

## 02. 迷你计算器. 逻辑处理

学习要点：

### 1. 逻辑处理

本节课我们来开始了解下迷你计算器中的逻辑判断处理。

#### 一. 逻辑处理

1. 这节课，我们处理几个问题：包括拒绝开始 0，重复小数点，父子通信等问题。
2. 先解决第一个问题，结果值在父组件上：resultValue : 0；让子组件读取到；

//计算器结果框组件

```
const resultBox = {  
  //父子通信  
  props : ['getValue'],  
  //计算属性  
  computed : {  
    value() {  
      return this.getValue  
    }  
  },  
  template : `  
    <input id="input-box" type="text" readonly v-model="value">  
  `,  
}
```

PS：通过 props 中的 getValue 去读取父组件的 resultValue；

```
<result-box :get-value="resultValue"></result-box>
```

3. 输入 0123456789 时，0 这个在开始是不需要反复输入累计的，需要过滤掉；

//方法

```
methods : {  
  //输入  
  input(param) {  
    //拒绝反复输出零  
    if (parseInt(this.resultValue) === 0  
        && !/^0\./.test(this.resultValue)  
    ) {  
      this.resultValue = param  
    } else {  
      this.resultValue += param  
    }  
  },  
}
```

4. 对于加减乘除等算术公式，另起一个方法实现；

```
//加减乘除符号
symbol(param) {
    this.resultValue += ' ' + param + ' '
},
```

5. 每个数值只能有一个小数点；

```
//处理"."问题
point() {
    //转换字符串，方便字符操作
    const strValue = this.toStr()

    //得到最后一个数字
    const lastNumber = strValue.substring(strValue.lastIndexOf(' '))

    //判断如果有.就不累加
    if (lastNumber.indexOf('.') !== -1) {
        return
    } else {
        this.resultValue += '.'
    }
},

//转换字符串方法
toStr() {
    return this.resultValue.toString()
}
```