

16. 组件的单向数据流

学习要点：

1. 单向数据流

本节课我们来开始学习 Vue 的组件 props 单向数据流的概念。

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；

//父组件

```
const app = new Vue({
  el: '#app',
  data: {
    count: 0
  },
  components: {
    'html-a': {
      data() {
        return {
          message: '子组件',
        }
      },
      template: `
        <button v-on:click="count++">{{count}}</button>
      `,
      props: ['count'],
    }
  },
})
```

1. 以上内容，会正确改变了 **props** 的值，也渲染到视图中，但控制台报错；
2. 意为：不可以直接修改 **props** 值，可以通过数据或计算属性来解决；

//通过使用 data 数据更改

```
components: {
  'html-a': {
    data() {
      return {
        message: '子组件',
        childCount: this.count
      }
    }
  }
}
```

```
    },  
    template : `  
      <button v-on:click="childCount++">{{childCount}}</button>  
    `,  
    props : ['count'],  
  }  
},
```

//通过使用计算属性和方法数据更改

```
components : {  
  'html-a' : {  
    data() {  
      return {  
        message : '子组件',  
        childCount : this.count  
      }  
    },  
    template : `  
      <button  
v-on:click="clickChildCount">{{changeChildCount}}</button>  
    `,  
    props : ['count'],  
    computed: {  
      changeChildCount() {  
        return this.childCount;  
      },  
    },  
    methods : {  
      clickChildCount() {  
        this.childCount++;  
      }  
    }  
  }  
},  
},
```