

微信小程序：

生命周期：

属性	类型	描述
<code>data</code>	Object	页面的初始数据
<code>onLoad</code>	Function	生命周期函数--监听页面加载
<code>onReady</code>	Function	生命周期函数--监听页面渲染完成
<code>onShow</code>	Function	生命周期函数--监听页面显示
<code>onHide</code>	Function	生命周期函数--监听页面隐藏
<code>onUnload</code>	Function	生命周期函数--监听页面卸载
其他	Any	开发者可以添加任意的函数或数据到 OBJECT 参数中，用 <code>this</code> 可以访问

简单描述下微信小程序的相关文件类型？

微信小程序项目结构主要有四个文件类型,如下：

一、WXML（WeiXin Markup Language）是框架设计的一套标签语言，结合基础组件、事件系统，可以构建出页面的结构。内部主要是微信自己定义的一套组件。

二、WXSS (WeiXin Style Sheets)是一套样式语言，用于描述 WXML 的组件样式，

二、js 逻辑处理，网络请求

三、json 小程序设置，如页面注册，页面标题及tabBar。

四、app.json

必须要有这个文件，如果没有这个文件，项目无法运行，因为微信框架把这个作为配置文件入口，整个小程序的全局配置。包括页面注册，网络设置，以及小程序的window背景色，配置导航条样式，配置默认标题。

五、app.js

必须要有这个文件，没有也是会报错！但是这个文件创建一下就行 什么都不需要写以后我们可以在这个文件中监听并处理小程序的生命周期函数、声明全局变量。

六、app.wxss

微信小程序的原理：

微信小程序采用JavaScript、WXML、WXSS三种技术进行开发

JavaScript: 首先JavaScript的代码是运行在微信App中的，并不是运行在浏览器中，因此一些H5技术的应用，需要微信App提供对应的API支持，而这限制住了H5技术的应用，且其不能称为严格的H5，可以称其为伪H5，同理，微信提供的独有的某些API，H5也不支持或支持的不是特别好。

WXML: WXML微信自己基于XML语法开发的，因此开发时，只能使用微信提供的现有标签，HTML的标签是无法使用的。

WXSS: WXSS具有CSS的大部分特性，但并不是所有的都支持，而且支持哪些，不支持哪些并没有详细的文档。

微信的架构，是数据驱动的架构模式，它的UI和数据是分离的，所有的页面更新，都需要通过对数据的更改来实现。

小程序分为两个部分webview和appService。其中webview主要用来展现UI，appService有来处理业务逻辑、数据及接口调用。它们在两个进程中运行，通过系统层JSBridge实现通信，实现UI的渲染、事件的处理

微信小程序的坑？

- 1、限制较多。页面大小不能超过1M。不能打开超过5个层级的页面。
- 2、样式单一。小程序的部分组件已经是成型了的，样式不可以修改。例如：幻灯片、导航。
- 3、推广面窄，不能分享朋友圈，只能通过分享给朋友，附近小程序推广。其中附近小程序也受到微信的限制。
- 4、依托于微信，无法开发后台管理功能。

你是怎么封装微信小程序的数据请求的？

- 一、将所有的接口放在统一的js文件中并导出
- 二、在app.js中创建封装请求数据的方法
- 三、在子页面中调用封装的方法请求数据

有哪些参数传值的方法？

- 一、给HTML元素添加data-*属性来传递我们需要的值，然后通过e.currentTarget.dataset或onload的param参数获取。但data-名称不能有大写字母和不可以存放对象
- 二、设置id的方法标识来传值通过e.currentTarget.id获取设置的id的值,然后通过设置全局对象的方式来传递数值
- 三、在navigator中添加参数传值

你使用过哪些方法，来提高微信小程序的应用速度？

- 一、提高页面加载速度
- 二、用户行为预测
- 三、减少默认data的大小
- 四、组件化方案

微信小程序与H5的区别？

第一条是运行环境的不同

传统的HTML5的运行环境是浏览器，包括webview，而微信小程序的运行环境并非完整的浏览器，是微信开发团队基于浏览器内核完全重构的一个内置解析器，针对小程序专门做了优化，配合自己定义的开发语言标准，提升了小程序的性能。

第二条是开发成本的不同

只在微信中运行，所以不用再去顾虑浏览器兼容性，不用担心生产环境中出现不可预料的奇妙BUG

第三条是获取系统级权限的不同

系统级权限都可以和微信小程序无缝衔接

第四条便是应用在生产环境的运行流畅度

长久以来，当HTML5应用面对复杂的业务逻辑或者丰富的页面交互时，它的体验总是不尽人意，需要不断的对项目优化来提升用户体验。但是由于微信小程序运行环境独立

怎么解决小程序的异步请求问题？

在回调函数中调用下一个组件的函数：

app.js

```
success: function (info) {  
  
    that.apirtnCallback(info)  
}
```

index.js

```
onLoad: function () {  
    app.apirtnCallback = res => {  
        console.log(res)  
    }  
}
```

小程序的双向绑定和vue哪里不一样?

小程序直接this.data的属性是不可以同步到视图的，必须调用：

```
this.setData({  
    noBind:true  
})
```

小程序的wxss和css有哪些不一样的地方?

一、wxss的图片引入需使用外链地址；二、没有Body；样式可直接使用import导入

小程序关联微信公众号如何确定用户的唯一性?

使用wx.getUserInfo方法withCredentials为 true 时 可获取encryptedData，里面有union_id。后端需要进行对称解密

如何实现下拉刷新?

用view代替scroll-view,,设置onPullDownRefresh函数实现

使用webview直接加载要注意哪些事项？

- 一、必须要在小程序后台使用管理员添加业务域名；
- 二、h5页面跳转至小程序的脚本必须是1.3.1以上；
- 三、微信分享只可以都是小程序的主名称了，如果要自定义分享的内容，需小程序版本在1.7.1以上；
- 四、h5的支付不可以是微信公众号的appid，必须是小程序的appid，而且用户的openid也必须是用户和小程序的。

小程序调用后台接口遇到哪些问题？

- 一、数据的大小有限制，超过范围会直接导致整个小程序崩溃，除非重启小程序；
- 2、小程序不可以直接渲染文章内容页这类型的html文本内容，若需显示要借住插件，但插件渲染会导致页面加载变慢，所以最好在后台对文章内容的html进行过滤，后台直接处理批量替换p标签div标签为view标签，然后其它的标签让插件来做，减轻前端的时间

webview的页面怎么跳转到小程序导航的页面？

小程序导航的页面可以通过switchTab，但默认情况是不会重新加载数据的。若需加载新数据，则在success属性中加入以下代码即可

```
success: function (e) {  
    var page = getCurrentPages().pop();  
    if (page == undefined || page == null) return;  
    page.onLoad();  
}
```

webview的页面，则通过

```
wx.miniProgram.switchTab({  
    url: '/pages/index/index'  
})
```

小程序和Vue写法的区别？

- 一、循环遍历的时候：小程序是wx:for="list"，而Vue是v-for="info in list"
- 二、调用data模型的时候：小程序是this.data.uinfo，而Vue是this.uinfo；给模型

赋值也不一样，小程序是`this.setData({uinfo:1})`，而Vue是直接`this.uinfo=1`