

iOS 10

畅销书全新升级，累计印数6万多册

- Swift和Objective-C双语讲解
- 新增手势识别、Quartz 2D绘图技术、动画技术、用户扩展、用户通知、Core Data等
- 数百个项目案例+一个真实项目开发全过程
- 涵盖测试驱动开发、性能优化、版本控制和程序调试等



iOS 开发指南

从Hello World到App Store上架

(第5版)

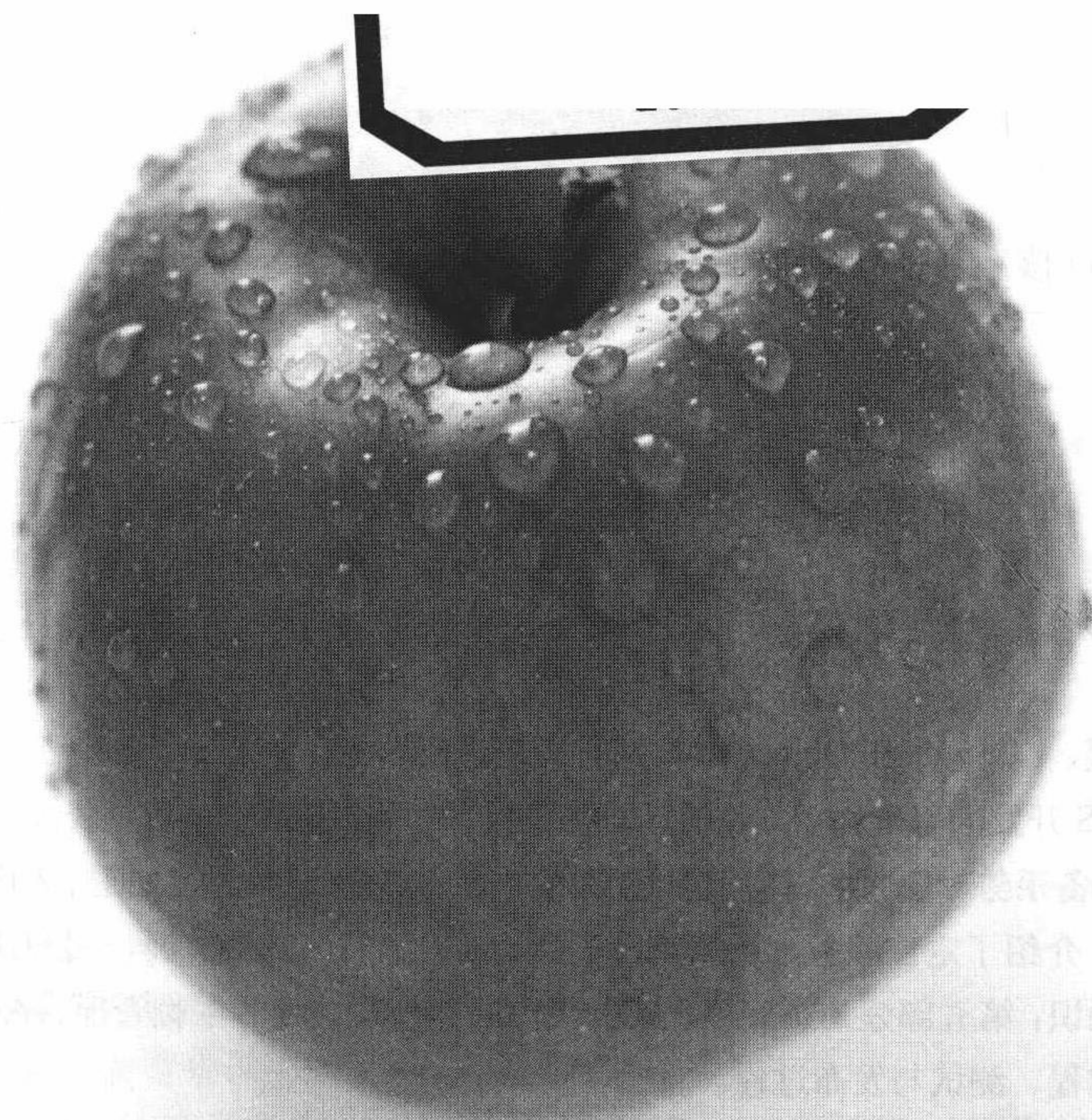
关东升 著



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



iOS 开发指南

从Hello World到App Store上架

(第5版)

关东升 著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

iOS开发指南：从Hello World到App Store上架 /
关东升著. -- 5版. -- 北京：人民邮电出版社，2017.6
(图灵原创)
ISBN 978-7-115-45063-0

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

中国版本图书馆CIP数据核字(2017)第040974号

内 容 提 要

本书是iOS开发权威教程，以Swift和Objective-C两种语言进行讲解。书中分5部分介绍如何从零起步编写并上线iOS应用：第一部分介绍iOS开发的基础知识，包括界面构建技术、基本控件、协议、表视图、界面布局、屏幕适配、导航、iPad应用开发、iOS设备手势、Quartz 2D、iOS动画等；第二部分介绍本地数据持久化、数据交换格式和Web Service，第三部分为进阶篇，介绍了定位服务、苹果地图、Contacts框架、应用扩展、用户通知等内容；第四部分介绍测试、调试和优化等相关知识；第五部分为实战篇，涵盖代码版本管理、项目依赖管理、App Store发布流程，以及一个真实iOS应用的设计、编程、测试与发布过程。

与上一版相比，本书不仅基于iOS 10进行了全面更新，还增加并修订了设备手势、Quartz 2D、动画、用户扩展、用户通知、Core Data等内容。

本书适合iOS开发人员阅读。

-
- ◆ 著 关东升
 - 责任编辑 王军花
 - 责任印制 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 三河市海波印务有限公司印刷
 - ◆ 开本：880×1230 1/16
 - 印张：46
 - 字数：1523千字
 - 印数：63 501 - 67 500册
 - 2017年6月第5版
 - 2017年6月河北第1次印刷
-

定价：119.00元

读者服务热线：(010)51095186转600 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字20170147号



站 在 巨 人 肩 上

Standing on Shoulders of Giants





站在巨人肩上

Standing on Shoulders of Giants



前言

北京时间2016年10月13日凌晨，苹果公司正式推出了iOS 10系统。此次，Swift 3.0也正式推出，Swift经过两年的发展已经非常健壮了，很多项目也转而使用Swift语言开发，但是很多老项目还使用Objective-C语言开发。在这个大背景下，我们原来编写的《iOS开发指南：从零基础到App Store上架》得到了广大读者的认可，很多读者希望我们将其升级为iOS 10版本，并且能够使用Swift和Objective-C两种语言进行讲解，并比较它们的不同之处。几个月过去了，我们终于在2016年年底将书稿提交给出版社。

内容和组织结构

本书是我们团队编写的iOS系列丛书中的一本，目的是使有Swift或Objective-C基础的程序员通过学习本书，从零基础学习如何在App Store上发布一款应用。全书共5部分。

第一部分为基础篇，共13章内容，介绍了iOS的一些基础知识。

第1章介绍了iOS的开发背景以及本书约定。

第2章使用故事板技术创建了HelloWorld，同时讨论了iOS工程模板、应用的运行机制和生命周期、视图的生命周期等，最后介绍了如何使用API帮助文档和官方案例。

第3章重点介绍了Cocoa Touch框架中构建界面的相关类，重点介绍了构建界面的三种技术：故事板、XIB和纯代码。

第4章重点介绍了标签、按钮、文本框、文本视图、开关、滑块、分段控件、网页控件、活动指示器、进度条、警告框、操作表、工具栏、导航栏等基本控件，而且每个示例都采用故事板和纯代码两种方式实现。

第5章首先介绍了数据源协议和委托协议，然后介绍了高级视图：选择器和集合视图。

第6章探讨了表视图的组成、表视图类的构成和表视图的分类，使我们对表视图有了一个整体上的认识。接下来，介绍了如何实现简单表视图和分节表视图，以及表视图中索引、搜索栏和分组的使用，然后学习了如何对表视图单元格进行删除、插入、移动等操作，最后介绍了表视图UI设计模式方面的内容。

第7章讲述了界面布局，其中介绍了iOS应用布局模式，并比较了传统界面布局和Auto Layout布局技术的区别。此外，还介绍了静态表布局和堆视图StackView等布局相关的内容。

第8章介绍了iOS多分辨率屏幕适配方法，其中主要介绍了Size Class技术。

第9章讨论了如何判断应用是否需要一个导航功能，并且知道在什么情况下选择平铺导航、标签导航、树形结构导航，或者同时综合使用这3种导航模式。

第10章首先介绍了iPhone和iPad设备使用场景上的差异，然后介绍了iPad树形结构导航、iPad模态视图和Popover视图，最后介绍了iOS分屏多任务。

第11章介绍了iOS设备手势识别，这些手势包括Tap（点击）、Long Press（长按）、Pan（拖动）、Swipe（滑动）、Rotation（旋转）、Pinch（手指的合拢和张开）和Screen Edge Pan（屏幕边缘平移）。

第12章主要介绍Quartz 2D绘图技术，其中包括UIKit绘图技术、绘制视图的路径、绘制图像和文本、坐标、Quartz坐标和坐标变换。

第13章介绍了iOS动画技术，其中包括视图动画和Core Animation框架等。

第二部分为数据与网络通信篇，共3章，介绍了iOS本地数据持久化、数据交换以及如何通过网络通信，实现数据传输。

第14章讨论了iOS本地数据持久化的问题。首先分析了数据存取的几种方式以及每种数据存取方式适合什么样的场景，然后分别举例介绍了每种存取方式的实现。

第15章重点介绍了数据交换格式，其中XML和JSON是主要的方式。

第16章介绍了Web Service，并重点讨论了REST风格Web Service，其中具体访问Web Service的API包括：NSURLSession、AFNetworking框架和Alamofire框架。

第三部分为进阶篇，共5章，介绍了iOS的一些高级知识。

第17章讨论了iOS中的定位服务技术，包括地理信息编码和反编码查询。此外，还介绍了苹果的微定位技术iBeacon。

第18章讨论了iOS苹果地图的使用，包括显示地图、添加标注以及跟踪用户位置变化等。最后，我们介绍了程序外地图的使用。

第19章首先介绍了如何在iOS中访问通讯录框架——Contacts框架，然后介绍了如何使用Contacts框架访问联系人信息、写入联系人信息等。最后，还介绍了如何使用ContactsUI框架提供系统界面实现选择联系人、显示和修改联系人以及创建联系人的操作。

第20章讨论了iOS 10应用扩展，首先介绍了应用扩展的概念，然后介绍了Today扩展、开发表情包和Message框架。

第21章介绍了iOS 10用户通知，其中首先介绍了用户通知的概念，接着介绍了如何开发本地通知和推送通知。

第四部分为测试、调试和优化篇，共4章，介绍了iOS高级内容，包括测试、调试和优化等相关知识。

第22章首先介绍了iOS中都有哪些调试工具并重点介绍了几个常用的工具，具体包括日志与断言的输出、异常栈报告分析，接下来讲解了如何在设备上调试应用，最后分析了Xcode设备管理工具的使用。

第23章讨论了测试驱动的iOS开发，学习了测试驱动开发流程，单元测试框架XCTest，以及如何基于分层架构进行单元测试。

第24章介绍了UI测试框架。在这一章最后，我们还介绍了基于分层架构进行UI测试。

第25章分析了iOS中的性能优化方法，其中包括内存优化、资源文件优化、延迟加载、持久化优化、使用可重用对象、并发处理与多核CPU等。

第五部分为实战篇，共4章，介绍了iOS项目开发过程中相关的技术，包括：代码版本管理、项目依赖管理，以及App Store发布流程，最后还从无到有地介绍了一个真实的iOS应用：MyNotes应用和2020东京奥运会应用。

第26章介绍了如何使用Git进行代码版本控制，其中包括Git服务器的搭建、Git常用命令和协同开发。此外，还介绍了在Xcode中如何配置和使用Git工具。

第27章讨论了iOS和macOS项目依赖管理工具，其中包括：CocoaPods和Carthage。

第28章探讨了如何在App Store上发布应用，介绍了应用的发布流程以及应用审核不通过的一些常见原因。

第29章介绍了完整的iOS应用分析设计、编程、测试和发布过程，其中采用了敏捷开发方法。此外，该项目采用分层架构设计，这对于学习iOS架构是非常重要的。

书中并没有包括多媒体等知识，我们会在另外一本iOS开发书中介绍，具体进展请读者关注智捷iOS课堂官方网站<http://www.51work6.com>。

本书服务网址

为了更好地为广大读者提供服务，我们专门为本书建立了一个服务网址www.51work6.com/book/ios15.php，大家可以查看相关出版进度，对书中内容发表评论，并提出宝贵意见。

源代码

书中包括了100多个完整的案例项目源代码，大家可以到本书网站www.51work6.com/book/ios15.php下载或者到图灵社区本书主页（www.ituring.com.cn/book/1932）免费注册下载。

免费相关视频

为了能够使广大读者快速学习本书，我们智捷课堂给广大读者提供了本书相关的免费视频，这些视频课程包括：从零开始学Swift、从零开始学Objective-C视频，以及部分iOS相关视频。读者凭书中夹带的书签下方优惠码，就可以购买价值50元内视频课程。具体使用说明：首先在智捷课堂视频平台（www.zhijieketang.com）注册并登录，然后找到相应课程，接着选择页面中的“购买课程”→“去支付”→“输入优惠码”→“使用”即可。

勘误与支持

我们在网站www.51work6.com/book/ios15.php中建立了一个勘误专区，可以及时地把书中的问题、失误和纠正反馈给广大读者。如果你发现了任何问题，均可以在网上留言，也可以发送电子邮件到eorient@sina.com，我们会在第一时间回复你。此外，你也可以通过新浪微博与我们联系，我的微博为@tony_关东升。

致谢

在此感谢图灵的王军花编辑给我们提供的宝贵意见，感谢智捷iOS课堂团队的赵志荣参与内容讨论和审核，感谢赵大羽老师手绘了书中全部草图，并从专业的角度修改书中图片，力求更加真实完美地奉献给广大读者。此外，还要感谢我的家人容忍我的忙碌，以及对我的关心和照顾，使我能抽出这么多时间，投入全部精力专心编写本书。

由于时间仓促，书中难免存在不妥之处，请读者原谅。

关东升
2016年11月于北京

欢迎加入

图灵社区 iTuring.cn

——最前沿的IT类电子书发售平台

电子出版的时代已经来临。在许多出版界同行还在犹豫彷徨的时候，图灵社区已经采取实际行动拥抱这个出版业巨变。作为国内第一家发售电子图书的IT类出版商，图灵社区目前为读者提供两种DRM-free的阅读体验：在线阅读和PDF。

相比纸质书，电子书具有许多明显的优势。它不仅发布快，更新容易，而且尽可能采用了彩色图片（即使有的书纸质版是黑白印刷的）。读者还可以方便地进行搜索、剪贴、复制和打印。

图灵社区进一步把传统出版流程与电子书出版业务紧密结合，目前已实现作译者网上交稿、编辑网上审稿、按章发布的电子出版模式。这种新的出版模式，我们称之为“敏捷出版”，它可以让读者以较快的速度了解到国外最新技术图书的内容，弥补以往翻译版技术书“出版即过时”的缺憾。同时，敏捷出版使得作、译、编、读的交流更为方便，可以提前消灭书稿中的错误，最大程度地保证图书出版的质量。

优惠提示：现在购买电子书，读者将获赠书款20%的社区银子，可用于兑换纸质样书。

——最方便的开放出版平台

图灵社区向读者开放在线写作功能，协助你实现自出版和开源出版的梦想。利用“合集”功能，你就能联合二三好友共同创作一部技术参考书，以免费或收费的形式提供给读者。（收费形式须经过图灵社区立项评审。）这极大地降低了出版的门槛。只要有写作的意愿，图灵社区就能帮助你实现这个梦想。成熟的书稿，有机会入选出版计划，同时出版纸质书。

图灵社区引进出版的外文图书，都将在立项后马上在社区公布。如果你有意翻译哪本图书，欢迎你来社区申请。只要你通过试译的考验，即可签约成为图灵的译者。当然，要想成功地完成一本书的翻译工作，是需要有坚强的毅力的。

——最直接的读者交流平台

在图灵社区，你可以十分方便地写文章、提交勘误、发表评论，以各种方式与作译者、编辑人员和其他读者进行交流互动。提交勘误还能够获赠社区银子。

你可以积极参与社区经常开展的访谈、乐译、评选等多种活动，赢取积分和银子，积累个人声望。

目 录

第一部分 基础篇

第 1 章 开篇综述.....2

- 1.1 iOS 概述.....2
 - 1.1.1 iOS 介绍.....2
 - 1.1.2 iOS 10 新特性.....2
- 1.2 开发环境及开发工具.....3
- 1.3 本书约定.....4
 - 1.3.1 案例代码约定.....4
 - 1.3.2 图示的约定.....5
 - 1.3.3 方法命名约定.....6
 - 1.3.4 构造函数命名约定.....7
 - 1.3.5 错误处理约定.....7

第 2 章 第一个 iOS 应用程序.....8

- 2.1 创建 HelloWorld 工程.....8
 - 2.1.1 通过 Xcode 创建工程.....9
 - 2.1.2 添加标签.....11
 - 2.1.3 运行应用.....13
 - 2.1.4 Xcode 中的 iOS 工程模板.....13
 - 2.1.5 应用剖析.....14
- 2.2 应用生命周期.....16
 - 2.2.1 非运行状态——应用启动场景.....18
 - 2.2.2 点击 Home 键——应用退出场景.....18
 - 2.2.3 挂起重新运行场景.....21
 - 2.2.4 内存清除：应用终止场景.....21
- 2.3 设置产品属性.....22
 - 2.3.1 Xcode 中的工程和目标.....22
 - 2.3.2 设置常用的产品属性.....24
- 2.4 iOS API 简介.....26
 - 2.4.1 API 概述.....26
 - 2.4.2 如何使用 API 帮助文档.....26
- 2.5 小结.....28

第 3 章 Cocoa Touch 框架与构建应用界面.....29

- 3.1 视图控制器.....29
 - 3.1.1 视图控制器种类.....29

- 3.1.2 视图的生命周期.....29
- 3.2 视图.....30
 - 3.2.1 UIView 继承层次结构.....30
 - 3.2.2 视图分类.....32
 - 3.2.3 应用界面构建层次.....33
- 3.3 使用故事板构建界面.....34
 - 3.3.1 什么是故事板.....34
 - 3.3.2 场景和过渡.....37
- 3.4 使用 XIB 文件构建界面.....38
 - 3.4.1 重构 HelloWorld.....38
 - 3.4.2 XIB 与故事板比较.....42
- 3.5 使用纯代码构建界面.....42
 - 3.5.1 重构 HelloWorld.....42
 - 3.5.2 视图的几个重要属性.....44
- 3.6 三种构建界面技术讨论.....46
 - 3.6.1 所见即所得.....46
 - 3.6.2 原型驱动开发.....46
 - 3.6.3 团队协作开发.....47
- 3.7 小结.....47

第 4 章 UIView 与视图.....48

- 4.1 标签与按钮.....48
 - 4.1.1 Interface Builder 实现.....48
 - 4.1.2 代码实现.....51
- 4.2 事件处理.....53
 - 4.2.1 Interface Builder 实现.....53
 - 4.2.2 代码实现.....55
- 4.3 访问视图.....57
 - 4.3.1 Interface Builder 实现.....57
 - 4.3.2 代码实现.....58
- 4.4 TextField 和 TextView.....60
 - 4.4.1 Interface Builder 实现.....60
 - 4.4.2 代码实现.....62
 - 4.4.3 键盘的打开和关闭.....64
 - 4.4.4 关闭和打开键盘的通知.....64
 - 4.4.5 键盘的种类.....65
- 4.5 开关控件、分段控件和滑块控件.....67

4.5.1 开关控件	67	6.4 添加搜索栏	138
4.5.2 分段控件	69	6.5 分节表视图	143
4.5.3 滑块控件	70	6.5.1 添加索引	143
4.6 Web 视图: WKWebView 类	72	6.5.2 分组	146
4.7 警告框和操作表	77	6.6 插入和删除单元格	147
4.7.1 使用 UIAlertController 实现警告框	77	6.6.1 Interface Builder 实现	149
4.7.2 使用 UIAlertController 实现操作表	79	6.6.2 代码实现	154
4.8 等待相关的控件与进度条	81	6.7 移动单元格	155
4.8.1 活动指示器 ActivityIndicatorView	81	6.8 表视图 UI 设计模式	157
4.8.2 进度条 ProgressView	83	6.8.1 分页模式	157
4.9 工具栏和导航栏	86	6.8.2 下拉刷新模式	158
4.9.1 工具栏	86	6.8.3 下拉刷新控件	159
4.9.2 导航栏	90	6.9 小结	160
4.10 小结	95	第 7 章 界面布局	161
第 5 章 委托协议、数据源协议与高级视图	96	7.1 界面布局概述	161
5.1 视图中的委托协议和数据源协议	96	7.1.1 表单布局模式	161
5.2 选择器	96	7.1.2 列表布局模式	161
5.2.1 日期选择器	96	7.1.3 网格布局模式	162
5.2.2 普通选择器	101	7.2 iOS 中各种“栏”	162
5.2.3 数据源协议与委托协议	106	7.3 传统界面布局问题	163
5.3 集合视图	108	7.4 Auto Layout 布局技术	164
5.3.1 集合视图的组成	108	7.4.1 在 Interface Builder 中管理 Auto Layout 约束	164
5.3.2 集合视图数据源协议与委托协议	109	7.4.2 案例: Auto Layout 布局	164
5.4 案例: 奥运会比赛项目	109	7.5 静态表布局	169
5.4.1 创建工程	110	7.5.1 什么是静态表	169
5.4.2 自定义集合视图单元格	110	7.5.2 案例: iMessage 应用登录界面	170
5.4.3 添加集合视图	112	7.6 使用堆视图 StackView	174
5.4.4 实现数据源协议	114	7.6.1 堆视图与布局	174
5.4.5 实现委托协议	115	7.6.2 案例: 堆视图布局	175
5.5 小结	115	7.7 小结	180
第 6 章 表视图	116	第 8 章 屏幕适配	181
6.1 概述	116	8.1 iOS 屏幕的多样性	181
6.1.1 表视图的组成	116	8.1.1 iOS 屏幕介绍	181
6.1.2 表视图的相关类	117	8.1.2 iOS 的 3 种分辨率	182
6.1.3 表视图分类	117	8.1.3 获得 iOS 设备的屏幕信息	183
6.1.4 单元格的组成和样式	119	8.2 Size Class 与 iOS 多屏幕适配	184
6.1.5 数据源协议与委托协议	121	8.2.1 在 Xcode 6 和 Xcode 7 中使用 Size Class 技术	184
6.2 简单表视图	122	8.2.2 Size Class 的九宫格	185
6.2.1 实现协议方法	122	8.2.3 Size Class 的四个象限	186
6.2.2 UIViewController 根视图控制器	123	8.2.4 在 Xcode 8 中使用 Size Class	187
6.2.3 UITableViewController 根视图控制器	129	8.2.5 案例: 使用 Size Class	190
6.3 自定义表视图单元格	133	8.3 资源目录与图片资源适配	192
6.3.1 Interface Builder 实现	133		
6.3.2 代码实现	137		

8.4 小结	195	10.4 Popover 视图	271
第 9 章 视图控制器与导航模式	196	10.4.1 Popover 相关 API	272
9.1 概述	196	10.4.2 PopoverViewSample 案例	272
9.1.1 视图控制器的种类	196	10.5 分屏多任务	276
9.1.2 导航模式	196	10.5.1 Slide Over 多任务	276
9.2 模态视图	197	10.5.2 分屏视图多任务	278
9.2.1 Interface Builder 实现	199	10.5.3 画中画多任务	280
9.2.2 代码实现	205	10.6 iPad 分屏多任务适配开发	280
9.3 平铺导航	206	10.6.1 分屏多任务前提条件	280
9.3.1 应用场景	206	10.6.2 分屏多任务适配	281
9.3.2 基于分屏导航的实现	208	10.7 小结	282
9.3.3 基于电子书导航的实现	213	第 11 章 手势识别	283
9.4 标签导航	219	11.1 手势种类	283
9.4.1 应用场景	219	11.2 手势识别器	284
9.4.2 Interface Builder 实现	220	11.2.1 视图对象与手势识别	284
9.4.3 代码实现	223	11.2.2 手势识别状态	285
9.5 树形结构导航	225	11.2.3 实例: 识别 Tap 手势	285
9.5.1 应用场景	225	11.2.4 实例: 识别 Long Press 手势	290
9.5.2 Interface Builder 实现	226	11.2.5 实例: 识别 Pan 手势	291
9.5.3 代码实现	233	11.2.6 实例: Swipe 手势	293
9.6 组合使用导航模式	235	11.2.7 实例: Rotation 手势	295
9.6.1 应用场景	235	11.2.8 实例: Pinch 手势	297
9.6.2 Interface Builder 实现	236	11.2.9 实例: Screen Edge Pan 手势	299
9.6.3 代码实现	243	11.3 小结	301
9.7 小结	246	第 12 章 Quartz 2D 绘图技术	302
第 10 章 iPad 应用开发	247	12.1 绘制技术基础	302
10.1 iPad 与 iPhone 应用开发的差异	247	12.1.1 视图绘制周期	302
10.1.1 应用场景不同	247	12.1.2 实例: 填充屏幕	302
10.1.2 导航模式不同	247	12.1.3 填充与描边	305
10.1.3 API 不同	247	12.1.4 绘制图像和文本	306
10.2 iPad 树形结构导航	248	12.2 Quartz 图形上下文	308
10.2.1 “邮件”应用中的树形结构导航	248	12.3 Quartz 路径	309
10.2.2 Master-Detail 应用程序模板	249	12.3.1 Quartz 路径概述	309
10.2.3 使用 Interface Builder 实现		12.3.2 实例: 使用贝塞尔曲线	311
SplitViewSample 案例	254	12.4 Quartz 坐标变换	312
10.2.4 使用代码实现 SplitViewSample		12.4.1 坐标系	312
案例	258	12.4.2 2D 图形的基本变换	315
10.3 iPad 模态视图	261	12.4.3 CTM 变换	317
10.3.1 “邮件”应用中的模态导航	261	12.4.4 仿射变换	322
10.3.2 iPad 模态导航相关 API	262	12.5 小结	322
10.3.3 使用 Interface Builder 实现		第 13 章 动画技术	323
ModalViewSample 案例	264	13.1 视图动画	323
10.3.4 使用代码实现 ModalViewSample		13.1.1 动画块	323
案例	268		

13.1.2 动画结束的处理	325	15.4 案例: MyNotes 应用 JSON 解码	387
13.1.3 过渡动画	326	15.5 小结	388
13.2 Core Animation 框架	329	第 16 章 REST Web Service	389
13.2.1 图层	329	16.1 概述	389
13.2.2 隐式动画	331	16.1.1 REST Web Service 概念	389
13.2.3 显式动画	333	16.1.2 HTTP 协议	389
13.2.4 关键帧动画	335	16.1.3 HTTPS 协议	390
13.2.5 使用路径	336	16.1.4 苹果 ATS 限制	390
13.3 小结	338	16.2 使用 NSURLSession	391
第二部分 数据与网络通信篇		16.2.1 NSURLSession API	392
第 14 章 数据持久化	340	16.2.2 简单会话实现 GET 请求	393
14.1 概述	340	16.2.3 默认会话实现 GET 请求	396
14.1.1 沙箱目录	340	16.2.4 实现 POST 请求	397
14.1.2 持久化方式	341	16.2.5 下载数据	398
14.2 实例: MyNotes 应用	341	16.3 实例: 使用 NSURLSession 重构 MyNotes	
14.3 属性列表	343	案例	400
14.4 使用 SQLite 数据库	349	16.3.1 插入方法	401
14.4.1 SQLite 数据类型	350	16.3.2 修改方法	402
14.4.2 添加 SQLite3 库	350	16.3.3 删除方法	402
14.4.3 配置 Swift 环境	351	16.4 使用 AFNetworking 框架	405
14.4.4 创建数据库	351	16.4.1 比较 ASIHTTPRequest、AFNetworking 和 MKNetworkKit	405
14.4.5 查询数据	353	16.4.2 安装和配置 AFNetworking 框架	405
14.4.6 修改数据	355	16.4.3 实现 GET 请求	407
14.5 iOS 10 中的 Core Data 技术	357	16.4.4 实现 POST 请求	408
14.5.1 对象关系映射技术	357	16.4.5 下载数据	408
14.5.2 添加 Core Data 支持	358	16.4.6 上传数据	410
14.5.3 Core Data 栈	359	16.5 使用为 Swift 设计的网络框架: Alamofire	412
14.6 案例: 采用 Core Data 重构 MyNotes 应用	360	16.5.1 安装和配置 Alamofire 框架	412
14.6.1 建模和生成实体	360	16.5.2 实现 GET 请求	413
14.6.2 Core Data 栈 DAO	366	16.5.3 实现 POST 请求	414
14.6.3 查询数据	367	16.5.4 下载数据	414
14.6.4 修改数据	369	16.5.5 上传数据	415
14.7 小结	370	16.6 反馈网络信息改善用户体验	416
第 15 章 数据交换格式	371	16.6.1 使用下拉刷新控件改善用户体验	416
15.1 XML 数据交换格式	372	16.6.2 使用活动指示器控件	419
15.1.1 XML 文档结构	372	16.6.3 使用网络活动指示器	420
15.1.2 解析 XML 文档	373	16.7 小结	421
15.2 案例: MyNotes 应用读取 XML 数据	374	第三部分 进阶篇	
15.2.1 使用 NSXML 解析	376	第 17 章 定位服务	424
15.2.2 使用 TBXML 解析	379	17.1 定位服务概述	424
15.3 JSON 数据交换格式	385	17.1.1 定位服务编程	425
15.3.1 JSON 文档结构	385	17.1.2 测试定位服务	428
15.3.2 JSON 数据编码/解码	386		

17.2 管理定位服务	432	20.2.1 使用 Today 应用扩展	479
17.2.1 应用启动与停止下的定位服务管理	432	20.2.2 实例：奥运会倒计时牌	480
17.2.2 视图切换下的定位服务管理	432	20.3 开发表情包	483
17.2.3 应用前后台切换下的定位服务管理	433	20.3.1 iMessage 应用	483
17.2.4 设置自动暂停位置服务	436	20.3.2 表情包	484
17.2.5 后台位置服务管理	436	20.3.3 实例：开发表情包	484
17.3 地理信息编码与反编码	437	20.4 Message 框架	488
17.3.1 地理信息反编码	437	20.4.1 Message 框架的主要 API	488
17.3.2 实例：地理信息反编码	437	20.4.2 消息布局	488
17.3.3 地理信息编码查询	438	20.4.3 消息扩展界面的收缩和展开	489
17.3.4 实例：地理信息编码查询	439	20.4.4 消息应用的生命周期	490
17.4 小结	440	20.4.5 消息会话	490
第 18 章 苹果地图应用	441	20.4.6 实例：高斯模糊滤镜	491
18.1 使用 iOS 苹果地图	441	20.5 小结	494
18.1.1 显示地图	441	第 21 章 重装上阵的 iOS 10 用户通知	495
18.1.2 显示 3D 地图	445	21.1 用户通知概述	495
18.2 添加标注	446	21.1.1 通知种类	495
18.2.1 实现查询	447	21.1.2 通知界面	495
18.2.2 在地图上添加标注	449	21.1.3 设置通知	497
18.3 跟踪用户位置变化	450	21.2 开发本地通知	498
18.4 使用程序外地图	451	21.2.1 开发本地通知案例	498
18.5 小结	454	21.2.2 请求授权	500
第 19 章 访问通讯录	455	21.2.3 通知的创建与发送	501
19.1 通讯录的安全访问设置	455	21.2.4 通知接收后的处理	503
19.2 使用 Contacts 框架读取联系人信息	456	21.3 开发推送通知	503
19.2.1 查询联系人	457	21.3.1 推送通知机理	504
19.2.2 读取单值属性	460	21.3.2 生成 SSL 证书	504
19.2.3 读取多值属性	461	21.3.3 iOS 客户端编程	513
19.2.4 读取图片属性	464	21.3.4 在 iOS 设备上运行客户端	515
19.3 使用 Contacts 框架写入联系人信息	464	21.3.5 内容提供者推送通知	518
19.3.1 创建联系人	465	21.4 小结	519
19.3.2 修改联系人	467	第四部分 测试、调试和优化篇	
19.3.3 删除联系人	468	第 22 章 找出程序中的 bug——调试	522
19.4 使用系统提供的界面	469	22.1 Xcode 调试工具	522
19.4.1 选择联系人	470	22.1.1 定位编译错误	522
19.4.2 显示和修改联系人	472	22.1.2 查看和显示日志	523
19.5 小结	475	22.1.3 设置和查看断点	524
第 20 章 iOS 10 应用扩展	476	22.1.4 调试工具	530
20.1 应用扩展概述	476	22.1.5 输出窗口	532
20.1.1 iOS 10 应用扩展种类	476	22.1.6 变量查看窗口	532
20.1.2 应用扩展工作原理	477	22.1.7 查看线程	533
20.1.3 应用扩展的生命周期	478	22.2 LLDB 调试工具	535
20.2 Today 应用扩展	478	22.2.1 断点命令	535

22.2.2 观察点命令.....	537	24.5.2 增加备忘录操作.....	580
22.2.3 查看变量和计算表达式命令.....	538	24.5.3 删除备忘录操作.....	581
22.3 日志与断言输出.....	541	24.5.4 显示备忘录详细信息操作.....	582
22.3.1 使用 NSLog 函数.....	541	24.6 小结.....	582
22.3.2 使用断言.....	542	第 25 章 让你的程序“飞”起来——性能 优化.....	583
22.4 异常栈报告分析.....	543	25.1 内存优化.....	583
22.4.1 跟踪异常栈.....	543	25.1.1 内存管理.....	583
22.4.2 分析栈报告.....	545	25.1.2 使用 Analyze 工具检查内存泄漏.....	583
22.5 在 iOS 设备上调试.....	546	25.1.3 使用 Instruments 工具检查内存泄漏.....	587
22.5.1 Xcode 设置.....	546	25.1.4 使用 Instruments 工具检查僵尸对象.....	592
22.5.2 设备设置.....	546	25.1.5 autorelease 的使用问题.....	594
22.6 Xcode 设备管理工具.....	548	25.1.6 响应内存警告.....	595
22.6.1 查看设备上的应用程序.....	548	25.2 优化资源文件.....	596
22.6.2 设备日志.....	550	25.2.1 图片文件优化.....	597
22.7 小结.....	551	25.2.2 音频文件优化.....	598
第 23 章 iOS 测试驱动与单元测试.....	552	25.3 延迟加载.....	599
23.1 测试驱动的软件开发概述.....	552	25.3.1 资源文件的延迟加载.....	599
23.1.1 测试驱动的软件开发流程.....	552	25.3.2 故事板文件的延迟加载.....	603
23.1.2 测试驱动的软件开发案例.....	553	25.3.3 XIB 文件的延迟加载.....	605
23.1.3 iOS 单元测试框架.....	555	25.4 数据持久化的优化.....	607
23.2 使用 XCTest 测试框架.....	555	25.4.1 使用文件.....	607
23.2.1 添加 XCTest 到工程.....	555	25.4.2 使用 SQLite 数据库.....	611
23.2.2 编写 XCTest 测试方法.....	558	25.4.3 使用 Core Data.....	612
23.2.3 运行测试用例目标.....	561	25.5 可重用对象的使用.....	613
23.2.4 分析测试报告.....	562	25.5.1 表视图中的可重用对象.....	614
23.3 异步单元测试.....	564	25.5.2 集合视图中的可重用对象.....	615
23.4 性能测试.....	567	25.5.3 地图视图中的可重用对象.....	617
23.4.1 测试用例代码.....	568	25.6 并发处理.....	618
23.4.2 分析测试结果.....	568	25.6.1 一些概念.....	618
23.5 小结.....	571	25.6.2 主线程阻塞问题.....	618
第 24 章 iOS 应用 UI 测试.....	572	25.6.3 选择 NSThread、NSOperation 还是 GCD.....	619
24.1 UI 测试概述.....	572	25.6.4 GCD 技术.....	619
24.2 添加 UI 测试到工程.....	572	25.7 小结.....	620
24.2.1 创建工程时添加 UI 测试框架.....	572	第五部分 实 战 篇	
24.2.2 在现有工程中添加 UI 测试用例目标.....	573	第 26 章 管理好你的程序代码——代码版本 控制.....	622
24.3 录制脚本.....	575	26.1 概述.....	622
24.3.1 录制之前的准备.....	575	26.1.1 版本控制历史.....	622
24.3.2 录制过程.....	575	26.1.2 基本概念.....	623
24.3.3 修改录制脚本.....	577	26.2 Git 代码版本控制.....	623
24.4 访问 UI 元素.....	577		
24.4.1 UI 元素的层次结构树.....	577		
24.4.2 UI 测试中相关 API.....	579		
24.5 表示逻辑组件测试最佳实践.....	579		
24.5.1 备忘录查询操作.....	580		

26.2.1	服务器搭建	623	28.4.1	功能问题	691
26.2.2	Gitolite 服务器管理	625	28.4.2	用户界面问题	691
26.2.3	Git 常用命令	627	28.4.3	商业问题	691
26.2.4	Git 分支	628	28.4.4	不当内容	691
26.2.5	Git 协同开发	632	28.4.5	其他问题	692
26.2.6	Xcode 中 Git 的配置与使用	634	28.5	小结	692
26.3	GitHub 代码托管服务	642	第 29 章 iOS 开发项目实战——2020 东京奥运会应用开发及 App Store 发布		
26.3.1	创建和配置 GitHub 账号	642	29.1	应用分析与设计	693
26.3.2	创建代码库	645	29.1.1	应用概述	693
26.3.3	删除代码库	646	29.1.2	需求分析	693
26.3.4	派生代码库	647	29.1.3	原型设计	694
26.3.5	管理组织	650	29.1.4	数据库设计	694
26.4	小结	653	29.2	任务 1: 创建应用工程	695
第 27 章 项目依赖管理			29.2.1	迭代 1.1: 创建工程	696
27.1	使用 CocoaPods 工具管理依赖	654	29.2.2	迭代 1.2: 发布到 GitHub	696
27.1.1	安装 CocoaPods	654	29.3	任务 2: 数据库与数据持久化逻辑组件开发	696
27.1.2	搜索库	655	29.3.1	迭代 2.1: 编写数据库 DDL 脚本	696
27.1.3	项目与第三方库搭配形式	656	29.3.2	迭代 2.2: 插入初始数据到数据库	697
27.1.4	实例: 静态链接库形式管理依赖	657	29.3.3	迭代 2.3: 数据库版本控制	697
27.1.5	实例: 框架形式管理依赖	659	29.3.4	迭代 2.4: 配置数据持久化逻辑组件	698
27.2	使用 Carthage 工具管理依赖	660	29.3.5	迭代 2.5: 编写实体类	698
27.2.1	安装 Carthage	660	29.3.6	迭代 2.6: 编写 DAO 类	700
27.2.2	项目与第三方库搭配形式	661	29.3.7	迭代 2.7: 数据库帮助类 DBHelper	704
27.2.3	Cartfile 文件	661	29.3.8	迭代 2.8: 发布到 GitHub	706
27.2.4	实例: 重构 MyNotes 依赖关系	662	29.4	任务 3: 表示逻辑组件开发	706
27.3	小结	664	29.4.1	迭代 3.1: 使用资源目录管理图片和图标资源	707
第 28 章 把应用放到 App Store 上			29.4.2	迭代 3.2: 根据原型设计初步设计故事板	708
28.1	收官	665	29.4.3	迭代 3.3: “首页”模块	709
28.1.1	在 Xcode 中添加图标	665	29.4.4	迭代 3.4: “比赛项目”模块	710
28.1.2	在 Xcode 中添加启动界面	668	29.4.5	迭代 3.5: “比赛日程”模块	714
28.1.3	调整 Identity 和 Deployment Info 属性	671	29.4.6	迭代 3.6: “倒计时”模块	718
28.2	为发布进行编译	672	29.4.7	迭代 3.7: “关于我们”模块	719
28.2.1	创建开发者证书	672	29.4.8	迭代 3.8: 发布到 GitHub	720
28.2.2	创建 App ID	675	29.5	任务 4: 收工	720
28.2.3	创建描述文件	676	29.5.1	迭代 4.1: 添加图标	720
28.2.4	发布编译	679	29.5.2	迭代 4.2: 设计和添加启动界面	720
28.3	发布上架	680	29.5.3	迭代 4.3: 性能测试与改善	721
28.3.1	创建应用	681	29.5.4	迭代 4.4: 发布到 GitHub	722
28.3.2	应用定价	683	29.5.5	迭代 4.5: 在 App Store 上发布应用	722
28.3.3	基本信息输入	684	29.6	小结	722
28.3.4	上传应用	687			
28.3.5	提交审核	689			
28.4	常见审核不通过的原因	691			

Part 1

第一部分

基础篇

本部分内容

- 第1章 开篇综述
- 第2章 第一个 iOS 应用程序
- 第3章 Cocoa Touch 框架与构建应用界面
- 第4章 UIView 与视图
- 第5章 委托协议、数据源协议与高级视图
- 第6章 表视图
- 第7章 界面布局
- 第8章 屏幕适配
- 第9章 视图控制器与导航模式
- 第10章 iPad 应用开发
- 第11章 手势识别
- 第12章 Quartz 2D 绘图技术
- 第13章 动画技术