

iOS 11, Xcode 9、Swift 4的新特性, 展现开发一款优秀iOS应用程序的核心技术与开发技巧
包括CoreML机器学习、ARKit增强现实、自动化测试、性能分析以及加密与安全等移动应用高级技术

移动开发丛书
多媒体视频教学
素材源代码下载



—— 使用iOS 11+Xcode 9+Swift 4开发 ——

iOS 移动开发 从入门到精通

第2版

李发展 编著

清华大学出版社





iOS 移动开发 从入门到精通

第2版

李发展 编著

清华大学出版社
北京

内 容 简 介

本书介绍使用 iOS 11、Xcode 9 和 Swift 4.0 开发移动应用所需要的主要内容，全书共分三篇，第一篇为基础篇（第 1~10 章），主要介绍了 iOS 开发基础知识，包括 iOS 系统的历史、Swift 4.0 语言、UI 界面构建技术、视图控制器、常用的控件、表和集合视图、视图和层、声音、视频、动画等；第二篇为进阶篇（第 11~23 章），主要介绍了 Alamofire 网络服务、数据的持久化、多线程、CoreText、CoreImage、Storyboard、CoreML 机器学习、ARKit 增强现实、自动化测试、性能分析、本地化、加密与安全等实用技术；第三篇为实例篇（第 24~31 章），通过对实体层、表现层和逻辑层三层相关技术的讲解，向读者详细展示了 MVC 设计模式在项目开发中的应用方式，并最终创建一个成熟的商业项目。本书还提供了丰富的源代码和教学视频，可以大幅提高读者的学习效率。

本书采用丰富的实例和直观的代码展示，向读者形象地讲解了在创建一款优秀的 iOS 应用程序时所需要的大部分技巧，可以帮助读者轻松掌握并高效开发 iOS 程序。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目（CIP）数据

iOS 移动开发从入门到精通/李发展编著. —2 版. —北京：清华大学出版社，2018

（移动开发丛书）

ISBN 978-7-302-51157-1

I. ①i… II. ①李… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字（2018）第 203239 号

责任编辑：王金柱

封面设计：王 翔

责任校对：闫秀华

责任印制：丛怀宇

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座

邮 编：100084

社总机：010-62770175

邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：北京嘉实印刷有限公司

经 销：全国新华书店

开 本：190mm×260mm

印 张：38.75

字 数：992 千字

版 次：2018 年 10 月第 2 版

印 次：2018 年 10 月第 1 次印刷

定 价：118.00 元

产品编号：079481-01

前言

编写本书的目的

2011年9月，我第一次接触了 macOS 平台，并且学习了 Xcode 工具和 Objective-C 语言。当时市面上的学习资料比较少，所以学习 iOS 开发时，即便是对于我这个曾经从事过 3 年 PHP 编程和 4 年 J2EE 开发的程序员来说，也是比较吃力的事情。

而对于现在的读者来说，几乎有数十种 iOS 入门的教程可供选择。不过令人感到困惑的是，还没有一本书籍能够作为初学者向更高阶内容前进的桥梁。

本书针对没有任何 iOS 应用程序开发基础的开发者，从基本的 iOS 程序语言 Swift 4.0 开始介绍，接着对苹果公司提供的 API 开发框架 Cocoa Touch 的原理进行分类和详细讲解，进而对 iOS 的顶尖技术进行梳理和讲解。最后结合实际案例整合出一款符合 MVC 设计模式并包含大量实用 iOS 开发技能的商业 App，让读者能够成为在 iOS 应用开发领域中驰骋的高手。

iOS 系统已经陪伴我们走到了第 11 个年头，iPhone、iPod Touch、iPad 这些搭载着 iOS 系统的设备，以其丰富的功能和出色的用户体验改变了我们日常的生活、工作和娱乐方式。

当全世界都在赞叹苹果的 iOS 时，iOS 开发的就业大门也面向全世界打开了。随着移动互联网渐入佳境，越来越多的组织和个人开始进入移动互联网领域。在这个充满机会的移动互联网时代，赶快加入 iOS App 的开发行列吧！

本书主要内容

本书内容全面，几乎涵盖了 iOS 11 应用开发所需要的内容，全书内容言简意赅，讲解方法通俗易懂，特别适合初学者学习使用。

本书内容共分三篇，第一篇为基础篇，主要介绍了 iOS 开发的基础知识，包括 iOS 系统的历史、Swift 4.0 语言、UI 界面构建技术、视图控制器、常用的控件、表和集合视图、视图和层、声音、视频、动画等；第二篇为进阶篇，主要介绍了 Alamofire 网络服务、数据的持久化、多线程、CoreText、CoreImage、Storyboard、CoreML 机器学习、ARKit 增强现实、自动化测试、性能分析、本地化、加密与安全等实用技术；第三篇为实例篇，通过对实体层、表现层和逻辑层三层相关技术的讲解，向读者详细展示了 MVC 设计模式在项目开发中的应用方式，并最终创建一个成熟的商业项目。

本书采用丰富的实例、直观的代码展示向读者形象地讲解了在创建一款优秀的 iOS 应用程序时所需要的大部分技巧,帮助读者轻松掌握并高效开发 iOS 程序。

本书特点

内容全面: 包含 Swift 4.0 语言、Cocoa Touch、视图控制器、控件、表格、网络、地图、音频、视频、动画、触摸、手势、故事板、CoreData、性能测试、机器学习、增强现实、加密和解密等众多功能模块。这些内容是 iOS 开发必知必会的内容,需要读者重点掌握。

由浅入深: 内容按照原理讲解、实现过程、代码解析的编排顺序讲解,使读者朋友更容易掌握知识点。同时对重点代码做了大量的注释和讲解,以便于读者更加轻松地学习。

代码精简: 为了在有限的篇幅内讲解更多的实用功能,在本书中只给出了每个实例的核心代码及分析结构性代码,保留关键代码和核心代码,以节省篇幅,让本书更超值。

原理图解: 为复杂的原理讲解配有生动、详细的图例,以帮助读者轻松理解相关概念。

重点讲解: 对每个实例的核心功能都给予了专门讲解,以便于读者更好地掌握。

书中的实例完全适用于全新的 iOS 11 开发平台,同时也兼容 iOS 10 等旧的开发平台。

读者对象

通过对本书的学习,读者可以基本建立起 iOS 11 技术的思想框架,并且对 iOS 11 开发的过程有一个全面的了解。因此,本书适合 iOS 开发初学者、iOS 程序员、iPhone 开发人员、iPad 开发人员学习,同时也可以作为相关培训机构和大专院校相关专业的教学用书。

练习素材和视频教程

为方便读者学习,本书提供了练习素材和视频教程,具体使用方法如下:

本书源代码中的文件夹名称,如 17.1.3CoreTextProject_RichText,名称的前三个数字表示第 17 章的第 1 节的第 3 个示例项目的源代码。CoreTextProject 表示当前为 CoreText 相关的示例项目,而 RichText 为示例项目的名称。

本书中的例子和演示代码下载地址: <http://coolketang.com/download.html> (注意区分数字和英文字母大小写),所有代码均可直接使用 Xcode 开发工具运行(我会根据读者的意见反馈,随时更新位于此地址的源代码)。同时,相比本书第一版,第二版提供了更多、更实用、更精彩的配套视频教程,包含 Xcode 9、Swift 4 和 iOS 开发热门开源项目的分析与讲解。

本书中大部分示例可以在 Xcode 9 的 iOS 模拟器中运行。部分示例如相机、定位设备的使用需要连接真机进行测试。此时，你需要一个苹果开发者账户来访问大部分工具和文档，并且需要一个开发人员许可证来在真机上运行应用程序，对此请参考第 1 章的内容进行账号的注册和设置。

如果下载有问题，请发送电子邮件至 booksaga@126.com 获得帮助，邮件标题为“iOS 移动开发从入门到精通配书文件”。

勘误和支持

书中所有章节的内容和源代码都根据 Swift 4.0 和 iOS 11.0 版本进行了更新。如果你遇到有关兼容性的问题请联系我，我会发布更新并进行修改。

如果你对本书的内容有任何建议，或者发现了本书的一些错误，希望尽快联系我，这将对本书的后续版本有很大的帮助。我非常愿意听取任何能使本书变得更加完善的建议，并会不断致力于让本书更加完美。

如果你有关于本书的任何评论或者疑问，请访问微信公众号 [coolketang](#) 联系我。

致谢

首先衷心地感谢清华大学出版社的王金柱老师及各位编辑对本书的编写给予的帮助和关注。

感谢酷课堂 iOS 移动开发交流群的小伙伴、广大读者朋友们及时提出的各种反馈建议。

感谢好友兼第一版合著者王亮，以及冉玉玲、李爱民、谢美仙、李晓飞、朱小娟、李红梅、翟海岗、金善众、蔡银珠、金依灵、郑大翰、戴永威等人在写作过程中给予的支持和鼓励！感谢我的爱人金兵兵女士耐心地帮助校对书稿，提出改进意见，感谢大儿子李金诚、小女儿李开颜带给我新的见解和创意，愿你们健康快乐成长，用自己的努力去实现人生的一个个梦想！

最后，感谢这个时代给予每位有理想的人实现人生价值的机会！

编者

2018 年 7 月 1 日

目 录

第 1 章 开始 iOS 开发之旅 1

- 1.1 iOS 移动开发现状 1
 - 1.1.1 iOS 的历史 2
 - 1.1.2 iOS 11 的十一大变化 4
 - 1.1.3 iOS 应用开发的八个特点 6
- 1.2 iOS 11 和 Xcode 9 新增的开发技能 6
 - 1.2.1 ARKit 增强现实 7
 - 1.2.2 CoreML 机器学习 7
 - 1.2.3 Face ID 7
 - 1.2.4 拖放操作 8
 - 1.2.5 Core NFC 8
 - 1.2.6 安全区布局指南 9
 - 1.2.7 Asset UIColor 9
 - 1.2.8 Wireless Development 9
 - 1.2.9 GitHub 10
- 1.3 成为一名 iOS 开发者 10
 - 1.3.1 硬件上的需求 10
 - 1.3.2 加入 Apple 开发者计划 11
 - 1.3.3 下载安装 Xcode 和模拟器 12
 - 1.3.4 创建开发和发布配置文件
Provisioning Profile 13
- 1.4 创建一个基本的 iOS 项目 16
 - 1.4.1 项目的创建 16
 - 1.4.2 设置项目的信息 18
 - 1.4.3 设置应用的 Icon 图标 19
 - 1.4.4 设置应用的 Launch 图片 20
 - 1.4.5 编译并运行项目 22
- 1.5 iOS 应用的生命周期 23
 - 1.5.1 iOS 应用程序的状态 23
 - 1.5.2 演示 iOS 应用程序的
各种状态 24
- 1.6 小结 25

第 2 章 Xcode 9 IDE 和 iOS 模拟器 27

- 2.1 Xcode 9 的使用 27
 - 2.1.1 工具栏的使用 28
 - 2.1.2 导航器的功能介绍 29
 - 2.1.3 项目导航器中的内容 29
 - 2.1.4 创建一个组和 Swift 文件 30
 - 2.1.5 在项目中搜索文件 33
 - 2.1.6 代码的查找和替换 36
 - 2.1.7 代码的 Debug 调试 36
- 2.2 iOS 模拟器的使用 39
 - 2.2.1 使用多种 iOS 模拟器 39
 - 2.2.2 iOS 模拟器方向的切换 40
 - 2.2.3 iOS 模拟器语言的切换 41
 - 2.2.4 截取模拟器屏幕 41
 - 2.2.5 应用程序的退出和删除 41
- 2.3 小结 42

第 3 章 Swift 4.0 的基础知识 43

- 3.1 Swift 的基本概念 43
 - 3.1.1 Swift 语言的特点 44
 - 3.1.2 Swift 常量和变量的定义 44
 - 3.1.3 常量和变量的输出 45
 - 3.1.4 注释 46
 - 3.1.5 Playground 47
- 3.2 数据类型与运算 49
 - 3.2.1 布尔类型 49
 - 3.2.2 整数类型及实例属性 50
 - 3.2.3 浮点类型及实例属性 50
 - 3.2.4 字符和字符串 51
 - 3.2.5 元组类型 54
 - 3.2.6 一元、二元和三元运算符 55
 - 3.2.7 比较运算符 56

3.3 循环语句和条件判断语句	56	4.3.8 类的引用特征	93
3.3.1 for 循环	56	4.3.9 类型检查	94
3.3.2 while 循环语句	58	4.3.10 Any 和 AnyObject	96
3.3.3 repeat-while 循环语句	58	4.4 扩展、协议和其他	97
3.3.4 if 条件判断语句	59	4.4.1 扩展	97
3.3.5 switch 条件判断语句	60	4.4.2 协议	99
3.3.6 continue 语句和 fallthrough 语句	60	4.4.3 问号和感叹号	101
3.4 数组和字典	61	4.5 小结	103
3.4.1 数组	61	第5章 视图 UIView 和层 CALayer	104
3.4.2 多维数组	63	5.1 视图 UIView	104
3.4.3 数组的遍历	64	5.1.1 UIView 概述	104
3.4.4 字典	65	5.1.2 UIView 的外观属性	106
3.5 小结	67	5.1.3 UIView 的几何属性	110
第4章 Swift 4.0 的进阶内容	68	5.1.4 UIView 的嵌套和层次关系	112
4.1 函数	68	5.1.5 UIView 的交互属性	114
4.1.1 函数的定义和调用	68	5.1.6 UIView 的变形操作	116
4.1.2 为参数设置默认值	69	5.1.7 自定义 UIView 视图	118
4.1.3 设置可变的函数参数数量	70	5.2 CALayer 层	122
4.1.4 函数作为参数和返回类型	71	5.2.1 CALayer 边框	122
4.1.5 元组作为函数的返回类型 实现多个返回值	72	5.2.2 CALayer 投影	123
4.1.6 使用函数类型	73	5.2.3 CALayer 圆角	124
4.1.7 函数的输入输出参数	74	5.2.4 CALayer 渐变	126
4.1.8 函数的嵌套	74	5.3 小结	128
4.1.9 函数的递归用法	75	第6章 视图控制器 UIViewController	129
4.1.10 常用的内置函数	76	6.1 UIViewController 视图控制器 概述	129
4.2 枚举和结构体	78	6.1.1 视图控制器与 MVC	130
4.2.1 枚举	78	6.1.2 创建一个视图控制器	131
4.2.2 结构体	81	6.1.3 以模态方式弹出视图 控制器	134
4.3 类	84	6.1.4 视图控制器的生命周期	138
4.3.1 类的创建	84	6.1.5 视图控制器和 UIScrollView	140
4.3.2 类属性的 set 和 get 方法	86	6.2 UITabBarController	144
4.3.3 类属性的 willSet 和 didSet 方法	87	6.2.1 创建一个 UITabBarController	144
4.3.4 类的静态方法	89	6.2.2 改变 UITabBarController 索引	146
4.3.5 类的析构	89		
4.3.6 类的继承	90		
4.3.7 类的延迟加载属性 lazy	92		

6.2.3 在 Tab 图标上显示角标	147	第 8 章 UITableView 和 UICollectionView	188
6.3 UINavigationController	148	8.1 UITableView 概述	188
6.3.1 创建一个 UINavigationController	148	8.1.1 UITableViewCell 单元格	189
6.3.2 UINavigationController 视图 入栈和出栈	151	8.1.2 创建一个简单的表格	191
6.3.3 导航栏和工具栏的显示和 隐藏	155	8.1.3 自定义单元格 UITableViewCell	195
6.3.4 导航栏样式的修改	155	8.1.4 自定义单元格的高度	197
6.4 导航控制器和标签控制器	156	8.1.5 给表格添加章节和索引	200
6.5 小结	159	8.1.6 单元格的选择与取消选择	203
第 7 章 UIKit 常用控件	160	8.1.7 单元格的插入和删除	206
7.1 UIKit 框架和 Foundation 框架	160	8.1.8 移动单元格的位置	211
7.1.1 UIKit 框架的结构	161	8.1.9 表格之间的嵌套	214
7.1.2 UIKit 框架和视图对象的 交互	163	8.2 UICollectionView 概述	221
7.2 UIButton 按钮控件	164	8.2.1 UICollectionView 的实现 原理	222
7.2.1 不同样式的 UIButton 按钮	164	8.2.2 创建一个 UICollectionView 对象并添加点击事件	223
7.2.2 UIButton 按钮的状态	166	8.3 小结	227
7.2.3 UIButton 按钮的事件响应	167	第 9 章 触屏事件和触摸手势	228
7.3 UILabel 标签控件	169	9.1 触摸事件的监测	228
7.3.1 创建一个 UILabel 标签	170	9.1.1 对触摸事件阶段的监测	229
7.3.2 长文本标签的换行	170	9.1.2 监测是否触摸到某个视图	231
7.3.3 标签文字样式的自定义	172	9.1.3 实现图像视图的拖动	232
7.4 UISwitch 开关控件	172	9.2 触摸手势的使用	234
使用开关控件	173	9.2.1 实现单点触摸手势	235
7.5 UITextField 文本输入控件的使用	174	9.2.2 实现双击触摸手势	237
7.5.1 自定义 UITextField 的外观 和键盘	174	9.2.3 实现长按手势的监测	237
7.5.2 响应 UITextField 的键盘 通知	177	9.2.4 捏合手势的监测	239
7.6 UIAlertController	178	9.3 3D Touch 的使用	241
7.6.1 创建一个警告窗口	179	9.4 小结	244
7.6.2 创建一个动作表单	180	第 10 章 动画、声音与视频	245
7.7 UIDatePicker	182	10.1 制作动画	245
7.8 UIStackView	184	10.1.1 UIView 的动画	245
使用 UIStackView 组织视图	184	10.1.2 使用 UIImageView 播放帧 动画	251
7.9 小结	187	10.1.3 位移关键帧动画的制作	253

10.2 音频的播放	255	12.2.2 设置网页的 CSS 样式	296
10.2.1 使用 System Sound Services 播放简短声音	257	12.2.3 添加插入笔记图片的 方法	297
10.2.2 使用 AVAudioPlayer 播放 音乐	258	12.2.4 添加删除图片和获得 笔记的方法	299
10.3 视频的播放	261	12.2.5 实现笔记本的用户界面	299
10.3.1 使用 AVPlayer 播放影片	262	12.3 小结	302
10.3.2 实现影片的画中画功能	263	第 13 章 数据的持久化	303
10.4 小结	266	13.1 解析 plist 文件	303
第 11 章 网络请求和 Alamofire	267	13.1.1 创建和解析 plist 文件	304
11.1 网络请求	267	13.1.2 将信息写入 plist 文件	306
11.1.1 GET 和 POST 请求	267	13.2 使用 CoreData 进行数据的 持久化	308
11.1.2 同步请求和异步请求	268	13.2.1 创建 Core Data 实体并插入 数据	309
11.1.3 使用 URLSession 进行网络 请求	268	13.2.2 检索 Core Data 中的数据	312
11.1.4 使用 URLSession 下载 图片并显示进度	270	13.2.3 修改和删除 Core Data 数据	313
11.2 Alamofire 的使用	273	13.3 小结	316
11.2.1 Alamofire 的安装	274	第 14 章 多线程技术	317
11.2.2 使用 Alamofire 进行 GET 和 POST 请求	275	14.1 多线程的概念	317
11.2.3 使用 Alamofire 上传图片	277	14.1.1 任务、进程和线程	318
11.2.4 使用 AlamofireImage 缓存 图像	278	14.1.2 线程的 Stack Space	318
11.2.5 使用 AlamofireObjectMapper 将 JSON 数据映射为对象	281	14.1.3 线程的优先级	318
11.3 小结	283	14.1.4 线程的生命周期	319
第 12 章 UIWebView 与 HTML 5 技术	284	14.1.5 线程和 RunLoop	319
12.1 UIWebView 的使用	284	14.2 使用 Thread	320
12.1.1 使用 UIWebView 加载 网页	285	14.2.1 创建 Thread 下载图片	320
12.1.2 使用 UIWebView 加载 HTML 字符串和 GIF 动画	287	14.2.2 给 Thread 加锁使线程 同步	322
12.1.3 使用 UIWebView 加载 本地网页	289	14.3 Operation 技术的使用	325
12.1.4 使用 JavaScript 回调原生 程序	293	14.4 Grand Central Dispatch 的使用	328
12.2 创建一个简单的笔记本	296	14.4.1 使用 GCD 查询 IP 地址 信息	329
12.2.1 创建一个可编辑的 div	296	14.4.2 DispatchGroup 调度组的 使用	331
		14.5 小结	333

第 15 章 MapKit 地图 334

15.1 MapKit 的概念 334

15.1.1 MKMapView 的显示
区域 33515.1.2 MKMapView 的跟踪和
定位 33615.1.3 MKMapView 的交互选项和
装饰物 33615.1.4 创建一个 MKMapView
地图 337

15.1.5 MKMapView 代理的使用 339

15.1.6 自定义 annotationView 和
calloutView 341

15.1.7 Camera 的使用 344

15.2 小结 346

第 16 章 iOS 的图文混排技术 347

16.1 Core Text 的使用 347

16.1.1 文字排版方面的一些
基本概念 348

16.1.2 列出设备中的所有字体 350

16.1.3 富文本的创建 351

16.1.4 图文混排的实现 353

16.2 Text Kit 的使用 358

16.2.1 使用 Text Kit 对文字进行
分栏 35816.2.2 使用 Text Kit 制作图文
混排效果 360

16.3 小结 362

第 17 章 iOS 的图像和视频处理技术 363

17.1 Core Image 概述 363

17.1.1 使用 Core Image 的三个类
制作单色调效果 36417.1.2 使用 Core Image 给图片
添加马赛克 36617.1.3 使用 Core Image 处理视频
画面 36717.1.4 输出 Core Image 的所有
滤镜 37217.1.5 使用 CIDetector 和 CIFeature
进行人脸检测 373

17.2 UIBlurEffect 和 CGBlendMode 375

17.2.1 使用 UIBlurEffect 实现
毛玻璃效果 37517.2.2 使用 CGBlendMode 实现
颜色混合效果 376

17.3 小结 378

第 18 章 iOS 常用设备的使用 379

18.1 照片和相机的使用 379

18.1.1 读取照片中的图片 380

18.1.2 调用相机拍摄照片 384

18.2 定位设备的使用 385

18.2.1 使用定位设备获取设备
地理位置 38618.2.2 将地理坐标转换为实际
地名 389

18.3 陀螺仪和加速计设备的使用 391

18.3.1 陀螺仪的原理和应用 391

18.3.2 加速计的原理和应用 392

18.3.3 检测 iOS 设备的晃动 392

18.4 小结 394

第 19 章 使用 Storyboard 开发用户
界面 395

19.1 Storyboard 概述 395

19.1.1 Storyboard 的优缺点 396

19.1.2 Storyboard 的组织形式 397

19.1.3 创建一个 Storyboard 398

19.1.4 更改 Storyboard 的初始
视图控制器 400

19.2 在 Storyboard 中使用标准控件 401

19.2.1 添加三个控件 401

19.2.2 对控件进行连接 403

19.3 在 Storyboard 中使用表格控件 406

19.3.1 在 Storyboard 中添加 Table
View 406

19.3.2 添加 Table View Cell 406

19.3.3 设置 Table View 的
数据源 408

19.3.4 修改 ViewController.swift 文件	409	20.3 小结	451
19.4 在 Storyboard 中使用集合控件	410	第 21 章 iOS 应用内存泄漏的检测	452
19.4.1 添加 Collection View	410	21.1 Allocations 工具的使用	452
19.4.2 设置 Collection View Cell	410	21.2 Leaks 工具的使用	459
19.4.3 设置 Collection View 的数据源和代理	412	21.3 小结	462
19.4.4 修改 ViewController.swift 文件	413	第 22 章 iOS 应用程序的本地化	463
19.5 Storyboard 常见的层次结构	414	22.1 应用程序的本地化	463
19.6 Storyboard 之间的页面跳转和参数传递	418	22.1.1 应用程序名称的本地化	464
19.6.1 单个 Storyboard 中页面之间的参数传递	418	22.1.2 应用程序文字内容的本地化	468
19.6.2 多个 Storyboard 中的页面跳转及参数传递	421	22.1.3 Storyboard 的本地化	472
19.7 使用约束功能适配多分辨率	426	22.1.4 图片素材的本地化	474
19.7.1 打开并运行示例项目	427	22.2 小结	477
19.7.2 给父视图 MainView 添加约束	428	第 23 章 加密与安全	478
19.7.3 给三条分隔线添加约束	430	23.1 加密与安全	478
19.7.4 给两个小箭头添加约束	432	23.1.1 对称加密和非对称加密	479
19.7.5 给“添加新的会员”按钮添加约束	434	23.1.2 3DES 加密	479
19.7.6 给图标添加约束	435	23.1.3 SHA1 加密	485
19.8 小结	437	23.1.4 MD5 加密	487
第 20 章 单元测试和界面测试	438	23.1.5 越狱情况的判断	489
20.1 单元测试概述	438	23.2 小结	491
20.1.1 Xcode 的 XCTest 框架	439	第 24 章 iOS 11 新增开发功能	492
20.1.2 对自定义类的方法进行单元测试	442	24.1 Assets.xcassets 的新特性	492
20.1.3 使用 XCTest 框架进行性能测试	445	24.1.1 在 Assets.xcassets 中添加 Color Set	493
20.2 UI Testing 界面测试	446	24.1.2 在 Assets.xcassets 中添加矢量图片	495
20.2.1 往项目中插入 UI Testing	446	24.2 UIPasteConfiguration 和 Drag and Drop	496
20.2.2 制作 UI 测试的程序	448	24.2.1 实现 Drag to paste text 功能	497
20.2.3 录制 UI Testing 的步骤	450	24.2.2 实现拖放粘贴图片功能	499
20.2.4 播放 UI Testing 的录制步骤	450	24.3 UISwipeActionsConfiguration 的使用	502
		24.4 ARKit 增强现实框架	505
		24.4.1 什么是 AR 增强现实	506

24.4.2 基于 ARKit 的应用程序	506
24.4.3 创建一个 ARKit 项目	507
24.5 CoreML 机器学习框架	511
24.6 小结	515
第 25 章 在 iTunes Connect 上	
创建 App	516
25.1 创建一款新 App	516
25.1.1 登录 iTunes Connect 并 创建 App	517
25.1.2 创建 In-App Purchase 内购 项目	519
25.1.3 为 IAP 内购添加沙箱技术 测试账号	521
25.2 小结	522
第 26 章 版本控制和开源类库安装	523
26.1 将项目添加到 GitHub	523
26.1.1 Git 和 GitHub	523
26.1.2 创建一个 GitHub 账号	524
26.1.3 将新建项目上传到 GitHub	525
26.2 使用 CocoaPods 安装第三方 类库	528
26.2.1 安装 CocoaPods	528
26.2.2 安装第三方开源类库	529
26.3 小结	531
第 27 章 App 项目结构的搭建	532
27.1 项目目录结构的创建	532
27.1.1 创建 New Group 新组	532
27.1.2 Utils 目录中的工具类	533
27.1.3 Models 目录中的实体类	535
27.2 完成项目的信息配置	536
27.2.1 设置项目的通用属性	536
27.2.2 设置应用程序的图标	537
27.2.3 设置应用程序的启动图	538
27.3 制作引导页	539
27.3.1 下载和安装 BWWalkthrough	539
27.3.2 创建引导页	539
27.4 制作动态标签控制器	545
27.4.1 下载和安装 RAMAnimatedTabBarController	545
27.4.2 创建 Tab Bar Controller	546
27.4.3 制作 Tab 动画效果	547
27.5 小结	549
第 28 章 开发笔试和随机答题功能	
应用	550
28.1 开发笔试答题功能	550
28.1.1 创建自定义视图	551
28.1.2 FoldingCell 的下载和 安装	554
28.1.3 在故事板中搭建界面	554
28.1.4 创建答题页面 QuestionCell	556
28.1.5 创建 QuizViewController	559
28.2 开发随机答题功能	562
28.2.1 在故事板中搭建界面	562
28.2.2 创建 RandomViewController	563
28.3 小结	565
第 29 章 开发错题集和面试题功能	
应用	566
29.1 开发错题集功能	566
29.1.1 在故事板中搭建错题集 界面	567
29.1.2 创建 FailedItemsViewController	567
29.2 开发面试题功能	570
29.2.1 在故事板中搭建面试题 界面	570
29.2.2 创建 InterviewItemsViewController	571
29.3 小结	574

第 30 章 开发 App 盈利模式和分享传播应用	575
30.1 App 的盈利模式和分享传播	575
30.1.1 谷歌 AdMob 广告的申请和植入	576
30.1.2 在故事板中搭建界面	579
30.1.3 使用 Alamofire 获取产品列表	580
30.1.4 实现 In App Purchases 内购功能	582
30.1.5 使用 ShareSDK 实现 App 的社会化分享	585
30.1.6 多种分辨率的适配	591
30.2 小结	594

第 31 章 应用程序的打包和发布	595
31.1 iOS 应用商店概述	595
31.1.1 App Store 的重大更新	596
31.1.2 在 iTunes Connect 配置一个 App	598
31.1.3 打包 App 为 IPA 文件	600
31.1.4 自动化打包 App	602
31.1.5 上传 App 至 App Store	604
31.1.6 配置构建版本并提交审核	605
31.2 小结	606

第 1 章

开始 iOS 开发之旅

在学习 iOS 移动开发之前，首先应该将开发环境配置完成，并且对需要使用的开发工具有一定程度的了解。

本章首先向读者介绍 iOS 11 相比之前系统的一些新特性，使读者对目前主流的 iOS 系统在宏观上有一定了解，后面将一步步演示开发环境的搭建，并向读者介绍开发工具 Xcode 的常用功能。

通过本章的学习，读者能够掌握：

- (1) 了解 iOS 11 的新特性和新功能。
- (2) 申请免费的 Apple ID 账号。
- (3) 使用 Xcode 开发工具创建 iOS 工程。
- (4) 使用 Xcode 开发工具编写与调试程序。
- (5) 熟悉 Xcode 工程结构。
- (6) 编写第一个程序 HelloWorld。
- (7) iOS 应用程序的生命周期。

1.1 iOS 移动开发现状

iOS 是 Apple 公司（简称 Apple）推出的一款操作系统，是用于 Apple 移动设备的移动操作系统，和 Apple 的 macOS 操作系统一样，属于类 UNIX 的商业操作系统。

在 2007 年 1 月 9 日的 Macworld 大会上，Apple 公司公布了这个系统。由于最初是设计给 iPhone 使用的，因此该系统原名为 iPhone OS，即“iPhone 运行 OS X”，该系统使用和 macOS 一样的 UNIX 内核。

iPhone OS 系统后来被陆续应用到 iPod touch、iPad 等产品上，这样 iPad、iPhone、iPod touch

三个系列的产品都在使用 iPhone OS 系统, 因此在 2010 年的 WWDC 大会上, Apple 宣布其 iPhone OS 改名为 iOS, 并获得了思科 iOS 的名称授权。

经历了十余年时间的发展, iOS 系统经历过多个版本, 支持多代的 iPhone、iPad 和 iPod 设备, 目前已经成长为市场上影响力最大、功能最丰富、生态最完整的移动操作系统之一。

1.1.1 iOS 的历史

2007 年, 当乔布斯 (史蒂夫·乔布斯, Apple 公司联合创始人, 别名“乔帮主”) 带来 iPhone 手机的同时, iOS 移动系统也跟大家见面了。

如今, 该系统已经升级到了 iOS 11, 它原来的掌舵人也从 Scott Forstall 变为了乔纳森。在图 1-1 中列出了从 iOS 1.0 至 iOS 11.0 共 11 个版本的 Home Screen 的界面变化。



图 1-1 从 iOS 1.0 至 iOS 11.0 共 11 个版本的 Home Screen (摘自 coolketang.com)

第一代 iPhone OS 面世时, 存在着各种各样的问题, 诸如无法发送 MMS、无法使用复制粘贴功能、全封闭的系统, 并且只有黑色的背景, 无法自定义壁纸等。但是这些问题并不能阻止它成为一个伟大的操作系统。iOS 1.0 拥有大量的创新功能, 它展现了革命性的人机交互方式, iPhone 之前的智能手机都是使用键盘或手写的方式进行人机交互, 而 iPhone 则实现了一种全手控操作模式。所以谷歌地图在第一代 iPhone 上的用户体验比当时在其他平台上的版本好很多。

一年后, 伴随着 iPhone 3G 的出现, iPhone OS 才算是得到了真正意义上的完整。2008 年 7 月, Apple 为 iOS 发布了 App Store, 并为第三方应用提供了一种可管理的标准模式, 用来开发、浏览、下载和安装应用。如今的 App Store 绝对是 iOS 发展史上最重要的杀手应用, 它不仅帮助 Apple 建立了一个庞大的应用生态, 而且让众多的开发者因此找到了自己的商业模式和商业机会, 而 Apple 也因此积累了庞大数量的精品应用。

在 iPhone 3GS 发布的同时, Apple 也带来了 iOS 3.0 系统。iPhone 3GS 型号中的 S 也代表速度 (Speed) 的意思, 由于硬件上的提升, 因此 3GS 在速度上有了不小的升级。从此, iPhone 手机每

两年进入一次大升级，中间一年推出的设备型号的尾部常被标以 S，如 iPhone 4S、iPhone 5S 等，表示在设备性能上进行了一些升级。

此时，自 iOS 3.2 开始被应用于 iPad，Apple 针对大屏幕进行了 UI 优化设计，并且专门针对大屏幕做了功能优化设计。

在 iPhone 4 时代，iPhone OS 被正式更名为 iOS，iOS 4 开始支持多任务技术，此时的多任务方式不会受到后台应用占用内存的影响，也能保证不错的续航能力。多任务的菜单是通过双击 Home 键的方式调出的。

Siri 是 iOS 5 最大的亮点，实现了语音控制输入的功能。用户可以通过 Siri 技术，使用语音提问的方式进行人机交互。Apple 在 iOS 5 中取消了对数据线的依赖，可以使用 Wi-Fi 进行设备的激活，这意味着 iPad、iPhone 已经彻底变成了个人移动终端，减轻了对桌面环境的依赖。

Siri 语音助手在 iOS 6 中得到了加强，iOS 6 用户已经可以使用语音打开应用程序，甚至可以发送状态到 Facebook 和 Twitter。iOS 6 也被称为一个告别谷歌的系统版本，最大的一个变化是不再使用从第一代开始一直内置的谷歌地图，而是采用了 Apple 自己开发的地图服务。

iOS 系统最突出的一次演变是在 iOS 7 时发生的。iOS 掌门人换成乔纳森后，iOS 7 也进行了大调整，大家熟悉的拟物化图标全都被拍扁了，多任务界面也发生了巨大的变化，变得更加直观，用卡片式替代了原有的应用图标，如图 1-2 所示。



图 1-2 左图为拟物风格，右图为扁平化风格

这个变化是从第一代 iPhone 问世以来最明显的不同。

iOS 8 正式版本的系统于 2014 年 9 月 17 日向用户推送，它支持 iPhone 4S、iPad 2、iPod-touch（第 5 代）及更新的设备。在这个版本中，Apple Pay 终于和大家见面了，只是当时的中国内地用户还暂时无法使用该功能。在 iOS 8 中，Apple 也对一直被诟病的输入法进行了升级，终于开放了对第三方输入法的支持，也正是从该版本开始，iOS 越狱版的用户变得越来越少。

iOS 9 系统比 iOS 8 更加稳定，功能也更加全面，而且还更加开放。iOS 9 加入了更多的新功能，包括更加智能的 Siri 和省电模式。Siri 变得更加智能，甚至在你开口之前，新的主动建议功能就可以帮你把事情办妥；Slide Over、Split View 和画中画功能可以让用户能够以全新的方式在 iPad