

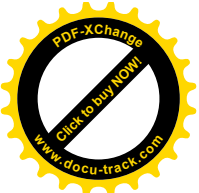
马上着手开发 iOS 应用程序

开发 iOS 应用程序既有趣又回报丰厚，如果您是一位新手，自然想知道从哪里入手。本路线图提供了 iOS 应用程序开发的绝佳起点。在 Mac 电脑上，您可以创建在 iPad、iPhone 和 iPod touch 上运行的 iOS 应用程序。遵循本路线图以了解如何取得开发工具，理解主要概念及最佳实践，并学会查找更多信息。

继续遵循此路线图，您将使用到 Xcode 和 iOS SDK (Apple 提供的开发工具)。您将了解 Objective-C (驱动所有 iOS 应用程序和框架的程序设计语言) 的编程基础知识，并将探索 Cocoa Touch 框架。您将创建一个简单的 iOS 应用程序，并学会在设备上进行测试。最后，您还会学到如何将应用程序提交到 App Store。



路线图中的每个页面，都介绍一个新主题，并链接到一篇或多篇有关该主题的简短文章。路线图只是提供基本的入门知识，最后一页“接下来做什么”包含您接下来应该阅读的文稿链接。完成了路线图，您就具备了进一步了解各个主题的能力，并且掌握到着手开发 iOS 应用程序的知识、工具和技能。



如果您是 Mac 开发者：您其实已经掌握了不少开发 iOS 应用程序的知识。可是，尽管 iOS 和 OS X 使用相同的开发工具和开发语言，两者仍然存在着一些重大的差异（您将在路线图的学习过程中发现这点）。有关这些平台差异的全面描述，请参阅 [iOS Technology Overview](#)（iOS 技术概述）中的“[Migrating from Cocoa](#)”（从 Cocoa 迁移）。

设置

您在开发应用程序时，会使用到 iOS 软件开发套件 (SDK) 以及 Xcode，即 Apple 的集成开发环境 (IDE)。

Xcode 包括源代码编辑器、图形用户界面编辑器及其他许多功能，为您开发完美的 iPhone、iPod touch 和 iPad 应用程序，提供了所需要的全部资源。大多数应用程序开发工具集中显示在一个窗口中，Xcode 称之为**工作区窗口**。在此窗口内，您可以顺畅地从代码编写转换到代码调试，再到用户界面设计。iOS SDK 扩展了 Xcode 工具集，包含 iOS 专用的工具、编译器和框架。

Mac App Store



Xcode 免费

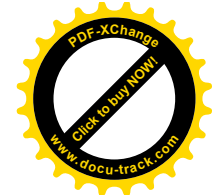
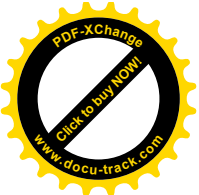


下载



开始之前：

1. 下载最新版本的 Xcode。



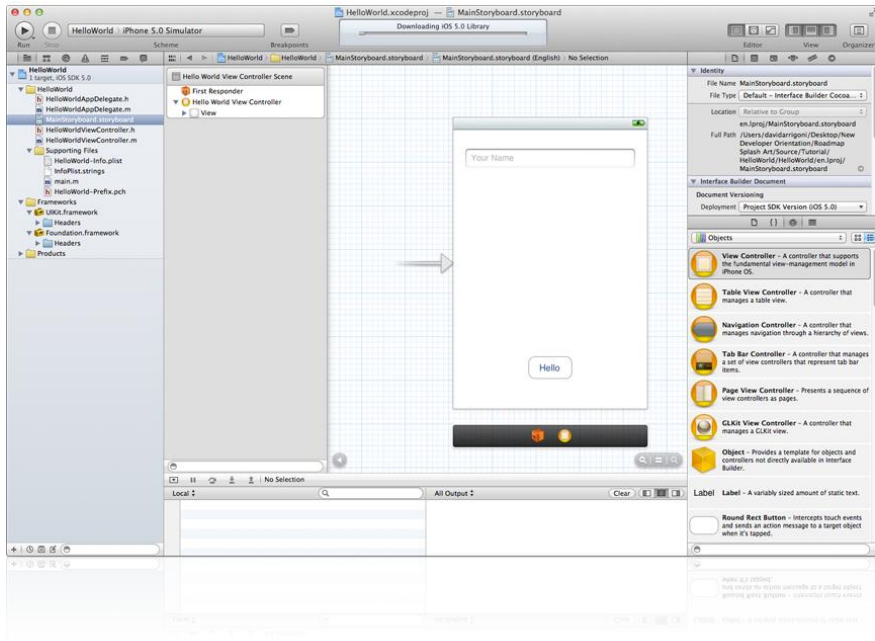
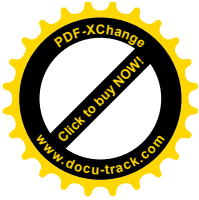
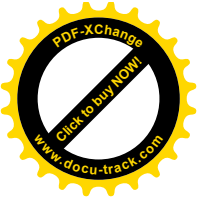
在 Mac 上打开 [Mac App Store](#) 应用程序，搜索 Xcode，然后点按“免费”按钮下载 Xcode。您下载的 Xcode 已包含 iOS SDK。（Mac OS X v10.7 以及更高版本已经预装 Mac App Store 应用程序。如果您使用的是较早版本的 Mac OS X，则需要升级。）

2. [加入 iOS Developer Program 成为 Apple 开发者](#)。

您无需加入该计划也可编写应用程序并在 iOS Simulator 中测试。但是，您只有加入该计划，才能在设备上测试与分发应用程序。加入该计划后，您还可以全权访问 iOS Dev Center 和 iOS Provisioning Portal。如果您现在加入，就可以执行路线图中的所有操作步骤，包括在设备上测试应用程序。

马上开始

开发优秀的 iOS 应用程序，需要大量的学习和实践。不过，有了这些工具和 iOS SDK，开发一个简单可用的程序并非难事。**您的首个 iOS 应用程序**教程，介绍了这些工具、基本设计模式和应用程序开发过程。通过这个教程，您将学习创建一个能接收用户文本输入并能将文本显示在屏幕上的应用程序。您还将学到如何在 Mac 上的 iOS Simulator 中运行这一程序。本教程中的简单步骤引入了一些简练实用的概念，将在今后的程序开发中不断地用到。



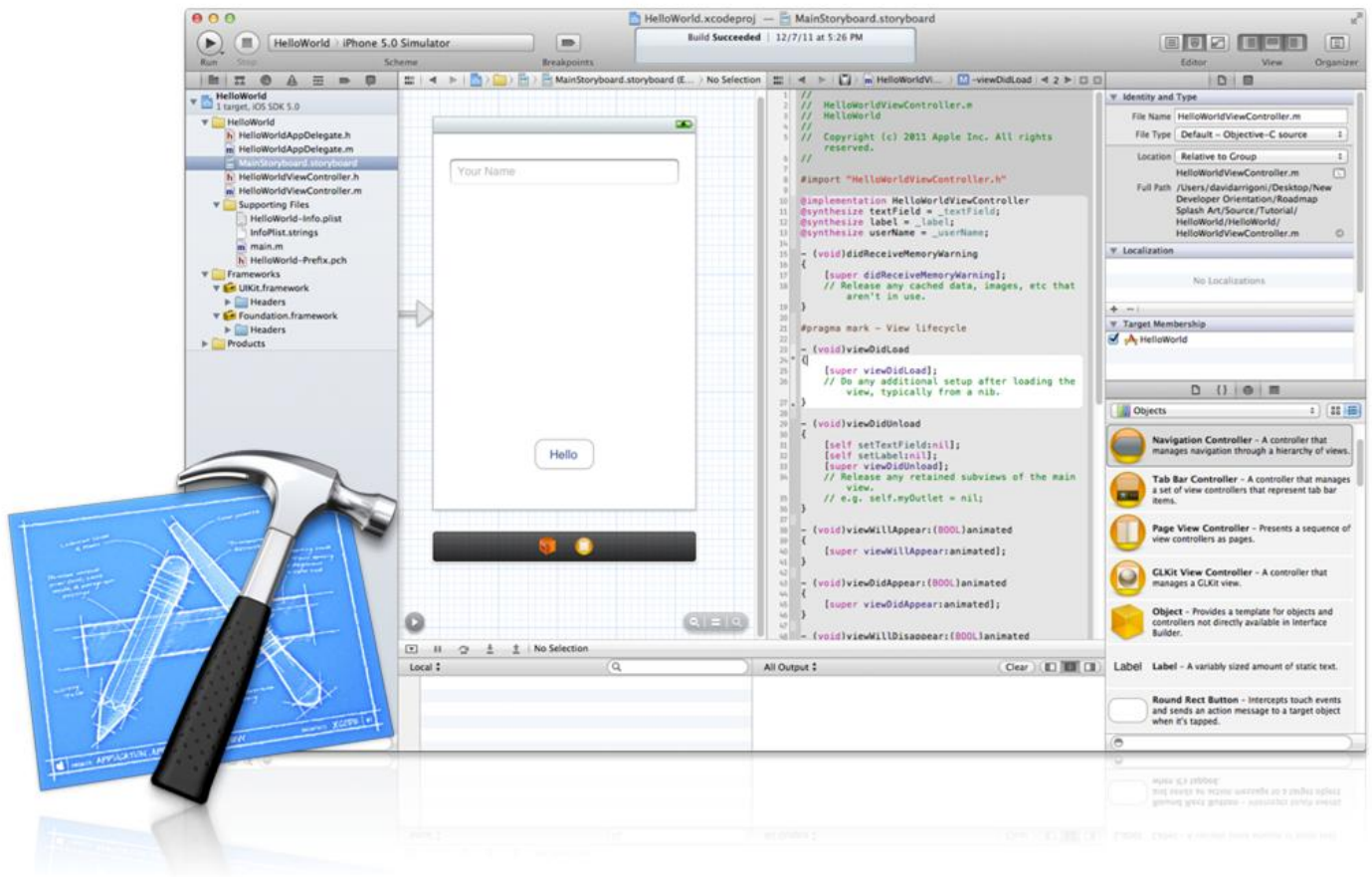
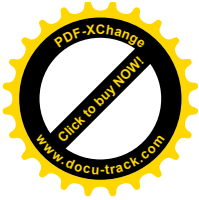
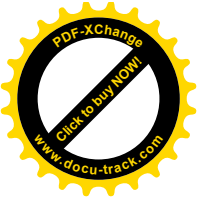
您的首个 iOS 应用程序是路线图中最长的文章，但请不要仓促地完成它。它为您在后面将详细学习的概念奠定了
基础。其他每篇文章，都是围绕当中某个概念再深入说明。



立即学习本教程：[您的首个 iOS 应用程序](#)告诉您，如何在 iOS 开发环境中，创建一个简单的应用程序。完成教程后，可返回本页继续学习[马上着手开发 iOS 应用程序](#)。

工具

Xcode 提供整套管理开发工作流程的工具——从创建应用程序、设计用户界面，到测试、优化，并将其提交到 App Store。您可以自定 Xcode，来符合您的工作风格，让您专注于手头的任务。

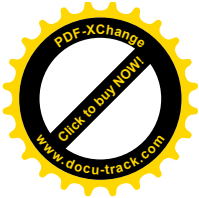
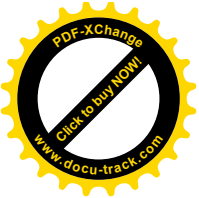


您的首个 iOS 应用程序向您展示了如何创建新项目、添加用户界面元素，以及编辑源代码。创建应用程序后，您可使用 Xcode 来测试和调试源代码、分析和改进应用程序的性能、执行源代码控制操作、归档并将应用程序提交到 App Store 等等。



立即阅读此文章：在 [Xcode 中管理工作流程](#) 向您展示 Xcode 用于 iOS 应用程序开发的重要功能。

程序设计语言



Objective-C 是一种简洁的、面向对象的程序设计语言，所有 iOS 应用程序都由它来驱动。您需要编写 Objective-C 代码来创建应用程序，同时您需要懂得该语言，才能使用大多数的框架。尽管您可以使用其他编程语言来开发，但不使用 Objective-C 就无法生成 iOS 应用程序。

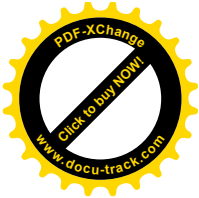


Objective-C 是一种简单的程序设计语言，其语法和规范简单易学。如果您有其他面向对象程序设计语言（例如 Java 或 C++）的编程经验，那么它对您来说，将更容易上手。如果您是 C 语言程序员，您会发现熟悉面向对象编程和 Objective-C 后，应用程序的设计和修改变得更加容易。



立即阅读此文章: [编写 Objective-C 代码](#) 教您 Objective-C 的基本语法，包括类声明、方法、发消息和声明属性。

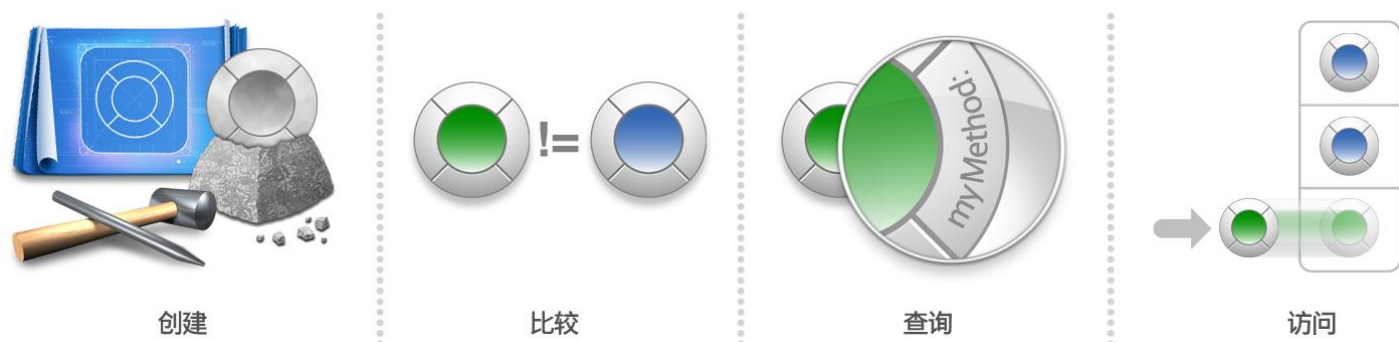
Mac 和 iOS 应用程序都可用 Objective-C 来开发。如果您是 Mac 开发者，那么基本语法对您来说已经不是问题了。



基本任务

现在，您学到的 Objective-C 的知识，足可用来阅读和编写基本代码，您可以开始以对象的方式来考虑问题了。

如同思考现实世界中的对象一样，您应该思考一个对象包含什么内容，可以用来做什么，以及如何与其他对象关联。



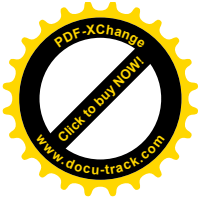
要创建 iOS 应用程序，您需要了解如何创建对象、比较对象、查询对象的相关信息，以及访问数据集（如数组）中的对象。这些任务在 iOS 应用程序中很常见。掌握这些技能后，您就可以编写更复杂的 Objective-C 代码了。



立即阅读此文章：[掌握基本的编程技能](#)描述 Objective-C 编程的基本任务。本文解释的概念在 Mac 和 iOS 开发中基本相同。

框架

应用程序由您编写的代码和 Apple 提供的框架组成。框架包含方法资源库，供您的应用程序调用。多个应用程序可同时访问一个框架资源库。



您开发的应用程序都会链接多种框架。您可以通过框架的应用编程接口 (API) 来利用框架。API (已发布在头文件中) 指定可用的类、数据结构和协议。Apple 编写的框架, 预计了您可能想要实现的基本功能。使用框架既省时省力, 又可确保代码高效、安全。系统框架是访问底层硬件的唯一途径。

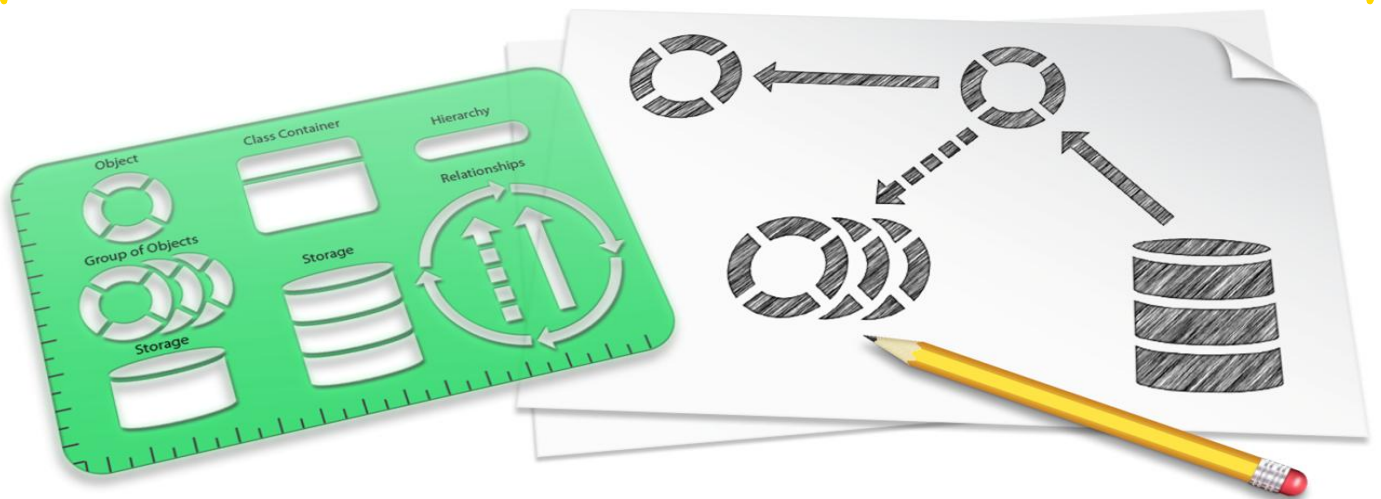
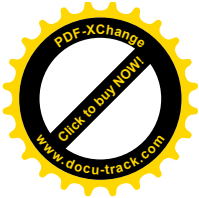
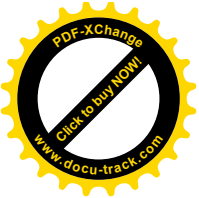


立即阅读以下文章：

- [研究主要框架](#)描述应用程序开发中最常用的框架。它还简要阐述了 OS X API 和 iOS API 之间的一些异同点。
- [将代码与框架整合](#)描述 Objective-C 框架中的方法种类, 并解释如何将应用程序代码与框架代码整合。尽管 OS X API 和 iOS API 之间存在不同之处, 但应用程序和框架之间的关系通常相同。

设计模式

设计模式可以解决常见的软件工程问题。模式是抽象设计, 而非代码。采用一种设计, 就是应用它的通用模式来满足具体需求。



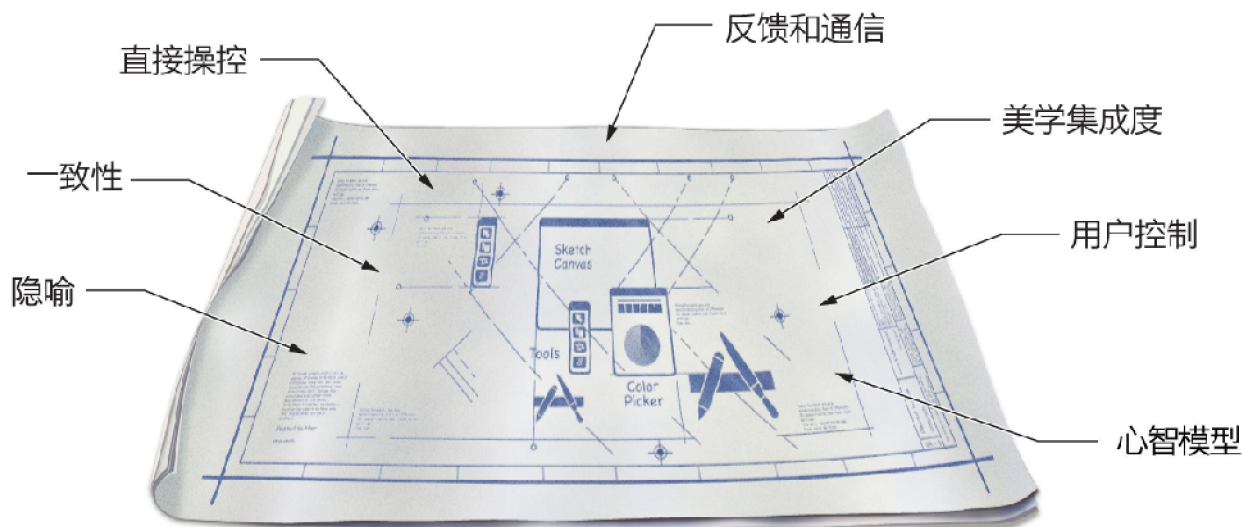
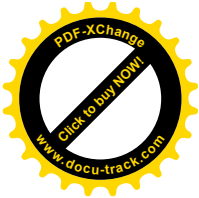
不管是创建哪种类型的应用程序，您都应该了解框架中使用的基本设计模式。了解设计模式有助于更高效地使用框架，并且可让您编写的程序复用程度更高、扩展能力更强和更容易修改。



立即阅读此文章：[采用设计模式使您的应用程序合理化](#)描述关键的设计模式，并解释如何在应用程序开发中使用这些模式。这些设计模式在 OS X 和 iOS 中基本相同。

用户界面设计

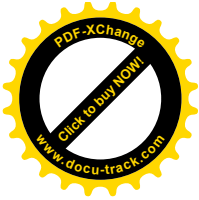
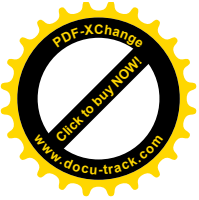
仅仅创建一个能用的应用程序是不够的，用户期望的 iOS 应用程序是直观易用、交互性强和引人入胜的。在设计应用程序时，从选取的功能到应用程序响应手势的方式，每个方面都要考虑用户体验。



用户或许不知道用户界面的设计原则，但是应用程序有没有遵循指南，他们是可以分辨出来的。请遵循 *iOS Human Interface Guidelines* (iOS 用户界面指南) 中清楚说明的原则和约定，为您的产品设计最好的用户界面和用户体验。

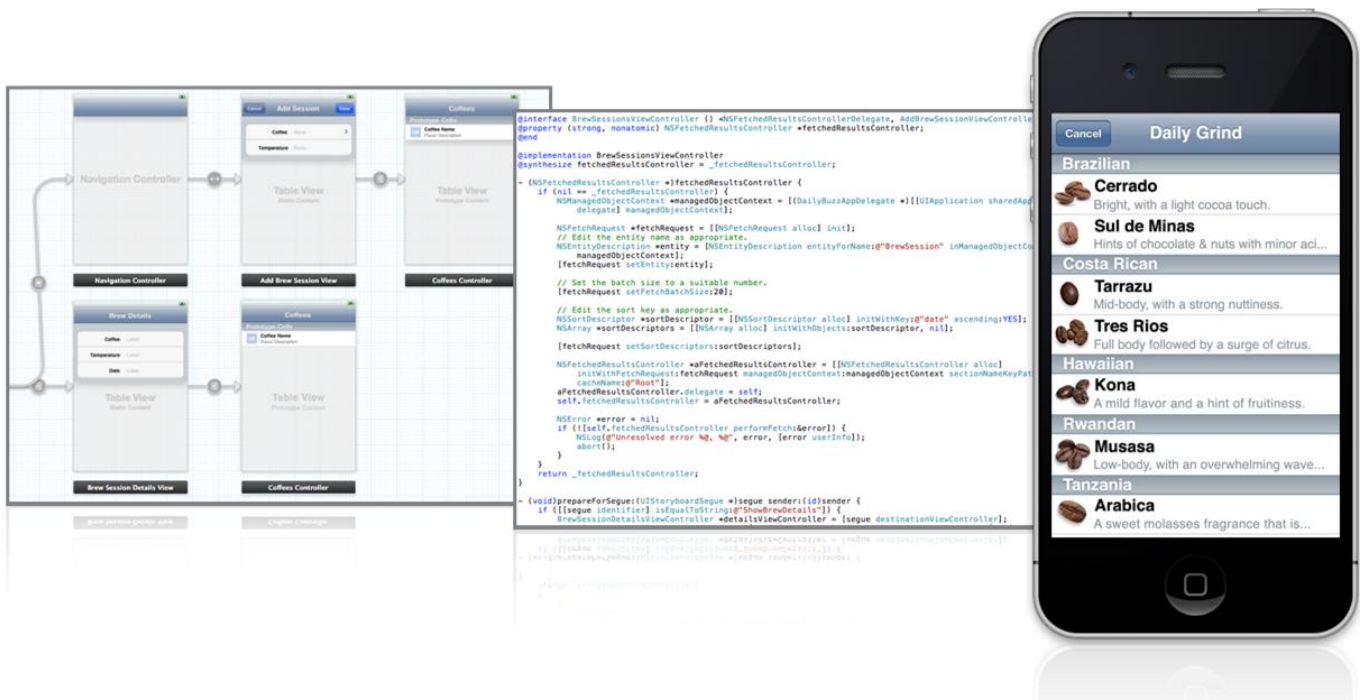


立即阅读此文章：[从用户角度进行设计](#)阐述了如何创建具有卓越用户体验的应用程序。本文稿介绍编程时需遵循的指南，但不涉及如何在代码中实现设计。大多数 iOS 指南，都与 OS X 的不同，因为移动设备上的应用程序，在本质上是不同的。



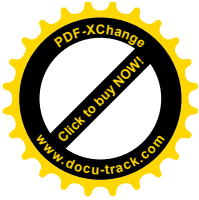
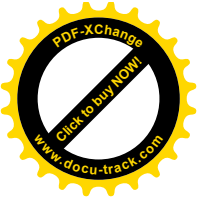
应用程序设计

在开始编写第一行代码之前，您应该做出一些必要的设计决策。应用程序的用途和功能，应该尽可能的具体。选取应用程序将使用的数据模型种类。决定应用程序的用户界面风格，例如，是应该遵循主从复合模式（master-detail pattern）还是实用工具应用程序（utility app）的模式？您要应用程序通用吗？也就是说，在 iPad、iPhone 和 iPod touch 上均可运行该应用程序吗？诸如此类的设计决策有助于构建应用程序的架构。



但在根据架构进行开发之前，请务必熟悉 Cocoa Touch 框架。毕竟，应用程序并不会凭空出现——您使用框架构建对象，然后才能生成应用程序。框架对象既是应用程序的基础构造，也是数据模型的组成部分，同时还向用户传递出应用程序的独特体验，应好好掌握它。

设计优良的应用程序，会吸引用户，而且具有适当和有用的功能。例如，应用程序可能适当而有效地使用动画；如果它允许用户选择对象，可能会允许用户拷贝、剪切和粘贴；又或者，它会根据不同的语言，来呈现不同的文本、图像和声音。



立即阅读以下文章：

- [用心设计您的应用程序](#)提出了一些您需要自己解决的初始设计问题。它还有助于理清如何将设计决策付诸实践。
- [了解您的应用程序的核心对象](#)叙述作为所有 iOS 应用程序的一部分的重要框架对象，以及这些对象如何协作。
- [将您的应用程序国际化](#)带您逐步完成将 HelloWorld 应用程序国际化和本地化的过程。

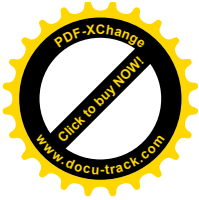
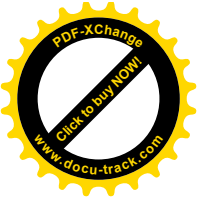
App Store

您目前所读到的信息主要描述在 Xcode 中创建应用程序的方法。然而，要在 App Store 上发布应用程序，您还需要进一步了解相关内容。



要制作出很好的应用程序，您需要在开发过程中用真实设备测试，而不只是借助 Simulator。要在基于 iOS 的设备上运行应用程序，您需要注册测试设备，创建证书来授权开发者给应用程序签名，以及创建应用程序 ID 来标识应用程序。

测试和改进应用程序后，您需要通过 iTunes Connect 提交应用程序。您的应用程序必须通过 App Store 审查人员审批后才能发布。



立即阅读此文章：[准备提交到 App Store](#) 描述为 App Store 开发应用程序的管理方面的任务。

查找信息

开发应用程序时，需要能轻易得到详细的技术信息。Xcode 可让您在编程时轻松查到所需信息。

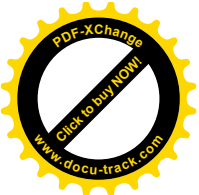


Xcode Quick Help 显示简明的参考信息，不会分散您对正在编辑的文件的注意力。请点按符号、界面对象或生成设置，以查看更多信息。按住 Control 键点按 Xcode 本身的各个区域，获得任务导向的说明，介绍如何执行常见操作。每篇帮助文章都提供逐步的指导，并且通常包括视频或插图来进一步说明。

Xcode 中的“Documentation”管理器，提供深入的编程指南、指导教程、示例代码、开发者工具使用手册、详细的框架 API 参考，以及由 Apple 工程师讲解的视频演示。“Documentation”管理器提供了一个一体化视图，可在其中搜索和浏览所有 Apple 开发者文稿。iOS Developer Library 也在网上提供。



立即阅读此文章：[快速查找文稿](#)图解如何使用这些重要的文稿资源。



接下来做什么

祝贺您，您已学完**马上着手开发 iOS 应用程序**中的所有内容，并阅读了所要求的关联文章，现在可以进行 iOS 应用程序的开发了。但是，要成为高效多产的 iOS 开发者，首先，应该[加入 iOS Developer Program](#)。然后，您就可以从 iOS Developer Library 下载示例应用程序项目，并从中了解各个方面是如何配合工作的，从而深化已学到的知识和技能。

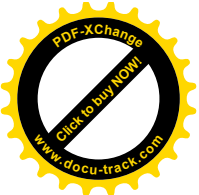
阅读以下文稿来增进您的 iOS 应用程序开发知识

以下文稿是任何 iOS 应用程序开发者都必不可少的读物：

- [iOS Technology Overview](#) (iOS 技术概述) 介绍可在 iOS 应用程序中使用的框架和其他技术。
- [iOS Human Interface Guidelines](#) (iOS 用户界面指南) 教您如何让您的应用程序符合 iOS 用户界面规范。
- [Developing for the App Store](#) (为 App Store 开发) 带您逐步完成这些过程：开发应用程序，预备测试设备，提交应用程序到 App Store。
- [Programming with Objective-C](#) (使用 Objective-C 编程) 描述如何使用 Objective-C 程序设计语言定义类、发送消息、封装数据，以及完成各种其他任务。
- [iOS App Programming Guide](#) (iOS 应用程序编程指南) 讲解在开发 iOS 应用程序时，您必须要了解并做到的基本事情。

阅读以下教程来探索 iOS 应用程序开发

学完这些教程以取得应用程序开发的其他方面的经验：



- [*App Store Submission Tutorial*](#)(向 App Store 提交应用程序教程)向您讲解预备设备、提交应用程序到 App Store 的过程。
- [*Your Second iOS App: Storyboards*](#) (您的第二个 iOS 应用程序：串联图) 向您讲解如何使用串联图，实现主从复合应用程序。
- [*Your Third iOS App: iCloud*](#) (您的第三个 iOS 应用程序：iCloud) 教您如何将 iCloud 集成到基于文稿的应用程序。

[苹果官方教程<编写 Objective-C 代码>](#)

苹果官方编写，原网址：

<http://developer.apple.com/library/ios/#referencelibrary/GettingStarted/RoadMapiOSCh/chapters/Introduction.html>