

白球手札：微信小程序学习摘要《二》.json，逻辑层

一：.json-配置

进行全局配置，
决定页面文件的路径、
窗口表现、
设置网络超时时间、
设置多 tab
...

app.json 配置项列表

属性	类型	必填	描述
pages	String Array	是	设置页面路径
window	Object	否	设置默认页面的窗口表现
tabBar	Object	否	设置底部 tab 的表现
networkTimeout	Object	否	设置网络超时时间
debug	Boolean	否	设置是否开启 debug 模式

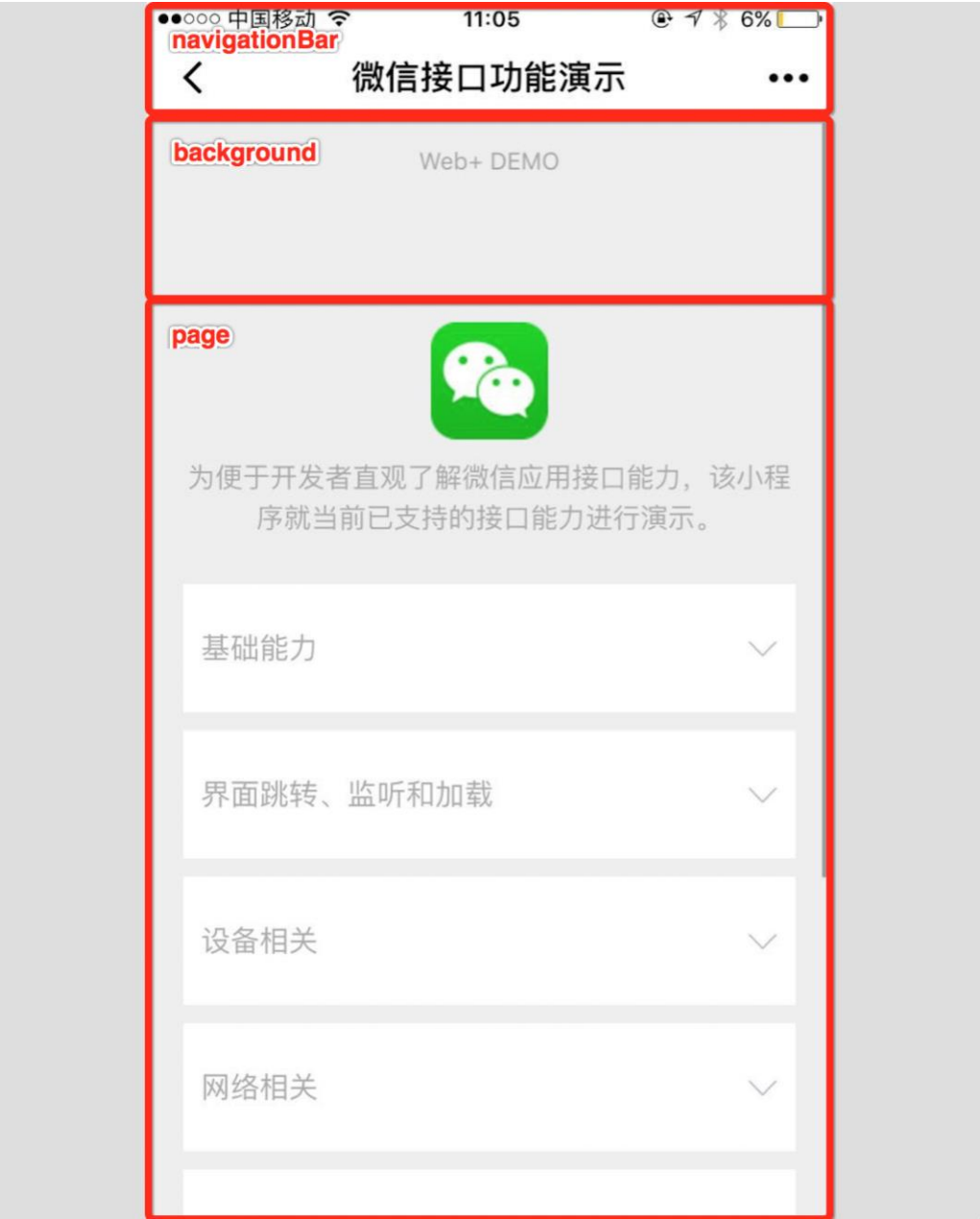
pages

小程序中新增/减少页面，都需要对 pages 数组进行修改。

文件名不需要写文件后缀，因为框架会自动去寻找路径.js,json,.wxml,.wxss 的四个文件进行整合

window

设置小程序的状态栏、导航条、标题、窗口背景色。



tabBar

如果我们的程序是一个多 tab 应用（客户端窗口的底部有 tab 栏可以切换页面），那么我们可以通过 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 是一个数组，只能配置最少 2 个、最多 5 个 tab，tab 按数组的顺序排序。

内容与显示分离

{[几个包含数据的对象], 统一的样式设置 1, 统一的样式设置 2, 统一的样式设置 3}



networkTimeout

可以设置各种网络请求的超时时间。

debug

可以在开发者工具中开启 debug 模式

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。

只

只是

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。

决定本页面的窗口表现

无需!!!写 **window** 这个键，如：

```
{
  "navigationBarBackgroundColor": "#ffffff",
  "navigationBarTextStyle": "black",
  "navigationBarTitleText": "微信接口功能演示",
  "backgroundColor": "#eeeeee",
  "backgroundTextStyle": "light"
}
```

二：逻辑层(App Service)

小程序开发框架的逻辑层是由 **JavaScript** 编写。

逻辑层将数据进行处理后发送给视图层，

同时接受视图层的事件反馈。

在 JavaScript 的基础上，我们做了一些修改，以方便地开发小程序：

增加 **App** 和 **Page** 方法，进行程序和页面的注册。

增加 `getApp` 和 `getCurrentPages` 方法，分别用来获取 App 实例和当前页面栈。

提供丰富的 **API**，如微信用户数据，扫一扫，支付等微信特有功能。

每个页面有独立的作用域，并提供**模块化**能力。

由于框架并非运行在浏览器中，所以 JavaScript 在 **web** 中一些能力都无法使用，如 `document`，`window` 等。

开发者写的所有代码最终将会打包成一份 **JavaScript**，并在小程序启动的时候运行，直到小程序销毁。类似 `ServiceWorker`，所以逻辑层也称之为 App Service。(RDD:类似于 JSX)

APP

App()

`App()` 函数用来注册一个小程序。接受一个 `object` 参数，其指定小程序的生命周期函数等。

object 参数说明：

属性	类型	描述	触发时机
<code>onLaunch</code>	<code>Function</code>	生命周期函数--监听小程序初始化	当小程序初始化完成时,会触发 <code>onLaunch</code> （全局只触发一次）
<code>onShow</code>	<code>Function</code>	生命周期函数--监听小程序显示	当小程序启动，或从后台进入前台显示，会触发 <code>onShow</code>
<code>onHide</code>	<code>Function</code>	生命周期函数--监听小程序隐藏	当小程序从前台进入后台,会触发 <code>onHide</code>
其他	<code>Any</code>	开发者可以添加任意的函数或数据到 <code>Object</code> 参数中，用 <code>this</code> 可以访问	

RDD 注解：

小程序的数据库。

小程序的宏观状态变化时的表现。

打开

后台 -> 前台

前台 -> 后台

getApp()

我们提供了全局的 `getApp()` 函数，可以获取到小程序实例。

```
// other.js
var appInstance = getApp()
console.log(appInstance.globalData) // I am global data
```

注意

唯一：

`App()` 必须在 `app.js` 中注册，且不能注册多个。

一次：

不要在定义于 `App()` 内的函数中调用 `getApp()`，使用 `this` 就可以拿到 `app` 实例。

生命周期：

不要在 `onLaunch` 的时候调用 `getCurrentPage()`，此时 `page` 还没有生成。

不干涉：

通过 `getApp()` 获取实例之后，不要私自调用生命周期函数。

三：微信小程序的一些特点

小

压缩之后不得大于 1M

内容限制

不允许做游戏（APPLE

微信的框架

比较封闭受限的环境

限制大部分对 DOM 的操作。

限制对外链接的数据。

限制多数的 API

(但对‘亲儿子’开放

比较粗糙的模块化。

阉割版的 require.js

比较粗糙的 MVVC 实现。

直接修改数据时并不会自动刷新，而要通过使用 `data.set({})` 来触发刷新