

微信小程序入门教程之三：脚本编程

作者：阮一峰

日期：2020年10月29日

这个系列教程的前两篇，介绍了小程序的[项目结构](#)和[页面样式](#)。

今天，接着往下讲，教大家为小程序加入 JavaScript 脚本，做出动态效果，以及如何跟用户互动。学会了脚本，就能做出复杂的页面了。

本篇的难度要大于前两篇，如果觉得不好理解，可以先跟着例子，动手做一遍，然后再读文字说明，可能就容易理解了。

所有示例的完整代码，都可以从 GitHub 的[代码仓库](#)下载。



一、数据绑定

前面的所有示例，小程序的页面都是写死的，也就是页面内容不会变。但是，页面数据其实可以通过脚本传入，通过脚本改变页面，实现动态效果。

小程序提供了一种特别的方法，让页面可以更方便地使用脚本数据，叫做"数据绑定"（data binding）。

所谓"数据绑定"，指的是脚本里面的某些数据，会自动成为页面可以读取的全局变量，两者会同步变动。也就是说，脚本里面修改这个变量的值，页面会随之变化；反过来，页面上修改了这段内容，对应的脚本变量也会随之变化。这也叫做 MVVM 模式。

下面看一个例子。打开 `home.js` 文件，改成下面这样。

```
Page({
  data: {
    name: '张三'
  }
});
```

上面代码中，`Page()` 方法的配置对象有一个 `data` 属性。这个属性的值也是一个对象，有一个 `name` 属性。数据绑定机制规定，`data` 对象的所有属性，自动成为当前页面可以读取的全局变量。也就是说，`home` 页面可以自动读取 `name` 变量。

接着，修改 `home.wxml` 文件，加入 `name` 变量。

```
<view>
  <text class="title">hello {{name}}</text>
</view>
```

上面代码中，`name` 变量写在 `{{...}}` 里面。这是小程序特有的语法，两重大括号表示，内部不是文本，而是 JavaScript 代码，它的执行结果会写入页面。因此，`{{name}}` 表示读取全局变量 `name` 的值，将这个值写入网页。

注意，变量名区分大小写，`name` 和 `Name` 是两个不同的变量名。

开发者工具导入项目代码，页面渲染结果如下。



可以看到，`name` 变量已经自动替换成了变量值"张三"。

这个示例的完整代码，可以查看[代码仓库](#)。

页面和脚本对于变量 `name` 是数据绑定关系，无论哪一方改变了 `name` 的值，另一方也会自动跟着改变。后面讲解到事件时，会有双方联动的例子。

二、全局数据

数据绑定只对当前页面有效，如果某些数据要在多个页面共享，就需要写到全局配置对象里面。

打开 `app.js`，改写如下。

```
App({
  globalData: {
    now: (new Date()).toLocaleString()
  }
});
```

上面代码中，`App()` 方法的参数配置对象有一个 `globalData` 属性，这个属性就是我们要在多个页面之间分享的值。事实上，配置对象的任何一个属性都可以共享，这里起名为 `globalData` 只是为了便于识别。

然后，打开 `home.js`，改成下面的内容，在页面脚本里面获取全局对象。

```
const app = getApp();

Page({
  data: {
    now: app.globalData.now
  }
});
```

上面代码中，`getApp()` 函数是小程序原生提供的函数方法，用于从页面获取 `App` 实例对象。拿到实例对象以后，就能从它上面拿到全局配置对象的 `globalData` 属性，从而把 `app.globalData.now` 赋值给页面脚本的 `now` 属性，进而通过数据绑定机制，变成页面的全局变量 `now`。

最后，修改一下页面代码 `home.wxml`。

```
<view>
  <text class="title">现在是 {{now}}</text>
</view>
```

开发者工具导入项目代码，页面渲染结果如下。



可以看到，页面读到了全局配置对象 `app.js` 里面的数据。

这个示例的完整代码，可以查看[代码仓库](#)。

三、事件

事件是小程序跟用户互动的主要手段。小程序通过接收各种用户事件，执行回调函数，做出反应。

小程序的[常见事件](#)有下面这些。

- **tap**: 触摸后马上离开。
- **longpress**: 触摸后，超过 `350ms` 再离开。如果指定了该事件的回调函数并触发了该事件，**tap**事件将不被触发。

- **touchstart**: 触摸开始。
- **touchmove**: 触摸后移动。
- **touchcancel**: 触摸动作被打断，如来电提醒，弹窗等。
- **touchend**: 触摸结束。

上面这些事件，在传播上分成两个阶段：先是捕获阶段（由上层元素向下层元素传播），然后是冒泡阶段（由下层元素向上层元素传播）。所以，同一个事件在同一个元素上面其实会触发两次：捕获阶段一次，冒泡阶段一次。详细的介绍，请参考我写的[事件模型解释](#)。

小程序允许页面元素，通过属性指定各种事件的回调函数，并且还能够指定是哪个阶段触发回调函数。具体方法是为事件属性名加上不同的前缀。小程序提供四种前缀。

- **capture-bind**: 捕获阶段触发。
- **capture-catch**: 捕获阶段触发，并中断事件，不再向下传播，即中断捕获阶段，并取消随后的冒泡阶段。
- **bind**: 冒泡阶段触发。
- **catch**: 冒泡阶段触发，并取消事件进一步向上冒泡。

下面通过一个例子，来看如何为事件指定回调函数。打开 **home.wxml** 文件，改成下面的代码。

```
<view>
  <text class="title">hello {{name}}</text>
  <button bind:tap="buttonHandler">点击</button>
</view>
```

上面代码中，我们为页面加上了一个按钮，并为这个按钮指定了触摸事件（**tap**）的回调函数 **buttonHandler**，**bind**: 前缀表示这个回调函数会在冒泡阶段触发（前缀里面的冒号可以省略，即写成 **bindtap** 也可以）。

回调函数必须在页面脚本中定义。打开 **home.js** 文件，改成下面的代码。

```
Page({
  data: {
```

```
    name: '张三'
  },
  buttonHandler(event) {
    this.setData({
      name: '李四'
    });
  }
});
```

上面代码中， `Page()` 方法的参数配置对象里面，定义了 `buttonHandler()`，这就是 `<button>` 元素的回调函数。它有几个地方需要注意。

(1) 事件回调函数的参数是事件对象 `event`，可以从它上面获取[事件信息](#)，比如事件类型、发生时间、发生节点、当前节点等等。

(2) 事件回调函数内部的 `this`，指向页面实例。

(3) 页面实例的 `this.setData()` 方法，可以更改配置对象的 `data` 属性，进而通过数据绑定机制，导致页面上的全局变量发生变化。

开发者工具导入项目代码，点击按钮后，页面渲染结果如下。



可以看到，点击按钮以后，页面的文字从"hello 张三"变成了"hello 李四"。

这个示例的完整代码，可以查看[代码仓库](#)。

这里要强调一下，修改页面配置对象的 `data` 属性时，不要直接修改

`this.data`，这不仅无法触发事件绑定机制去变更页面，还会造成数据不一致，所以一定要通过 `this.setData()` 去修改。`this.setData()` 是一个很重要的方法，有很多细节，详细介绍可以读一下[官方文档](#)。

四、动态提示 Toast

小程序提供了很多组件和方法，用来增强互动效果。比如，每次操作后，都显示一个动态提示，告诉用户操作的结果，这种效果叫做 **Toast**。

打开 `home.js` 文件，为 `this.setData()` 加上第二个参数。


```
Page({
  data: {
    name: '张三'
  },
  buttonHandler(event) {
    this.setData({
      name: '李四'
    }, function () {
      wx.showToast({
        title: '操作完成',
        duration: 700
      });
    }),
  },
});
```

上面代码中，`this.setData()` 方法加入了第二个参数，这是一个函数，它会在页面变更完毕后（即 `this.setData()` 执行完）自动调用。

这个参数函数内部，调用了 `wx.showToast()` 方法，`wx` 是小程序提供的原生对象，所有客户端 API 都定义在这个对象上面，`wx.showToast()` 会展示微信内置的动态提示框，它的参数对象的 `title` 属性指定提示内容，`duration` 属性指定提示框的展示时间，单位为毫秒。

开发者工具导入项目代码，点击按钮后，页面渲染结果如下。



过了700毫秒，提示框就会自动消失。

这个示例的完整代码，可以查看[代码仓库](#)。

五、对话框 Modal

下面，我们再用小程序提供的 `wx.showModal()` 方法，制作一个对话框，即用户可以选择"确定"或"取消"。

打开 `home.js` 文件，修改如下。

```
Page({
  data: {
    name: '张三'
  },
  buttonHandler(event) {
    const that = this;
```

```
wx.showModal({
  title: '操作确认',
  content: '你确认要修改吗?',
  success (res) {
    if (res.confirm) {
      that.setData({
        name: '李四'
      }, function () {
        wx.showToast({
          title: '操作完成',
          duration: 700
        });
      });
    } else if (res.cancel) {
      console.log('用户点击取消');
    }
  }
});
```

上面代码中，用户点击按钮以后，回调函数 `buttonHandler()` 里面会调用 `wx.showModal()` 方法，显示一个对话框，效果如下。



`wx.showModal()` 方法的参数是一个配置对象。 `title` 属性表示对话框的标题（本例为"操作确认"）， `content` 属性表示对话框的内容（本例为"你确认要修改吗？"）， `success` 属性指定对话框成功显示后的回调函数， `fail` 属性指定显示失败时的回调函数。

`success` 回调函数里面，需要判断一下用户到底点击的是哪一个按钮。如果参数对象的 `confirm` 属性为 `true`，点击的就是"确定"按钮， `cancel` 属性为 `true`，点击的就是"取消"按钮。

这个例子中，用户点击"取消"按钮后，对话框会消失，控制台会输出一行提示信息。点击"确定"按钮后，对话框也会消失，并且还会去调用 `that.setData()` 那些逻辑。

注意，上面代码写的是 `that.setData()`，而不是 `this.setData()`。这是因为 `setData()` 方法定义在页面实例上面，但是由于 `success()` 回调函数

不是直接定义在 `Page()` 的配置对象下面，`this` 不会指向页面实例，导致 `this.setData()` 会报错。解决方法就是在 `buttonHandler()` 的开头，将 `this` 赋值给变量 `that`，然后在 `success()` 回调函数里面使用 `that.setData()` 去调用。关于 `this` 更详细的解释，可以参考[这篇教程](#)。

这个示例的完整代码，可以查看[代码仓库](#)。

今天的教程就到这里，对于初学者来说，已经讲了很多东西，可能需要慢慢消化。如果对网页开发和 `JavaScript` 语言不熟悉，你也许会觉得不容易完全理解，不用担心，初学者只需要知道加入脚本的方法，以及脚本可以达到的效果就可以了，后面做到实际的项目，慢慢就会加深理解。

有了脚本以后，就可以通过小程序 API，去调用微信的各种内置能力。[下一篇教程](#)将重点讲解如何使用小程序 API。

（完）

文档信息

- 版权声明：自由转载-非商用-非衍生-保持署名（[创意共享3.0许可证](#)）
- 发表日期：2020年10月29日

算法训练营体验课

7 天掌握大厂算法面试 必考知识点

知识点精讲

- 时间复杂度
- 数组、链表、跳表
- 树、二叉树、二叉搜索树
- 递归

大厂面试题解析

- 美团
- 滴滴出行
- 京东商城

社群活动多多

- 2020 最新面经分享
- 每周一次讲师直播
- 班主任督学
- 助教作业精讲 + 答疑

大厂面试题解析

- 美团
- 滴滴出行
- 树
- 链表
- 跳表
- 递归
- 数组

相关文章

- **2020.12.13:** [《SSH 入门教程》发布了](#)

SSH 是登录 Linux 服务器的必备工具，只要你在做互联网开发，多多少少都会用到它。

- **2020.11.02:** [微信小程序入门教程之四：API 使用](#)

今天是这个系列教程的最后一篇。

- **2020.10.27:** [微信小程序入门教程之二：页面样式](#)

这个系列的上一篇教程，教大家写了一个最简单的 Hello world 微信小程序。

- **2020.10.26:** [微信小程序入门教程之一：初次上手](#)

微信是中国使用量最大的手机 App 之一，日活跃用户超过3亿，月活跃用户超过11亿（2019年底统计），市场极大。



Weibo | Twitter | GitHub

Email: yifeng.ruan@gmail.com