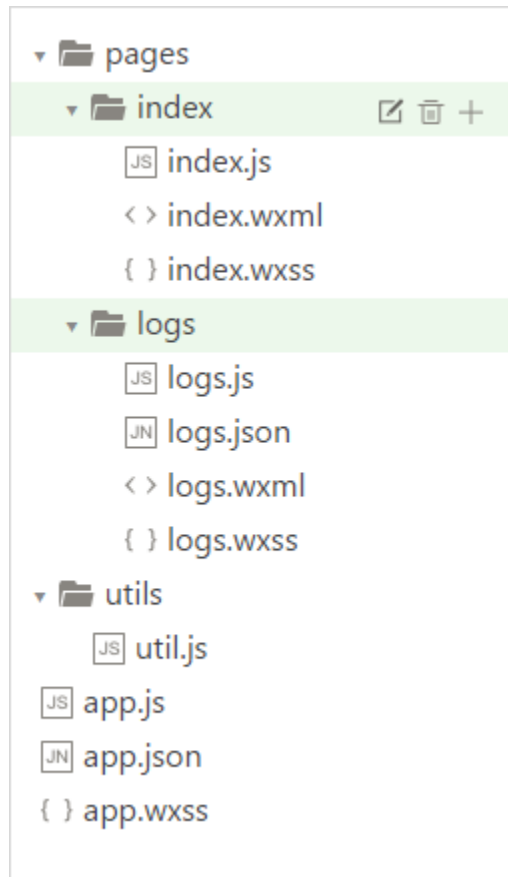


梦断难寻新手入门篇《一》：熟悉项目结构

微信小程序创建之后会生成一个项目模板，如下图所示（基本如此，但并不局限于此）



现在分别来说明一下每个文件及目录的意思

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 })
```

程序的入口文件，必须存在。

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。调用框架提供的丰富的 API，如本例的同步存储及同步读取本地数据。

app.json

```
{
  "pages": [
    "pages/index/index",
    "pages/logs/logs"
  ],
  "window": {
    "backgroundTextStyle": "light",
    "navigationBarBackgroundColor": "#fff",
    "navigationBarTitleText": "WeChat",
    "navigationBarTextStyle": "black"
  }
}
```

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

app.json

```
/**app.wxss**/
.container {
  height: 100%;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: space-between;
  padding: 200px 0;
  box-sizing: border-box;
}
```

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。

Pages（开发者自行创建）目录是你的页面目录，但不是一定需要叫 Pages

Utils（开发者自行创建）在本例中是工具类库目录，不是一定需要存在

在开发过程中最好遵循以上目录结构，特别 Pages，每个页面独立一个文件夹，因为每个页面有自己独立的样式或者 Js。未来方便管理。但是，如果有公共样式的存在，请写在 **app.wxss**