

# 新手入手教程微信小程序上手开发和使用总结

## 一、模板

WXML 提供模板组件给我们使用，可以在模板定义公用的代码片段，然后在需要引用的地方进行调用。

### 定义模板

定义模板使用 **name** 属性作为模板的名字，然后在 **template** 标签中定义代码片段：

```
1<template name="mytemplate1">
2  <view>
3    <text>firstName:{{firstName}}</text>
4    <text>lastName: {{lastName}}</text>
5  </view>
6</template>
7<template name="mytemplate2">
8  <view wx:for="{{persons}}">
9    <text>firstName:{{item.firstName}}</text>
10   <text>lastName: {{item.lastName}}</text>
11  </view>
12</template>
```

### 使用模板

使用模板我们用 **is** 属性引用定义好的模板，然后把模板所需要的值通过 **data** 属性传给模板。比如需要遍历 **persons** 数组，我们可以将整个 **persons** 作为对象传给模板，也可以遍历 **persons** 后将每个对象传给模板，具体取决于所应用的场景。

```
1<view wx:for="{{persons}}">
2  <template is="mytemplate1" data="{{...item}}"></template>
3</view>
4<template is="mytemplate2" data="{{persons}}"></template>
```

需要的数据结构如下

```
1Page({
2    data:{
3        persons:[
4            {firstName: 'Hulk', lastName: 'Hu'},
5            {firstName: 'Shang', lastName: 'You'},
6            {firstName: 'Gideon', lastName: 'Lin'}
7        ]
8    }
9})
```

注意：

**is** 属性也可以使用 **Mustache** 语法进行动态渲染，决定使用哪个模板

**data** 如果传的是对象类型的数据，需要用“...”进行“解构”，在模板中可以直接调用对象的属性；如果是其他数据类型则不需要用“...”

## 二、事件

什么是事件呢，简单来说，事件就是逻辑层到逻辑层的通讯方式。就是在页面上通过触发某个操作（就是我们说的事件），在逻辑层进行一系列的操作，最终来改变数据。

比如在一个输入框中用户输入了一段文字，但是 **data** 中的数据并没有随之改变，因此我们需要在输入框上绑定对应的输入事件来更改数据。

## 事件分类

事件也有分类，可以分为冒泡事件和非冒泡事件。“冒泡”这个词很形象的表现了事件向上传递的过程，这

两种事件的区别也在于是否会向父节点进行传递。

一些常用的冒泡事件，除以下的事件外都是非冒泡事件：

名称	触发
touchstart	手指开始触摸
touchmove	手指触摸后移动
touchend	手指触摸动作结束
touchcancel	触摸被打断，比如来点，弹框等
tap	触摸后离开，有点像点击 click
longtap	长按，超过 350ms 才离开

## 事件对象

当事件函数被调用时，从逻辑层有一个默认的事件对象传到函数中，不同的事件所包含的事件对象的属性有所区别，一些常用的事件对象的属性如下：

属性	类型	说明
type	String	事件类型
timeStamp	Int	从页面加载到事件触发的时间戳
target	Object	触发事件的组件的一些属性值集合
currentTarget	Object	当前组件的属性值集合
touches	Array	触摸点信息的数组
detail	Object	额外的信息

## target 和 currentTarget 区别

当不存在嵌套时，**target** 和 **currentTarget** 没有区别。但是当嵌套触发事件是，**current** 和 **currentTarget** 的区别就体现出来了。

```
1<view class="A" bindtap="handle1">
2  outer
3    <view class="B" bindtap="handle2">
4      inner
5    </view>
6 </view>
```

点击组件 B，当触发 **handle2** 事件时，收到 **target** 和 **currentTarget** 对象是一样，都指向组件 B；而当点击组件 B 触发 **handle1** 事件时，**target** 对象指向了组件 B，**currentTarget** 对象则组件 A。总结一下：

**target** 对象指向了所触发事件的对象

**currentTarget** 对象指向了绑定事件所在的对象

## 向 detail 中添加内容

在组件中定义数据，当触发事件时，这些数据通过事件对象传给逻辑层。书写规则：以“**data-**”开头，多个字符用“-”连接，不能含有大写，可以绑定多个 **data** 值。例如 **data-element-name**，最终会在 **event.currentTarget.dataset** 中转为 **elementName** 属性，属性的值就是定义的数据。

## 三、异步数据

小程序还提供发送异步的方法 **request(object)**，发起的是 **https** 请求。一个小程序，同时只有有 5 个网络请求链接。**object** 的参数如下：

参数名	类型	说明
url	String	服务器接口地址
data	Object	请求的参数
header	Object	设置请求头 header，header 不能设置 Referer
method	String	请求方式，默认 GET
success	Function	请求成功的回调方法
fail	Function	请求失败的回调方法
complete	Function	请求完成的回调方法(请求成功、失败都会调用)

跟 jQuery 不同的是，小程序请求的数据不是直接在 `success` 方法的 `res` 中(`res` 是一个对象，还包括请求成功的状态码等)，而是在 `res.data` 中。示例代码如下：

```
1wx.request({
2  url: '/url',
3  data: {
4    x: '',
5    y: ''
6  },
7  header: {
8    'Content-Type': 'application/json'
9  },
10 success: function(res) {
11   console.log(res.data) //接收到的数据
12 }
13})
```