

17. 组件的 props 验证

学习要点：

1. props 验证

本节课我们来开始学习 Vue 的组件 props 数据类型的验证功能。

一. props 验证

1. 之前 props 选项通信采用的是数组方式，其实还有第二种方式：对象；
2. 通过对象模式进行传递数据，可以对数据类型进行验证，支持的类似有：
String.Number.Boolean.Function.Object.Array.Symbol

```
const app = new Vue({
  el : '#app',
  data : {
    num : 100,           //改成非数值，控制台将警告
    str : 'Mr.Lee',
    age : '100',
    arr : [1, 2, 3, 4, 5],
    obj : {
      name : 'Mr.Lee',
      age : 100,
      gender : '男'
    },
    price : 100
  },
  components : {
    'html-a' : {
      data() {
        return {
          message : '子组件'
        }
      },
      template : `
        <div>{{childNum}} {{childAge}} {{childStr}} {{childArr}}
        {{childObj.name}} {{childPrice}}</div>
      `
    },
    props : {
      childNum : Number,           //仅仅支持数值
      childAge : [Number, String], //支持数值和字符串
      childStr : {                 //设置类型和默认值
        type : String,             //[Number, String]
        default : 'no data'        //当没有 v-bind 绑定则显示
      }
    }
  }
})
```

```
    },
    childArr : {
      type : Array,
      default() { //数组和对象的默认值必须用方法
        return [];
      },
      required : true //必须 v-bind 绑定，不能不传
    },
    childObj : {
      type : Object,
      default() {
        return {
          name : 'no name' //给对象的某个属性设置默认值
        }
      }
    },
    childPrice : {
      validator(value) {
        return value > 90; //自定义验证，不成立报错
      }
    }
  }
},
});
```

```
<div id="app">
  <html-a v-bind:child-num="num"
    v-bind:child-str="str"
    v-bind:child-age="age"
    v-bind:child-arr="arr"
    v-bind:child-obj="obj"
    v-bind:child-price="price"
  ></html-a>
</div>
```

PS: `null` 和 `undefined` 会通过任何类型验证