

# 绿色小苹果的入门教程：从 0 开始创建

## 一：创建第一个小程序

### 1、创建或导入

< 返回

添加项目

AppID

填写小程序AppID

无 AppID

选择无 AppID

项目名称

项目目录  选择

随便填写

取消 添加项目

目前只有 200 个内测名额，所以选择无 AppID（注：此 AppID 非公众号或服务号 appid）

### 2、填写完成

AppID 无 AppID 部分功能受限

[返回填写小程序AppID](#)

项目名称 wxapp

项目目录 C:\...wxapp 选择

☒ 在当前目录中创建 quick start 项目

可选

取消 添加项目

quick start 选择此选择会默认创建一个 demo, 点击【添加项目】会创建默认的小程序。到此，小程序的创建完成，灰常简单。

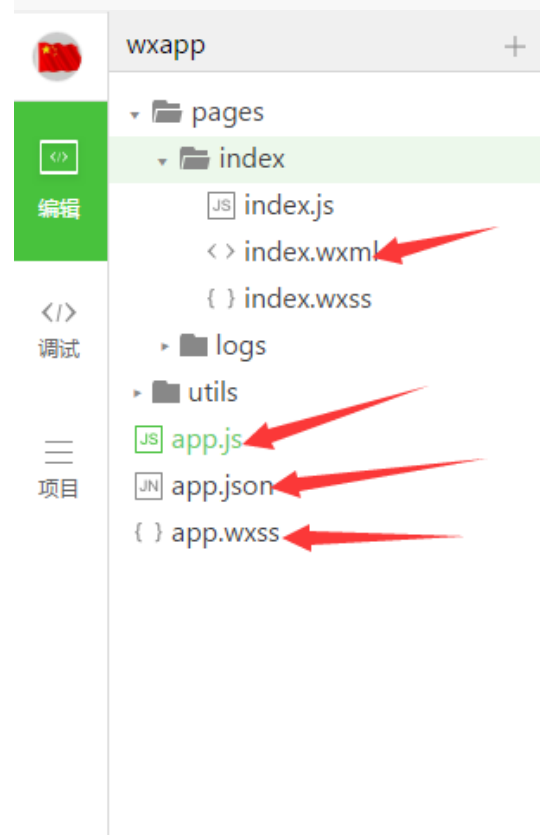
### 3)、quick 生成的小程序



## 二：每个文件的含义

了解一门新的语言一定要了解它的文件，比如 .java 是 java 文件 .txt 是文本文件 .html 是网页文件 .jpg 是图片文件。那么小程序中都有哪些文件呢。

以 quick start 程序为例，其他程序同理



每个小程序都包括这四种文件

**.wxml**：是网页文件（同 .html, MVC 中的 V）

**.js** : 是脚本文件 ( 可以理解成 java 中的.java 文件 , MVC 中的 C )

**.json** : 是配置文件 ( 类似 .xml 或者 .pom )

**.wxss** : 是样式表文件 ( 同 .css 文件 )

其中 , 每个页面**必须包含 .js 和 .wxml 文件**

### **三 : app.js 文件**

app.js 是小程序的脚本代码。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。

app.js ×

```
1 //app.js
2 App({
3   onLaunch: function () {
4     //调用API从本地缓存中获取数据
5     var logs = wx.getStorageSync('logs') || []
6     logs.unshift(Date.now())
7     wx.setStorageSync('logs', logs)
8   },
9   getUserInfo:function(cb){
10    var that = this
11    if(this.globalData.userInfo){
12      typeof cb == "function" && cb(this.globalData.userInfo)
13    }else{
14      //调用登录接口
15      wx.login({
16        success: function () {
17          wx.getUserInfo({
18            success: function (res) {
19              that.globalData.userInfo = res.userInfo
20              typeof cb == "function" && cb(that.globalData.userInfo)
21            }
22          })
23        }
24      })
25    }
26  },
27  globalData:{
28    userInfo:null
29  }
30 })
```

跟标准js基本没区别

配置全局变量

[点击查看详情](#)

#### 四：app.json 文件

app.json 是对整个小程序的全局配置。我们可以在这个文件中配置小程序是由哪些页面组成，配置小程序的窗口背景色，配置

导航条样式，配置默认标题。注意该文件不可添加任何注释（#### 不知道在搞什么，竟然不能添加注释）

app.json ×

```
1  {
2    "pages": [
3      "pages/index/index",
4      "pages/logs/logs"
5    ],
6    "window": {
7      "backgroundTextStyle": "light",
8      "navigationBarBackgroundColor": "#fff",
9      "navigationBarTitleText": "WeChat",
10     "navigationBarTextStyle": "black"
11   }
12 }
13
```

配置页面，所有页面必须配置在pages中，无需添加后缀

配置窗口

[点击查看详情](#)

## 五：app.wxss 文件

app.wxss 是整个小程序的公共样式表。我们可以在页面组件的 class 属性上直接使用 app.wxss 中声明的样式规则。如果页面有自己的样式表，则会覆盖公共样式表

用法跟标准 css 文件无异，支持 .class #id。

app.wxss ×

```
1  /**app.wxss**/  
2  .container {  
3      height: 100%;  
4      display: flex;  
5      flex-direction: column;  
6      align-items: center;  
7      justify-content: space-between;  
8      padding: 200rpx 0;  
9      box-sizing: border-box;  
10 }  
11
```

[点击查看详情](#)

## 六 : .wxml

页面结构文件，用来显示页面。但结构不是标准 html 的结构，如

```
<html>  
  
  <head>  
  
    <title>title</title>  
  
  </head>  
  
  <body>  
  
    this is a wxapp  
  
  </body>
```



```
</html>
```

这样的标签都是错误的，小程序不支持。微信有自己的一套组件

index.wxml ×

```
1  <!--index.wxml-->
2  <view class="container">
3    <view bindtap="bindViewTap" class="userinfo">
4      <image class="userinfo-avatar" src="{{userInfo.avatarUrl}}" background-size="cover"></image>
5      <text class="userinfo-nickname">{{userInfo.nickName}}</text>
6    </view>
7    <view class="usermotto">
8      <text class="user-motto">{{motto}}</text>
9    </view>
10 </view>
11
```