue({
  el: "#app",
  data: {
  },
  store: myStore
});
```

