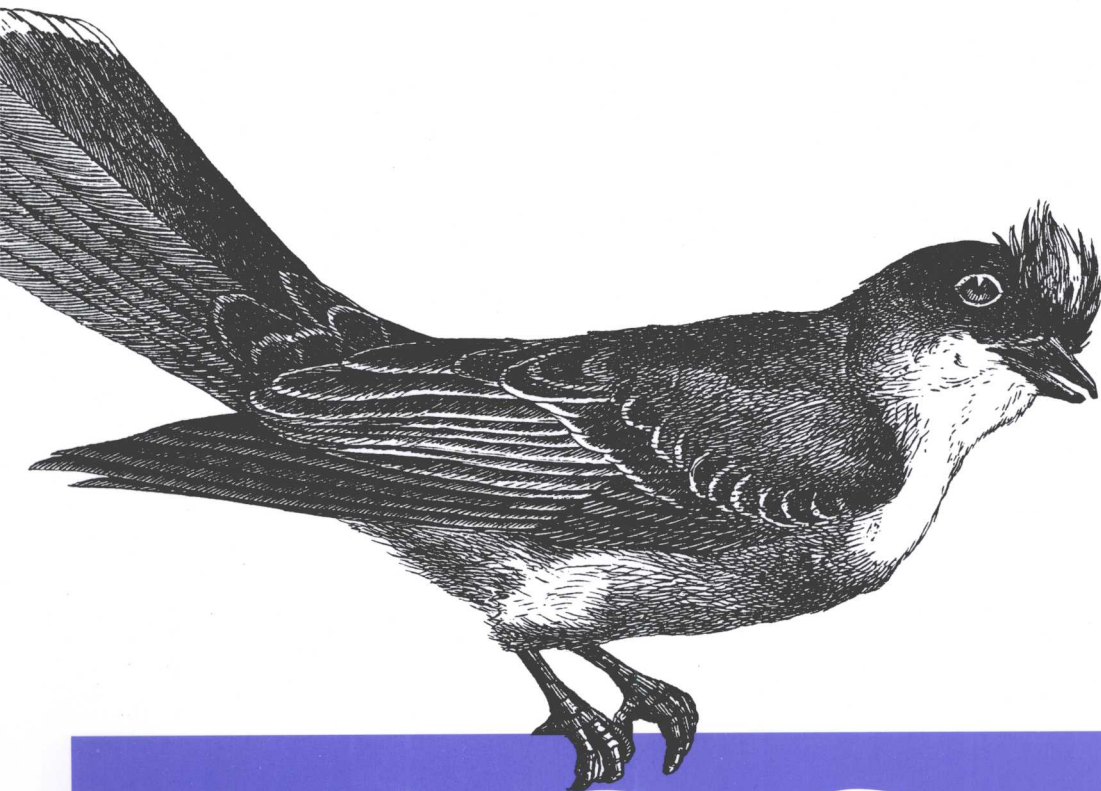




Programming iOS 5
系统学习iPhone/iPad应用开发的经典著作

第2版



iOS

程序设计

O'REILLY®



机械工业出版社
China Machine Press

Matt Neuburg 著
陈建设 刘洋 等译

013032952

TN929. 53
471

iOS 程序设计 (原书第 2 版)

Matt Neuburg 著

陈建设 刘洋 薛晓东 李明 译



O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'Reilly Media, Inc. 授权机械工业出版社出版



机械工业出版社
China Machine Press



北航

C1640722

TN929.53
471

图书在版编目 (CIP) 数据

869520810

iOS 程序设计 (原书第 2 版) / (美) 诺伊堡 (Neuburg, M.) 著; 陈建设, 刘洋等译. —北京: 机械工业出版社, 2013.4

(O'Reilly 精品图书系列)

书名原文: Programming iOS 5, Second Edition

ISBN 978-7-111-41847-4

I. i… II. ① 诺… ② 陈… ③ 刘… III. 移动电话机—应用程序—程序设计
IV. TN929.53

中国版本图书馆 CIP 数据核字 (2013) 第 053023 号

北京市版权局著作权合同登记

图字: 01-2012-5212 号

© 2012 by O'Reilly Media, Inc.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Machine Press, 2013.
Authorized translation of the English edition, 2012 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版 2012。

简体中文版由机械工业出版社出版 2013。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

封底无防伪标均为盗版

本书法律顾问

北京市展达律师事务所

书 名/ iOS程序设计 (原书第2版)

书 号/ ISBN 978-7-111-41847-4

责任编辑/ 谢晓芳

封面设计/ Karen Montgomery, 张健

出版发行/ 机械工业出版社

地 址/ 北京市西城区百万庄大街22号 (邮政编码 100037)

印 刷/ 北京市荣盛彩色印刷有限公司印刷

开 本/ 178毫米×233毫米 16开本 49.5印张

版 次/ 2013年4月第1版 2013年4月第1次印刷

定 价/ 129.00元 (册)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010)88378991; 88361066

购书热线: (010)68326294; 88379649; 68995259

投稿热线: (010)88379604

读者信箱: hzsj@hzbook.com

iOS 程序设计（原书第 2 版）

Programming iOS 5, Second Edition

O'Reilly Media, Inc. 介绍

O'Reilly Media通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自1978年开始，O'Reilly一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源码峰会，以至于开源软件运动以此命名；创立了Make杂志，从而成为DIY革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项O'Reilly的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

译者序

正如多年前苹果公司开创了个人 PC 的时代一样，iPhone 开创了移动互联网的时代，苹果公司再次以梦幻的舞步重新回到了舞台的中心。锐意创新和追求完美的苹果公司，在让用户得到美妙而极致体验的同时，也将 Objective-C 和 Cocoa 带入到更多程序员的视野中。很少有开发平台能够与 iOS 相提并论，其一流的系统和框架设计，能够帮助开发者迅速开发出非常优秀的作品。

现在，国内有越来越多的程序员投身到 iOS 应用和游戏开发的队伍中，无论是新手入门还是需要巩固现有技能，本书都是一本非常好的参考书籍。本书作者本着“授之以鱼不如授之以渔”的态度，深入浅出地介绍了 iOS 开发中许多必备的基础知识和高级技巧，以理论和实践相结合的方式，并辅以详尽的说明，指导读者掌握 iOS 的开发技能。作为译者和开发人员，在翻译的过程中，我也学习到了大量的知识。本书并没有为读者准备照搬就可以运行的代码，但是书中所讲的内容都旨在帮助读者理解每一步背后的含义，彻底掌握 iOS 开发技术。

本书的作者 Matt Neuburg 有着多年的开发经验，并且是专业技术杂志的特约编辑，有丰富的培训和咨询经验，并在大学中任教。由这样具备实战和教学经验的作者倾力打造的书，自然是一本不可多得的好书。本书涵盖了 iOS 的方方面面，从 Objective-C 语言基础和 Xcode 开发环境介绍，到对 Cocoa 框架和视图动画的深入剖析，再到线程和网络等高级的主题，本书作者洋洋洒洒、丰富细腻地描绘了 iOS 这个宏大的主题，可以说真正做到了授之以渔。

本书由陈建设、刘洋、薛晓东、李明合作翻译。感谢机械工业出版社的吴怡老师在整个翻译过程中耐心而细致的工作，及时的沟通协调，我们才得以顺利地完成本书的翻译工作。感谢机械工业出版社的谢晓芳编辑细心严谨的审稿，进一步保证了译文的质量。虽然我们在翻译过程中竭力以信达雅为目标，但是囿于经验和能力，书中的问题和疏漏在所难免，恳请广大读者批评指正。

陈建设

前言

苹果公司通过推出第 2 版 iPhone 系统做到了一件了不起的事：用已有的 Cocoa 计算机程序设计框架来为内存和速度都受限的小型触摸屏设备编写应用程序。结果，由此诞生的 Cocoa Touch 在很多方面比原来的 Cocoa 还要好。

编程框架都有自己的个性，框架的整体感觉能让人窥见设计者的目标和心态。第一次接触 Cocoa Touch 时，我对它的评价是：“哇，这真是个聪明人的杰作！”苹果公司虽然故意只在 iPhone 系统中提供了很少几个内置的界面部件，但它们却比 Mac OS X 中的对应部件强大而灵活得多，比如 UITableView 等。更重要的是，苹果公司还为开发者提供了一种非常完美的方法（UIViewController），帮助程序员用基于层次结构的方法对界面的显示、隐藏和替换进行控制。只能在 iPhone 小屏幕上运行的应用程序因此能够不断展开为多个界面，用户也不至于受到困扰。

更妙的是，苹果公司借此机会把 Cocoa 打造成全新的 Cocoa Touch，并证明了它的优势。Cocoa 是一个古老的框架，它的出现甚至早于 Mac OS X，在 NeXTStep 中就存在了。Cocoa 日积月累逐渐壮大，为向后兼容性保持了传统的设计。现在，苹果公司紧紧抓住了 Cocoa Touch 的出现这一机会，抛弃了陈旧的设计。

因此，Cocoa Touch 虽然在概念上基于 Mac OS X Cocoa，却与后者截然不同，也不由后者定义或限制。Cocoa Touch 是一个独立的框架，是一个更精练、更简单、更智能的 Cocoa。我能举出无数例子来赞美 Cocoa Touch 中经过仔细斟酌的系统性设计（以及对奥卡姆剃刀原则的严格遵循）。^{译注 1}例如 Mac OS X 的动画层是后来才匆匆加入视图中的，而 Cocoa Touch 中的所有视图从一开始就有与之对应的动画层。再如 Cocoa Touch 的内存管理策略更简单而清晰，当 nib 文件载入时，顶层对象的管理方法更明确。这样的例子数不胜数。

但是，Cocoa Touch 归根结底还是来源于 Cocoa。它仍然要求开发者具有 Objective-C 的知识。这不是一门脚本语言，它当然也不是针对非程序员的，诸如 HyperCard 的 HyperTalk 或苹果公司的 AppleScript 这样的语言。它是一门庞大而复杂的语言，学起来并不轻松。

iPhone 和 iPad 的流行以及该平台上大量价格低廉甚至免费的应用程序已经为苹果公司带来了大批新程序员，而且新人还会持续增多。他们在 iOS 平台上看到了 Mac OS X 中没有的机会。苹果公司年度的 WWDC 开发者大会也反映了这种趋势，苹果逐渐将重心从 Mac OS X 转移到 iOS 上。

虽然广大程序员对 iOS 的热忱令人欣喜，但这也造成了大量程序员急功近利，走路都没学会就开始跑。iOS 给了程序员强大的主动权，让他们能做想象中的任何事情。但是，iOS

译注 1：即“如无必要，勿增实体”。

同样要求程序员具有必备的基础知识。我经常在网上看到有人提出一些基础问题，他们明显已经沉浸在了开发新应用的乐趣中，却由于不熟悉基础知识而备受打击。

正是这样的现状让我想要写这本书，为读者介绍 iOS 的基础知识。我热爱 Cocoa，早就希望为它写点东西，iOS 的流行坚定了我这么做的决心。在写作过程中，本书的主题是“Cocoa Touch 编程基础”。在本书中，我将尝试用便于教学的欧几里德式的逻辑顺序提纲挈领地阐述健壮 iOS 编程所需的基础原则。本书介绍了 Objective-C 的基础知识（从 C 语言出发），面向对象编程原则，工具的使用建议，Cocoa 对象实例化、引用、相互通信及生命周期管理的完整流程，以及主要界面部件的常见用法与其他常见任务。和我的前一本书一样，我希望你能从本书发现新大陆而兴致勃勃地从头至尾阅读本书，也可将它放到手边以备随时参考。

写本书的目的不是为了替代苹果公司自己的文档和示例工程。苹果公司提供了很棒的在线文档，它们还在不断完善和改进。在准备写作的过程中，我非常依赖于这些资料。同时我还发现它们无法替代有组织且条理性强教程。由于无法保证读者会按固定的顺序访问在线文档，文档作者也不知道读者在阅读时已经储备了哪些知识。这让在线文档更适合作为参考资料，而不是教程。示例工程中的注释写得再完美，也仍然让人难以理解——它只是为了演示，无法进行教学。

本书则不是这样，本书由章节组成，按页码排序。因而我能够先介绍 C 语言之后再介绍 Objective-C。除了基础知识，本书还会介绍具体的开发经验。贯穿全书你都能看到提及“初学者常犯的错误”，这其中大部分是我自己犯过的错误。我会尝试说明哪儿有陷阱，否则你可能也会像我一样轻易就掉了进去。在阅读过程中，你能看到我一点点建立起很多完整的示例，或者从稍大的应用程序中拿出一小块进行分析和解释。能教会你编程的并不是一个完整的大项目，而是开发项目的思维过程。我希望你能从本书中获得的正是掌握这种思维过程。

iOS 是一个很大的框架，根本无法把所有内容浓缩到一本书中，就连这本大厚书也不行。其实把全部内容压缩到本书中也是不必要且不合适的。本书省略了 Cocoa Touch 中的若干内容，其中一些内容需要完整的一本书才能清晰描述，而另一些则是在你有了一定基础之后就能轻松地从苹果公司文档中学会的知识。本书只是一个开头，介绍的是基础知识。我希望本书的内容能够为你打下坚实的基础，足以让你应付今后有价值而且有趣的 iOS 编程中所遇到的各种问题。

本书约定

下面是关于本书印刷字体方面的一些约定：

斜体 (Italic)

表示新的术语、URL、电子邮件地址、文件名以及文件扩展名。

等宽字体 (Constant width)

用于列出程序清单，也用在正文中表示程序中的变量名、函数名、数据库、数据类型、环境变量、语句和关键字等内容。

等宽粗字 (**Constant width**)

表示需要由用户键入的命令或文本。

等宽斜体字符 (*Constant width*)

表示需要由用户提供的值或由上下文确定的值替换的内容。

注意：这个符号表示提示、建议或一般的注解。

警告：这个符号表示警告。

示例代码的使用

本书的目的是帮助你。一般来说，可以在自己的程序和文档中使用本书的代码。除非复制了相当数量的代码，否则你并不需要联系我们以征得同意。例如，在程序中使用本书中的几段示例代码不必获得许可，但销售或分发包含 O'Reilly 书籍示例代码的 CD-ROM 则必须取得我们的授权。引用文书的文字或代码回答问题无需授权，但在自己的产品文档中加入大量本书的示例代码则需要经过我们的许可。

我们欢迎你标明出处，但并不强求。标明出处时通常需要包含书名、作者、出版社以及 ISBN，例如，“*Programming iOS 5* by Matt Neuburg(O'Reilly). Copyright 2012 Matt Neuburg, 978-1-4493-1934-2”。如果你认为你对代码示例的使用已经超出以上的许可范围，我们很欢迎你通过 permissions@oreilly.com 联系我们。

Safari® 联机丛书

Safari 联机丛书是应需而变的数字图书馆，它允许你轻松检索超过 7500 本科技及创新性的参考书籍和视频资料。在此你能快速找到需要的答案。

获得订阅后，你就能访问联机丛书中的全部页面和视频。你能在手机或移动设备上阅读，能在新书上市前就开始阅读，还能访问仍在创作中的图书并向作者提供反馈。你能复制粘贴代码示例，组织收藏夹，下载部分章节，标记关键信息，做笔记，打印页面，还能使用大量其他帮你节省时间的功能。

O'Reilly Media 已经将本书加入 Safari 联机丛书服务中。要访问本书的电子版或来自 O'Reilly 及其他出版社的相关图书，请到 <http://my.safaribooksonline.com> 免费注册。

如何联系我们

请将本书的评论及疑问发给出版社。

美国：O'Reilly Media, Inc.

1005 Gravenstein Highway North

Sebastopol, CA 95472

中国：北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室（100035）

奥莱利技术咨询（北京）有限公司

我们还为本书建立了一个网页，其中包含勘误、示例以及其他信息。其地址是：

<http://shop.oreilly.com/product/0636920023562.do>

要评论本书或咨询相关技术问题，请发送电子邮件至：

bookquestions@oreilly.com

要获取更多有关我们书籍、课程、会议及新闻的信息，请访问 <http://www.oreilly.com>。

我们的 Facebook 页面：<http://facebook.com/oreilly>

在 Twitter 上关注我们：<http://twitter.com/oreillymedia>

在 YouTube 上观看我们的视频：<http://www.youtube.com/oreillymedia>

第 1 版致谢

工欲善其事，必先利其器。我在本书的写作过程中利用了很多非常优秀的工具。衷心感谢我的 Unicomp Model M 键盘（<http://pckeyboard.com>），没有它我完全无法想象如何才能轻松地敲出这么多文字。很多优秀的软件同样也给了我帮助，包括 TextMate（<http://macromates.com>）以及 AsciiDoc（<http://www.methods.co.nz/asciidoc>）。BBEdit（<http://www.barebones.com>）的 diff 显示方式也很有帮助。书中出现的屏幕截图使用 Snapz Pro X（<http://www.ambrosiasw.com>）以及 GraphicConverter（<http://www.lemkesoft.com>）制作。图表使用 OmniGraffle（<http://www.omnigroup.com>）绘制。

O'Reilly 优秀的制作工序能在我工作时就把 AsciiDoc 文本文件转化为 PDF 格式，让我在模拟的书籍样式中进行检查。要是没有它，以及 Early Program 程序帮助我在写作过程中定期向读者提供书籍内容更新，我从一开始就不会接受这个工作。我还特别要感谢工具专家 Abby Fox 一直以来给我提供的帮助。

我采纳了 Dave Smith 和 David Rowland 这两位技术审校者提出的很多建议。我还从很多读者中得到了物质和精神上的帮助，他们向我提供勘误，给我鼓励。我尤其有幸与编辑 Brian Jepson 合作，他热情地指导我使用 O'Reilly 工具，调整电子版格式，仔细检查图书中的错误，给予了我充分的信任。在我需要与 O'Reilly 各个部门进行交流时，他还受累充当了中间人的角色。我在 O'Reilly 写的所有书都离不开 Nancy Kotary 的帮助，她将我充斥着大量短句的写作风格进行了改善，今后也请多关照。本书中出现的所有错误当然由我来承担责任。

第 2 版致谢

我无疑要再次感谢我的编辑 Brain Jepson，是他促使我写出新版本。Brain，现在可以放下鞭子了！同样感谢 O'Reilly 团队，他们在各方面给予了我帮助。感谢热忱而忠实的读者给了我许多鼓励和建议。

第 2 版说明

为了介绍本书第 2 版和第 1 版的关系，我首先需要概括一下最近 iOS 和 Xcode 的版本历史。

我刚开始写本书第 1 版时，iPhone 的系统版本为 3.1.3，iPad 为 3.2。在写作过程中，iOS 4 和 iPhone 4 发布了，但 iOS 4 并不能在 iPad 上运行。后来，苹果公司发布了 iOS 4.2，这是第一版同时支持 iPhone 和 iPad 的系统。与此同时，Xcode 更新到了 3.2.5。iOS 4 也是第一版支持多任务的系统，开发人员不得不修改应用程序使其适应新世界的规则。

正当我最后一次修订第 1 版前，Xcode 3.2.6 和 iOS 4.3 发布了，同时还发布了期待已久的 Xcode 4 的第 1 个公开版本。Xcode 4 可以说是一个全新的集成开发环境，它在菜单、窗口、首选项等各方面都与 Xcode 3.2.x 不相同。一台 Mac 中能同时安装 Xcode 4 与 Xcode 3.2.x（最高支持 Snow Leopard），它们也能用于开发同一个工程。Xcode 4 移除了 Xcode 3.2.x 中的一些功能，因而一些开发者继续使用旧版本。这样的局面让我很难明确地描述开发过程。对于 iOS 开发来说，我还是推荐使用 Xcode 4，本书第 1 版就假设读者已经使用了 Xcode 4。

这就是 2011 年 5 月的情况。当时本书第 1 版正式出版，介绍了在 iOS 4 上的编程。

2011 年 10 月，苹果发布了 iOS 5。iOS 5 中的一些新功能对开发的细枝末节产生了很大的影响，本书因此有必要进行全面的修订。苹果同时还发布了 Xcode 4.2。由于这是官方支持 iOS 5 开发的最早 Xcode 版本，因此本书假设你正在使用这一版本（或较新版）的 Xcode。（通过一系列的技巧，更早的 Xcode 或许也可以用于 iOS 5 开发，但这种行为并不支持。）本书第 1 版包含 Xcode 3.2.x 中的一些菜单命令和其他界面说明，但这一版删除了这些内容。Xcode 4.2 有两个版本，分别在开发机器上运行 Snow Leopard（Mac OS X 10.6）和 Lion（Mac OS X 10.7）。虽然它们本应完全相同，但各版本都有自己的 bug，你可以两种都试试。

正当我在 2012 年 2 月完成本书第 2 版时，Xcode 4.3 发布了（仅支持 Lion）。它的主要创新在于磁盘文件的组织上：下载的 Xcode 4.3 不再是安装程序，不会在顶层 Developer 文件夹中包含众多附属文件及文件夹。它将整个 Developer 文件夹包含在文件包中（能用 Finder 的显示包内容命令查看）。当说到 Developer 文件夹时，你要知道这实际上指的是 `/Applications/Xcode.app/Content/Developer`。除此以外，我还没有找到 Xcode 4.2 和 Xcode 4.3 的其他重大区别。在本书中我有时会说“Xcode 4.2”，而这指的是 Xcode 4.2 或更新的版本。

这一版的主要目的是为 iOS 5 更新书中的内容。由于读者可能首次接触 iOS 开发，本书假设读者没有任何在 iOS 4 或者更老版本的系统上开发的知识。但有的读者可能像我一样从 iOS 4 过渡到 iOS 5，因而本书对 iOS 5 中的新功能重点进行了介绍。这些内容也能帮助新的 iOS 开发人员写出可以在 iOS 4 中运行的应用程序。但是，我这样做的目的并不是为读者提供过时的累赘信息。绝大多数支持 iOS 4 的设备都已经更新到了 iOS 5，我们能比较放心地假设大量用户使用的是 iOS 5。因此，不是一定要让应用程序能在旧版本 iOS 系统中运行的。从教学的角度上看，介绍老旧的方法也没什么好处。对于本书来说，当你确实想了解旧版系统的相关内容时还可以参考前一版。因此，从这一版中删除了 iOS 4.2 之前的内容。

这里可以举个例子，说明我在本书这一版中对 iOS 5 中新功能的态度以及教学方法。iOS

5 支持的 ARC（自动引用计数）彻底改变了 Objective-C 程序员管理对象内存的方法。甚至可以说，它把 Objective-C 变成了一门全新的语言。虽然可以选择在 iOS 编程中不使用 ARC，但是由于它对开发有极大的促进作用，我在本书中还是会假设你启用了 ARC。第 12 章介绍了内存管理，这里会像前一版那样向读者描述 ARC 出现前内存如何管理。但是在其他所有章节中，除非特别说明，所有的示例代码均应该在开启 ARC 的情况下执行。在 Xcode 中使用 File → New Project（文件→新工程）菜单项建立新的工程，随后选择任何 iOS 应用程序模板，如果在第 2 步中选复选框“Use Automatic Reference Counting”，则会启用 ARC。

iOS 5 还提供了串联圈这一新功能。串联圈类似于 nib 文件，允许开发者直接“绘制”界面。二者的主要区别在于一个串联圈文件能够完成多个 nib 文件的工作。它们在一些地方有所区别，使用方法也不尽相同。但由于它们性质相似，当我在本书中笼统地提到 nib 文件时，指的是 nib 或串联圈文件。虽然我会尽量使用完整的名称，但是当我忘了说 nib 或串联圈时希望各位读者能够谅解我。

最后，我还想对向本书前一版慷慨地提出了批评意见的读者说几句。他们或多或少认为：

1. 这不是一本“食谱书”^{译注 2}。
2. 这本书居然用了好几百页来讲基础知识。
3. 书里没有提供让读者能够立即上手的示例应用程序，甚至没有“Hello World”教程。

这些意见都相当中肯，我承认确实如此。但本书的目录以及这个前言一直试图说明：本书根本就不是这些意见中提到的那类书。直截了当地说，本书就是这样的书，而不是其他什么书。这才是我写这本书的最初目的。这些评论希望本书应该是其他什么类型的。但在我看来，那样的书已经够多了，稀缺的正是本书这样的。和我以前进行写作的原因一样，写这本书的原因是我找不到我需要的书，我不得不自己写一本。我希望本书能对那些确实需要这样一本书的人提供帮助。希望本书是其他类型的读者应该购买别的书，而不是在本书的网站上批评它与想象中相差甚远。我从一开始就没有打算把本书写成那样。

本书会按顺序分重点介绍实际 iOS 开发所需要的相关知识。这正好与“食谱书”完全相反。本书没有为读者准备简单的教程，只要填入自己的代码就能创建一个应用程序。我认为授之以鱼不如授之以渔。市面上充斥着大量草率地忽略基础知识的书，它们让读者自己学习 Objective-C，鼓励大量开发者在尚未打好基础时就尝试进行开发。根据这些开发者提出的问题以及产生的迷惑，我相信他们这时完全不明白自己在做什么。

本书就想纠正这种现象，这要求我介绍大量的基础知识。本书并未强迫你从头至尾阅读所有的基础知识。如果你真的认为已经掌握了所有有关 C、Objective-C、Xcode、Cocoa、视图、绘图以及其他东西的知识（你真的确定吗？），尽管跳过开头的几章吧！但请不要对我需要的人讲解基础知识所费的笔墨而不快。

这就解释了为什么本书没有立即让读者上手编程。本书不是写给那些只能鹦鹉学舌般照着教程写代码的“程序员”的。我的书所讲的是真正的知识，是历经辛苦才能学到的知识。书中所讲的内容能帮你理解进行 iOS 5 开发的每一步究竟做了什么。这要求投入大量的时

译注 2：通过按步骤写出的教程指导读者编写应用程序的书籍。

间和精力，在开始做任何有实用价值的演示前，我需要介绍大量基础知识。

或许他们误解的部分原因在于没有注意或理解，我之前说的本书会以“便于教学的欧几里德式的逻辑顺序”书写。有些人不知道“欧几里德式”是什么意思。欧几里德写出了第一本世代相传的数学教科书。该书的特色之一是：如果概念 B 依赖于概念 A，就会先介绍概念 A。没有什么会推迟，欧几里德绝不会说：“我会在之后解释（证明、讨论）这个东西，现在先姑且相信我说的话。”我试着复制欧几里德的模型。这里举一个例子。每一个 iOS 应用程序都会用视图控制器，因此，读者在读到第 19 章描述视图控制器的内容前不会知道真实的 iOS 应用程序会涉及哪些东西。但要理解视图控制器，需要先理解它控制的究竟是什么；因而第 14 章以及第四部分剩下的内容都在讨论视图。要掌握视图控制器和视图间的关系，需要先学会 Cocoa 的架构模式，例如，生命期事件以及响应者链；这就是第 11 章以及第三部分讨论的内容。视图控制器的视图通常会从 nib 中载入，这一点在第 7 章中讨论。以上所有这些都要求首先掌握操作它们时用到的编程语言 Objective-C，因此有了第 3 章。最后，由于 Objective-C 是 C 语言的超集，这样就有了第 1 章。可以看出，要更早期地接触到视图控制器几乎是不可能的。我的论证到此结束。

另外，还有读者抱怨本书没有任何可运行的代码。很奇怪，书中到处都有示例代码啊！所有这些代码都能从我的 GitHub 网站下载（<https://github.com/mattneub>）。你可以获取这些代码，在 Xcode 中运行，尽情进行研究。你可以一边阅读一边运行相关的代码，而我也建议你这么做。但是，这些代码并不是为了好玩才写的，它们的作用是帮助你理解书中相应的概念。

在任何情况下，完美的“Hello World”教程一点也不稀奇，网上到处都有，包括苹果公司自己的网站（<http://developer.apple.com/library/ios/#documentation/iPhone/Conceptual/iPhone101/Articles/>）。根本不需要我来演示写一个简单的 iPhone 应用程序是多么有趣和轻松。

不过，有人强烈建议我提供“Hello World”程序，否则我的工作就没有做完。既然如此，下面我就给出一个例子，你们可以根据以下步骤一步一步完成：

1. 安装 Xcode，启动 Xcode。
2. 选择 File → New → New Project（文件→新建→新工程）。
3. 在 Choose a template 对话框中，单击左侧的“iOS”（不是“Mac OS X”）下的“Application”。在对话框右侧，单击“Empty Application”。单击 Next 按钮。
4. 在“Product Name”中输入 Hello。如果“Company Identifier”为空，则输入公司标识符，例如 com.yourLastName.yourFirstName。在“Device Family”中选择 iPhone。确保 3 个复选框都没有选中。单击 Next 按钮。
5. 导航到桌面。不要选中“Create local git repository”复选框。单击 Create 按钮。
6. 这时工程窗口将会打开。按 Command+I 组合键。在窗口左半部分单击 *AppDelegate.m*。
7. 现在需要在窗口中部的编辑器中工作。在代码中找到“Override point for customization after application launch.”这一行。将光标移动到该行最右侧并按几次 Return 键，创建一些空格。在空格中的某处单击，键入下面的代码：

```
CGRect f = label.frame;
f.origin = CGPointMake(100,100);
label.frame = f;
[self.window addSubview:label];
```

8. 按 Command+R 组合键，如果有窗口弹出询问是否保存，请接受。
9. 稍等片刻，iOS Simulator 程序会自动启动，里面会有一个白色的窗口，其中显示了“Hello, world!”。

恭喜！你完成了第一个“Hello world”程序。这难道不简单吗？难道不无聊吗？难道还有什么重要的意义吗？实际上，你自己还能做得比写刚刚那个程序更好吧？如果你想知道这个问题的答案，想学到足够的知识以便走得更远，继续阅读这本书吧！

作者介绍

Matt Neuburg 博士 曾在多所大学及学院任教。他从1968年起就开始了关于计算机的程序设计，开发了Mac OS X和iOS的多个应用程序。他曾是MacTech杂志的编辑，现在是TidBITS的长期特约编辑。他以前所著的O'Reilly图书包括《Frontier: The Definitive Guide》、《REALbasic: The Definitive Guide》和《AppleScript: The Definitive Guide》。他通过撰写图书、文章、软件文档，编写应用程序，提供顾问和培训服务谋生。

封面介绍

本书封面上的动物是王霸鹟（kingbird），是北美洲13种鸣禽构成的霸鹟属（tyrannus）中的一种。王霸鹟的主要食物是昆虫，它们经常从栖息地俯冲到半空中捕食昆虫。此外，浆果和水果也是它们的食物。它们具有长而尖的翅膀，雄性王霸鹟会通过复杂的飞行动作进行求偶。这种鸟类的属名和物种名都暗示它们具有对自己的地盘、繁殖区域以及配偶的攻击性保护行为。有记录表明它们会攻击红尾鹰（体型超过它们自身的20倍），将蓝松鸦撞离树木，驱赶乌鸦和渡鸦。（由于它具有抗争比自己大得多的其他鸟类的习性，把灰王霸鹟用在波多黎各民族主义者的标志里）。

“王霸鹟”通常指的是东王霸鹟（T. tyrannus），体型大小在平均水准（7.5～9英寸长，翼展13～15英寸），广泛分布在整个北美洲。这种常见鸟类的头部和背部颜色较深，喉部、胸部和腹部为白色。它难得一见的头冠上有红点斑。它们断断续续的尖锐嗡嗡叫声经常被描述为类似“电线间的火花”或电网。

推荐阅读



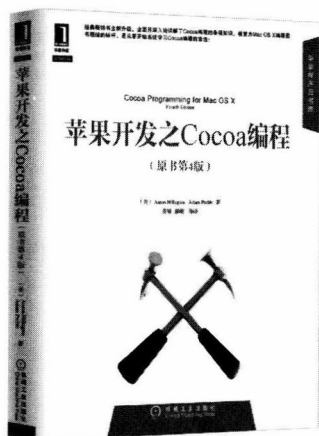
内容全面，系统讲解开发企业级iPhone应用所需掌握的各项核心技术，以及各种工具和框架的用法，包含大量技巧和最佳实践

实战性强，不仅为各个知识点精心设计了能辅助读者理解的小案例，而且还有能指导读者实践的大案例，具备极强的可操作性



Amazon五星级畅销书，作者权威，在全球iOS/Mac开发者社区享有盛誉！

完美地展现了测试驱动开发方法与iOS开发的结合，能使iOS开发者在产品需求、软件设计、测试有效性与开发效率之间达成很好的平衡



经典畅销书全新升级，系统且深入地讲解了Cocoa编程的各项知识，被誉为Mac OS X编程图书领域的标杆，被公认为从零开始学习Cocoa的首选！



资深iOS开发工程师撰写，Amazon五星级畅销书，国际Mac和iPhone开发者社区CocoaHeads联合创始人Mark Dalrymple等数位专家联袂推荐！

推荐阅读



深入Android应用开发：核心技术解析与最佳实践

作者：苗忠良 等著 ISBN: 978-7-111-37957-7 定价：79.00元

内容简介

如何才能真正进阶为Android应用开发高手？必须深入理解Android核心技术的底层原理和在开发中总结并使用各种最佳实践，别无他法！本书以Android的源代码为主，SDK为辅，针对应用开发者的需求，对各种核心技术的使用方法、底层原理和实现细节进行了深入而详细的讲解，同时辅之以大量案例和最佳实践，为开发者的进阶修炼和开发高质量的应用提供了绝佳指导。

Android开发精要

作者：范怀宇 著 ISBN: 978-7-111-39058-9 定价：69.00元

内容简介

这如何才能写出贴近Android设计理念、能够更加高效和可靠运行的Android应用？通过Android的源代码去了解其底层实现细节是最重要的方法之一！

AIR Android应用开发实战

作者：邱彦林 著 ISBN: 978-7-111-39177-7 定价：69.00元

内容简介

本书由资深Adobe技术专家兼资深Android应用开发工程师亲自执笔，既系统全面地讲解了如何利用Adobe AIR技术开发Android应用，又细致深入地讲解了如何将已有的基于PC的AIR应用移植到Android设备上。不仅包含大量实战案例，而且还包括大量建议和最佳实践，是系统学考书。



北航

C1640722