

## 半桶水技术分享《三》：wx:key 理解

个人感觉官方给出的例子不是很明确，官方解释如下：

wx:key

如果列表中项目的位置会动态改变或者有新的项目添加到列表中,并且希望列表中的项目保持自己的特征和状态（如 `<input/>` 中的输入内容，`<switch/>` 的选中状态），需要使用 `wx:key` 来指定列表中项目的唯一的标识符。

`wx:key` 的值以两种形式提供

1. 字符串,代表在 for 循环的 array 中 item 的某个 property,该 property 的值需要是列表中唯一的字符串或数字,且不能动态改变。
2. 保留关键字 `*this` 代表在 for 循环中的 item 本身,这种表示需要 item 本身是一个唯一的字符串或者数字,如：

当数据改变触发渲染层重新渲染的时候，会校正带有 key 的组件，框架会确保他们被重新排序，而不是重新创建，以确保使组件保持自身的状态，并且提高列表渲染时的效率。

如不提供 `wx:key`，会报一个 warning，如果明确知道该列表是静态，或者不必关注其顺序，可以选择忽略。

示例代码：

```
<switch wx:for="{{objectArray}}" wx:key="unique" style="display: block;"> {{item.id}} </switch>
```

```
<button bindtap="switch"> Switch </button>
```

```
<button bindtap="addToFront"> Add to the front </button>
```

```
<switch wx:for="{{numberArray}}" wx:key="*this" style="display: block;"> {{item}} </switch>
```

```
<button bindtap="addNumberToFront"> Add to the front </button>
```

```
Page({
```

```
  data: {
```

```
    objectArray: [
```

```
      {id: 5, unique: 'unique_5'},
```

```
      {id: 4, unique: 'unique_4'},
```

```
      {id: 3, unique: 'unique_3'},
```

```
      {id: 2, unique: 'unique_2'},
```

```
      {id: 1, unique: 'unique_1'},
```

```
      {id: 0, unique: 'unique_0'},
```

```
],  
  numberArray: [1, 2, 3, 4]  
},  
switch: function(e) {  
  const length = this.data.objectArray.length  
  for (let i = 0; i < length; ++i) {  
    const x = Math.floor(Math.random() * length)  
    const y = Math.floor(Math.random() * length)  
    const temp = this.data.objectArray[x]  
    this.data.objectArray[x] = this.data.objectArray[y]  
    this.data.objectArray[y] = temp  
  }  
  this.setData({  
    objectArray: this.data.objectArray  
  })  
},
```

```
addToFront: function(e) {  
    const length = this.data.objectArray.length  
    this.data.objectArray = [{id: length, unique: 'unique_' +  
length}].concat(this.data.objectArray)  
    this.setData({  
        objectArray: this.data.objectArray  
    })  
},  
addNumberToFront: function(e){  
    this.data.numberArray = [ this.data.numberArray.length + 1 ].concat(this.data.numberArray)  
    this.setData({  
        numberArray: this.data.numberArray  
    })  
}  
})
```

这里写下个人的理解 ,有什么不对的地方希望大家指正 :以<switch></switch>为例 ,如果没有 wx:key ,

选中其中的某个按钮的时候 ,改变其顺序 或添加选项的时 ,选中的按钮时不回跟随 上个按钮改变顺序的 ,  
会一直在固定位子 ,如果如果有 wx:key 则相反 ,适用于列表或其他标签可以改变顺序或添加项目的情况