

微信小程序基础Day01

今日目标:

- 能够知道如何创建小程序项目
- 能够清楚小程序项目的基本组成结构
- 能够知道小程序页面由几部分组成
- 能够知道小程序中常见的组件如何使用
- 能够知道小程序如何进行协同开发和发布

讲解目录:

- 1.小程序简介
- 2.第一个小程序
- 3.小程序代码的构成
- 4.小程序的宿主环境
- 5.协同工作和发布
- 6.总结

1.小程序简介

1.1.小程序与普通网页开发的区别

- 运行环境的不同
 - 网页运行在浏览器
 - 小程序运行在微信环境
- API 不同
 - 小程序无法调用 DOM 和 BOM 的 API
 - 但是小程序可以调用微信环境提供一些 API
 - 地理定位
 - 扫码
 - 支付
 -
- 开发模式不同
 - 网页的开发模式: 浏览器+代码编辑器
 - 小程序开发模式
 - 申请小程序开发账号
 - 安装小程序的开发者工具
 - 创建和配置小程序项目

1.2.体验微信小程序



2.第一个小程序

2.1.注册微信开发者账号, 获取 AppID

- 使用浏览器打开 <https://mp.weixin.qq.com/> 网址, 点击右上角的“立即注册”, 进入小程序开发账号注册流程



- 选择注册账号的类型

注册

请选择注册的帐号类型



订阅号

具有信息发布与传播的能力
适合个人及媒体注册



服务号

具有用户管理与提供业务服务的能力
适合企业及组织注册



小程序

具有出色的体验，可以被便捷地获取与传播
适合有服务内容的企业和组织注册



企业微信
原企业号

具有实现企业内部沟通与协同管理的能力
适合企业客户注册

• 填写账号信息

小程序注册

① 帐号信息 — ② 邮箱激活 — ③ 信息登记

每个邮箱仅能申请一个小程序

已有微信小程序？[立即登录](#)

邮箱

作为登录帐号，请填写未被微信公众平台注册，未被微信开放平台注册，未被个人微信号绑定的邮箱

密码

字母、数字或者英文符号，最短8位，区分大小写

确认密码

请再次输入密码

验证码  [换一张](#)

☒ 你已阅读并同意《微信公众平台服务协议》及《微信小程序平台服务条款》

[注册](#)

• 提示邮箱激活

小程序注册

① 帐号信息 — ② 邮箱激活 — ③ 信息登记



激活小程序帐号

感谢注册！确认邮件已发送至你的注册邮箱：liu@itcast.cn。请进入邮箱查看邮件，并激活小程序帐号。

没有收到邮件？

1. 请检查邮箱地址是否正确，你可以[返回重新填写](#)。
2. 检查你的邮件垃圾箱
3. 若仍未收到确认，请尝试[重新发送](#)

• 点击链接激活账号



• 选择主体类型

① 帐号信息

② 邮箱激活

③ 信息登记

用户信息登记

微信公众平台致力于打造真实、合法、有效的互联网平台。为了更好的保障你和广大微信用户的合法权益, 请你认真填写以下登记信息。
为表述方便, 本服务中, “用户”也称为“开发者”或“你”。

用户信息登记审核通过后:
1. 你可以依法享有本微信公众平台帐号所产生的权利和收益;
2. 你将对本微信公众平台帐号的所有行为承担全部责任;
3. 你的注册信息将在法律允许的范围内向微信用户展示;
4. 人民法院、检察院、公安机关等有权机关可向腾讯依法调取你的注册信息等。

请确认你的微信公众平台主体类型属于政府、媒体、企业、其他组织、个人, 并请按照对应的类别进行信息登记。
点击查看微信公众平台信息登记指引。

1

注册国家/地区

中国大陆

2

主体类型

如何选择主体类型?

个人

企业

政府

媒体

其他组织

继续

• 主体信息登记

主体类型 如何选择主体类型?

个人 企业 政府 媒体 其他组织

个人类型包括：由自然人注册和运营的公众帐号。
帐号能力：个人类型暂不支持微信认证、微信支付及高级接口能力。

主体信息登记

身份证姓名

信息审核成功后身份证姓名不可修改；如果名字包含分隔号“”，请勿省略。

身份证号码

请输入您的身份证号码。一个身份证号码只能注册5个小程序。

管理员手机号码 获取验证码

请输入您的手机号码。一个手机号码只能注册5个小程序。

短信验证码 无法接收验证码?

请输入手机短信收到的6位验证码

管理员身份验证

 请用管理员本人微信扫描二维码。本验证方式不扣除任何费用。注册后，扫码的微信号将成为该帐号的管理员微信号。

继续

- 获取小程序的 AppID

小程序

文档 社区 工具

开发

开发设置

开发者ID

开发者ID 操作

AppID(小程序ID) wx5eda4566d93729d6 创建小程序项目的时候，需要用到此 AppID

AppSecret(小程序密钥) 重置

小程序代码上传 开发者可基于配置信息调用微信开发者工具提供的代码上传模块。查看详情

配置信息 操作

开发

成长

2.2.微信开发者工具的安装和使用

2.2.1.安装

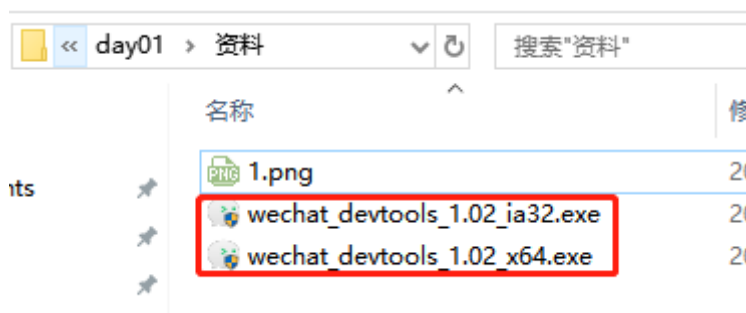
- 下载最新版本

1 | <https://developers.weixin.qq.com/miniprogram/dev/devtools/stable.html>



或者使用资料文件夹中的安装文件

在第一天的资料文件夹中, 有对应系统的安装文件, 如下



- 安装

一路点击下一步即可完成安装

2.2.2.使用

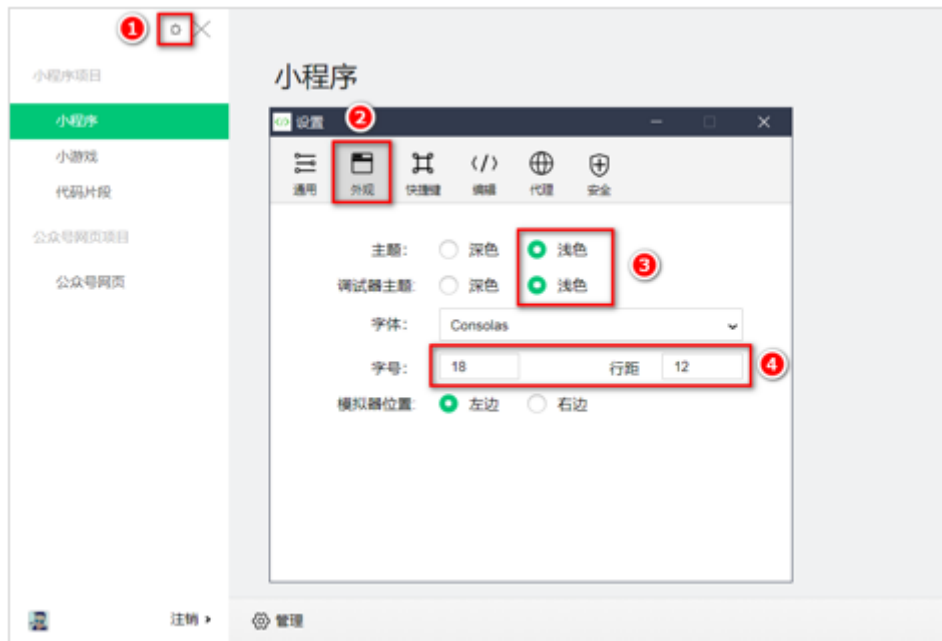
- 扫码登录

在完成安装安装之后, 将会打开微信开发者工具, 此时会提示扫码登录, 需使用自己的微信进行扫码

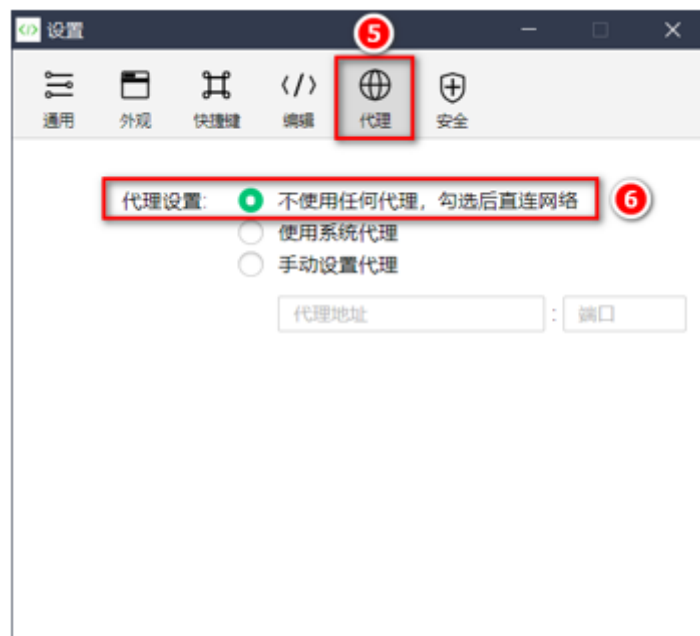


- 工具的设置

- 设置外观



- 设置代理



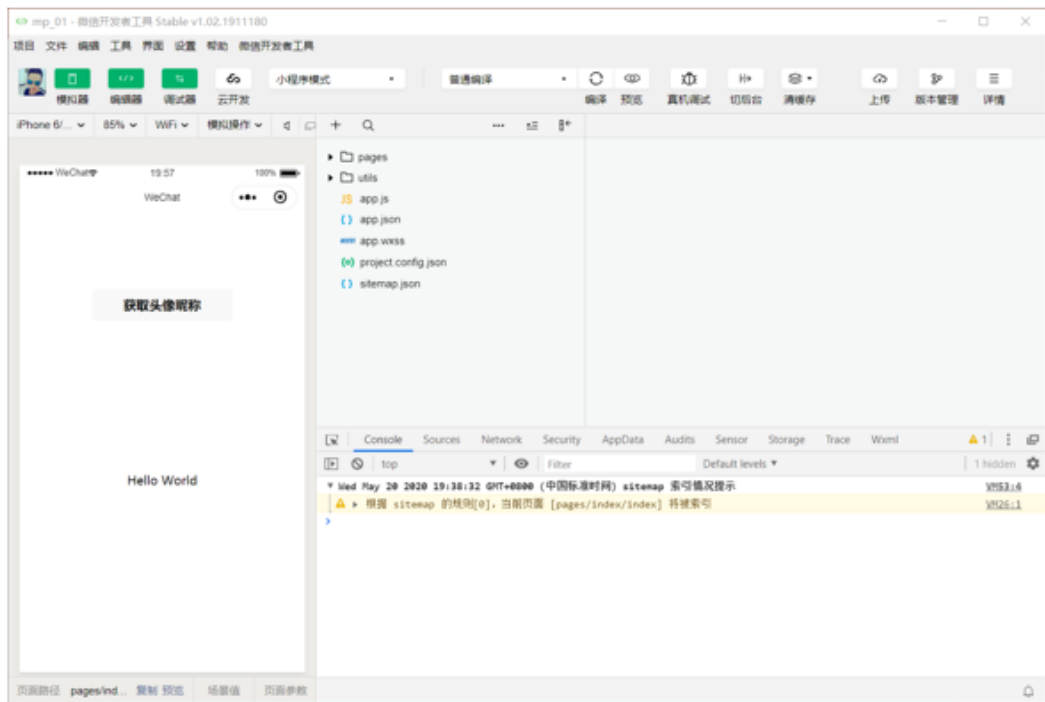
- 项目的创建
 - 点击“加号”按钮



- 填写项目信息



- 项目创建完成



- 微信开发者工具的介绍
 - 菜单栏
 - 帮助
 - 开发者文档
 - 设置
 - 通用设置
 - 工具
 - 编译
 - 刷新
 - 插件
 - 构建 npm
 - 工具栏
 - 模拟器/编辑器/调试器，点击切换显示隐藏
 - 编译/预览按钮
 - 详情-项目详情
 - 模拟器
 - 机型：建议 iPhone6/7/8
 - 缩放85%
 - 底部页面路径/页面参数/场景值
 - 代码编辑器
 - 调试区

3.小程序代码的构成

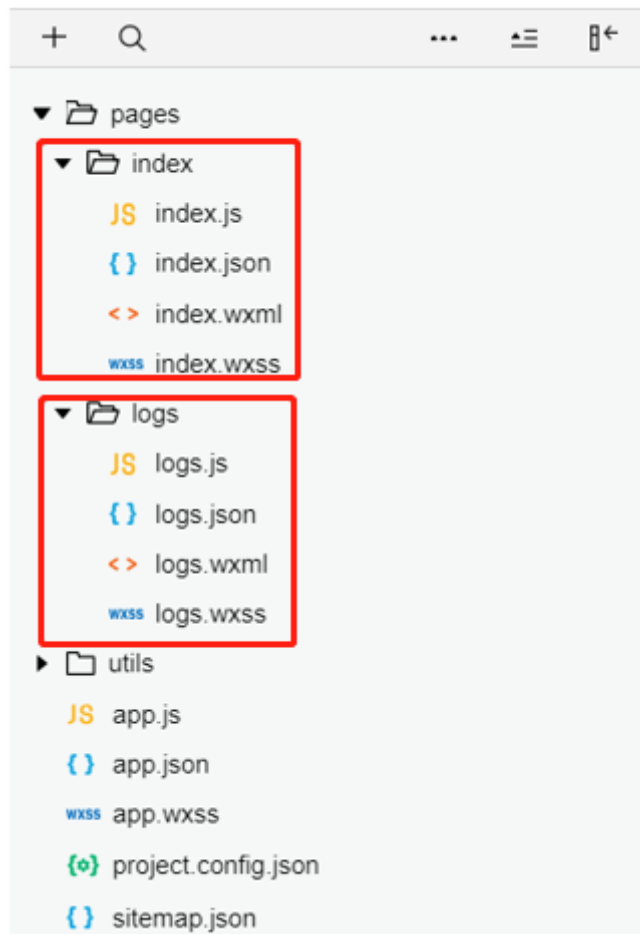
3.1.项目的基本组成结构

1. `pages` 用来存放所有小程序的页面，每个页面都是一个单独的文件夹
2. `utils` 用来存放工具性质的模块
3. `app.js` 整个小程序项目的入口文件
4. `app.json` 小程序项目的全局配置文件
5. `app.wxss` 小程序项目的全局样式文件

6. `project.config.json` 项目的配置文件
7. `sitemap.json` 用来配置小程序及其页面是否允许被微信索引

3.2. 小程序页面的组成部分

小程序官方建议把所有小程序的页面，都存放在 `pages` 目录中，以单独的文件夹存在，如图所示：



每个页面由 4 个基本文件组成，它们分别是：`.js` 文件（页面的脚本文件，存放页面的数据、事件处理函数等）`.json` 文件（当前页面的配置文件，配置窗口的外观、表现等）`.wxml` 文件（页面的模板结构文件）`.wxss` 文件（当前页面的样式表文件）

总结而言,就是一个页面的结构,样式,功能和配置都被分别放到了单独的文件夹中进行维护

3.3. JSON 配置文件

`JSON` 是一种数据格式，在实际开发中，`JSON` 总是以配置文件的形式出现。小程序项目中也不例外：通过不同的 `.json` 配置文件，可以对小程序项目进行不同级别的配置。小程序项目中有 4 种 `json` 配置文件，分别是：

- 项目根目录的 `app.json` 配置文件
- 项目根目录种的 `project.config.json` 配置文件
- 项目根目录中的 `sitemap.json` 配置文件
- 每个页面文件夹中的 `.json` 配置文件

3.3.1. `app.json` 配置文件

`app.json` 是当前小程序的**全局配置**，包括了小程序的所有页面路径、窗口外观、界面表现、底部 `tab` 等。Demo 项目里边的 `app.json` 配置内容如下：

```

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   "style": "v2",
13   "sitemapLocation": "sitemap.json"
14 }

```

简单了解下这 4 个配置项的作用：`pages`：用来记录当前小程序所有页面的路径 `window`：全局定义小程序所有页面的背景色、文字颜色等 `style`：全局定义小程序组件所使用的样式版本 `sitemapLocation`：用来指明 `sitemap.json` 的位置

目前大家先对 `pages` 和 `window` 节点多多关注, 后续会不断的在这两个节点中进行配置

3.3.2. project.config.json 配置文件

`project.config.json` 是项目配置文件，用来记录我们对小程序开发工具所做的个性化配置

- `setting`
保存了编译相关的配置
- `projectname`
保存的是项目名称
- `appid`
保存的是小程序的账号 ID

同学们这里需要将 `project.config.json` 和 `app.json` 配置文件的作用区分开来,

`project.config.json` 更多的是对微信开发者工具的配置, 而 `app.json` 是对项目代码功能层面上的配置.

3.3.3. sitemap.json 配置文件

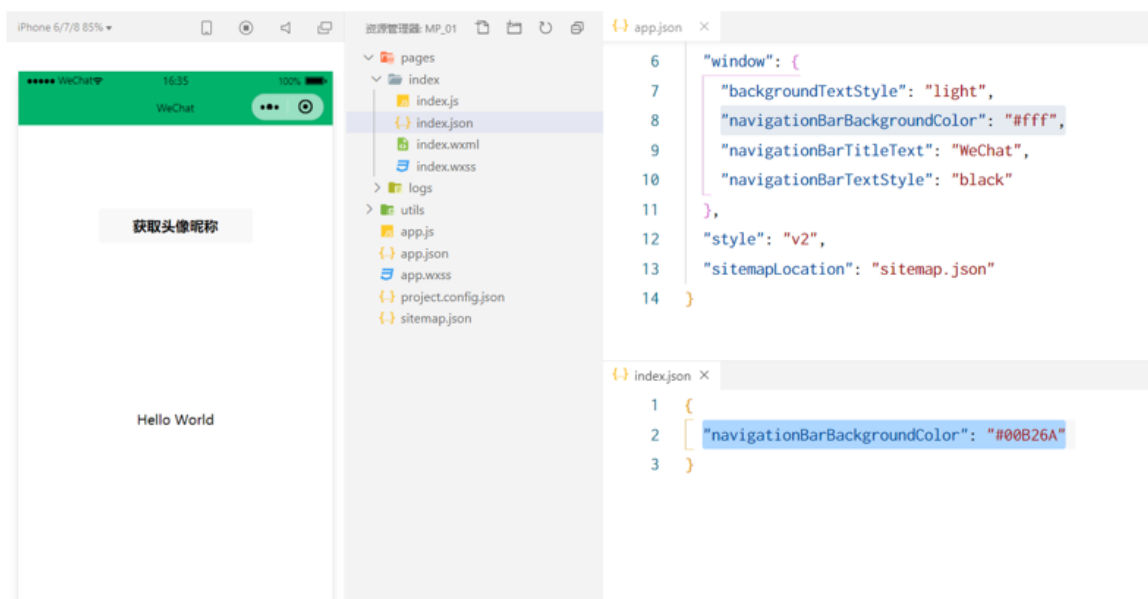
- 微信现已开放小程序内搜索，效果类似于 PC 网页的 `SEO`。`sitemap.json` 文件用来配置小程序页面是否允许微信索引。
- 当开发者允许微信索引时，微信会通过爬虫的形式，为小程序的页面内容建立索引。当用户的搜索关键字和页面的索引匹配成功的时候，小程序的页面将可能展示在搜索结果中。



注意：`sitemap` 的索引提示是默认开启的，如需要关闭 `sitemap` 的索引提示，可在小程序项目配置文件 `project.config.json` 的 `setting` 中配置字段 `checkSiteMap` 为 `false`

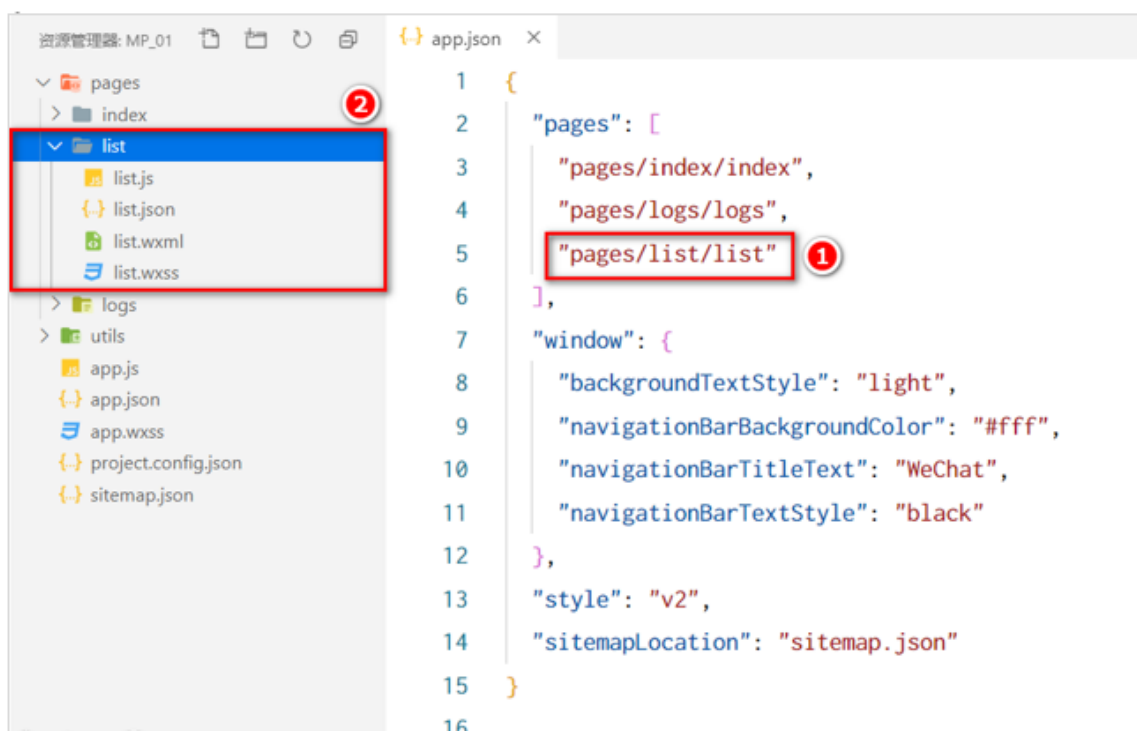
3.3.4.页面的 .json 配置文件

小程序中的每一个页面，可以使用 .json 文件来对本页面的窗口外观进行配置，**页面中的配置项会覆盖 app.json 的 window 中相同的配置项**，当然，这指的是当前页面被展示出来的时候。



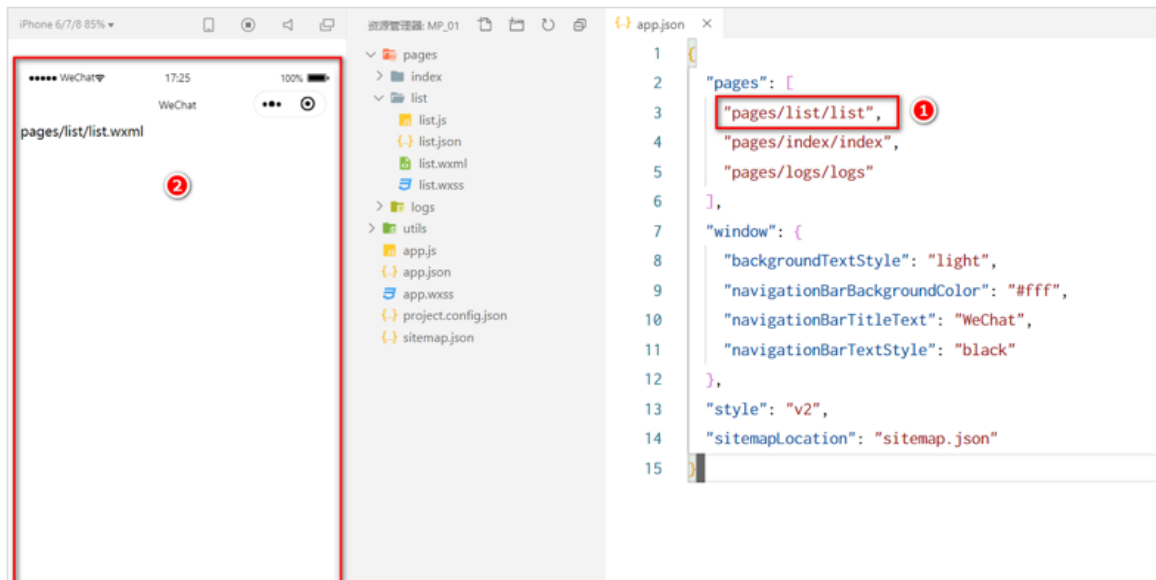
3.4.新建小程序页面

在 app.json->pages 中新增页面的存放路径，小程序开发中工具就可帮我们自动创建对应的页面文件



3.5.修改项目首页

只需要调整 app.json -> pages 数组中页面路径的前后顺序，即可修改项目的首页。小程序会把排在第一位的页面，当作项目首页进行渲染。



3.6. WXML

WXML 是小程序框架的标签语言,用来构建小程序页面的结构,其作用类似于网页开发中的 HTML。他们主要存在下面几点的差别

1. 标签名不同

1. HTML (div)
2. WXML(view, text, image, navigator)

2. 属性节点不同

1. ` `
2. `<navigator url=""></navigator>`

3. 提供了类似于 vue 中的模板语法

1. 数据绑定
2. 列表渲染
3. 条件渲染

3.7. WXSS

WXSS (Weixin Style Sheets) 是一套样式语言,用于描述 WXML 的组件样式,类似于网页开发中的 CSS。他们主要存在下面几点的差别

① 新增了 rpx 尺寸单位

- CSS 中需要手动进行像素单位换算,例如 rem
- WXSS 在底层支持新的尺寸单位 rpx,在不同大小的屏幕上小程序会自动进行换算

② 提供了全局的样式和局部样式

- 项目根目录中的 app.wxss 会作用于所有小程序页面
- 局部页面的 .wxss 样式仅对当前页面生效

③ WXSS 仅支持部分 CSS 选择器

- .class 和 #id
- element
- 并集选择器、后代选择器
- ::after 和 ::before 等伪类选择器

3.8.js 文件的分类

小程序中的 JS 文件分为三大类，分别是：

① app.js

- 是整个小程序项目的入口文件，通过调用 App() 函数来启动整个小程序

② 页面的 .js 文件

- 是页面的入口文件，通过调用 Page() 函数来创建并运行页面

③ 普通的 .js 文件

- 是普通的功能模块文件，用来封装公共的函数或属性供页面使用

4.小程序的宿主环境

4.1.一些概念

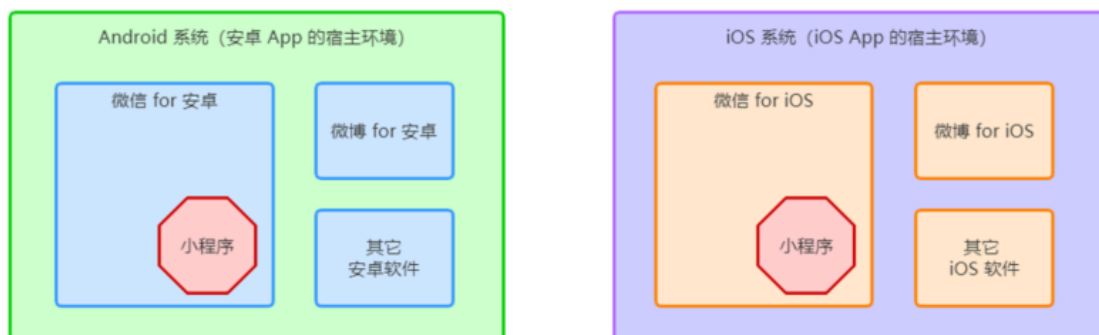
- 宿主环境

宿主环境 (host environment) 指的是程序运行所**必须的依赖环境**。例如：

Android 系统和 iOS 系统是两个不同的宿主环境。安卓版的微信 App 是不能在 iOS 环境下运行的，所以，

Android 是安卓软件的宿主环境，脱离了宿主环境的软件是没有任何意义的！

手机微信是小程序的宿主环境，如图所示：



小程序借助宿主环境提供的能力，可以完成许多普通网页无法完成的功能，例如：

微信扫码、微信支付、微信登录、地理定位、etc...

小程序的宿主环境包含以下几部分内容

① 通信模型

② 运行机制

③ 组件

④ API

- 通信模型

小程序中通信的主体是渲染层和逻辑层，其中：

① WXML 模板和 WXSS 样式工作在渲染层

② JS 脚本工作在逻辑层



而所谓的通信模型指的是:

- ① 渲染层和逻辑层之间的通信
- ② 逻辑层和第三方服务器之间的通信

这两部分的通信都是通过微信客户端进行转发



• 运行机制

小程序的运行机制主要讲两部分

① 小程序启动的过程

- 把小程序的代码下载到本地
- 解析 `app.json` 全局配置文件
- 执行 `app.js` 小程序入口文件, 调用 `App()` 创建小程序实例
- 渲染小程序首页
- 小程序启动完成

② 某个页面渲染的过程

- 加载解析页面的 `.json` 配置文件
- 加载页面的 `.wxml` 模板和 `.wxss` 样式
- 执行页面的 `.js` 文件, 调用 `Page()` 创建页面实例
- 页面渲染完成

4.2.组件

4.2.1.常用的组件

小程序中的组件也是由宿主环境提供的, 开发者可以基于组件快速搭建出漂亮的页面结构。官方把小程序的组

件分为了 9 大类, 分别是:

① 视图容器

② 基础内容

③ 表单组件

④ 导航组件

⑤ 媒体组件

⑥ map 地图组件

⑦ canvas 画布组件

⑧ 开放能力

⑨ 无障碍访问

我们在课程中主要涉及到标红的4中组件, 其中, 对于视图容器组件而言, 常用的有以下3个

- `view`

普通视图区域

类似于 HTML 中的 `div`, 是一个块级元素

常用来实现页面的布局效果

- `scroll-view`

可滚动的视图区域

常用来实现滚动列表效果

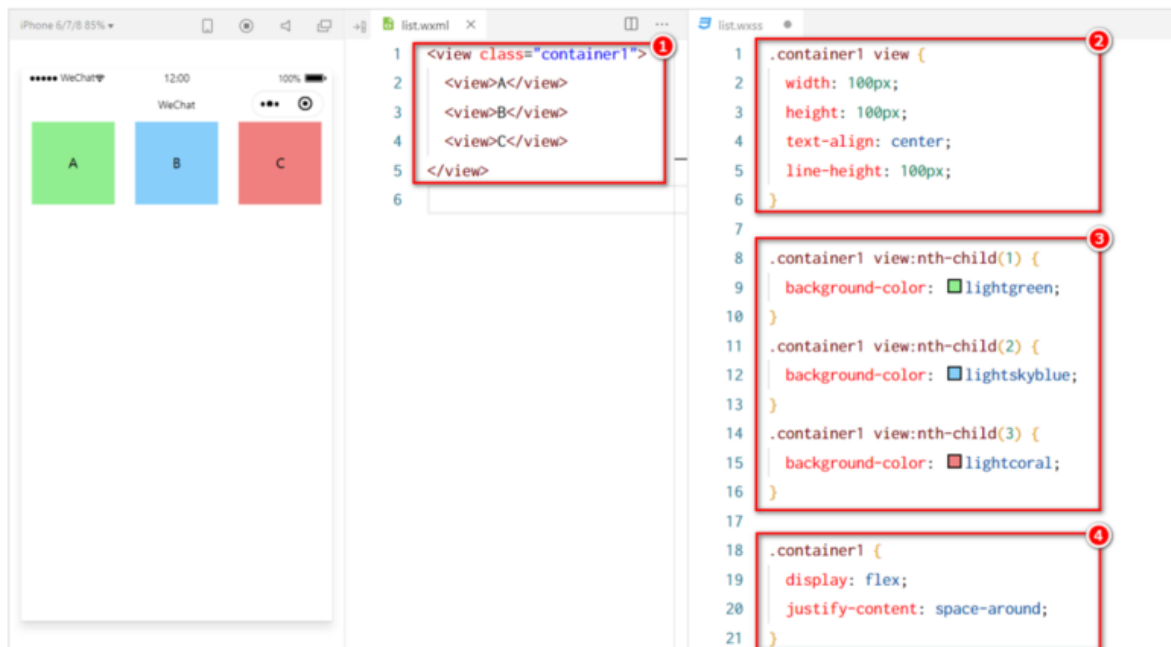
- `swiper` 和 `swiper-item`

轮播图容器组件 和 轮播图 `item` 组件

接下来我们就来学习一下这3个视图容器组件的使用.

4.2.2. `view` 组件的使用

实现如图的 `flex` 横向布局效果



代码如下:

```
1 <!--pages/list/list.wxml-->
2 <view class="container1">
3   <view>A</view>
4   <view>B</view>
5   <view>C</view>
6 </view>
```

```
1 /* pages/list/list.wxss */
2 .container1 {
3   display: flex;
4   justify-content: space-around;
5   align-items: flex-start;
6 }
7 .container1 view {
8   width: 100px;
9   height: 100px;
10  line-height: 100px;
11  text-align: center;
12  background-color: yellowgreen;
13  color: #fff;
14 }
```

4.2.3. scroll-view 组件的使用

scroll-view 主要来实现元素内部的滚动, 注意, 元素内部的滚动和整个页面的滚动不是同一回事。

代码如下:

```
1 <!-- scroll-y 允许纵向滚动 -->
2 <!-- scroll-x 允许横向滚动 -->
3 <scroll-view class="container1" scroll-y>
4   <view>A</view>
5   <view>B</view>
6   <view>C</view>
7 </scroll-view>
```

```

1  /* pages/list/list.wxss */
2  .container1 {
3      /* display: flex;
4      justify-content: space-around;
5      align-items: flex-start; */
6      border: 2px dashed lightgreen;
7      /* 给父盒子添加固定高度，内部的元素就可以滚动 */
8      width: 100px;
9      height: 130px;
10 }
11 .container1 view {
12     width: 100px;
13     height: 100px;
14     line-height: 100px;
15     text-align: center;
16     color: #fff;
17 }
18 .container1 view:nth-child(1) {
19     background-color: yellowgreen;
20 }
21 .container1 view:nth-child(2) {
22     background-color: lightskyblue;
23 }
24 .container1 view:nth-child(3) {
25     background-color: lightcoral;
26 }

```

4.2.4. swiper 组件的使用

swiper 组件主要是用来实现轮播图效果，swiper 组件相当于一个大盒子，swiper-item 指的是轮播图的每一项

代码如下：

```

1  <!-- 轮播图的结构 -->
2  <swiper class="swiper-container">
3      <!-- 第一个轮播图 -->
4      <swiper-item>
5          <view class="item">A</view>
6      </swiper-item>
7      <!-- 第二个轮播图 -->
8      <swiper-item>
9          <view class="item">B</view>
10     </swiper-item>
11     <!-- 第三个轮播图 -->
12     <swiper-item>
13         <view class="item">C</view>
14     </swiper-item>
15 </swiper>

```

```

1  .swiper-container {
2      height: 150px;
3  }
4  .item {
5      height: 100%;
6      line-height: 150px;

```

```

7   text-align: center;
8   }
9   swiper-item:nth-child(1) .item {
10    background-color: lightcoral;
11   }
12  swiper-item:nth-child(2) .item {
13    background-color: lightseagreen;
14   }
15  swiper-item:nth-child(3) .item {
16    background-color: lightpink;
17   }

```

swiper 还有其他相应的属性, 如下:

- indicator-dots 是否显示面板指示点
- indicator-color 指示点颜色
- indicator-active-color 激活指示点颜色
- autoplay 是否自动切换
- interval 自动切换时间间隔
- circular 是否采用衔接滑动

4.2.5. text 组件和 rich-text 组件

- text 组件

文本组件

类似于 HTML 中的 span 标签, 是一个行内元素

- rich-text 组件

富文本组件 支持把 HTML 字符串渲染为 WXML 结构

通过 text 组件的 selectable 属性, 实现长按选中文本内容的效果

```

1  <!-- 只有在text组件中添加selectable 属性支持长按选中效果 -->
2  <view>
3    <text selectable>手机号码支持长按选中效果: 13333333333</text>
4  </view>

```

注意: 在苹果 iOS 系统设备上, 有可能是长按是选中不了文字, 此时需要改变一下调试基础库的版本, 具体如下:



通过 rich-text 组件的 nodes 属性节点, 把 HTML 字符串渲染为对应的 UI 结构

```
1 <!-- rich-text组件中添加 nodes节点可以把 HTML 字符串渲染为对应的 UI 结构 -->
2 <rich-text nodes="<h1 style='color: red;'>标题</h1>">
3 </rich-text>
```

注意: 小程序默认是识别不了 HTML 的, 小程序默认只识别 WXML, 所以如果我们从服务器上获取到了包含 HTML 字符串的数据, 并且想展示出来, 应该使用到 rich-text 组件的 nodes 属性

4.2.6. button 组件

button 是按钮组件, 它的功能比 HTML 中的 button 按钮丰富, 我们可以通过 open-type 属性可以调用微信提供的各种功能 (客服、转发、获取用户授权、获取用户信息等, 关于 open-type 的使用在后续的讲解课程中会用到).

基本使用的代码如下:

```
1 <!-- 通过type属性指定按钮颜色类型 -->
2 <button>普通按钮</button>
3 <button type="primary">主色调按钮</button>
4 <button type="warn">警告按钮</button>
5 <!-- 通过size属性指定按钮尺寸 -->
6 <button size="mini">普通按钮</button>
7 <button size="mini" type="primary">主色调按钮</button>
8 <button size="mini" type="warn">警告按钮</button>
9 <!-- 通过plain属性指定按钮的镂空效果 -->
10 <button plain>普通按钮</button>
11 <button plain type="primary">主色调按钮</button>
12 <button plain type="warn">警告按钮</button>
```

4.2.7. image 组件

image 是图片组件 (注意, 不是 img 哦, 是 image), 用来展示图片的, 它也是通过 src 属性来指明图片的路径.

`image` 组件默认宽度约 300px、高度约 240px

关于 `image` 组件, 还有一个属性叫做 `mode`, 它是用来控制图片的裁剪和缩放形式, 具体的可选值如下:

mode值	说明
<code>scaleToFill</code>	(默认值)缩放模式, 不保持纵横比缩放图片, 使图片的宽高完全拉伸至填满 <code>image</code> 元素
<code>aspectFit</code>	缩放模式, 保持纵横比缩放图片, 使图片的长边能完全显示出来可以完整的将图片显示出来
<code>aspectFill</code>	缩放模式, 保持纵横比缩放图片, 只保证图片的短边能完全显示出来图片通常旨在水平或垂直方向是完整的, 另一个方向将发生截取
<code>widthFix</code>	缩放模式, 宽度不变, 高度自动变化, 保持原图宽高比不变
<code>heightFix</code>	缩放模式, 高度不变, 宽度自动变化, 保持原图宽高比不变

4.3. 小程序 API 的3大分类

小程序中的 `API` 是由宿主环境提供的, 通过这些丰富的小程序 `API`, 开发者可以方便的调用微信提供的能力,

例如: 获取用户信息、本地存储、支付功能等。

小程序 `API` 分为3大类, 分别是:

① 事件监听 API

- 特点: 以 `on` 开头, 用来监听某些事件的触发
- 举例: `wx.onWindowResize(function callback)` 监听窗口尺寸变化的事件

② 同步 API

- 特点1: 以 `Sync` 结尾的 `API` 都是同步 `API`
- 特点2: 同步 `API` 的执行结果, 可以通过函数返回值直接获取, 如果执行出错会抛出异常
- 举例: `wx.setStorageSync('key', 'value')` 向本地存储中写入内容

③ 异步 API

- 特点: 类似于 `jQuery` 中的 `$.ajax(options)` 函数, 需要通过 `success`、`fail`、`complete` 接收调用的结果
- 举例: `wx.request()` 发起网络数据请求, 通过 `success` 回调函数接收数据

5.协同工作和发布

5.1.为什么需要有权限管理

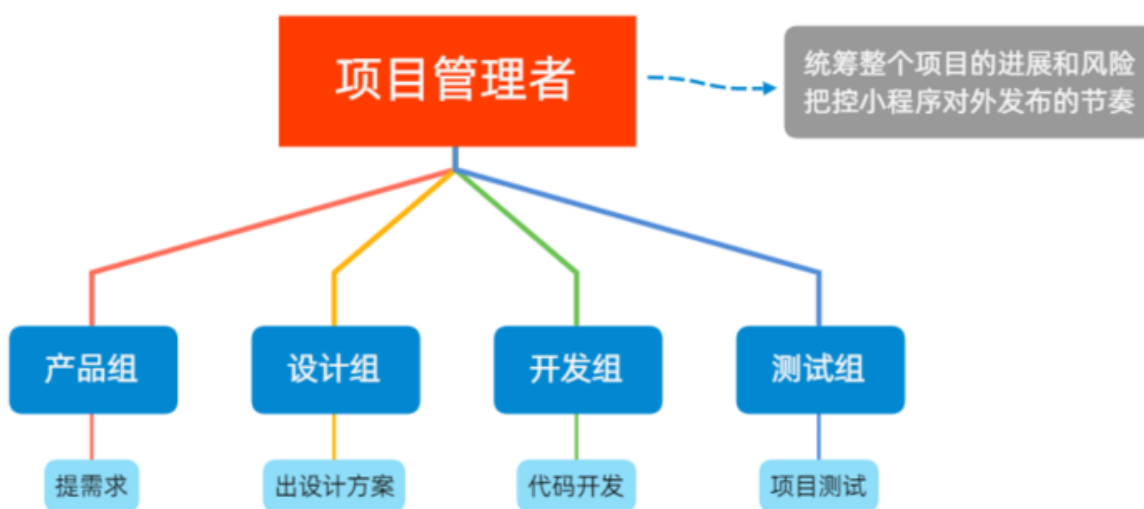
在中大型的公司里, 人员的分工非常仔细: 同一个小程序项目, 一般会有不同岗位、不同角色的员工同时参与设计与开发。

此时出于管理需要, 我们迫切需要对不同岗位、不同角色的员工的权限进行边界的划分, 使他们能够高效的进行协同工作。

5.2.项目成员的组织结构

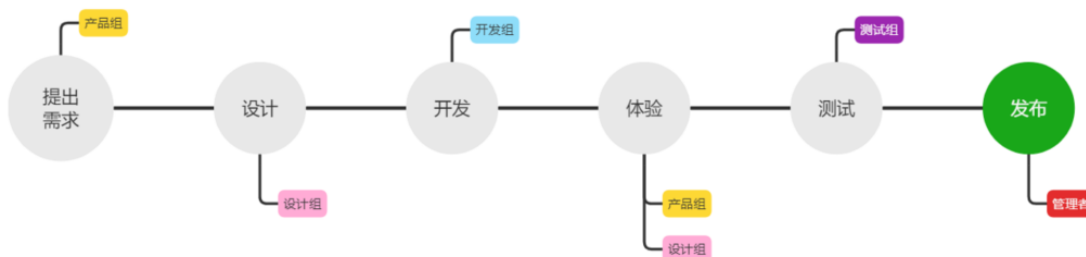
- 项目管理者
 - 产品组
 - 提需求
 - 设计组
 - 出设计方案
 - 开发组
 - 代码开发
 - 测试组
 - 项目测试

其中, 项目管理者需要统筹整个项目的进展和风险, 把控小程序对外发布的节奏
 组织结构图如下所示:



5.3.小程序的开发流程

小程序的大致开发流程图如下所示:



5.4.小程序成员管理

小程序成员管理体现在**管理员**对小程序**项目成员**及**体验成员**的管理:

① 项目成员:

- 表示参与小程序开发、运营的成员
- 可登录小程序管理后台
- 管理员可以添加、删除项目成员, 并设置项目成员的角色

② 体验成员：

- 表示参与小程序内测体验的成员
- 可使用体验版小程序，但不属于项目成员
- 管理员及项目成员均可添加、删除体验成员

5.4.1.不同项目成员对应的权限

不同项目成员对应的权限如下所示:

权限	运营者	开发者	数据分析者
开发者权限		√	
体验者权限	√	√	√
登录	√	√	√
数据分析			√
微信支付	√		
推广	√		
开发管理	√		
开发设置		√	
暂停服务	√		
解除关联公众号	√		
腾讯云管理		√	
小程序插件	√		
游戏运营管理	√		

我们更多关注的是开发者所具备的权限

开发者所具备的权限有: 开发权限, 体验者权限, 登录权限, 开发设置, 腾讯云管理

- 开发者权限：可使用小程序开发者工具及对小程序的功能进行代码开发
- 体验者权限：可使用体验版小程序
- 登录权限：可登录小程序管理后台，无需管理员确认
- 开发设置：设置小程序服务器域名、消息推送及扫描普通链接二维码打开小程序
- 腾讯云管理：云开发相关设置

5.4.2.添加项目成员和体验成员

具体操作如下图所示:



5.5.小程序的版本

在软件开发过程中，根据时间节点的不同，会产生不同的软件版本，例如：

- ① 开发者编写代码的同时，对项目代码进行自测（**开发版本**）
- ② 直到程序达到一个稳定可体验的状态时，开发者把体验版本给到产品经理和测试人员进行**体验测试**
- ③ 最后修复完程序的 Bug 后，发布**正式版**供外部用户使用

具体的版本说明如下图所示:

版本阶段	说明
开发版本	使用开发者工具，可将代码上传到开发版本中。开发版本只保留每人最新的一份上传的代码。点击提交审核，可将代码提交审核。开发版本可删除，不影响线上版本和审核中版本的代码。
体验版本	可以选择某个开发版本作为体验版，并且选取一份体验版。
审核中的版本	只能有一份代码处于审核中。有审核结果后可以发布到线上，也可直接重新提交审核，覆盖原审核版本。
线上版本	线上所有用户使用的代码版本，该版本代码在新版本代码发布后被覆盖更新。

5.6.发布上线

1. 小程序发布上线的整体步骤

一个小程序的发布上线，一般要经过上传代码 -> 提交审核 -> 发布这三个步骤。

2. 上传代码

- ① 点击开发者工具顶部工具栏中的“上传”按钮
- ② 填写版本号以及项目备注

3. 在后台查看上传之后的版本

登录小程序管理后台 -> 管理 -> 版本管理 -> 开发版本，即可查看刚才提交上传的版本了

4. 提交审核

- 为什么需要提交审核：为了保证小程序的质量，以及符合相关的规范，小程序的发布是需要经过腾讯官方审核的。
- 提交审核的方式：在开发版本的列表中，点击“**提交审核**”按钮之后，按照页面提示填写相关的信息，就能把小程序提交到腾讯官方进行审核。

5. 发布

审核通过之后，管理员的微信中会收到小程序通过审核的通知，此时在审核版本的列表中，点击“发布”按钮

之后，即可把“审核通过”的版本发布为“线上版本”，供所有小程序用户访问和使用。

6. 基于小程序码进行推广

相对于普通二维码来说，小程序码的优势如下：

- ① 在样式上更具辨识度和视觉冲击力
- ② 能够更加清晰地树立小程序的品牌形象
- ③ 可以帮助开发者更好地推广小程序

获取小程序码的 5 个步骤：

登录小程序管理后台 -> 设置 -> 基本设置 -> 基本信息 -> 小程序码及线下物料下载

5.7.运营数据

查看小程序运营数据的两种方式

- 在“**小程序后台**”查看
 - 登录小程序管理后台
 - 点击侧边栏的“统计”
 - 点击相应的 tab 可以看到相关的数据
- 使用“**小程序数据助手**”查看
 - 打开微信
 - 搜索“小程序数据助手”
 - 查看已发布的小程序相关的数据

6.总结

- ① 能够知道如何创建小程序项目

微信开发者工具的使用、`appID` 的获取

- ② 能够清楚小程序项目的基本组成结构

`app.js`、`app.json`、`app.wxss`、`pages` 文件夹

- ③ 能够知道小程序页面由几部分组成

`wxml`、`wxss`、`json`、`js`

- ④ 能够知道小程序中常见的组件如何使用

`view`、`text`、`image`

- ⑤ 能够知道小程序如何进行协同开发和发布

成员管理、发布小程序、查看运营数据

