

蓝狐锅锅：微信小程序实用教程系列《一》数组操作

前言

相信在小程序推出公测之后，很多小伙伴都已经抓紧申请注册小程序了。在开发阶段中也碰到了很多的问题，例如 `wx.request` 数据请求不成功，在数组操作时，不知道如何往数组里 `push` 数据，`input` 如何监听用户输入的状态，`css` 的 `background-image` 无法获取本地资源等等，本博客会出一个专题，给碰到这些问题的小伙伴解决思路。

数组操作

今天我们主要讲的是，数组操作。

相信对于用过 `vuejs`、`reactjs` 等 `mvvm` 框架的童鞋，微信小程序的数组操作就显得很简单啦，原理是一样的。

返回

数组操作

1、应季鲜果

删除 修改

2、精致糕点

删除 修改

3、全球美食烘培原料

删除 修改

4、无辣不欢生猛海鲜

删除 修改

向前插入数组

向后插入数组

清空数组

返回

数组操作

1、向前增加数据--1479046625352

删除 修改

2、修改了内容1479046629602

删除 修改

3、精致糕点

删除 修改

4、修改了内容1479046623111

删除 修改

5、无辣不欢生猛海鲜

删除 修改

6、向后增加数据--1479046624402

删除 修改

向前插入数组

向后插入数组

清空数组

"

这是简单做的一个 demo，已经上传到 github，到时大伙可以直接下载。我们主要讲在数组中常用的一些操作方法，对数组向前向后插入新的数据、修改数组、删除数组、清空数组，其他的一些操作方式，我会给大伙学习思路。

demo 数组操作的例子路径：

/pages/array/array.wxml

向前向后插入新的数据

```
Page({  
  
  data: {  
  
    list:[{  
  
      id:1,  
  
      name:'应季鲜果',  
  
      count:1  
  
    },{  
  
      id:2,  
  
      name:'精致糕点',  
  
      count:6  
  
    },{
```

```
      id:3,

      name:'全球美食烘培原料',

      count:12

    },{

      id:4,

      name:'无辣不欢生猛海鲜',

      count:5

    }]

  }

})
```

我们初始化一些数据，我们需要对 list 的添加新的数据里，需要用到 JavaScript `concat()` 的方法，`concat()` 方法用于连接两个或多个数组，该方法不会改变现有的数组。 其实我们的所说的向前向后插入数据，其实也就是把两个数组合并起来，组成一个新的数组，然后再使用 `this.setData()` 即可渲染到页面上。 我们看一下在微信小程序里的代码。

向前插入数据 demo

```
//向前增加数据

add_before:function (){
```

//要增加的数组

```
var newarray = [{
```

```
  id:6,
```

```
  name:'向前增加数据--'+new Date().getTime() ,
```

```
  count:89
```

```
  }];
```

//使用 concat()来把两个数组合并起来

```
this.data.list = newarray.concat(this.data.list);
```

//将合并之后的数据，发送到视图层，即渲染页面

//大伙请记录，修改了数据后，一定要再次执行`this.setData()`，页面才会渲染数据的。

```
this.setData({
```

```
'list': this.data.list

});

}
```

向后插入数据 demo

```
//向后增加数据

add_after:function (){

    //要增加的数组

    var newarray = [{

        id:5,

        name:'向后增加数据--'+new Date().getTime() ,

        count:89

    }

    ];
```


细心的小伙伴应该会发现，这两段在用 `concat()` 合并数据时，`concat` 括号里的数据是不一样的。向前向后插入数据的区别就在这里了。

```
//假设这一段是我们要新增的数组

var newarray = [{

    id:5,

    name:'向后增加数据--'+new Date().getTime() ,

    count:89

}];

//向前--用 newarray 与 this.data.list 合并
```

```
var newarray = [{
    id:5,
    name:'向后增加数据--'+new Date().getTime() ,
    count:89
}];

//向前--用 newarray 与 this.data.list 合并
```

```
//向前--用 newarray 与 this.data.list 合并
```

```
this.data.list = newarray.concat(this.data.list);
```

```
//向后--用 this.data.list 与 newarray 合并
```

```
this.data.list = this.data.list.concat(newarray);
```

修改数组

对展示的数据进行修改，在开发过程是常见得不在常见的事情啦。

```
//修改数组
```

```
edit:function (e){
```

```
//这个参数 “e” 的具体作用，请参考微信小程序官方提供的说明，地址为 https://mp.weixin.qq.com/debug/wxadoc/dev/framework/view/wxml/event.html?t=20161107
```

```
var dataset = e.target.dataset;
```

```
var Index = dataset.index; //在通过在 wxml 页面里使用 data-index="{{index}}"传递过来的，是为识别正在编辑修改哪个数组。
```

```
//我们要修改的数组
```

```
this.data.list[Index].name = '修改了内容'+new Date().getTime();
```

```
//将合并之后的数据，发送到视图层，即渲染页面
```

//大伙请记录，修改了数据后，一定要再次执行`this.setData()`，页面才会渲染数据的。

```
this.setData({  
  list:this.data.list  
});
```

}

删除某条数据

有增有改也必有删。删除需要用到 JavaScript splice() 方式，splice() 方法向/从数组中添加/删除项目，然后返回被删除的项目。

//删除

```
remove: function (e){
```

```
var dataset = e.target.dataset;
```

```
var Index = dataset.index;
```

```
//通过`index`识别要删除第几条数据，第二个数据为要删除的项目数量，通常为 1
```

```
this.data.list.splice(Index,1);
```

```
//渲染数据
```

```
this.setData({
```

```
  list:this.data.list
```

```
});
```

```
}
```

清空数据

增修删之后，还得再来一个清空数据呀。

```
//清空
```

```
clear:function (){
```

//其实就是让数组变成一个空数组即可

```
this.setData({
```

```
  list: {}
```

```
});
```

```
}
```

总结

今天我们主要讲了增修删清空，其实对数组的操作还有很多方式的，可以看以下截图。

Array 对象方法

方法	描述
<u>concat()</u>	连接两个或更多的数组，并返回结果。
<u>join()</u>	把数组的所有元素放入一个字符串。元素通过指定的分隔符进行分隔。
<u>pop()</u>	删除并返回数组的最后一个元素
<u>push()</u>	向数组的末尾添加一个或更多元素，并返回新的长度。
<u>reverse()</u>	颠倒数组中元素的顺序。
<u>shift()</u>	删除并返回数组的第一个元素
<u>slice()</u>	从某个已有的数组返回选定的元素
<u>sort()</u>	对数组的元素进行排序
<u>splice()</u>	删除元素，并向数组添加新元素。
<u>toSource()</u>	返回该对象的源代码。
<u>toString()</u>	把数组转换为字符串，并返回结果。
<u>toLocaleString()</u>	把数组转换为本地数组，并返回结果。
<u>unshift()</u>	向数组的开头添加一个或更多元素，并返回新的长度。
<u>valueOf()</u>	返回数组对象的原始值

”

具体的每一个的操作方法，可以去国内的 w3s 多学习下。

http://www.w3school.com.cn/jsref/jsref_splice.asp

demo 地址：

https://github.com/bluefox1688/wxapp_study