

# 小程序

# 小程序

## STEPS

1. 小程序简介；
2. 微信小程序基础；
3. 解读初始化项目；
4. 项目实践
5. 项目发布

# 小程序简介

# 移动端开发



Android



iOS



Windows

传统 App（Android / iOS）的开发和运营成本很高

# 小程序简介

- 最早推出小程序的是微信团队
- 小程序的发展历程
  - 2016年1月9日，小程序立项
  - 2016年9月21日，开始内测
  - 2017年1月9日，正式上线
- 本节只讨论微信小程序的内容

# 小程序简介

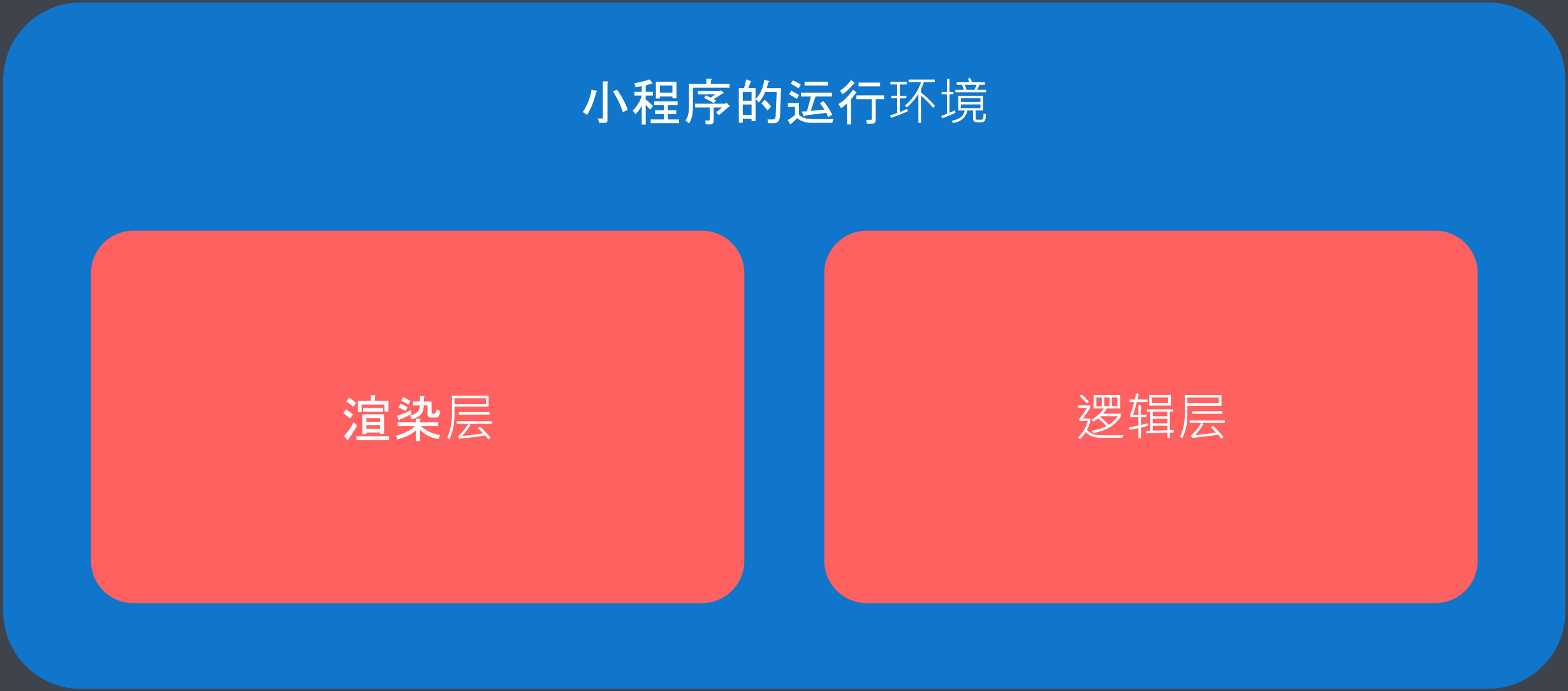
- 相对原生 App（Android / iOS）而言，小程序无需单独下载
  - 只需要搜一搜，或扫一扫，就能直接使用小程序
- 小程序寄宿在 App 中，App 自带流量，小程序营销成本低
  - 例如：跳一跳，利用朋友圈成绩排名，迅速爆红
- 小程序简单易学
  - 只需要掌握前端技术栈

# 小程序与网页开发的区别

- 小程序中不能使用 HTML
- 小程序中可以使用 CSS
- 小程序的主要开发语言是 JavaScript
  - 但是 DOM 和 BOM 在小程序中不可用
  - 基于 DOM 和 BOM 的 JS 库也不能用 – jQuery
- 小程序的运行环境与网页开发不同
  - 网页开发的运行环境是浏览器
  - 微信小程序的运行环境是操作系统（Android 和 iOS）上的微信客户端



# 小程序的运行环境



## 小程序的运行环境



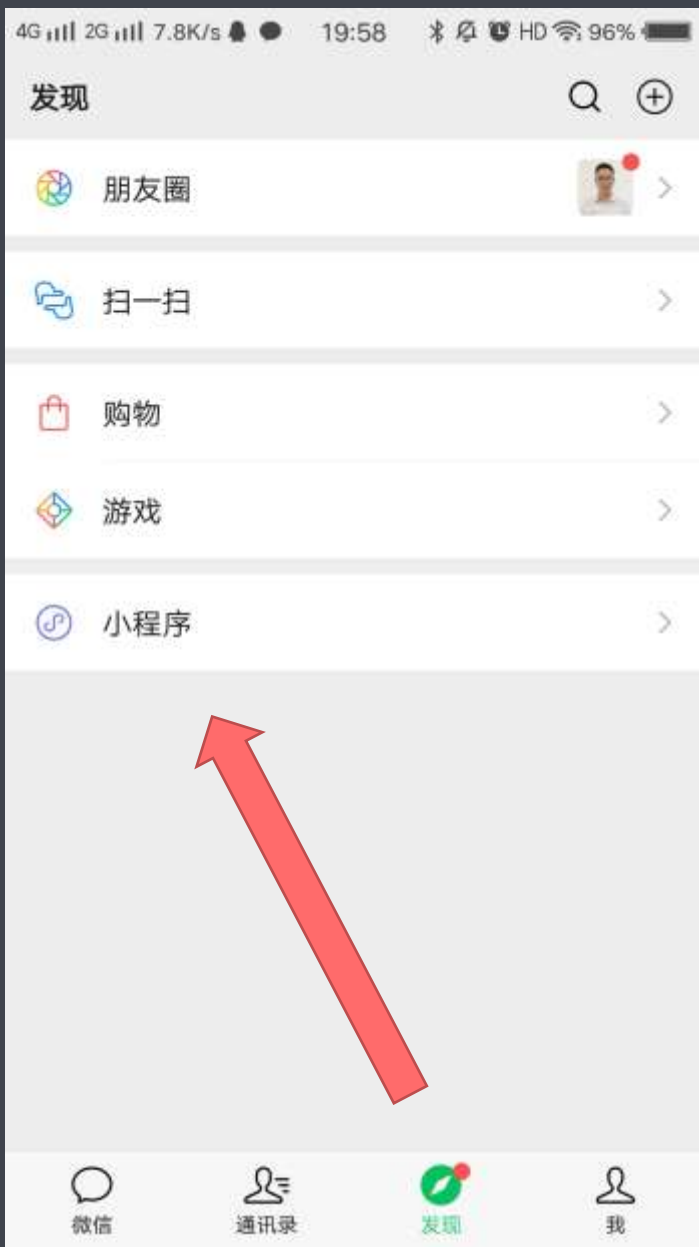
# 小程序的运行环境

运行环境	逻辑层	渲染层
iOS	JSCore	WKWebView
安卓	V8	Chromium 定制内核

# 小程序的运行环境



## 微信小程序入口



# 微信小程序基础

# 微信小程序基础

## STEPS

1. 注册小程序账号；
2. 搭建开发环境；
3. 初始化小程序；

# 微信小程序基础

注册小程序账号



# 注册小程序账号

- 在小程序官网注册
  - <https://mp.weixin.qq.com/wxopen/waregister?action=step1>
- 注册过微信公众号的邮箱，不能再注册微信小程序账号
- 注册成功后，在后台，获取小程序的 AppID
  - 设置 -> 基本设置 -> 账号信息

帐号信息

AppID(小程序ID)    wxd8

# 微信小程序基础

搭建开发环境

# 下载微信开发者工具

- 去小程序官网，下载微信开发者工具

- <https://developers.weixin.qq.com/miniprogram/dev/devtools/download.h>



# 微信小程序基础

初始化小程序

⚙️

✕

小程序项目

小程序

小游戏

代码片段

公众号网页项目

公众号网页

新建项目

导入项目

项目名称

start

目录

D:\lagou\code\miniprogram\start

⌵

AppID

wx

⌵

若无 AppID 可 [注册](#) 或使用 [测试号](#) ?

开发模式

小程序

⌵

后端服务

☐ 小程序·云开发


小程序·云开发为开发者提供数据库、存储和云函数等完整的云端支持。无需搭建服务器，使用平台提供的 API 进行核心业务开发，即可实现小程序快速上线和迭代。 [了解详情](#)

☒ 不使用云服务

语言

JavaScript

⌵

 注销

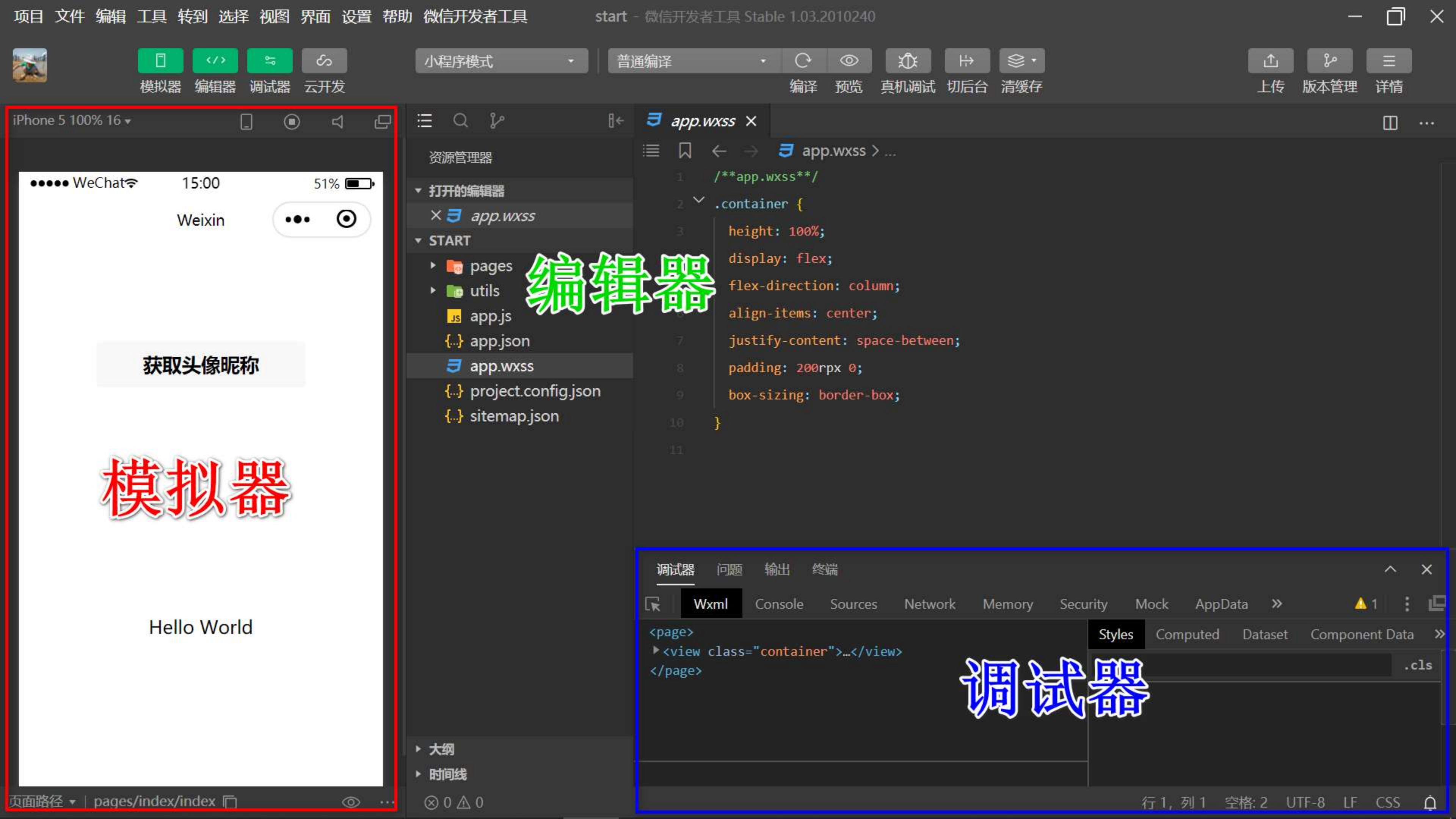
取消

新建

# 初始化微信小程序

- 项目名称：自定义
- 目录：设置本地空目录（用来存储初始化后的小程序代码）
- AppID：登录小程序后台查看（设置 -> 基本设置 -> 账号信息）
- 开发模式：小程序
- 后台服务：不使用云服务
- 语法：JavaScript





模拟器

编辑器

调试器

## 进入小程序

获取头像昵称

### 微信授权

 申请获得以下权限

- 获得你的公开信息（昵称、头像、地区及性别）

拒绝

允许

Weixin



初六

Hello World



# 解读初始化项目

# 解读初始化项目

## STEPS

1. 目录结构；
2. 生命周期；
3. app 代码；
4. index 页面；
5. logs 页面

# 解读初始化项目

目录结构

## 全局文件

app.js

小程序入口文件

app.json

小程序全局配置

app.wxss

小程序全局样式

## 页面文件

index.js

页面入口

index.json

页面配置

index.wxml

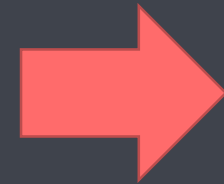
页面内容

index.wxss

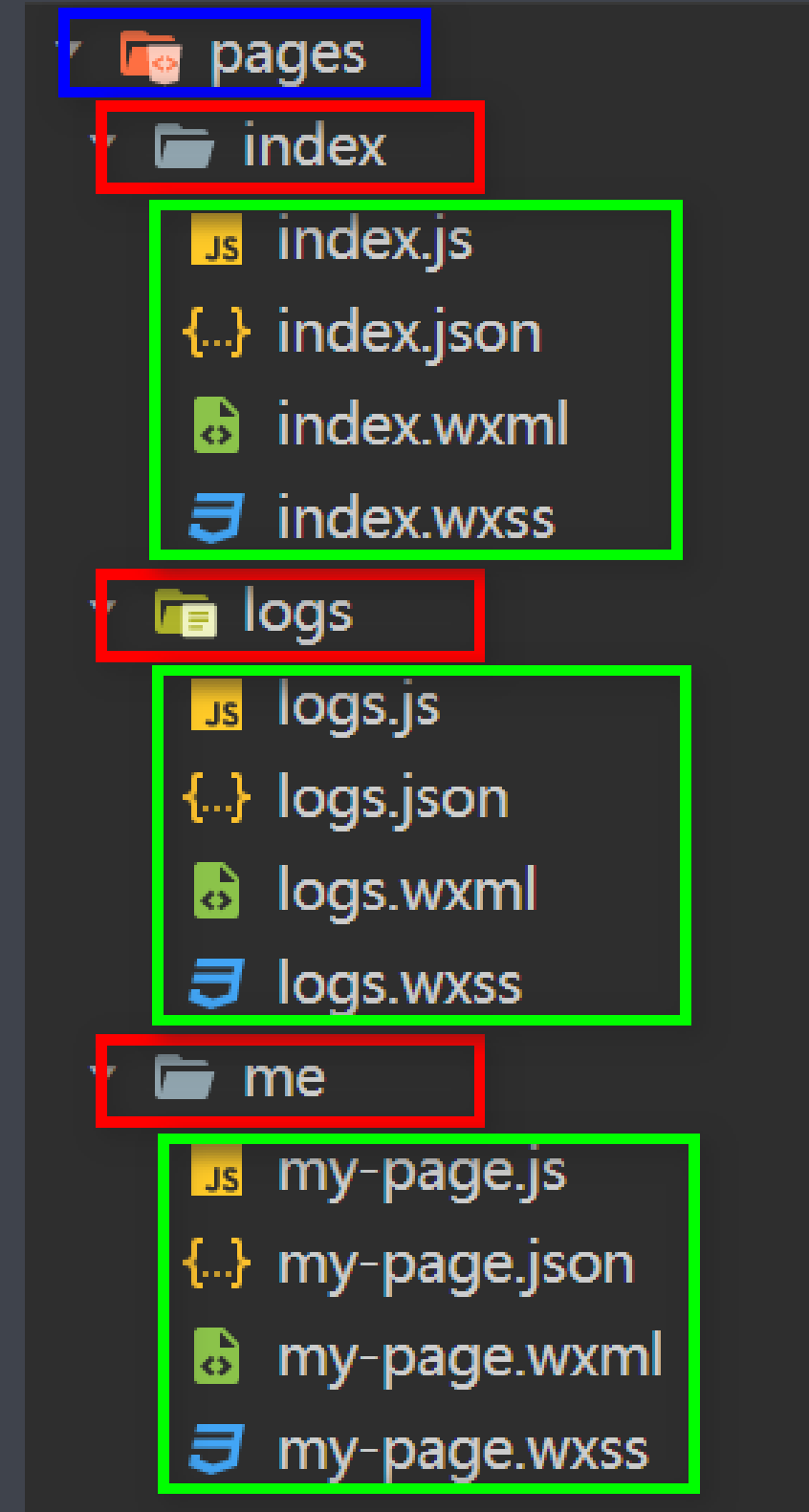
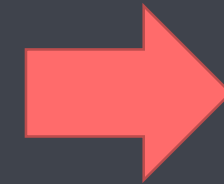
页面样式

## 全局配置 - 路由与页面

app.json



```
"pages": [
  "pages/index/index",
  "pages/logs/logs",
  "pages/me/my-page"
]
```



# 解读初始化项目

生命周期

## 生命周期

小程序的生命周期

app.js

页面的生命周期

pageName.js



## 小程序生命周期 (app.js)

onLaunch

小程序启动 (全局只调用一次)

onShow

初始化完成, 从后台切换到前台

onHide

从前台切换到后台

## 小程序生命周期



## 页面生命周期 ( pagename.js)

onLoad

页面加载（一次）

onShow

页面显示（切前台）

onReady

页面就绪（一次）

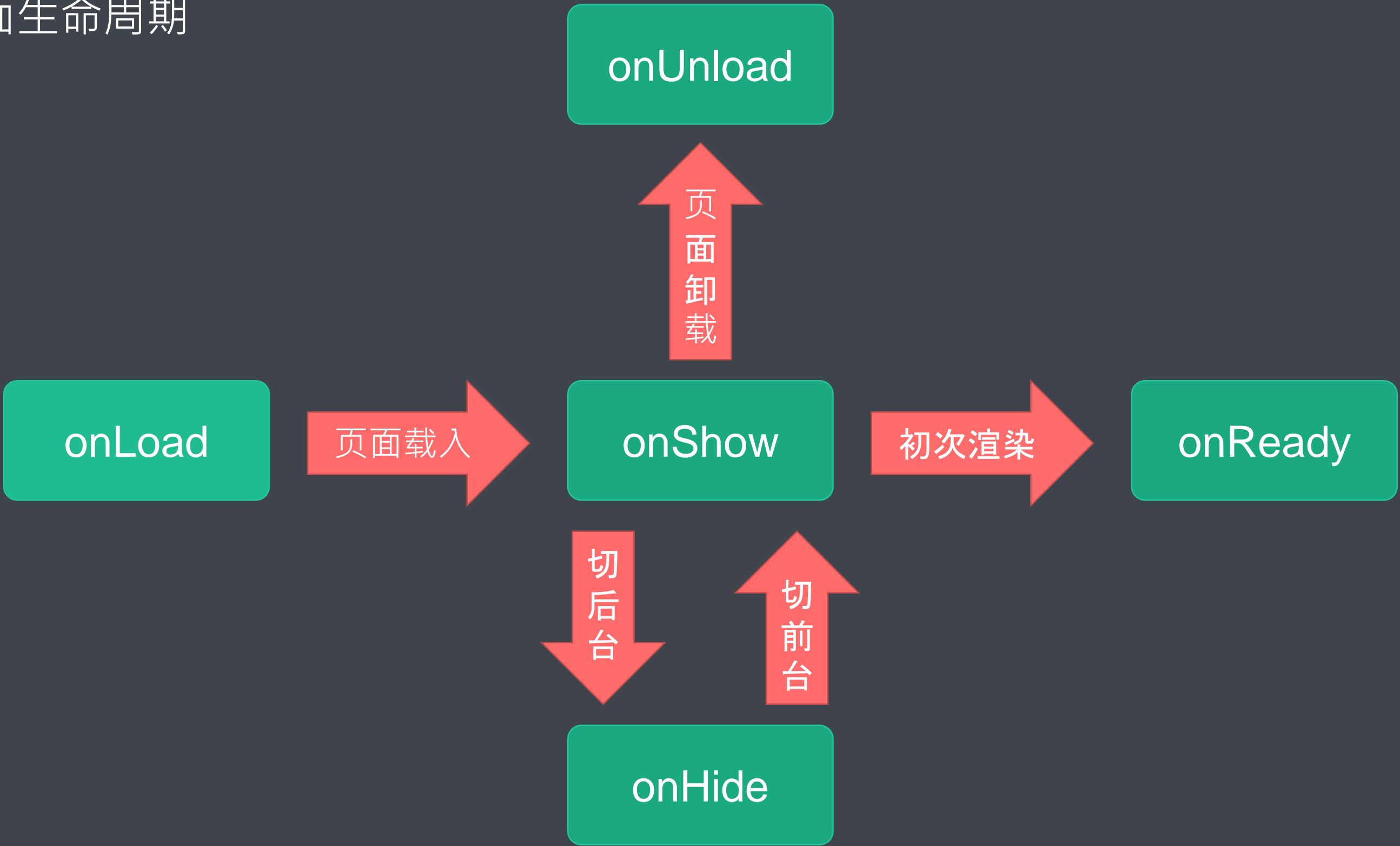
onHide

页面隐藏（切后台）

onUnload

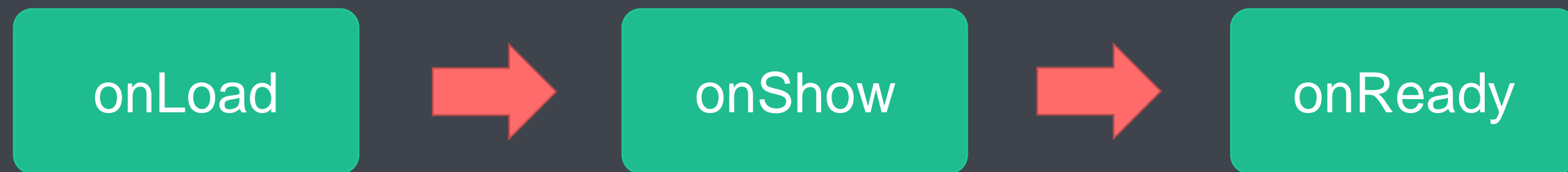
页面卸载（一次）

页面生命周期

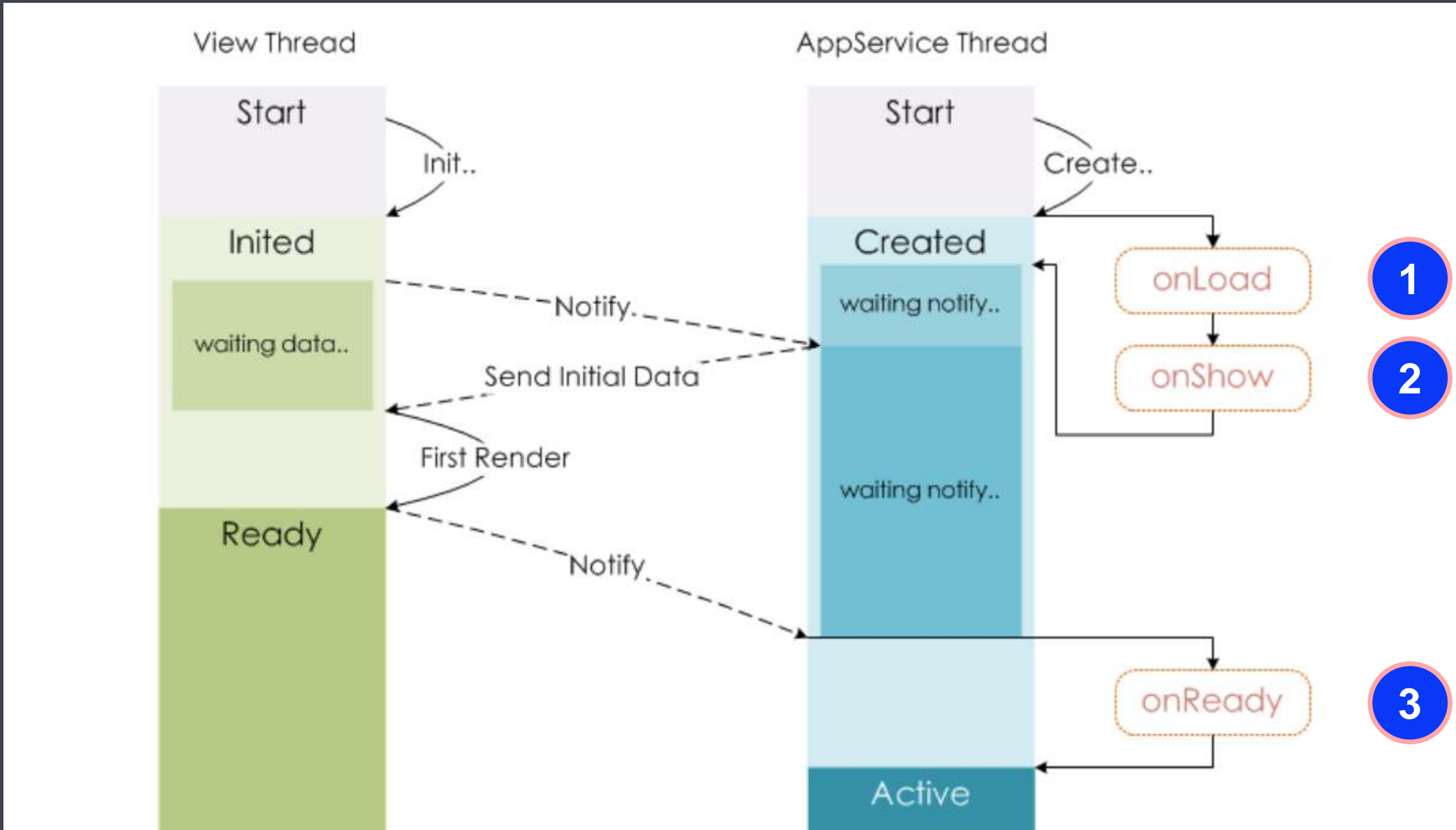


## 页面生命周期函数的调用顺序

onLoad	- 监听页面加载	①
onShow	- 监听页面显示	②
onReady	- 监听页面初次渲染完成	③

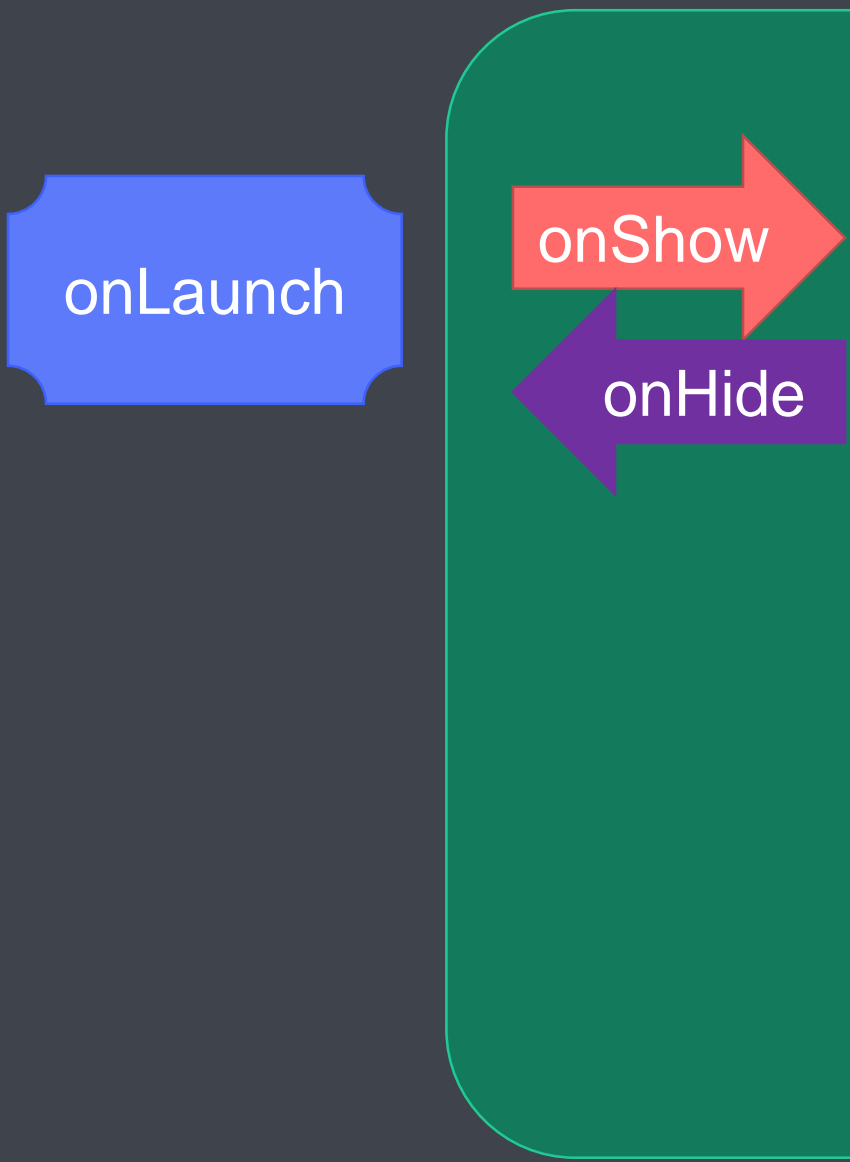


## 页面生命周期函数的调用顺序

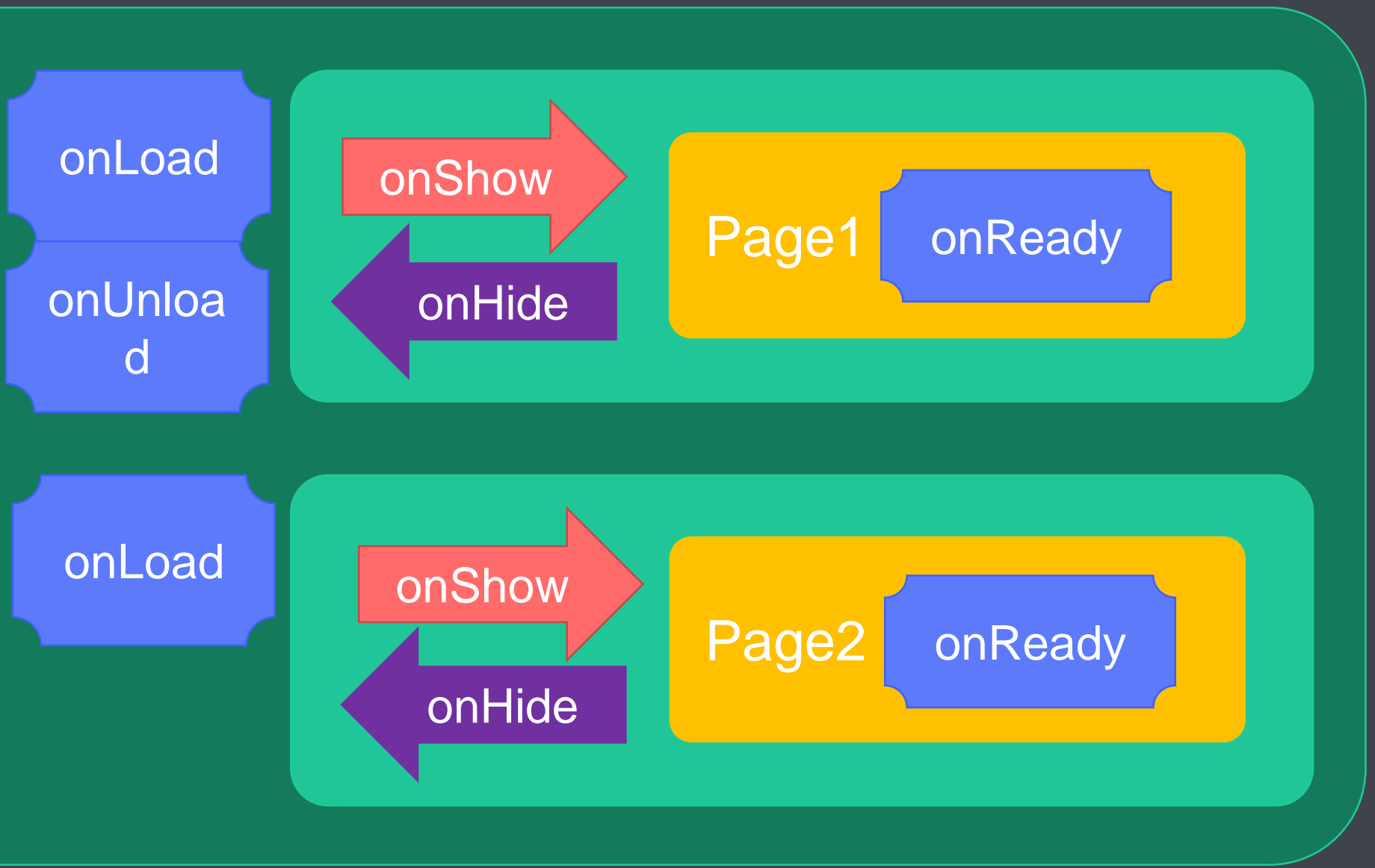


# 微信小程序启动流程

## 小程序生命周期



## 页面生命周期



# 解读初始化项目

app 代码



## 全局文件

app.js

云 API

app.json

全局配置

app.wxss

弹性布局

# 云 API

- 官方推出的，供小程序调用的，云端 API
  - 不需要自己开发
  - 不需要自己部署
  - <https://developers.weixin.qq.com/miniprogram/dev/api/>

# 解读初始化项目

index 页面

## 页面文件

index.js

数据绑定

index.json

页面配置

index.wxml

组件 + 条件渲染

index.wxss

页面样式

# WXML

- WXML (WeiXin Markup Language) 是框架设计的一套标签语言
- WXML 中展示内容的具体标签, 我们称为组件

# 组件

作用	小程序组件	HTML 标签
展示区块	view	div
展示图片	image	img
展示文本	text	p
链接导航	navigator	a
.....	.....	.....

WXML 与组件之间的关系，就相当于 HTML 与标  
签

## 官方组件

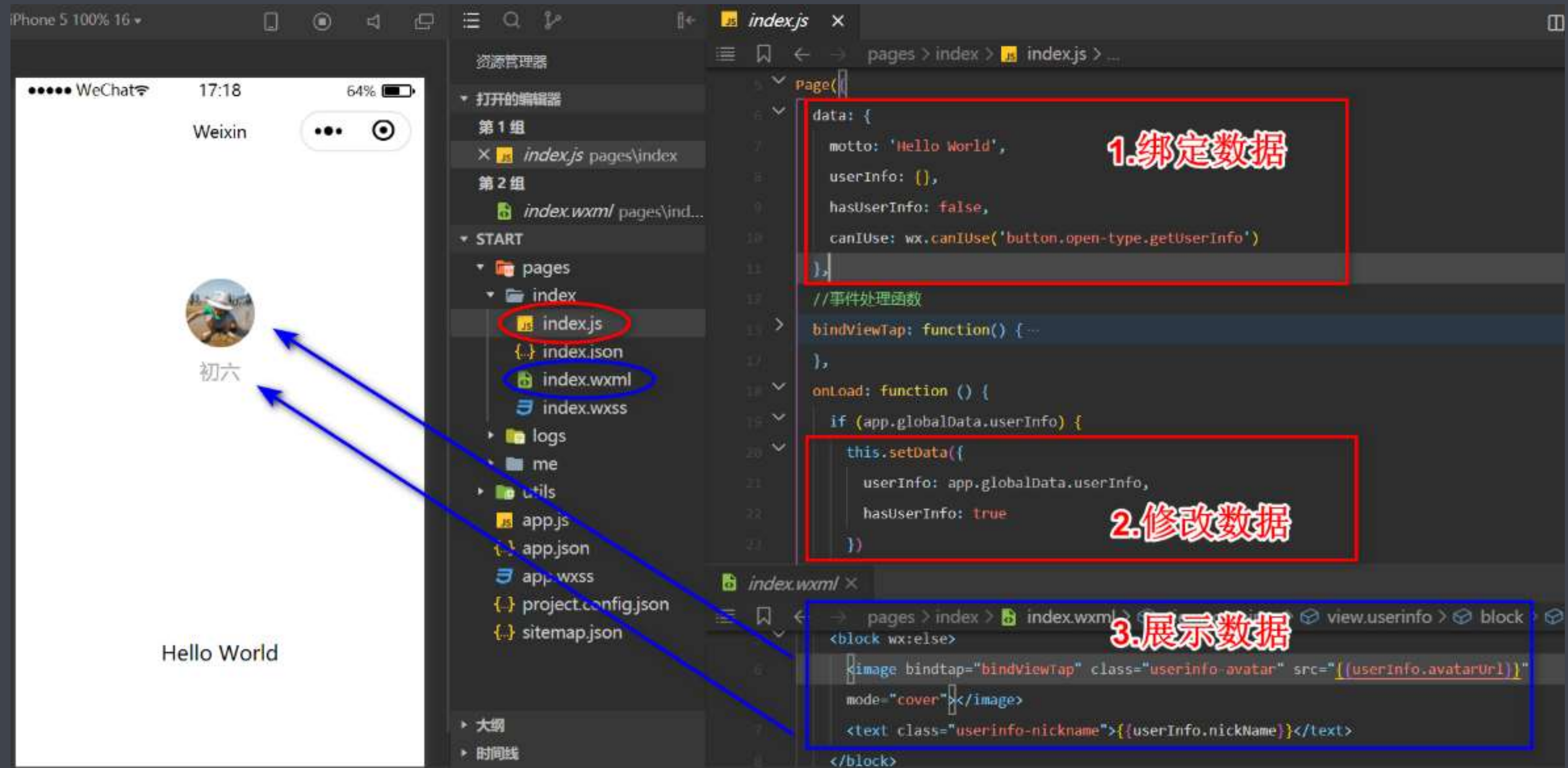




# WXML

- WXML (WeiXin Markup Language) 是框架设计的一套标签语言
- WXML 中展示内容的具体标签, 我们称为组件
- WXML 相当于小程序中的**模板引擎**
  - 数据绑定
  - 数据展示 (条件渲染、列表渲染)

## 数据绑定



# WXML

- WXML (WeiXin Markup Language) 是框架设计的一套标签语言
- WXML 中展示内容的具体标签，我们称为组件
- WXML 相当于小程序中的**模板引擎**
  - 数据绑定
  - 数据展示（条件渲染、列表渲染）
  - WXML:  
<https://developers.weixin.qq.com/miniprogram/dev/reference/wxml/>
  - 组件：<https://developers.weixin.qq.com/miniprogram/dev/component/>

# 解读初始化项目

logs 页面

# logs 页面

- 模块化开发
  - 小程序中的模块化开发遵循 CommonJS 规范（exports、require）
- WXML 的列表渲染
  - wx:for
- 页面级别的配置
  - .json 文件配置页面
  - .wxss 文件设置样式

# 项目实践

# 项目实践

## STEPS

1. 数据接口；
2. 项目界面；
3. 功能组合；

# 项目实践

数据接口



# 数据接口

## STEPS

1. 申请接口；
2. 调试接口；
3. 调用接口；

# 项目实践

数据接口：申请接口

## 申请数据接口

后端工程师

例如：Express 开发的

模拟接口（Mock API）

例如：rap2

第三方接口

例如：和风天气

# 和风天气接口

## STEPS

1. 注册账号；
2. 创建应用并申请密钥（key）；
3. 开发集成

# 和风天气接口

- 注册账号
  - <https://id.qweather.com/#/register>
- 创建应用并申请密钥（key）
  - key 是调用接口的凭证
- 开发集成（开发文档）
  - 请求接口的语法
  - 返回数据的示例

控制台 | 和风天气

控制台和风天气控制台

我的控制台

应用管理

资源包

插件管理

数据统计

订单管理

消费记录

发票管理

联系我们

下载APP

和风天气

开发文档支持购买

我的工单w\*\*\*@163.com

应用管理

1 创建应用

wechat

开发版 20 创建

2 添加数据KEY 设置

KEY名称	Public ID	KEY	类型	操作
weekend	HE1902	343k	Web API	编辑 删除
sinasae	HE1902	5a7	Web API	编辑 删除
lgmp	HE2010	687	Web API	编辑 删除

# 项目实践

数据接口：调试接口

# 接口调试工具



postman



loadrunner



insomnia



# insomnia

- 下载 (Insomnia Core)
  - <https://insomnia.rest/>

## 下载 Insomnia



### Insomnia Designer

The Collaborative API Design Tool for designing, testing and managing OpenAPI specifications.

[↓ Latest Release](#) **NEW**

[Changelog](#)



### Insomnia Core

The Desktop API client for REST and GraphQL. Make requests, inspect responses.

[↓ Latest Release](#)

[Changelog](#)

# insomnia

- 下载 (Insomnia Core)
  - <https://insomnia.rest/>
- 配置
  - 接口调用中使用的公共内容，应该提升为变量，例如：key

## 接口调试中的公共内容

`https://devapi.qweather.com/v7/weather/3d?key=xxx1234xxx&location=116.29,39.95`

`https://devapi.qweather.com/v7/air/now?key=xxx1234xxx&location=104.29,36.21`

`https://devapi.qweather.com/v7/indices/1d?key=xxx1234xxx&location=98.56,34.67`



```
{  
  "url": "https://devapi.qweather.com/v7"  
  "key": "xxx1234xxx"  
}
```



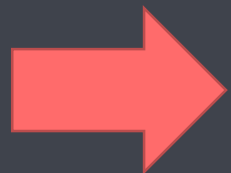
```
url/weather/3d?key=key&location=116.29,39.95  
url/air/now?key=key&location=104.29,36.21  
url/indices/1d?key=key&location=98.56,34.67
```

# insomnia

- 下载 (Insomnia Core)
  - <https://insomnia.rest/>
- 配置
  - 接口调用中使用的公共内容，应该提升为变量，例如：key
  - 不同的项目，具有不同的环境变量

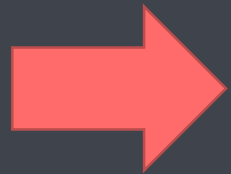
## 接口调试中的环境

和风天气 (qweather)



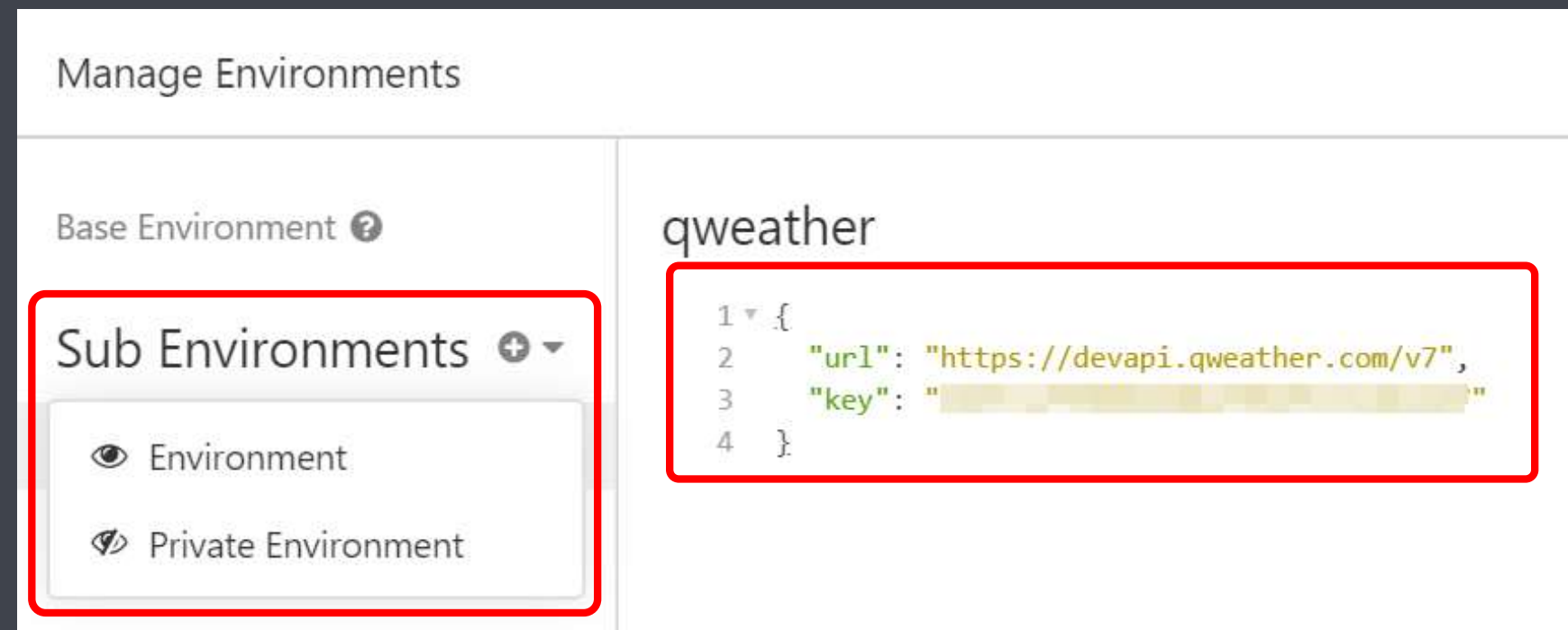
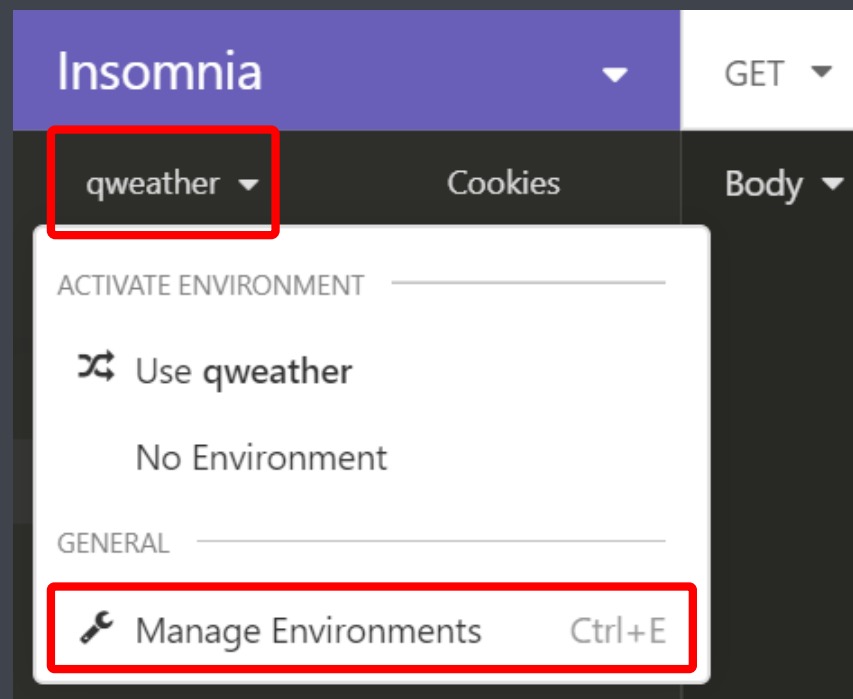
```
{  
  "url":  
    "https://devapi.qweather.com/v7"  
  "key": "xxxx1234xxxx"  
}
```

淘宝商城 (taobao)



```
{  
  "url": "https://api.taobao.com/v7"  
  "key": "xxxx6688xxxx"  
}
```

## Insomnia 配置环境变量

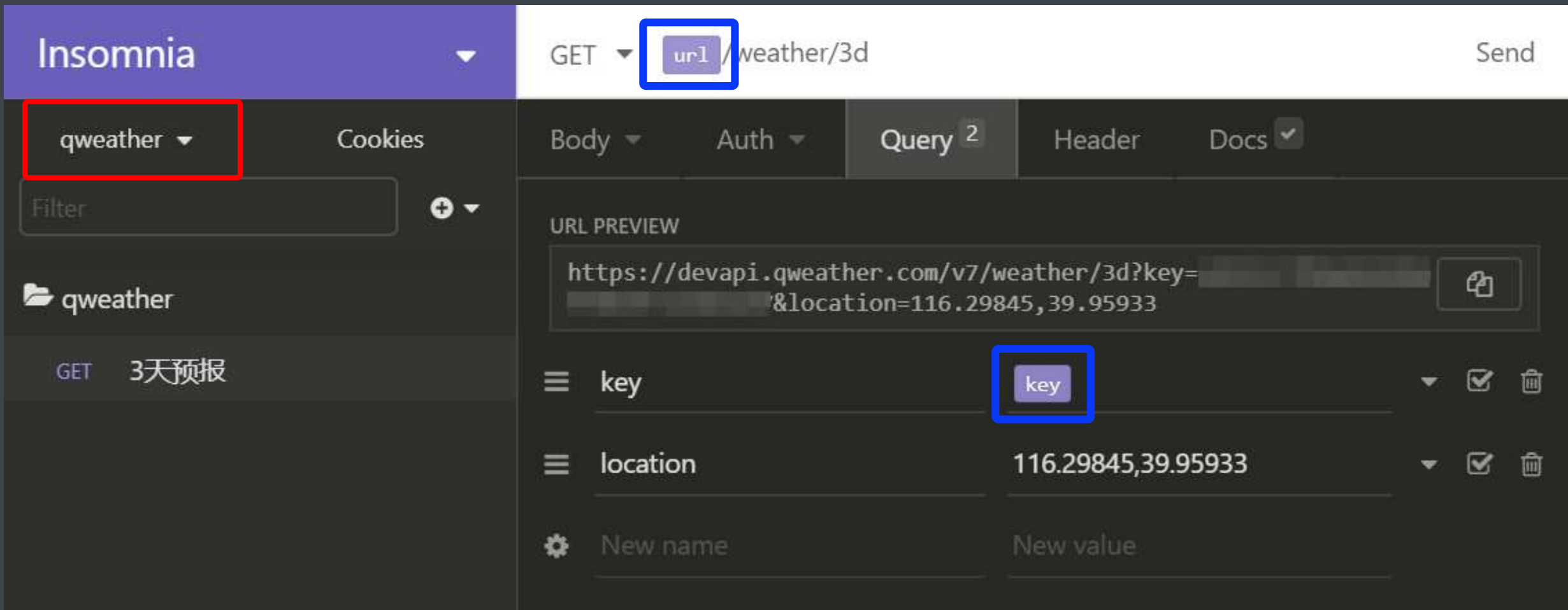


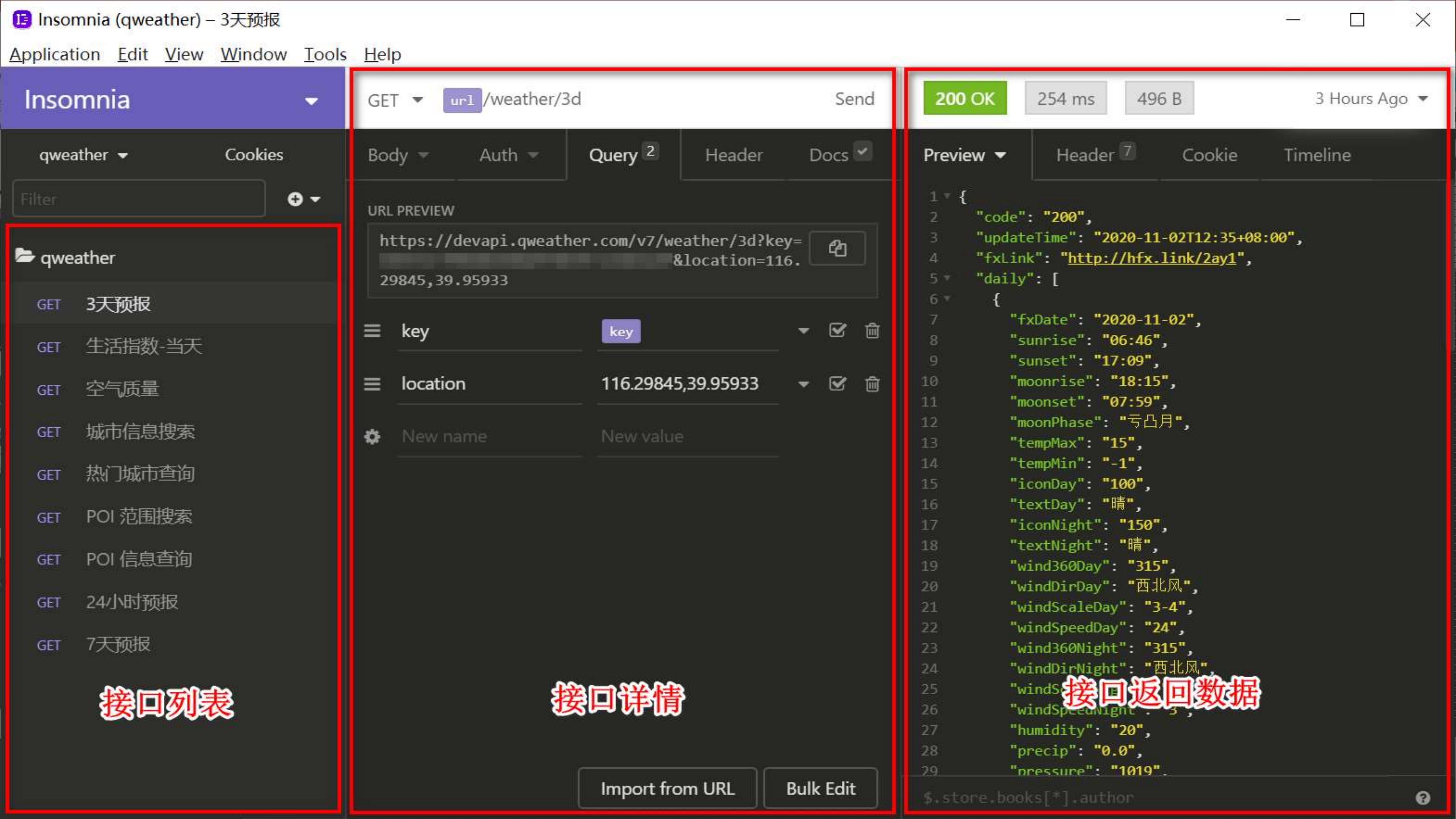
# insomnia

- 下载 (Insomnia Core)
  - <https://insomnia.rest/>
- 配置
  - 接口调用中使用的公共内容，应该提升为变量，例如：key
  - 不同的项目，具有不同的环境变量
- 使用
  - 请求方法 + 请求地址 + 请求参数



## Insomnia 使用环境变量





Insomnia

GET

url /weather/3d

Send

200 OK

254 ms

496 B

3 Hours Ago

qweather

Cookies

Filter

+ -

qweather

GET 3天预报

GET 生活指数-当天

GET 空气质量

GET 城市信息搜索

GET 热门城市查询

GET POI 范围搜索

GET POI 信息查询

GET 24小时预报

GET 7天预报

接口列表

Body

Auth

Query

Header

Docs

URL PREVIEW

https://devapi.qweather.com/v7/weather/3d?key=  
&location=116.  
29845,39.95933



key

key



location

116.29845,39.95933



New name

New value

Import from URL

Bulk Edit

Preview

Header

Cookie

Timeline

```
1 {  
2   "code": "200",  
3   "updateTime": "2020-11-02T12:35+08:00",  
4   "fxLink": "http://hfx.link/2ay1",  
5   "daily": [  
6     {  
7       "fxDate": "2020-11-02",  
8       "sunrise": "06:46",  
9       "sunset": "17:09",  
10      "moonrise": "18:15",  
11      "moonset": "07:59",  
12      "moonPhase": "亏凸月",  
13      "tempMax": "15",  
14      "tempMin": "-1",  
15      "iconDay": "100",  
16      "textDay": "晴",  
17      "iconNight": "150",  
18      "textNight": "晴",  
19      "wind360Day": "315",  
20      "windDirDay": "西北风",  
21      "windScaleDay": "3-4",  
22      "windSpeedDay": "24",  
23      "wind360Night": "315",  
24      "windDirNight": "西北风",  
25      "windScaleNight": "3-4",  
26      "windSpeedNight": "24",  
27      "humidity": "20",  
28      "precip": "0.0",  
29      "pressure": "1019"
```

接口返回数据

\$.store.books[\*].author



# 项目实践

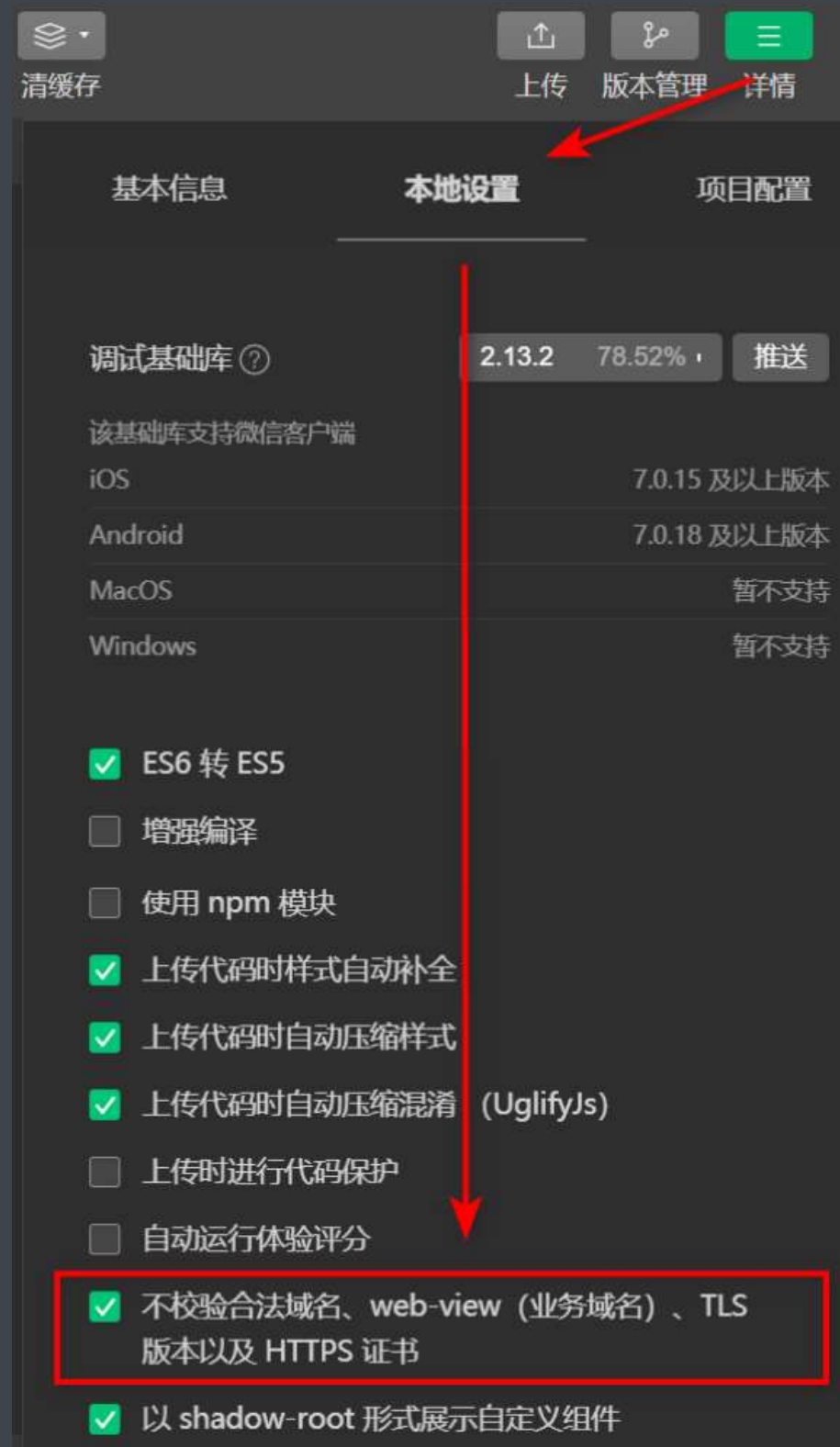
数据接口：小程序中调用接口

wx.request()

# 接口调用报错

- 报错内容
  - XXXXX 不在以下 request 合法域名列表中
- 解决方案
  - 开发阶段解决方案：不校验合法域名
  - 上线阶段解决方案：绑定请求接口域名

## 不校验合法域名



## 绑定请求接口域名

开发

运维中心

开发设置

开发者工具

接口设置

安全中心



服务器域名

服务器配置

说明

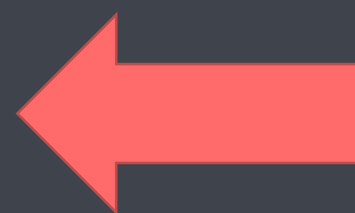
request合法域名

https://devapi.qweather.com



# 获取实时地理位置

- wx.getLocation()
  - <https://developers.weixin.qq.com/miniprogram/dev/api/location/wx.getLocation.html>
- 解决授权提示



```
"permission": {  
  "scope.userLocation": {  
    "desc": "尝试获取位置信息"  
  }  
},
```



# 项目实践

项目界面：官方组件 + 第三方组件

## 官方组件与第三方组件

官方组件

基础功能（HTML标签）

第三方组件

组合功能（Bootstrap）

# WeUI

- 下载

- <https://github.com/Tencent/weui-wxss>

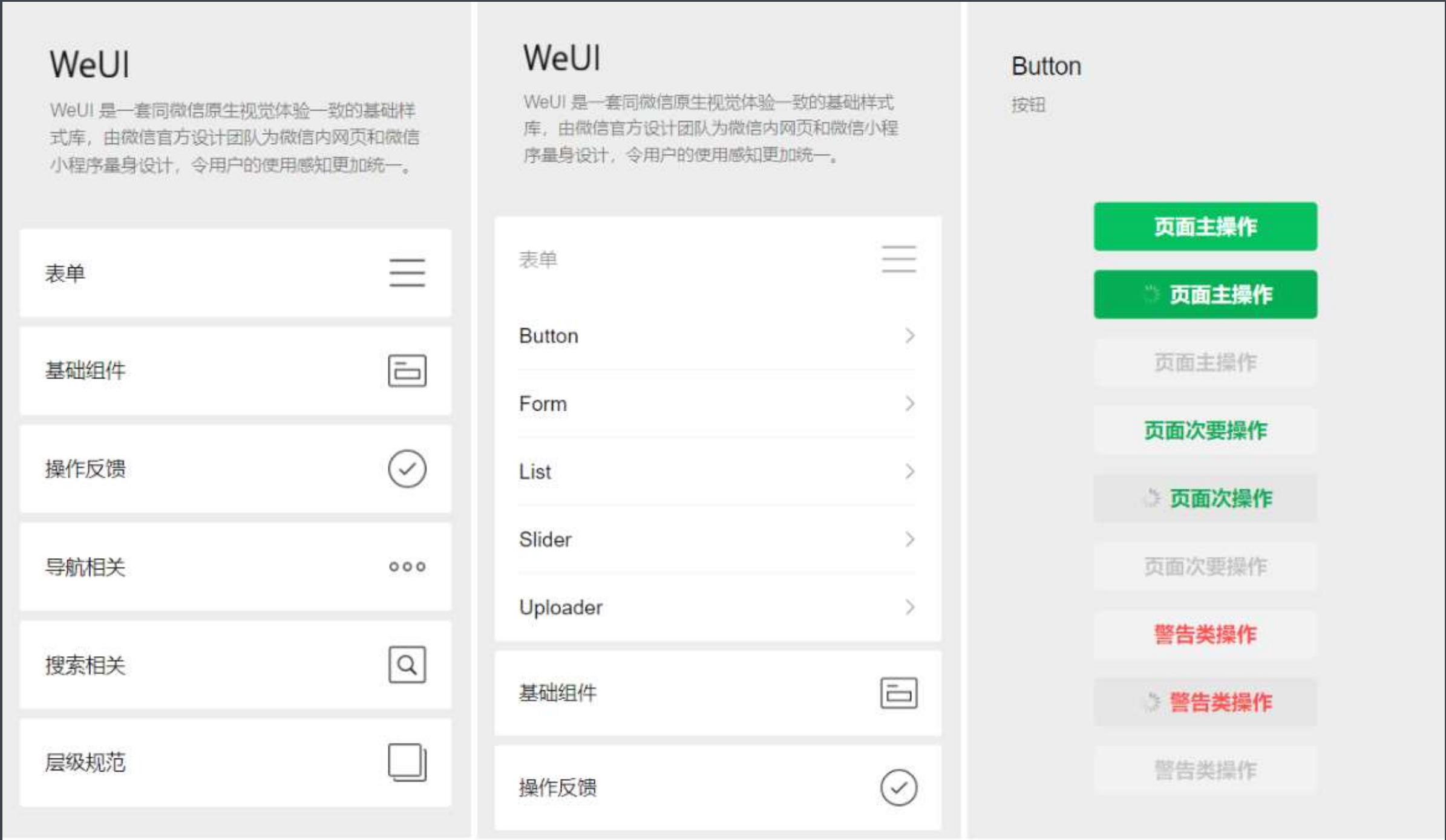
- 引入

- 解压后，找到 dist/style/ 目录下的 weui.wxss 文件
  - 将 weui.wxss 复制到小程序的根目录中
  - 在全局样式文件 app.wxss 中加入 @import "weui.wxss"

- 使用

- 通过微信开发者工具打开解压后的 dist 目录

WeUI - <https://weui.io/>



# 项目实践

功能组合：数据接口 + 项目界面

# 功能组合



# 项目实践

功能组合：模板 + 移动端 API

## WXML 的模板

声明模板

```
<template name="tplname">
  {{mydata}}
  ....
</template>
```



调用模板

```
<import src="path/mytpl.wxml"></import>
<template is="tplname" data="{{mydata}}"/>
```



## 小程序移动端的 API

去拍照

调用摄像头，并返回照片信息

扫一扫

识别二维码的内容

.....

.....

# 项目实践

发布小程序

# 发布小程序

## STEPS

1. 上传代码；
2. 提交审核；
3. 发布版本；

## 上传代码

### 开发版本

版本号

1.0.0

开发者

初六

提交时间

2020

项目备注

我的小程序代码

提交审核



管理->版本管理->开发版本

普通编译



编译



预览



真机调试



切后台



清缓存



上传



版本管理



详情

## 提交审核

### 审核版本

版本号

1.0.0

审核中

开发者

初六

提交审核时间

2020-

项目备注

我的小程序代码

详情



### 开发版本

版本号

1.0.0

开发者

初六

提交时间

2020-

项目备注

我的小程序代码

提交审核



## 提交审核

### 线上版本

版本号

1.0.1

发布者

初六

发布时间

2020

项目备注

添加工具类目

详情



### 审核版本

版本号

1.0.1

审核通过待发布

开发者

初六

提交审核时间

2020

项目备注

添加工具类目

提交发布



# 微信小程序总结

- 基础
  - 简介、环境搭建、初始化小程序
- 解读初始化小程序
  - 目录结构、app 代码、页面、生命周期、云 API、组件 .....
- 项目实践
  - 数据接口、项目界面、功能组合、发布

结束