

# 第 1 篇

## 基础知识

本篇是微信小程序开发的基础知识。首先介绍小程序的发展历程，然后讲述了小程序的优势与劣势以及应用范围，最后对小程序账号注册和项目的创建进行讲解，带领读者快速步入微信小程序的世界。

读者在学习完本篇后将会了解到小程序的发展史、小程序特性等基本内容，并且能够掌握小程序账号注册以及项目创建的方法，为后面更深入地学习小程序开发打下坚实的基础。

- 第 1 章 了解小程序
- 第 2 章 第一个微信小程序

# 第1章

## 了解小程序



### 本章概述

本章从微信小程序的定义、发展历程、应用场景及意义等不同角度进行了介绍与分析，告知了读者小程序的优缺点，帮助读者对小程序有一个全面的认知与了解，使读者可以根据应用的需求与应用场景来确定项目是否适合使用小程序进行开发，从而避免读者在实际进行小程序开发时出现小程序功能不能满足项目需求的情况，让读者能够方便有效使用小程序进行项目开发。



### 知识导读

本章要点（已掌握的在方框中打钩）

- 了解微信小程序
- 微信小程序账号的注册
- 微信开发者工具的下载与安装

## 1.1 小程序简介

微信小程序是一种新兴的应用程序，它与传统的 App 相比有很大区别，它是一种跨平台的应用程序，不区分 Android 版本与 iOS 版本，具有更好的移植性和更快的开发效率。

目前手机上的 App 琳琅满目，同类型的应用也比比皆是。在日常生活中，一些应用软件在最初下载时使用频率会高一点，随着时间的推移，这些软件的使用频率会逐渐降低，有时几周都不会使用。这些使用频率较低的应用软件一直占用用户手机的存储空间，造成了手机内存浪费。一些用户为了避免这种内存浪费的情况发生，对于使用频率较低的软件会在需要使用时进行下载，使用完毕后就删除，这种方式减少了手机内存的浪费，但是在应用下载过程中会消耗时间与流量。那么是否有一种更好的方式来解决这种现象呢？答案是：使用小程序。小程序具有无须安装、即开即用、不会额外占用用户手机内存的特点，它可以完美解决我们日常生活中低频率软件的内存占用问题。

### 1.1.1 什么是小程序

微信小程序，简称为小程序，英文名为 Wechat Mini Program。它是一种依赖微信环境的跨平台应用程

序，用户无须下载安装相应的小程序，点开即可使用，使用完毕退出即可，无须卸载，也不会额外占用手机内存。并且同一版本的小程序，在 Android 与 iOS 上都能良好运行，可以降低开发成本，提高开发效率。

### 1.1.2 小程序的发展历程

微信小程序最初并不完善，也没有对外开放，它仅是微信团队内部进行一些业务处理的 JS API，一些外部开发者发现这些相关的 API 后，按照相应的规则使用这些接口，随着时间的推移与外部开发者数量的增多，这些 API 形成了一种业内默认的规则与标准。2015 年初，微信团队对拍摄、录音、语音识别、二维码、地图、支付、分享、卡券等几十个 API 进行整理，发布了一整套的网页开发工具包，也被称为 JS-SDK，供所有开发者使用，使一些外部开发者也能够方便使用微信的原生功能，这就是小程序的雏形。

2016 年 1 月 11 日，“微信之父”张小龙对微信功能进行分析，提出了“一切以用户价值为依据”“让创造体现价值”“让用户用完即走”“让商业化存在于无形之中”的四大价值观。也指出了越来越多产品采用公众号方式实现面临的问题，虽然公众号具有开发方便、获取用户和传播成本低等特点，但是公众号拆分出的服务号并不能很好地解决服务所面临的问题，因此需要探索一种全新形态的应用，也就是微信小程序。

2016 年 9 月 21 日，微信小程序开始内测，在微信生态中，微信小程序“用完即走”的模式引发广泛关注，吸引了大批开发者参与测试。

2017 年 1 月 9 日，微信小程序测试结束，首批微信小程序正式上线运营，用户可以通过微信小程序体验相应的服务。

2017 年 12 月 28 日，微信更新的 6.6.1 版本中开放了小游戏程序，作为其中代表的“跳一跳”小程序，以游戏的方式引起了众多关注，收到了许多用户的好评，由此确定小游戏是小程序的另一种发展形态。

随着小程序版本的更新与优化，先后增加了小程序投诉、收藏与分享、广告、功能服务搜索、后台数据分析与管理等功能，使得第三方很容易开发出一款功能完善的小程序。时至今日，小程序因为使用简便、开发成本低等特点，已经成为商家应用的首选，各种各样的小程序为人们的衣食住行提供服务。

### 1.1.3 小程序的特征与应用场景

以下是“微信之父”张小龙对微信小程序的定义：“无须下载、安装触手可及，用完即走。”小程序依赖于微信生态，因此除了上面的一些特点以外，还天然具备一些其他优势。例如，引流、唯一性、入口众多、便于推广等。

(1) 引流——微信小程序上线后可以免费开通附近的小程序，微信用户使用附近小程序功能查找小程序时，会将附近 5 千米内的小程序按照距离排名。

(2) 唯一性——微信小程序的名称与域名相似，具有唯一性，一个小程序名称创建后，其他小程序不能使用相同的名称，微信用户可以通过搜索小程序名称进入小程序。

(3) 入口众多——微信小程序入口目前有 60 多种，其中常用的入口有以下几种：微信聊天界面下拉、附近的小程序、微信用户分享、小程序码、公众号关联、客服消息等。

(4) 便于推广——微信小程序在线上可以通过用户分享、广告、结合公众号等方式进行推广。在线下可以借助二维码或者小程序码的方式进行推广，从而实现线上、线下同步推广。

微信小程序具有的这些特征，使得越来越多的商户和产品加入到小程序的队伍中，小程序同传统的 App 相比既有优势也有劣势，它们之间具体的差异如表 1-1 所示。



表 1-1 小程序与 App 的区别

对比项	小程序	App
安装方式	无须下载安装	需要下载安装
应用占用空间	极小，基本可以忽略不计	一般占用空间较大
开发方式	只需开发一个版本	需要开发 Android 与 iOS 两个版本
开发成本	低	高
推广	借助微信流量，推广方便，成本低	原生流量，推广困难，成本高
应用功能	简单	丰富
用户体验	一般	较好

小程序目前虽然已经拥有丰富的产品应用，具备海量的用户基数。但是小程序简单方便的设计理念和体量限制，使得小程序不能像 App 一样具有完善丰富的功能，而且 App 在某些方面能够为用户提供更好的使用体验。因此小程序不能完全替代 App，它们之间不仅仅是竞争关系，还具有补充关系。对于功能简单、快进快出场景下的低频应用，采用小程序的方式是非常明智的，不仅可以降低开发难度、节省开发成本、提高开发效率，也可以减轻用户的使用负担，用户无须进行安装与下载操作，只需使用小程序提供的核心功能即可。对于一些高频、功能复杂的应用，使用小程序可能无法实现应用中的复杂功能，给用户带来一个较差的使用体验，面对这种情况，应用的核心功能应通过 App 方式实现，小程序用来作为一个引流入口，将微信的用户流量引入到 App 中。

小程序与 App 在不同的场景有不同的表现，目前小程序与 App 主流的应用场景如图 1-1 所示。



图 1-1 小程序与 App 的应用场景

在图 1-1 中第一象限：用户刚需且高频使用的应用，这个象限中基本都是行业内的巨头，需要使用 App 来为用户提供丰富的功能和高品质的用户体验。例如，购物类（淘宝、京东等）、出行类（百度地图、高德地图等）、支付类（支付宝、微信支付、各大银行支付）、社交类（微信、QQ 等）。

第二象限：用户刚需但是低频使用的应用，这个象限中包含大量的服务类产品，例如旅游、票务、教育、医疗等。对这些应用，用户可能使用过一次就不再使用，或者间隔很长时间才会使用一次。这类应用使用 App 方式实现，不仅开发复杂，成本较高，用户的自然增长也较慢，需要花费大量的人力、物力进行推广宣传。使用小程序的方式实现，开发简便，成本低，可以借助微信的用户流量进行推广宣传，非常适合创业公司或者小规模公司产品的试错，提升产品竞争力，提高产品成功率。

第三象限：用户非刚需且低频使用的应用，这个象限中大多是一些小众产品，一般是出于情怀的个人

兴趣产品和工具产品，用户普遍较少，收益较低，基本不作为商业产品。这一类型的产品对于具有产品设计与开发能力的企业或公司可以考虑 App 开发。但是对于不具备产品设计与开发能力的公司应当优先考虑小程序开发。

**第四象限：** 用户非刚需但会高频使用的应用，这个象限中大多是一些休闲娱乐类的应用，这些应用通常功能比较复杂，但不是用户刚需的，使用这些应用的用户使用频率会比较高。其中的代表应用：抖音、快手、修图软件、休闲游戏等。小程序不能完美实现这些应用功能，但是可以作为一个引流入口，将一些微信用户引流到各自的平台中，增加 App 的用户量。

#### 1.1.4 小程序的发展前景

自从微信小程序正式发布以后，互联网巨头 BAT 中的其余两家阿里巴巴和百度也先后发布各自版本的小程序，进行小程序的布局与规划。随着小程序的发展，企业的需求量也在不断增加，越来越多企业投入到小程序开发与建设的浪潮中，微信庞大的用户基数也为后期小程序的推广提供了便利。2020 年，微信小程序数量已经突破 380 万，日均活跃用户超过 4 亿，覆盖超过 200 个细分行业。并且腾讯一直在和高校教育机构合作，大力推广微信小程序进入大学课堂，越来越多的学生开始学习微信小程序的相关知识，参与到小程序的开发。

## 1.2 小程序开发的准备工作

进行项目开发前需要做一些准备工作，例如 Java、Python 等编程语言，需要先进行开发环境的设置。微信小程序的运行依赖于微信，虽然不用进行开发环境的设置，但是需要进行小程序的注册、设置、信息完善等工作。

#### 1.2.1 小程序的注册

注册一个微信小程序需要经过以下几个步骤：

(1) 开发者在微信公众平台，单击“立即注册”按钮，进行小程序账号的注册，具体操作如图 1-2 所示。



图 1-2 微信公众平台界面



(2) 选择账号类型，微信公众平台提供了四种类型的账号，分别是订阅号、服务号、小程序和企业微信。这四种账号的类型及功能如表 1-2 所示。

表 1-2 微信公众平台账号的四种类型及其功能

账号类型	功能介绍
订阅号	偏向于向用户传递信息（类似于报纸和杂志），认证前后每天只能群发 1 条信息（适用于个人和组织）。如果仅是简单发送消息进行宣传，可以选择订阅号
服务号	与订阅号相似但是更偏向于服务交互（类似银行、114、提供服务查询），认证前后每月只能群发 4 条消息（不适合个人使用）。需要提供更多功能，例如，微信支付可以选择服务号
小程序	一种新的开发能力，开发者可以快速开发一个小程序，具有无须安装、触手可及、用完即走的特点。小程序可以在微信中便捷获取与传播，也具有较好的使用体验
企业微信	企业微信原名企业号，是一个面向企业级市场的产品，是一个独立 App 的、好用的基础办公沟通工具，拥有最基础和最实用的功能服务，专门提供给企业使用的 IM 产品（适用于企业、政府、事业单位或其他组织）。用来管理内部企业员工、团队，对内使用可以选择企业微信

我们需要注册一个小程序账号，因此账号类型选择“小程序”，具体操作如图 1-3 所示。

(3) 接下来正式进入小程序账号的注册界面，小程序账号的注册流程分为三步，分别是账号信息、邮箱激活、信息登记。账号信息界面需要填写邮箱、账号密码、验证码等信息。其中邮箱需要满足一些条件。

- ① 邮箱没有申请过其他小程序。
- ② 邮箱未被公众平台和微信开放平台注册。
- ③ 邮箱未绑定个人微信号。

账号信息界面具体内容如图 1-4 所示。



图 1-3 选择要注册的账号类型

每个邮箱仅能申请一个小程序

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:  edrd 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**

图 1-4 账号信息界面

按照要求填写信息后，单击“注册”按钮，跳转到邮箱激活界面，进行小程序账号注册的第二步邮箱激活操作，具体界面如图 1-5 所示。

此时，我们填写的邮箱会收到一封来自微信团队发送的邮件，单击“登录邮箱”按钮，跳转到邮箱界面，查看邮件，邮件中包含一个激活链接，单击这个链接可以完成邮箱的激活操作，具体操作如图 1-6 所示。



图 1-5 邮箱激活界面



图 1-6 邮件信息

接下来进行小程序账号注册的最后一步，也是最重要的一步，进行小程序的信息登记。在小程序信息登记界面需要对小程序所在的地区与主体信息进行填写，这些信息填写完成后，后期进行变更需要审核，操作也比较麻烦，一定要认真填写。微信小程序的主体分为 5 类，分别是个人、企业、政府、媒体和其他组织，这几个主体类型之间的区别如表 1-3 所示。

表 1-3 微信小程序主体类型

主体类型	说 明	账 号 功 能	注 册 要 求
个人	18 周岁以上，具有国内身份信息的微信实名用户	暂不具备微信支付功能及高级接口能力，不支持微信认证，开发者上限 10 人	管理员身份证号、管理员本人微信进行身份验证（免费）
企业	个人独资企业、个体工商户、个体经营、分支机构、合伙企业等	支持微信认证，可以使用微信支付功能和高级接口能力，开发者上限认证前 10 人，认证后 20 人	企业名称、营业执照注册号、管理员身份证号、微信认证（300 元认证费）、管理员本人微信进行身份验证
政府	事业单位（事业单位分支、派出机构、部队医院等）、政府机关（政协组织、人民解放军、武警部队、其他机关等）	支持微信认证，可以使用微信支付功能和高级接口能力，开发者上限认证前 10 人，认证后 20 人	政府全称、管理员身份证号、微信认证、管理员本人微信进行身份验证



续表

主体类型	说 明	账号功能	注册要求
媒体	事业单位媒体、其他媒体、电视广播、报刊、杂志、网络媒体等	支持微信认证，可以使用微信支付功能和高级接口能力，开发者上限认证前 10 人，认证后 20 人	组织名称、组织机构代码、组织机构代码证件照、管理员身份证号、微信认证、管理员本人微信进行身份验证
其他组织	免费类型（基金会、政府机构驻华代表处）、社会团体（社会团体分支、群众团体等）、民办非企业、学校、医院、其他组织（宗教活动场所、村民委员会、居民委员会等）	支持微信认证，可以使用微信支付功能和高级接口能力，开发者上限认证前 10 人，认证后 20 人	组织名称、组织机构代码、组织机构代码证件照、管理员身份证号、微信认证（300 元认证费）

我们这里注册的小程序账号仅用来学习和个人使用，因此主体类型选择个人，按照要求填写相关信息，具体操作如图 1-7 所示。

The screenshot shows the 'Subject Information Registration' step of the WeChat Mini Program registration process. It includes fields for 'Identity Card Name', 'Identity Card Number', 'Manager's Phone Number', 'SMS Verification Code', and 'Administrator Identity Verification'. A red arrow points to the QR code area, with the text 'Use the administrator's WeChat to scan this QR code to complete the administrator identity verification' overlaid.

图 1-7 信息登记界面

单击“继续”按钮，会弹出一个弹窗来让开发者确定小程序账号的主体，一旦确定，小程序账号的主体信息就不能再随便修改。具体操作如图 1-8 所示。

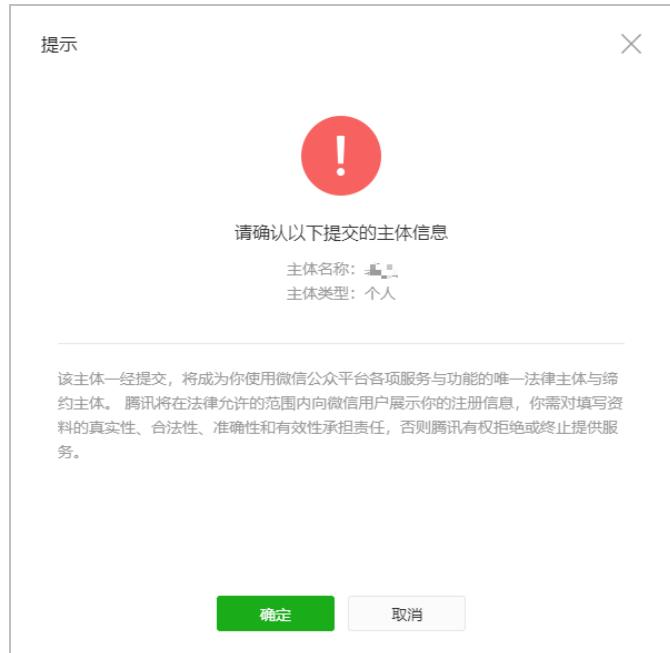


图 1-8 小程序账号主体确定界面

单击“确定”按钮后，小程序账号就创建成功了，如图 1-9 所示。

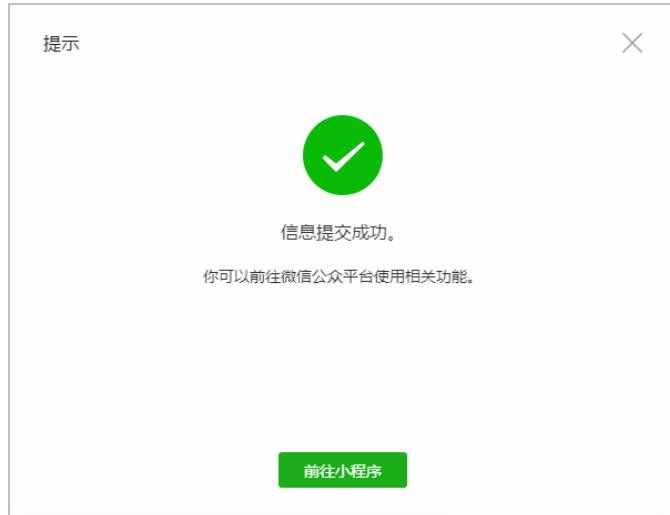


图 1-9 小程序账号创建成功界面

## 1.2.2 完善小程序信息

小程序账号创建完成后，可以通过单击图 1-9 所示的“前往小程序”按钮，或者访问微信公众平台界面使用账号密码或使用管理员微信扫描二维码的方式进行登录，进入小程序账号后台管理界面，可以在这个界面进行小程序基本信息的完善、小程序管理、版本发布等。小程序账号后台管理界面的内容如图 1-10 所示。

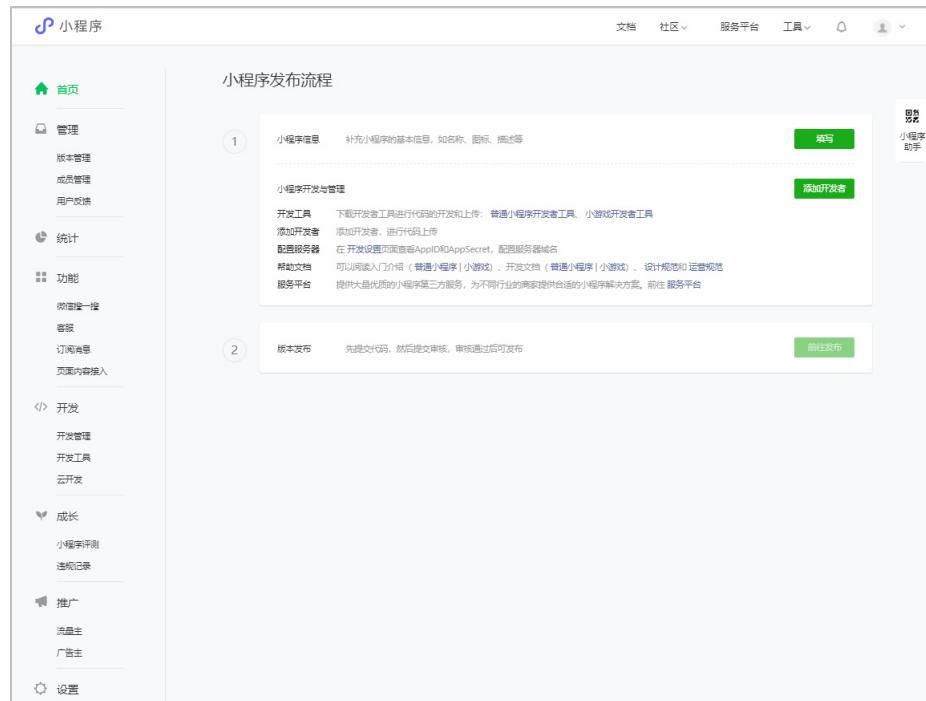


图 1-10 小程序账号后台管理界面

在这个界面中我们要完善小程序的基本信息，单击“填写”按钮会跳转到小程序信息填写页面，填写的内容包括小程序名称、小程序简称、小程序头像、小程序介绍、服务类目。这些信息的作用与要求如表 1-4 所示。

表 1-4 小程序信息作用与要求

内 容	说 明	填 写 要 求	修 改 要 求
小程序名称	用户用来搜索小程序的名称，具有唯一性	名称长度为 4~30 个字符(1 个中文字为 2 个字符)，必填	发布前有 2 次修改机会，发布后个人主体的小程序每年有 2 次修改机会，非个人主体的小程序需要先进行微信认证（认证费 300 元）才能修改
小程序简称	从小程序名称中按顺序截取的字段，可以在客户端任务栏向用户展示，可重名	简称长度为 4~10 个字符，选填	发布前有 2 次修改机会，发布后每年有 2 次修改机会，小程序修改名称期间无法修改简称，删除简称也会消耗 1 次修改机会
小程序头像	用户使用小程序时展示给用户的头像	头像不允许涉及政治敏感与色情，图片格式为 png、bmp、jpeg、jpg、gif，图片大小小于 2MB，建议图片尺寸为 144px*144px	每月可以修改 5 次
小程序介绍	向用户介绍小程序的主要功能与服务类型	长度为 4~120 个字符，并且介绍内容中不得含有国家相关法律法规禁止内容	每月可以修改 5 次
服务类目	根据小程序功能选择相关类目	服务类目有两级，每一级都必须填写，一个小程序最少要填写 1 个服务类目，最多填写 5 个服务类目，特殊行业需要额外提供资质证明	每月可以修改 1 次

小程序信息填写界面如图 1-11 所示。



图 1-11 小程序信息填写界面

### 1.2.3 设置小程序成员

完成小程序的基本信息设置后，就可以对小程序成员进行管理。小程序成员分为三类，分别是管理员、项目成员、体验成员。

- (1) 管理员——小程序注册使用的微信账号，是整个小程序权限最高的成员。
- (2) 项目成员——进行小程序开发、运营的人员，根据职责和权限可以细分为开发者、运营者、数据分析师。项目成员只能由管理员添加、删除。
- (3) 体验成员——进行小程序体验版体验、测试的人员，可以由管理员或者项目成员进行添加、删除。项目成员中开发者、运营者、数据分析师的权限如表 1-5 所示。

表 1-5 不同类型项目成员具有的权限

权 限	开 发 者	运 营 者	数 据 分 析 者
登录小程序管理后台	是	是	是
版本发布	是	否	否
数据分析	否	否	是
开发能力	否	是	否
修改小程序介绍	是	否	否
暂停/恢复服务	是	否	否
设置可被搜索	是	否	否
解除关联移动应用	是	否	否
解除关联公众号	是	否	否
管理体验者	是	是	是
体验者权限	是	是	是
微信支付	是	否	否
小程序插件管理	是	否	否
游戏运营管理	是	否	否
推广	是	否	否

主体为个人的小程序可以设置 15 个项目成员和 15 个体验成员；未认证、未发布的非个人小程序可以设置 30 个项目成员和 30 个体验成员；已认证、未发布或未认证、已发布的非个人小程序可以设置 60 个项目成员和 60 个体验成员；已认证、已发布的非个人小程序可以设置 90 个项目成员和 90 个体验成员。

## 1.3 微信开发者工具

微信开发者工具是微信团队提供的编译器软件，具有公众号、小程序调试与开发功能，类似于其他编程语言中的 IDE。

### 1.3.1 微信开发者工具的下载与安装

下载微信开发者工具需要访问微信开放文档 (<https://developers.weixin.qq.com/miniprogram/dev/devtools/download.html>)，进入微信开发者工具的下载界面，微信开发者工具分为开发版（Nightly Build）、预发布版（RC Build）、稳定版（Stable Build）。其中稳定版经过大量测试，性能稳定，BUG 较少，推荐选择这一版本。用户需要根据自己计算机的操作系统选择对应链接进行下载（Windows 操作系统仅支持 Windows 7 以上的版本），本书选用 Windows 64 位的稳定版本的微信开发者工具，具体操作如图 1-12 所示。



图 1-12 微信开发者工具下载界面

微信开发者工具下载完需要进行安装，安装步骤如下所示。

(1) 运行下载好的.exe 应用程序，具体操作分别如图 1-13 和图 1-14 所示。



图 1-13 安装向导

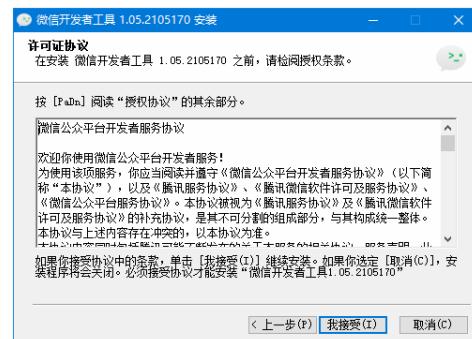


图 1-14 授权许可

(2) 设置安装路径，进行安装，具体操作分别如图 1-15 和图 1-16 所示。

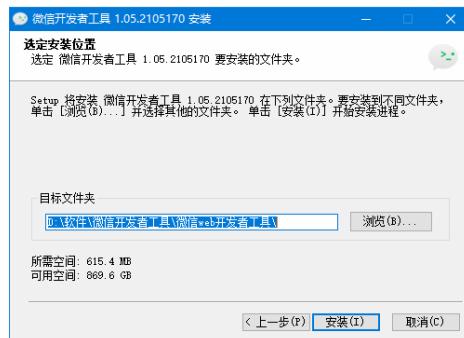


图 1-15 设置安装路径

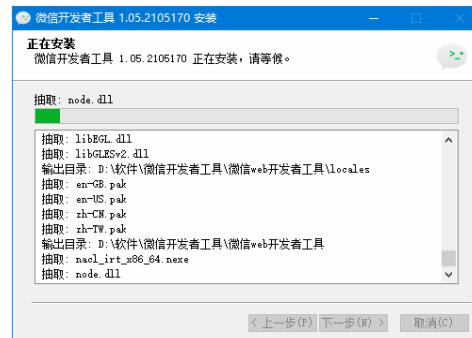


图 1-16 安装

安装完成的效果如图 1-17 所示。



图 1-17 安装完成

### 1.3.2 微信开发者工具功能介绍



图 1-18 扫描登录微信开发者工具

使用微信开发者工具时，需要使用微信账号扫描二维码进行登录，这个微信账号必须是小程序账号后台管理中设置的开发成员的微信账号，如图 1-18 所示。

登录后，创建新项目或者打开原有项目可以进入微信开发者工具界面，整个界面可以划分为菜单栏、工具栏、模拟器区域、目录树区域、编辑器区域和调试器区域六大部分。

(1) 菜单栏——包含项目（项目的新建、导入、查看、关闭等）、文件（新建文件、保存、关闭等）、编辑（查看编辑相关的操作和快捷键）、工具（编译、刷新、前后台切换、清除缓存等）、转到（切换编辑器、切换组等）、选择（全选、重复选择、选择所有匹配项等）、视图（外观、编辑器布局、资源管理器等）、界面（工具栏、调试器、模拟器等）、设置、帮助、微信开发者工具（检查更新、切换账号、退出等）等选项，可以进行项目的新建、导入、界面设置、代码编辑与调试等操作。

(2) 工具栏——包括用户中心（显示用户头像可以进行开发用户的切换）、模拟器（控制模拟器区域的显示与隐藏）、编辑器（控制编辑器区域的显示与隐藏）、调试器（控制调试器区域的显示与隐藏）、可视化、模式选择（小程序模式与插件模式）、编译（运行小程序）、预览（在手机预览小程序功能）、真机调试（在手机上进行小程序测试与调试）、缓存清理（清理小程序运行过程中产生的缓存）、上传（上传小程序项目代码）、版本管理（小程序项目代码管理）、详情（小程序信息、本地设置、项目配置）等选项，可以进行界面布局，代码的编译、调试、发布等操作。

(3) 模拟器区域——用来模拟移动端设备的运行界面，可以设置移动端的设备类型（iPhone、iPad、Nexus 等）、显示尺寸、字体大小、网络类型（WiFi、4G 等）等。

(4) 目录树区域——显示项目目录结构，进行项目文件夹及项目文件的创建与删除操作。

(5) 编辑器区域——用来显示项目代码，进行项目代码编辑操作。

(6) 调试器区域——显示项目运行信息，进行项目代码调试，查看网络请求信息等。

微信开发者工具界面如图 1-19 所示。

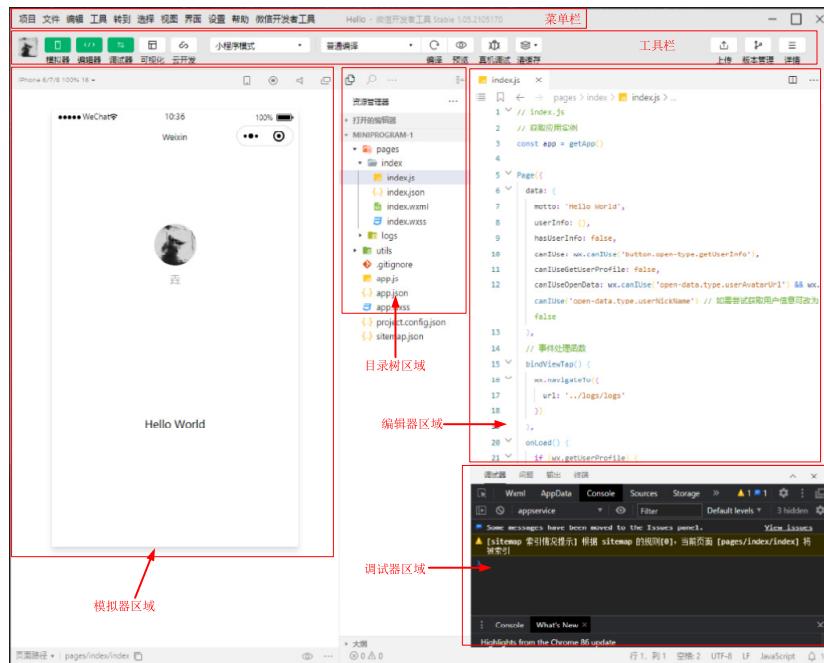


图 1-19 微信开发者工具界面

## 1.4 就业面试技巧与解析

工作通过一些常见的面试及笔试题来加深读者对微信小程序的理解，帮助读者更好地应对面试或笔试，从容发挥，取得心仪的 offer。

### 1.4.1 面试技巧与解析（一）

**面试官：**什么是微信小程序？

**应聘者：**微信小程序是由“微信之父”张小龙提出的一种新型公众号实现方式，具有无须下载安装、即开即用、用完即走的特点。并且微信小程序依赖于微信生态，具备开发成本低、自带推广流量、跨平台等优势。但是由于微信小程序体量与设计理念的限制，在功能丰富度与用户体验方面有所欠缺。

### 1.4.2 面试技巧与解析（二）

**面试官：**微信小程序、App 和 H5 的区别是什么？

**应聘者：**微信小程序、App 和 H5 都是用来实现用户功能的应用程序，但是它们的实现方式与理念有所区别，具体体现在以下几个方面：

(1) 运行环境不同。微信小程序运行在微信中，H5 运行在浏览器上，App 运行在 Android 系统或 iOS 系统中。

(2) 开发成本不同。H5 开发需要兼容不同的浏览器，App 开发需要区分 Android 版本与 iOS 版本，因此微信小程序的开发成本最低，H5 次之，App 最高。

(3) 获取系统权限不同，App 和微信小程序都能获取系统级权限，而 H5 不能。



- (4) 推广方面。微信小程序可以借助微信流量，推广方便；App 和 H5 只能通过原生流量，推广困难。
- (5) 用户体验方面。App 的用户体验最好，微信小程序和 H5 的用户体验稍差。

### 1.4.3 面试技巧与解析（三）

面试官：简述小程序原理。

应聘者：微信小程序由 `webView` 和 `appService` 两部分组成，其中 `webView` 用来展示 UI 界面，`appService` 用来处理业务逻辑、进行接口调用和数据处理操作，它们在两个不同的进程中运行，由系统层 `JSBridge` 实现通信、完成 UI 渲染、事件处理。

# 第2章

## 第一个微信小程序



### 本章概述

本章我们开始进入微信小程序的开发阶段，要学习使用微信开发者工具、创建第一个微信小程序、了解微信小程序的项目结构。

通过本章内容的学习，我们可以学会使用微信开发者工具创建一个微信小程序，深入认识微信小程序的项目结构，为读者在未来的小程序开发奠定坚实的基础。



### 知识导读

本章要点（已掌握的在方框中打钩）

- 创建一个微信小程序
- 微信小程序的项目结构
- 小程序应用的生命周期
- 小程序页面的生命周期

## 2.1 创建第一个微信小程序

本章要创建第一个微信小程序，通过这个小程序来认识小程序的项目结构，学习与小程序框架相关的知识。

创建微信小程序需要借助微信开发者工具。

### 2.1.1 新建项目

运行微信开发者工具，选择“小程序”按钮，点击右侧界面上的+号进行小程序项目的创建。具体操作如图 2-1 所示。

设置小程序项目名称（该名称仅是本地存储的项目文件名称，与小程序上线后显示的名称无关）、项目本地存储路径、AppID（需要创建微信小程序账号，从微信小程序账号管理后台获取）、后端服务（云开发，在后面章节中会进行介绍）。具体操作如图 2-2 所示。



图 2-1 选择“小程序”按钮



图 2-2 创建小程序项目

若是企业或个人不具备资质或者没有注册小程序账号，没有相应的 AppID，在创建小程序时可以在 AppID 输入框后边选择测试号，通过测试号方式创建的小程序，仅能用来进行功能测试，不能进行项目发布、上传，也不能使用云开发功能。

## 2.1.2 获得 AppID

微信小程序的 AppID 是在每一个小程序账号创建成功后自动生成的一个唯一 ID，这个 ID 相当于小程序的身份证，在获取用户信息、调用接口、使用云开发功能时都需要这个 ID。查看 AppID 需要登录微信小程序账号管理后台，按照“设置—基本设置—账号信息”的步骤查看，具体如图 2-3 所示。



图 2-3 查看微信小程序 AppID

## 2.2 微信小程序项目结构

在创建微信小程序项目时，微信开发者工具会自动生成一个项目 demo，这个 demo 中通常由 pages、utils、app.js、app.json、app.wxss、project.config.json、sitemap.json 几部分组成，其中 pages 属于项目页面文件；utils 属于项目其他文件；app.js、app.json、app.wxss 这几个以 App 开头的文件属于项目主体文件；project.config.json 与 sitemap.json 属于项目配置文件。小程序的目录结构如图 2-4 所示。



图 2-4 小程序的目录结构

### 2.2.1 小程序页面文件

每一个小程序项目中都存在一个 pages 文件夹，这是小程序页面文件的根目录文件夹，其中每一个页面由一个单独的文件夹保存。例如，pages 文件夹下的 index 文件夹中存放小程序的首页界面文件，首页界面文件包含 index.wxml（页面结构文件）、index.js（页面逻辑文件）、index.json（页面配置文件）、index.wxss（页面样式文件）。

#### 1) WXML 文件

index.wxml 文件使用的 WXML 语言是微信团队仿照 HTML 语言设计的一种新语言，它们用法相似，只是一些标签和属性有所区别，WXML 的代码格式如下所示。

```
<!--index.wxml-->
<view class="container">
<view class="userinfo">
<block wx:if="{{canIUseOpenData}}">
<view class="userinfo-avatar" bindtap="bindViewTap">
<open-data type="userAvatarUrl"></open-data>
```



```
</view>
<open-data type="userNickName"></open-data>
</block>
<block wx:elif="{{ !hasUserInfo }}">
  <button wx:if="{{ canIUseGetUserProfile }}" bindtap="getUserProfile">获取头像昵称</button>
  <button wx:elif="{{ canIUse }}" open-type="getUserInfo" bindgetuserinfo="getUserInfo">获取头像昵称</button>
  <view wx:else>请使用 1.4.4 及以上版本基础库</view>
</block>
<block wx:else>
  <image bindtap="bindViewTap" class="userinfo-avatar" src="{{ userInfo.avatarUrl }}" mode="cover">
    </image>
  <text class="userinfo-nickname">{{ userInfo.nickName }}</text>
</block>
</view>
<view class="usermotto">
  <text class="user-motto">{{ motto }}</text>
</view>
</view>
```

## 2) JS 文件

`index.js` 文件使用的是微信团队在 JavaScript 语言基础上改进后的 JS，用来实现页面逻辑功能，相比原本在浏览器上运行的 JS，缺少了 DOM 与 BOM 对象，而且还新增了一些在微信小程序中的专属用法，JS 的代码格式如下所示。

```
//index.js
//获取应用实例
const app = getApp()
Page({
  data: {
    motto: 'Hello World',
    userInfo: {},
    hasUserInfo: false,
    canIUse: wx.canIUse('button.open-type.getUserInfo'),
    canIUseGetUserProfile: false,
    canIUseOpenData:wx.canIUse('open-data.type.userAvatarUrl')&&wx.canIUse('open-data.type.userNickName')
    //如需尝试获取用户信息可改为 false
  },
  //事件处理函数
  bindViewTap() {
    wx.navigateTo({
      url: '../logs/logs'
    })
  },
  onLoad() {
    if (wx.getUserProfile) {
      this.setData({
        canIUseGetUserProfile: true
      })
    }
  },
  getUserProfile(e) {
    //推荐使用 wx.getUserProfile 获取用户信息，开发者每次通过该接口获取用户个人信息均需用户确认，
    //开发者妥善保管用户快速填写的头像昵称，避免重复弹窗
    wx.getUserProfile({
      desc: '展示用户信息', //声明获取用户个人信息后的用途，后续会展示在弹窗中，请谨慎填写
      success: (res) => {
        console.log(res)
        this.setData({
          userInfo: res.userInfo,
          hasUserInfo: true
        })
      }
    })
  }
})
```

```
        }
    })
},
getUserInfo(e) {
//不推荐使用 getUserInfo 获取用户信息，自 2021 年 4 月 13 日起，getUserInfo 不再弹出弹窗，
//并直接返回匿名的用户个人信息
    console.log(e)
    this.setData({
        userInfo: e.detail.userInfo,
        hasUserInfo: true
    })
}
})
```

### 3) JSON 文件

`index.json` 文件遵循 JSON 的语法格式，仅用来进行页面 `window` 的相关属性设置，具体代码格式如下所示。

```
{  
  "usingComponents": {}  
}
```

#### 4) WXSS 文件

`index.wxss` 文件是页面的样式文件，使用的该样式文件 WXSS 与 CSS 用法一致，可通过标签名、类名、`id` 名等方式为指定的标签设置相应样式，具体代码格式如下所示。

```
/**index.wxss*/
.userInfo {
  display: flex;
  flex-direction: column;
  align-items: center;
  color: #aaa;
}
.userInfo-avatar {
  overflow: hidden;
  width: 128rpx;
  height: 128rpx;
  margin: 20rpx;
  border-radius: 50%;
}
.usermotto {
  margin-top: 200px;
}
```

首页界面在微信开发者工具模拟器上的运行效果如图 2-5 所示。

另外，也可以点击工具栏的“预览”或“真机调试”按钮，使用微信账号扫描生成的二维码实现小程序在手机上的显示，只是真机调试模式不仅可以显示小程序的界面，还可以查看小程序的网络请求和数据交互信息，因此真机调试模式需要使用具有该小程序开发权限的微信账号；而预览模式只能显示小程序的界面，因此不需要使用具有开发权限的微信账号，它们的显示效果分别如图 2-6 与图 2-7 所示。

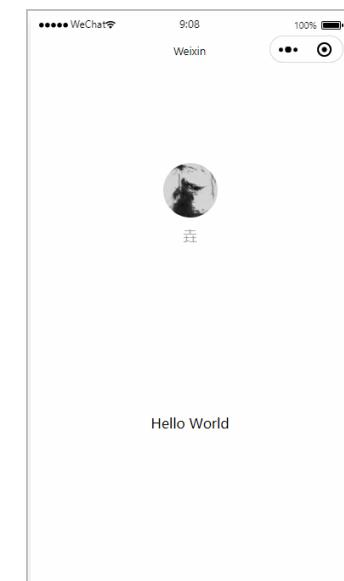


图 2-5 首页界面在模拟器上的运行效果



图 2-6 预览



图 2-7 真机调试

## 2.2.2 小程序其他文件

utils 属于小程序的其他文件夹，小程序中经常使用的方法可以进行提取整理形成一个模块存放到 utils 文件夹，之后其他页面使用时可以直接导入进行应用，也可以存放一些第三方插件用来完成实现小程序的特殊功能。

## 2.2.3 小程序主体文件

小程序主体文件具有 app 前缀，包含 app.js、app.json、app.wxss 三个文件，用来进行小程序全局的一些设置与方法实现。

### 1) app.js 文件

app.js 文件是小程序的全局逻辑文件，它与页面逻辑文件的写法与作用相似，只是 app.js 文件中的变量与方法作用于整个小程序，在任何一个页面中都可以使用，而页面逻辑文件中的变量与方法仅能在本页面使用，在其他页面中无法使用。

### 2) app.wxss 文件

app.wxss 文件是小程序的全局样式文件，其设定的样式作用于整个小程序，每个页面都会具有相应的样式。

### 3) app.json 文件

app.json 文件是小程序的全局配置文件，主要有 pages、window、sitemapLocation 三个配置项，其中 pages 配置项用来设置小程序的页面路径，运行小程序时默认显示第一个页面路径对应的页面；window 配置项可以设置小程序的状态栏、导航栏、标题、窗口背景色等与窗口相关的属性设置；sitemapLocation 配置项用来指定 sitemap.json 文件的路径。app.json 文件的代码格式如下所示。

{

```

"pages": [
    "pages/index/index",
    "pages/logs/logs"
],
>window": {
    "backgroundTextStyle": "light",
    "navigationBarBackgroundColor": "#fff",
    "navigationBarTitleText": "Weixin",
    "navigationBarTextStyle": "black"
},
"style": "v2",
"sitemapLocation": "sitemap.json"
}

```

app.json 文件中常用的配置项如表 2-1 所示。

表 2-1 app.json 文件中常用的配置项

配置项	类型	是否必填	说明
entryPagePath	string	否	小程序默认启动首页
pages	string[]	是	页面路径列表
window	Object	否	全局的默认窗口
tabBar	Object	否	小程序底部标签栏
debug	boolean	否	是否开启 debug 模式， 默认关闭
sitemapLocation	string	是	指明 sitemap.json 文件的路径
usingComponents	Object	否	全局自定义组件配置
plugins	Object	否	使用的插件
subpackage	Object[]	否	分包结构配置
functionalPages	boolean	否	是否启用插件功能页， 默认关闭

app.wxss 与 app.json 文件是小程序全局的样式与属性配置文件，当小程序页面没有设置相应的样式与 window 相关属性时，会默认使用 app.wxss 与 app.json 文件中的设置；当小程序页面设置相应的样式与 window 相关属性后，会替代全局的样式与 window 属性，优先使用页面中的设置。

## 2.2.4 小程序配置文件

小程序配置文件用来完成小程序项目在微信开发者工具中的一些个性化配置以及小程序能否被搜索到的权限设置。小程序配置文件包含 project.config.json 文件和 sitemap.json 文件。

### 1) project.config.json 文件

project.config.json 文件的本质是微信开发者工具的配置文件，用来记录小程序在开发者工具中的一些个性化配置。例如，编辑器的颜色，代码上传时的样式自动补全、自动压缩样式，小程序使用基础库的版本，小程序名称、AppID 等内容。该文件内容一般不需要改动，如果要进行改动一般不直接对该文件内容进行更改，而是通过单击开发者工具中工具栏的“详情”按钮，选择相应的信息选项进行修改。具体操作如图 2-8 所示。

在图 2-8 中，图（a）是小程序的基本信息选项，主要进行小程序基本信息的修改操作，可以更换小程序的 AppID，修改小程序的项目名称；图（b）是小程序的本地设置选项，可以进行小程序基础库的更改，小程序上传代码样式与校验合法域名等功能的开启和关闭；图（c）是小程序的项目配置选项，可以查看小程序配置的运行信息、项目包的大小以及分包情况，但是对域名的更改与分包操作需要在小程序账号



管理后台完成。

(a) 小程序基本信息

(b) 小程序本地设置

(c) 小程序项目配置

图 2-8 小程序项目配置的修改与查看

## 2) sitemap.json 文件

sitemap.json 文件用来进行小程序能否被搜索到的权限设置，可以设置小程序全部页面都能被索引到，也可以设置小程序仅有部分页面能被索引到，一般情况下该文件内容不需要改动，默认小程序全局都可被索引到，若该文件缺失也默认小程序全局可以被索引。示例代码如下所示。

```
{
  "rules": [
    {
      "action": "allow",
      "page": "path/to/page",
      "params": ["a", "b"],
      "matching": "exact"
    },
    {
      "action": "disallow",
      "page": "path/to/page"
    }
  ]
}
```

- ①path/to/page?a=0&b=1，会被优先索引。
- ②path/to/page，不会被索引。
- ③path/to/page?a=0，不会被索引。
- ④path/to/page?a=0&b=1&c=2，不会被索引。
- ⑤除 path/to/page 页面以外的其他页面都默认可以被索引。

在 sitemap.json 文件中通过 rules 配置项进行参数的配置，action 设置页面是否可以被索引，默认值为 allow，表示命中规则的页面可以被索引，disallow 表示命中该规则的页面不可以被索引。page 设置页面的路径，\*表示全部的页面，path/to/page 表示某一个具体的页面的路径。params 表示页面路径被规则匹配时携带有参数。matching 表示页面路径按照规则匹配时的匹配方式，matching 的取值说明如表 2-2 所示。

表 2-2 matching 的取值说明

值	说 明	示 例
exact	当小程序页面的参数列表等于 params 时，规则命中	"params": ["a", "b"], 符合规则的页面路径为“path/to/page?a=0&b=1”
inclusive	当小程序页面的参数列表包含 params 时，规则命中	"params": ["a", "b"], 符合规则的页面路径为“path/to/page?a=0&b=1”
exclusive	当小程序页面的参数列表与 params 交集为空时，规则命中	"params": ["a", "b"], 符合规则的页面路径为“path/to/page”
partial	当小程序页面的参数列表与 params 交集不为空时，规则命中	"params": ["a", "b"], 符合规则的页面路径为“path/to/page?a=0”

## 2.3 微信小程序的生命周期

微信小程序主要由视图层（View）与逻辑层（App Service）组成，其中视图层通常通过 WXML 文件和 WXSS 文件进行页面的布局与样式实现，逻辑层由 JS 文件实现。JS 文件从等级上可以分为 app.js（小程序应用等级）和 pages.js（小程序页面等级），每个小程序在运行时都先执行应用等级的 app.js 文件，然后才会执行页面等级的 pages.js 文件。

### 2.3.1 小程序应用的生命周期

app.js 文件是小程序全局逻辑文件，属于小程序应用等级，每一个小程序都要注册一个 App 实例对象，整个小程序中有且仅有一个 App 实例对象，这个 App 实例对象在小程序的每个页面都是共享的，都可以调用 App 实例对象中的全局数据和方法。app.js 文件中 App 实例对象的注册方法如下所示。

```
App({
  onLaunch (options) {
    //Do something initial when launch.
  },
  onShow (options) {
    //Do something when show.
  },
  onHide () {
    //Do something when hide.
  },
  onError (msg) {
    console.log(msg)
  },
 .globalData: {
    data:'I am global data',
  }
})
```

在 App 实例对象中有一些微信团队封装好的函数，这些函数用来执行小程序的生命周期或者完成某种特定的功能。App 实例对象中常用的函数如表 2-3 所示。



表 2-3 App 实例对象中常用的函数

函数名称	说 明
onLaunch	生命周期回调——监听小程序初始化，全局只触发一次
onShow	生命周期回调——监听小程序启动或切前台，小程序启动或者从后台进入前台显示时触发
onHide	生命周期回调——监听小程序切后台，小程序从前台进入后台时触发
onError	错误监听函数，小程序发生脚本错误或者 API 调用报错时触发
onPageNotFound	页面不存在监听函数，小程序要打开的页面不存在时触发
onUnhandledRejection	未处理的 Promise 拒绝事件监听函数，小程序有未处理的 Promise 拒绝时触发
onThemeChange	监听系统主题变化，系统切换主题时触发

App 实例对象中除了可以使用微信团队提供的函数以外，也可以由开发者自定义一些函数，具体操作如下所示。

```
App ({  
  //微信团队提供的函数  
  ...  
  //开发者自定义的函数  
  custom: function (params) {  
    //具体方法内容  
    ...  
  },  
  //全局数据对象  
  globalData: {  
    data:'I am global data',  
  }  
})
```

**小程序前后台机制：**用户打开一个小程序时分为冷启动与热启动，冷启动时需要重新进行加载，指用户首次打开或者小程序销毁后再次打开的情况；热启动是指用户之前打开一个小程序，在一定时间内再次打开的情况，此种情景小程序只需从后台切换到前台。用户点击右上角胶囊按键关闭小程序，或者通过设备上的 home 按键、手势操作等方式退出小程序，小程序并不会立即销毁，而是处于一种后台挂起状态，只有超过一定时间（目前是 5 分钟）或小程序占用资源过高时，才会被微信客户端销毁进行回收。一般用户退出小程序时会触发 onHide 方法，状态为前台→后台。用户再次打开小程序时会触发 onShow 方法，状态为后台→前台。

微信团队为了区分进入小程序的路径，设置了场景值，常用的场景值如表 2-4 所示。

表 2-4 常用的场景值

场景值 ID	说 明	场景值 ID	说 明
1000	其他	1011	扫描二维码
1001	发现栏小程序主入口	1014	小程序订阅消息
1005	微信首页顶部搜索框的搜索结果页	1025	扫描一维码
1006	发现栏小程序主入口搜索框的搜索结果页	1037	小程序打开小程序
1007	单人聊天会话中的小程序消息卡片	1043	公众号模版消息
1008	群聊会话中的小程序消息卡片	1047	扫描小程序码
1010	收藏夹	1058	公众号文章

小程序应用的生命周期函数有 `onLaunch`、`onShow`、`onHide`，具体的小程序应用生命周期如图 2-9 所示。

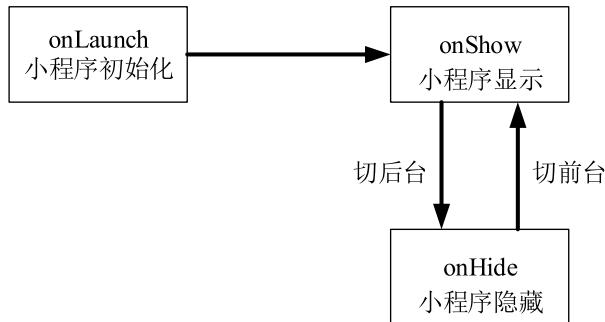


图 2-9 小程序应用生命周期

### 2.3.2 小程序页面的生命周期

`pages.js` 文件是小程序页面逻辑文件，属于小程序页面等级，每一个页面都要在其对应 JS 文件中注册一个 `Page` 实例对象，这个实例对象可以用来指定页面的初始数据、生命周期回调函数、事件处理函数等。`pages.js` 文件中 `Page` 实例对象的注册方法如下所示。

```

Page({
  data: {
    text: "This is page data."
  },
  onLoad: function(options) {
    // 页面创建时执行
  },
  onShow: function() {
    // 页面出现在前台时执行
  },
  onReady: function() {
    // 页面首次渲染完毕时执行
  },
  onHide: function() {
    // 页面从前台变为后台时执行
  },
  onUnload: function() {
    // 页面销毁时执行
  },
  onPullDownRefresh: function() {
    // 触发下拉刷新时执行
  },
  onReachBottom: function() {
    // 页面触底时执行
  },
  onShareAppMessage: function () {
    // 页面被用户分享时执行
  },
  onPageScroll: function() {
    // 页面滚动时执行
  },
  onResize: function() {
    // 页面尺寸变化时执行
  },
  onTabItemTap(item) {
    // tab 点击时执行
    console.log(item.index)
    console.log(item.pagePath)
    console.log(item.text)
  }
})

```



```
    },  
})
```

在 Page 实例对象中有一些微信团队封装好的属性方法，这些属性方法用来执行小程序页面的生命周期或者完成某种特定的功能。Page 实例对象常用的函数如表 2-5 所示。

表 2-5 Page 实例对象常用的函数

函数名称	说 明
onLoad	生命周期回调——监听页面加载，一个页面只会调用一次
onShow	生命周期回调——监听页面显示，页面显示/切入前台时触发
onHide	生命周期回调——监听页面隐藏，小程序从前台进入后台时触发，wx.navigateTo（保留当前页面跳转到其他页面）或底部 tab 切换到其他页面，小程序切入后台
onReady	生命周期回调——监听页面初次渲染完成，页面初次渲染完成时触发，一个页面只会调用一次，代表页面已经准备妥当，可以和视图层进行交互
onUnload	生命周期回调——监听页面卸载，页面卸载时触发，例如，wx.redirectTo（关闭当前页面，跳转到其他页面）或者 wx.navigateBack（关闭当前页面，返回上一页或多级页面）
onPullDownRefresh	监听用户下拉刷新事件
onReachBottom	监听用户上拉触底事件，在触发距离滑动期间，本事件只会被触发一次
onShareAppMessage	监听用户点击页面内转发按钮操作
onShareTimeline	监听用户点击右上角转发到朋友圈按钮的操作
onAddToFavorites	监听用户点击右上角收藏按钮操作
onPageScroll	监听用户滑动页面事件
onResize	页面尺寸改变时触发，屏幕发生旋转时触发
onTabItemTap	当前是 tab 页时，点击 tab 时触发

Page 实例对象中除了使用微信团队提供的属性方法以外，也可以由开发者自定义一些方法，具体操作如下所示。

```
Page({  
  //微信团队提供的函数  
  ...  
  //开发者自定义的函数  
  //事件响应函数  
  viewTap: function() {  
    this.setData({  
      text: 'Set some data for updating view.'  
    }, function() {  
      //this is setData callback  
    })  
  },  
})
```

page.js 文件也可以获取 App 实例对象中的全局数据或方法，在获取全局数据时需要先在 Page 实例对象外通过 getApp() 方法获取 App 实例对象，具体操作如下所示。

```
//获取 App 实例对象  
var app = getApp()  
Page({  
  ...  
})
```

在小程序应用生命周期中调用 onShow 函数会跳转进入小程序页面的生命周期，小程序页面生命周期

的函数有 `onLoad`、`onShow`、`onReady`、`onHide`、`onUnload`，具体的页面生命周期如图 2-10 所示。

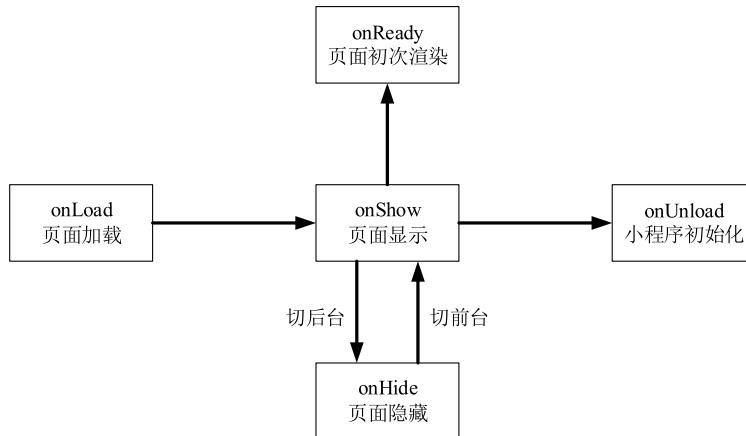


图 2-10 小程序页面生命周期

## 2.4 就业面试技巧与解析

在学习了微信小程序项目结构与框架后，下面选取了一些经常出现的面试题，对读者的学习内容进行一个简单检测，帮助读者加深对知识的理解与记忆。

### 2.4.1 面试技巧与解析（一）

**面试官：**微信小程序的项目结构以及主要文件类型都有哪些？

**应聘者：**微信小程序项目由页面文件、其他文件、主体文件、配置文件组成。页面文件存放在 `pages` 文件夹中，其他文件存放在 `utils` 文件夹中，主体文件存放在项目文件夹中，配置文件存放在项目文件夹中。

页面文件夹中，每个页面用一个单独的文件保存，每个页面由 `WXML`、`WXSS`、`JSON`、`JS` 四种文件组成。

(1) `WXML`——页面模板文件，通过 `WXML` 语言来编写页面内容。

(2) `JSON`——页面配置/设置文件，仅有 `window` 配置项，可以进行标题、`tabbar`、背景颜色等的设置。

(3) `WXSS`——页面样式文件，进行页面的布局和样式设置，也可以通过 `import` 导入一些其他样式文件来使用。

(4) `JS`——页面脚本逻辑文件，可以进行一些逻辑处理，网络请求操作，也可以监听页面的生命周期函数。

主体文件存放在项目文件夹下，由 `app` 开头的文件组成，包含 `app.json`、`app.js`、`app.wxss` 文件。

①`app.json`——配置文件入口，小程序的全局配置文件，在该文件中可以设置网络超时时间、页面底部 `tab`、页面路径、`window`，其中 `window` 配置项可以设置小程序所有页面的顶部背景颜色、文字颜色等。

②`app.js`——小程序的全局逻辑文件，可以在里边设置全局变量，监听小程序应用的生命周期函数。

③`app.wxss`——全局的样式文件，每个页面都可以使用这个全局样式，也可以使用自己页面样式来替代这个全局样式效果。

其他文件夹中主要存放一些 `JS` 文件，存放一些小程序页面通用的函数或者插件。

配置文件存放在项目文件夹下，包含 `project.config.json` 与 `sitemap.json` 文件。



(1) `project.config.json`——该文件可以查看、修改小程序的基本信息、本地设置、项目配置。

(2) `sitemap.json`——该文件用来设置小程序能否被搜索的权限。

一个基础的微信小程序项目目录结构如图 2-11 所示。

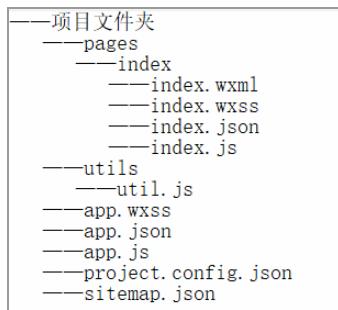


图 2-11 小程序项目的目录结构

## 2.4.2 面试技巧与解析（二）

**面试官：**简述微信小程序应用生命周期函数与页面生命周期函数的区别。

**应聘者：**微信小程序的生命周期分为两个层面，一个是应用等级，即 `app.js` 文件；一个是页面等级，即 `page.js` 文件。

小程序应用生命周期函数：

- (1) `onLaunch`——小程序初始化，全局仅调用一次。
- (2) `onShow`——监听小程序启动或切前台，小程序启动或者从后台进入前台显示时触发。
- (3) `onHide`——监听小程序切后台，小程序从前台进入后台时触发。
- (4) `onError`——错误监听函数，小程序发生脚本错误或者 API 调用报错时触发。

小程序页面生命周期函数：

- (1) `onLoad`——页面加载，每个页面调用一次。
- (2) `onShow`——页面显示，每次打开页面都调用。
- (3) `onReady`——初次渲染完成，每个页面调用一次。
- (4) `onHide`——页面隐藏，当 `navigateTo` 或底部 tab 切换时调用。
- (5) `onUnload`——页面卸载，当 `redirectTo` 或 `navigateBack` 时调用。