

微信小程序学习笔记：目录结构详解及数据绑定

目录结构

- pages 所有页面
- app.js 全局js
- app.json 注册页面，配置小程序
- app.wxss 全局样式文件

pages 存放页面文件

每个页面一个文件夹，下面.xml和.js文件是必需的。 .wxss和.json文件不是必需的，如果没有的话就用全局的 app.wxss 和 app.json 如果有的话就覆盖掉全局的用自己的

app.json 页面

```
{ //页面注册，无论有几个页面，所有都要在 pages 里面注册 "pages":[ "pages/index/index",  
"pages/logs/logs", "pages/main/main", "pages/main1/main1", "pages/main2/main2",  
"pages/main3/main3", ],  
  
//上导航配置及 app 背景颜色设置 "window":{ //是否开启下拉刷新 "enablePullDownRefresh":true,  
  
//窗口背景颜色，在上拉刷新，下拉刷新，navigate 切换页面时可以看见 "backgroundColor":"#eee",
```

```
//下拉背景字体、loading 图的样式 ,仅支持 dark/light,( string 形式 ) "backgroundTextStyle":"light",

//上导航条背景颜色 "navigationBarBackgroundColor": "orange",

//上导航标题文字 "navigationBarTitleText": "上导航标题文字",

//上导航标题字体颜色 , 仅支持 white 和 black "navigationBarTextStyle":"white" },

//底部导航配置 "tabBar": { "color": "#a9b7b7",//导航字体默认颜色

"selectedColor": "#eb4f38",//导航字体选中时颜色

"borderStyle": "black", //上 border 颜色 , 仅支持 black/white ( string 形式 )

"backgroundColor": "rgba(0,0,0,0.4)",//底部导航背景色

"list": [ //导航列表 2-5 个 { "pagePath": "pages/index1/index1", //每个导航的网页对应的路径

"text": "首页", //标题 "iconPath": "images/wechat.png", //默认小图标路径 "selectedIconPath":

"images/wechatHL.png" //选中时的小图标路径 },

{ "pagePath": "pages/index2/index2", "text": "日记", "iconPath": "images/pause.png",

"selectedIconPath": "images/play.png" },

{ "pagePath": "pages/index3/index3", "text": "指南", "iconPath": "images/pause.png",

"selectedIconPath": "images/play.png" }, ] },

//网络请求超时时间 "networkTimeout": { "request": 10000, "downloadFile": 9000,
```

```
"uploadFile":8000, "connectSocket":7000 },
```

```
//是否开启 debug 模式 "debug": true }
```

//说明：根据官方文档，上述 color 和 bg-color 类型为 HexColor 即 16 进制类型，（指定仅支持

black/white，string 形式的除外），但实测 rgb,rgba 和 red 这些类型目前也支持，但是建议大家还是

按文档要求来设置为 16 进制类型即: #ffffff 类型；//使用时仅需将文件复制下来,根据需求改动即可；

//app.json 为标准的 json 文件，所以不能存在有注释，使用的时候需把注释去掉；

app.js 页面

```
App({ //当小程序初始化完成时，会触发 onLaunch（全局只触发一次） onLaunch: function () {
```

```
//onLaunch 时调用 API 从本地缓存中获取数据 var logs = wx.getStorageSync('logs') || []
```

```
logs.unshift(Date.now()) wx.setStorageSync('logs', logs) },
```

```
//获取用户登录信息 getUserInfo:function(cb){ var that = this if(this.globalData.userInfo){ typeof
```

```
cb == "function" && cb(this.globalData.userInfo) }else{ //调用登录接口 wx.login({ success:
```

```
function () { wx.getUserInfo({ success: function (res) { that.globalData.userInfo = res.userInfo
```

```
typeof cb == "function" && cb(that.globalData.userInfo) } } } }, //获取后台数据
```

```
onshow:function(){ wx.request({ url: 'test.php', data: { x: 'xxx', y: 'yyy' }, header: { 'Content-Type':
```

```
'application/json' }, success: function(res) { console.log(res.data) } })
```

},

//设置全局数据 //本页面通过 this.globalData 即可取得数据 globalData:{ userInfo:null } //其他页面
可以通过 getApp() 获取到小程序实例 //其他页面可以通过 getApp().globalData.xxx 获取到全局数据 })

app.wxss 页面

.common { height: 100%; display: flex; flex-direction: column; align-items: center; justify-content:
space-between; padding: 200rpx 0; box-sizing: border-box; }

//仅支持: 类:.common ID:#common 标签 : common 多个标签 : common1,common2 伪类::after
伪类::before //不支持层级, 如.common > view 或者.common view

//引入新的单位 rpx: rpx (responsive pixel) : 可以根据屏幕宽度进行自适应。规定屏幕宽为 750rpx。

如在 iPhone6 上, 屏幕宽度为 375px, 共有 750 个物理像素, 则 $750\text{rpx} = 375\text{px} = 750$ 物理像素,
 $1\text{rpx} = 0.5\text{px} = 1$ 物理像素。

及新的 rem: rem (root em) : 规定屏幕宽度为 20rem ; $1\text{rem} = (750/20)\text{rpx}$ 。

//写在 app.wxss 里面的样式将被设置为公有的, 对于其他页面都是可用的

数据绑定添加事件

视图层在标签中添加 以 bind 或者 catch 开头 bind 是冒泡, catch 阻止冒泡 示例 :

```
<div> bindtap="showName">    {{name}}</div>
```

```
//js 中事件函数"showName": function(){ //事件处理代码 console.log("hello");}
```

事件类型

事件有：

- touchstart
- touchmove
- touchend
- touchcancel
- tap 点触摸 相当于 pc 的 click
- longtap 触摸后 350ms 再离开

事件对象

- type 事件类型
- timeStamp 事件戳
- target 触发事件的组件 (DOM 元素),也就是事件源
- currentTarget 当前组件、
- detail 事件相关的其他信息

项目地址：<https://github.com/shunrong/WeixinApp/tree/master/WeixinProgram>