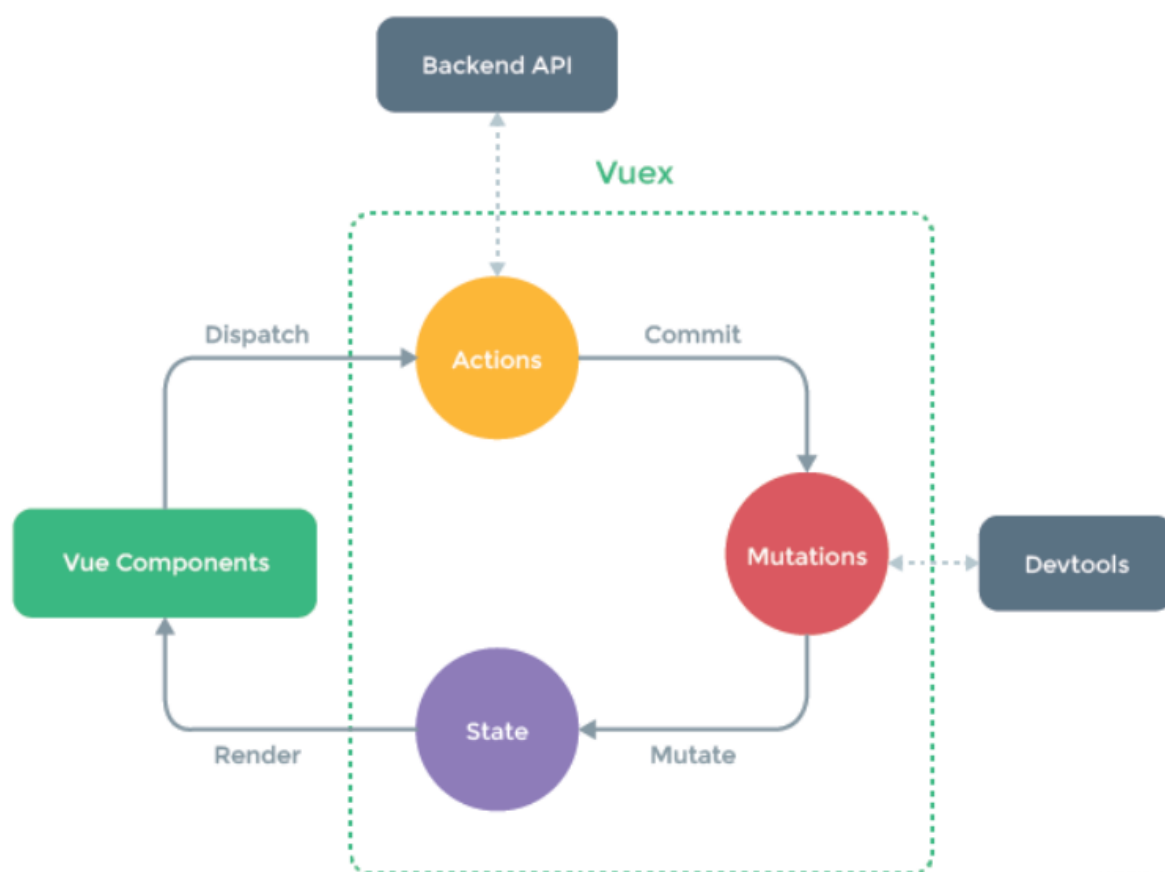


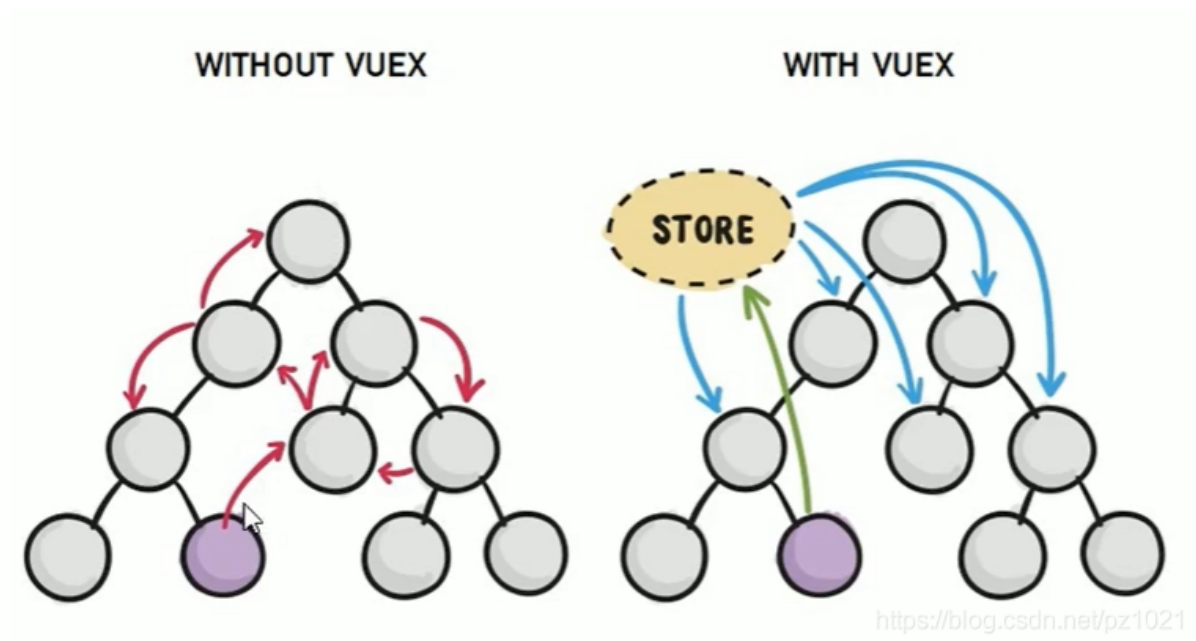
Vuex-共享单车



一、什么是vuex



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。他可以方便实现组件之间的数据共享



二、vuex的五个核心模块

- **state:** state 提供唯一的数据资源,所有的共享的数据都要统一放到store 中的state中进行存储

- 组件中访问state第一种方式:

```
this.$store.state.全局数据名称
```

- 组件中访问state第二种方式:

1. 从vuex中按需求导入mapState函数

```
import {mapState} from 'vuex'
```

通过刚才导入的mapState函数，将当前组件需要的全局数据，映射为当前组件的computed计算属性

2. 将全局数据，映射为当前组件的计算属性

```
computed :{ ...mapState(['count']) }
```

- **mutation:**mutation用于变更store中的数据

- ①只能通过mutation更Store数据，不可以直接操作Store中的数据。
- ②通过这种方式虽然操作起来稍微繁琐一些,但是可以集中监控所有数据的变化。

- `this.$store.commit()` 是触发mutations的第一种方式

```
//store.js中定义mutations
const store = new Vuex.Store({
  state: {
    count: 0
  },
  mutations: {
```

```

    add(state) {
      //更改状态
      state.count++
    }
  }
})
//组件中接收触犯mutations
methods: {
  add() {
    this.$store.commit('add')
  },

  },
//mutations可以传递参数
mutations: {
  add2(state,step) {
    state.count += step
  }
}
//触发带参数的mutation
methods: {
  add() {
    this.$store.commit('add',3)
  },
},

```

- 触发mutations的第二种方式：通过以函数映射的方式

```

1. 从vuex中按需求导入mapState函数
import {mapMutations} from 'vuex'
通过刚才导入的mapState函数，将当前组件需要的全局数据，映射为当前组件的computed计算属性
2. 将指定的mutations函数，映射为当前组件的methods函数
methods: { ...mapMutations(['add']) }

```

- **action:用于处理异步任务,如果通过异步操作变更数据，必须通过Action,而不能使用Mutation,但是在Action中还是要通过触发Mutation的方式间接变更数据。**

- `this.$store.dispatch` 是触发actions的第一 种方式

```

mutations: {
  add(state,step) {
    state.count += step
  }
},
actions: {
  addAsync(context,step) {
    setTimeout(()=> {
      context.commit('add',step)
    },1000)
  }
}
//触发action

```

```
add () {
  this.$store.dispatch('addAsync',5)
}
```

- 触发actions的第二种方式:通过以函数映射的方式

```
1. 从vuex中按需求导入mapState函数
import {mapActions} from 'vuex'
通过刚才导入的mapState函数, 将当前组件需要的全局数据, 映射为当前组件的computed计算属性
2. 将指定的mutations函数, 映射为当前组件的methods函数
methods: { ...mapActions(['addAsync']) }
```

- **getter:Getter是对Store中的数据进行加工处理形成新的数据。他不会修改state中的原始数据起到的是包装数据的作用**

- ①Getter 可以对Store中已有的数据加工处理之后形成新的数据,类似Vue的计算属性。
- ②Store 中数据发生变化, Getter 的数据也会跟着变化。

- `this.$store.getters.名字` 是的调用getters第一种方式

```
//组件中调用
{{ $store.getters.showNum }}
//store.js中定义
getters: {
  showNum (state){
    return `当前最新的数据${state.count}`
  }
}
```

- 通过以函数映射的方式

```
1. 从vuex中按需求导入mapState函数
import {mapGetters} from 'vuex'
通过刚才导入的mapState函数, 将当前组件需要的全局数据, 映射为当前组件的computed计算属性
2. 将指定的mutations函数, 映射为当前组件的methods函数
computed : { ...mapGetters(['showNum']) }
```

- **modules: store的子模块, 为了开发大型项目, 方便状态管理而使用的。一般用于导入公共js模块**

三、vuex实现购物车功能

