

程序员开发实战系列《二》微信小程序架构篇

一、小程序架构

每个小程序的结构都是由两个主要部分构成：主体部分 + 各个页面。

类似于许多框架，主体部分主要用于核心的配置，各个页面主要用于不同业务场景。

1.1，主体部分主要由 3 个文件构成

- 1) app.js: 小程序逻辑，初始化 APP
- 2) app.json : 小程序配置，比如导航、窗口、页面 http 请求跳转等
- 3) app.wxss: 公共样式配置

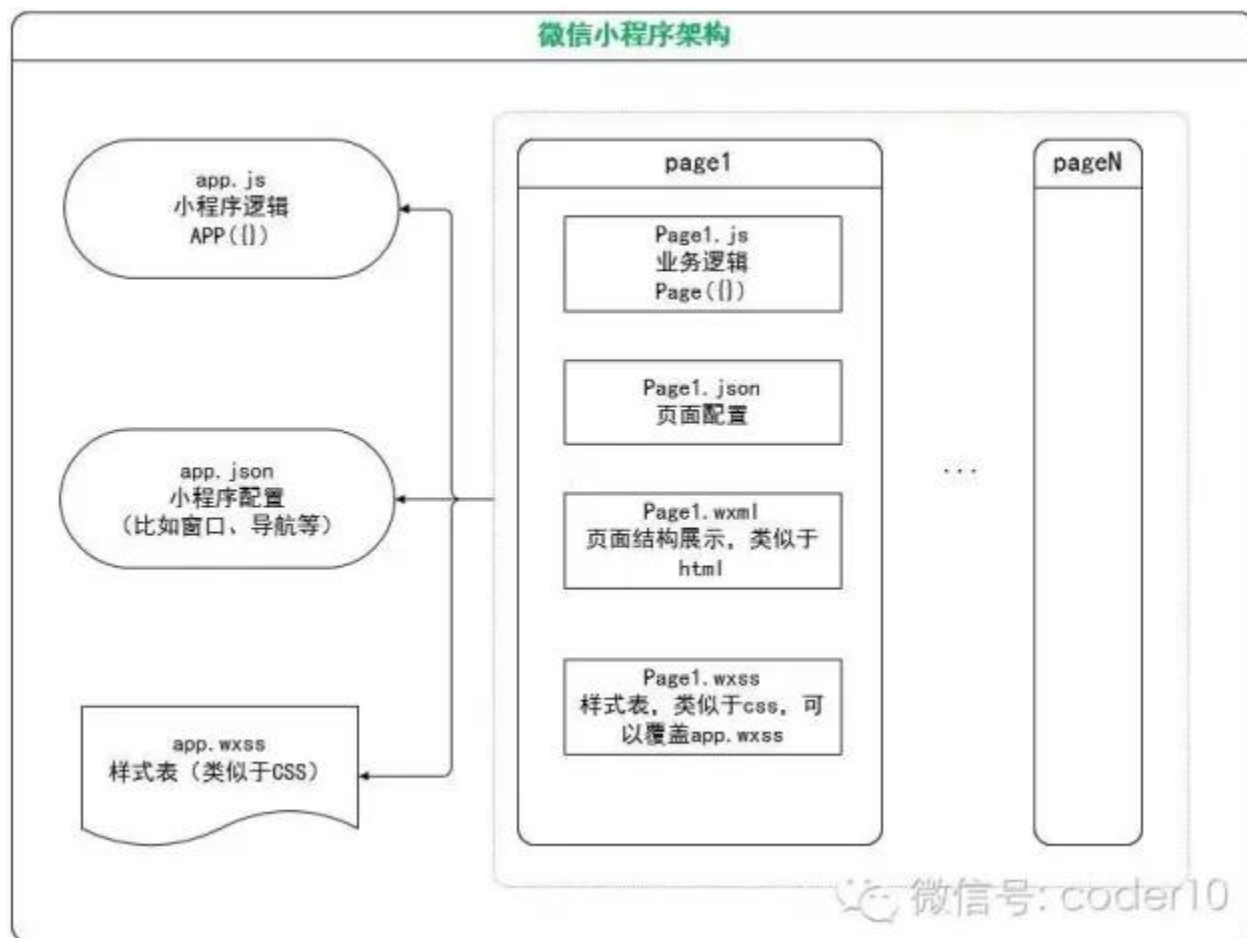
主体配置完成之后，就是对应的业务开发了，也就是开发者最常操作的页面。小程序页面设计基本上也是遵循 MVC 结构进行构建。

MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC 被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

1.2，页面由 4 个文件构成

- 1) js: 页面逻辑，相当于控制层（C）；也包括部分的数据（M）
- 2) wxml: 页面结构展示，相当于视图层（V）
- 3) wxss: 页面样式表，纯前端，用于辅助 wxml 展示
- 4) json: 页面配置，配置一些页面展示的数据，充当部分的模型（M）

如图



代码结构参考



二、配置 app.json

简单介绍核心架构配置，详细内容请参考小程序开发文档。

app.json

全局配置页面文件的路径、窗口表现、设置网络超时时间、设置多 tab 等。

pages

接受一个数组，每一项都是字符串，来指定小程序由哪些页面组成

pages	String Array	设置页面路径	<pre> "pages": ["pages/index/index" "pages/logs/logs"] </pre>  微信号: coder10
-------	--------------	--------	---

window

用于设置小程序的状态栏、导航条、标题、窗口背景色。

window	Object	设置默认页面的窗口表现	<pre> "window": { "navigationBarBackgroundColor": "#ffffff", "navigationBarTextStyle": "black", "navigationBarTitleText": "微信接口功能演示", "backgroundColor": "#eeeeee", "backgroundTextStyle": "light" } </pre>
navigationBarBackgroundColor	HexColor (#000000)	导航栏背景颜色，如"#000000"	
navigationBarTextStyle	String (white)	导航栏标题颜色，仅支持 black/white	
navigationBarTitleText	String	导航栏标题文字内容	
backgroundColor	HexColor (#ffffff)	窗口的背景色	
backgroundTextStyle	String (dark)	下拉背景字体、loading 样式，仅支持 dark/light	
enablePullDownRefresh	Boolean (false)	是否开启下拉刷新	

tabBar

通过 tabBar 配置项指定 tab 栏的表现，以及 tab 切换时显示的对应页面。

tabBar 配置数组，只能配置最少 2 个、最多 5 个 tab，tab 按数组的顺序排序。

tabBar	Object	设置底部tab的表现	<pre> "tabBar": { "selectedColor": "red", "list": [{ "pagePath": "pages/index/index", "text": "首页" }, { "pagePath": "pages/logs/logs", "text": "日志" }] } </pre>
color	HexColor	tab上的文字默认颜色	
selectedColor	HexColor	tab上的文字选中时的颜色	
backgroundColor	HexColor	tab的背景色	
borderStyle	String(black)	tabbar上边框的颜色, 仅支持 black/white	
list	Array	tab的列表, 详见list属性说明, 最少2个、最多5个	
position	String(bottom)	可选值 bottom、top	
list			
pagePath	String	页面路径, 必须在 pages 中先定义	
ext	String	tab 上按钮文字	
conPath	String	图片路径, icon 大小限制为40kb	
selectedIconPath	String	选中时的图片路径, icon 大小限制为40kb	

networkTimeout

可以设置各种网络请求的超时时间。

networkTimeout	Object	设置网络超时时间	<pre> "networkTimeout": { "request": 1000, "connectSocket": 1000, "uploadFile": 5000, "downloadFile": 5000 } </pre>
request		Number	wx.request的超时时间，单位毫秒
connectSocket		Number	wx.connectSocket的超时时间，单位毫秒
uploadFile		Number	wx.uploadFile的超时时间，单位毫秒
downloadFile		Number	wx.downloadFile的超时时间，单位毫秒

debug

true/false

可以在开发者工具中开启 **debug** 模式，在开发者工具的控制台面板，调试信息以 **info** 的形式给出，其信息有 **Page** 的注册，页面路由，数据更新，事件触发 。 可以帮助开发者快速定位一些常见的问题。

page.json

每一个小程序页面也可以使用.json 文件来对本页面的窗口表现进行配置。 页面的配置比 app.json 全局配置简单得多，只是设置 app.json 中的 **window** 配置项的内容，页面中配置项会覆盖 app.json 的 **window** 中相同的配置项。

debug	Boolean	是否调试模式	"debug": true
page. json	Object	页面json配置	<p>页面的.json只能设置 window 相关的配置项，以决定本页面的窗口表现，所以无需写 window 这个键，如：</p> <pre> { "navigationBarBackgroundColor": "#ffffff", "navigationBarTextStyle": "black", "navigationBarTitleText": "微信接口功能演示", "backgroundColor": "#eeeeee", "backgroundTextStyle": "light" } </pre>

