

## 18. 组件单向数据流

学习要点：

### 1. 单向数据流

本节课我们来开始学习 Vue3.x 的组件中 Prop 的单向数据流。

#### 一. 单向数据流

1. 父组件的 data 值更新后通过 props 选项交给子组件进行渲染，反之则不行；
2. 这就是单向数据流(单向下行绑定)，不能通过子组件来改变父组件的状态；
3. 这样做的是为了防止父组件发生改变后，数据流变得难以理解；
4. 父组件更新时，子组件所有 props 值也会更新，你不能改变子组件的 props 值；
5. 通过控制台输入 vm.message 赋值，子组件的自动渲染刷新；

```
vm.message = '改变父组件'
```

5. 我们可以通过设置一个父组件的计数器属性，并且通过子组件去更改它；

```
data() {  
  return {  
    message : '父组件 Vue3.x~',  
    counter : 100  
  }  
},  
  
//prop 通信  
props : ['parentMessage', 'parentCounter'],  
  
// 模板  
template : `  
  <div>{{parentCounter}}</div>  
  <button v-on:click="parentCounter++">counter+</button>  
`,
```

PS: 此时控制台会告诉你，Props 是只读，无法修改；

6. 那么，我们如何改变子组件这个值呢？可以考虑只改变子组件的值，父组件不变；

```
// 数据  
data() {  
  return {  
    message : '子组件 Vue3.x~',  
    childCounter : this.parentCounter // this 可以调用 props 内的属性  
  }  
},
```

```
// 模板
template : `
  <div>{{counter}}</div>
  <div>{{childCounter}}</div>
  <button v-on:click="childCounter++">counter+</button>
`,
```

PS: 上面的写法就是通过使用子组件的 **data** 属性，让 **props** 父组件的值先赋值过来；

PS: 然后通过操作这个子组件的 **childCounter** 实现累加，而并未修改父组件的 **Counter**；

PS: 当然，通过计算属性 **computed** 或方法 **methods** 来修改 **childCounter** 也是可以的；