

14. 组件的 props 通信

学习要点：

1. 组件的 props

本节课我们来开始学习 Vue 的组件 props 通信的方法。

一. 组件的 props

1. 组件的父子关系中，当设置一个独立组件时，这个组件就是 `new Vue()` 的子组件；
2. 当我们需要通过子组件显示父组件 `data` 值的时候，需要通过 `props` 属性传值；

//这个组件是 Vue 实例的子组件

```
const childComponent = {
  template : `
    <div>{{text}}</div>
  `,
  props : ['text']
};
```

//根组件

```
const app = new Vue({
  el : '#app',
  data : dataObj,
  components : {
    'html-a' : childComponent
  },
});
```

```
<!--1 : <div>world</div>-->
<!--2 : <div>Hello, Vue!</div>-->
<div id="app">
  <html-a text="world"></html-a>
  <html-a v-bind:text="message"></html-a>
</div>
```

PS: 使用 `v-bind` 才能将父组件的 `data` 值绑定到子组件使用；

3. 在 JS 操作 HTML 属性名有一个大小写的问题：两个单词组合，使用如下写法：

```
const childComponent = {
  template : `
    <div>{{textTitle}}</div>
  `,
  props : ['textTitle']
};
```

```
<div id="app">
  <html-a text-title="world"></html-a>
  <html-a v-bind:text-title="message"></html-a>
</div>
```

4. 如果设置一个全局组件，并未绑定 `new Vue()` 实例，那它也是 `Vue` 实例的子组件；

//全局组件(子组件)

```
Vue.component('html-a', {
  data() {
    return {
      message : '全局组件(子组件)'
    }
  },
  template : `
    <div>{{text}} {{message}}</div>
  `,
  props : ['text'],
});
```

PS: 子组件和父组件都有 `message` 属性，同时调用并不冲突；顺序也可控；