

21. 组件双向绑定

学习要点：

1. 双向绑定
2. 组件双向

本节课我们来开始学习 Vue3.x 的组件使用双向绑定的方法。

一. 双向绑定

1. 首先，回顾一下双向绑定的使用方法，使用 `v-model` 即可实现；

```
<input type="text" v-model="message">
```

PS：双向绑定即可实现内容的响应式更新；

2. 当然，它的原理就是通过 `input` 事件来进行内容的替换，如果改成完全形式如下：

```
<input type="text" v-bind:value="message"
      v-on:input="message = $event.target.value">
```

PS：通过 `input` 事件来获取输入的值，再赋值给 `message`，而 `<input>` 再通过 `v-bind` 更新值；

二. 组件双向

1. 首先，在组件调用的地方改写成如下格式：

```
<html-a v-model:parent-message="message"></html-a>
```

PS：将 `v-bind:` 改成 `v-model` 即双向绑定，而模板内部需要绑定对应的父组件属性；

2. 然后，根据第一个要点的写法，我们编写相应的 `input` 代码；

```
// 模板
template : `
  <div><input type="text" v-bind:value="parentMessage"
              v-on:input="$emit('update:parentMessage',
                                $event.target.value)">
  </div>
  <div>{{parentMessage}}</div>
  <div>{{message}}</div>
`
```

PS：这里通过触发 `input` 事件来执行自定义事件，并抛出数据；

3. 也可以通过计算属性中的 `get()` 和 `set()` 来改写组件中的双向绑定；

```
template : `
  <div><input type="text" v-model="value"></div>
`
```

```
// 计算属性
computed : {
  value : {
    get() {
      return this.parentMessage
    },
    set(value) {
      this.$emit('update:parentMessage', value)
    }
  }
}
```