

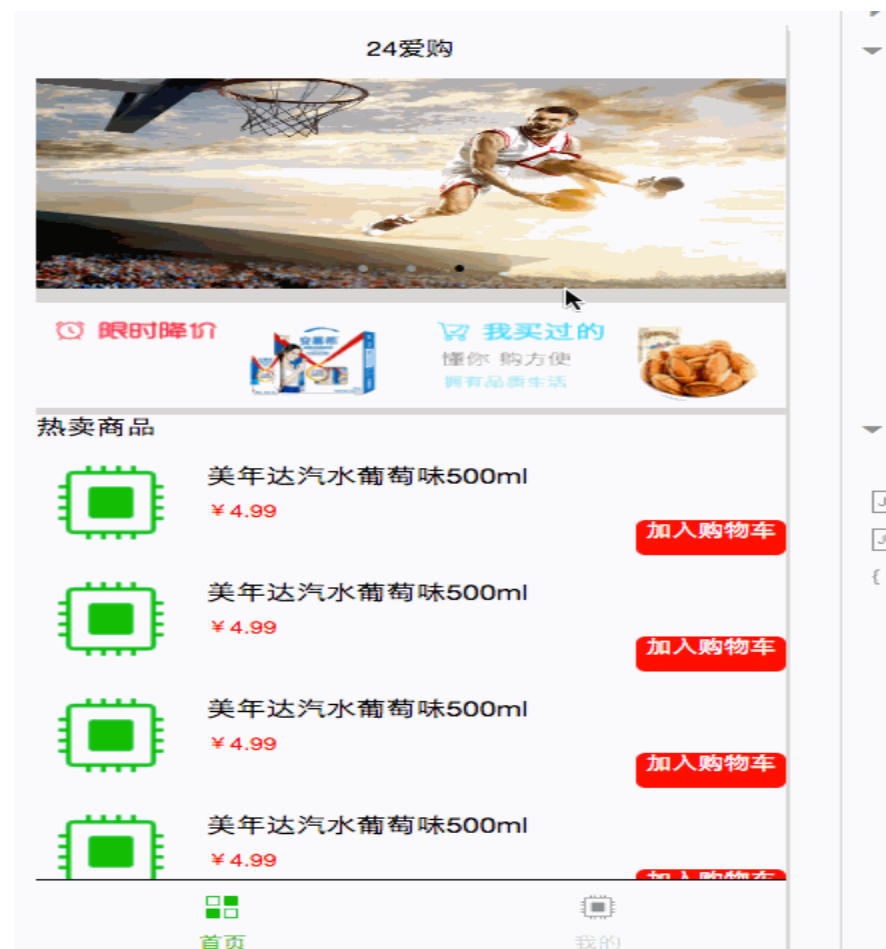
奇董的微信小程序--iOS 弱鸡摸黑入门

写给咱 iOS 开发者的小程序入门，肯定要按照 iOS 开发的流程走了啊

小程序文档 <https://mp.weixin.qq.com/debug/wxadoc/dev/index.html>

自己瞎写的 demo : <https://github.com/dongqihouse/wxappDemo>

文件下载 : [wxappDemo-master.zip](#)



首先，我们 iOS 程序程序的入口是 main 函数制定 appDelegate 进入程序循环=====在小程序中我们可以创建一个 app.js 文件，在其中调用 app () 的方法

app.js

//app.js

```
App({  
  onLaunch: function () {  
  
  },  
  
  getUserInfo:function(cb){  
  
  },  
  
  globalData:{  
  
    userInfo:null  
  
  }  
})
```

看到 onLaunch 我们很容易联想到 appDelegate 中的- (BOOL)application:(UIApplication

*)application didFinishLaunchingWithOptions:(nullable NSDictionary *)launchOptions 都是程序

加载的嘛

上面是 js 的语法，我稍微解释一下 app()是程序的入口传入的 app 的一个对象

在 js 中

```
{  
  onLaunch: function () {  
  
  },  
  
  getUserInfo:function(cb){  
  
  },  
  
  globalData:{  
  
    userInfo:null  
  
  }  
}
```

这就是一个对象， onLaunch 不用说，下面的 2 个方法都是我们为这个对象添加的方法。

在其他地方我们可以获取这个全局对象

```
var app = getApp()
```

这个就和[UIApplication sharedApplication]差不多。

不过这里区别的是 iOS 是指定了代理，而小程序这边是自己创建一个 app 的对象。

app.json

这个 json 文件就相当于配置文件，针对一些全局配置

```
{  
  "pages": [  
    "pages/index/index",  
    "pages/mine/mine"  
  ],  
  "window": {  
    "backgroundTextStyle": "light",  
    "navigationBarBackgroundColor": "#fff",  
    "navigationBarTitleText": "24 爱购",  
    "navigationBarTextStyle": "black"  
  },  
  "tabBar": {  
    "color": "#dddddd",  
    "selectedColor": "#3cc51f",
```

```
"borderStyle": "black",

"backgroundColor": "#ffffff",

"list": [{

  "pagePath": "pages/index/index",

  "text": "首页",

  "iconPath": "images/icon_component.png",

  "selectedIconPath": "images/icon_component_HL.png"

},

{

  "pagePath": "pages/mine/mine",

  "text": "我的",

  "iconPath": "images/icon_API.png",

  "selectedIconPath": "images/icon_API_HL.png"

}]

},

"networkTimeout": {
```

```
"request": 10000,  
  
"connectSocket": 10000,  
  
"uploadFile": 10000,  
  
"downloadFile": 10000  
  
},  
  
"debug": true  
  
}
```

1.pages 和 iOS 不同，在小程序中所有的页面都需要注册。不注册跳转的就会报错。

2.window 和 tabbar 这个就不用多说了，大家看上面的字面意思就能理解

3.networkTimeout 网络请求的超时时间

app.wxss

这个就是一个 css 的加强版，在我们 iOS 中就是 layout 的代码。

在这里写的 css 都是全局的

开始撸页面

我们知道我们在写 iOS 程序的时候一般都是分 MVC 的，在小程序对于一个页面一般有 3 类文件

wxml,js,wxss . 对于 wxml 就等同于我们的 view , js 就等同于我们的 model 和 controller , 而 wxss 就

是把 view 中的 layout 单独提取出来。

这样分，我感觉应该很清楚了。

我们现在就撸一个 tabbleview 练练手



按照我们 iOS 的套路直接一个 tableview 写好 cell 搞定。

可是在小程序中没有列表这种 view，我们只有用 for 循环来渲染啦

```
<view class="mine-item-list">
```

```
  <block wx:for="{{itemArray}}">
```

```
    <view class="mine-item">
```

```
      <image class="mine-item-img" src="{{item['img']}}"></image>
```

```
      <text class="mine-item-name">{{item["name"]}}</text>
```

```
        <image class="mine-item-arrow"
src="../../images/arrowright.png"></image>

    </view>

</block>

</view>
```

对了，我要提一下。这个 wxml 就和 html 没什么区别，view 看成 div