

08. 数组检测和 key

学习要点：

1. 数组检测
2. key 问题

本节课我们来开始学习 Vue 的 v-for 中的数组检测和 key 的问题。

一. 数组检测

1. 为了更好的渲染视图，Vue 提供了如下的变异方法，执行后可渲染视图更新：
push()/pop()/shift()/unshift()/splice()/sort()/reverse();

PS：在控制台输入 `app.array.push({city : '重庆'})`，视图立马渲染更新；

PS：所谓变异方法，就是直接改变源数组的数据，直接渲染更新；

2. 关于响应式，变异方法是可响应的，而其它是无法响应的，比如：
`app.array[1] = {city : '重庆'};` //无法立刻渲染刷新，可在 tool 刷新查看
`app.array.length = 1;` //同上
3. 替换解决方案，除了 splice() 变异方法外，可以使用 Vue.set 方法；
`Vue.set(app.array, 1, {city : '重庆'});` //app.\$set(...) 别名
`app.array.splice(1,1,{city : '重庆'})`;
`app.array.splice(1);` //length=1

二. key 问题

1. 首先，在 v-for 中不添加 key 时，会发生什么问题？

```
<button v-on:click="add">添加</button>                //按钮，添加数据
```

//复选框，提示目前的选项

```
<li v-for="item in array"><input type="checkbox"> {{item.city}}</li>
```

//每个数据增加一个 id

```
array : [  
  {  
    id : 1,  
    city : '北京'  
  },  
  {  
    id : 2,  
    city : '上海'  
  },  
  {  
    id : 3,
```

```
      city : '深圳'
    },
  ]

  methods : {
    add() {
      this.array.unshift({id : 4, city : '重庆'})
    }
  }
}
```

PS：我们给开头的位置添加一个数据，复选框会自动向上选定；其实它位置没变，只是数据变了而已。但我们的需求是跟着变动，否则会错乱。

```
//在 v-for 指派 key，以唯一值 id 作为提示
<li v-for="item in array" v-bind:key="item.id">
```