

<<iOS编程>>

图书基本信息

书名：<<iOS编程>>

13位ISBN编号：9787560986050

10位ISBN编号：7560986056

出版时间：2013-3-1

出版时间：华中科技大学出版社

作者：Joe Conway,Aaron Hillegass

译者：夏伟频,李骏

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<iOS编程>>

前言

要成为一名优秀的iOS程序员，需要攻克以下三项难题。

必须学会Objective-C语言。

Objective-C是C语言的扩展，小巧简单。

读完本书的前4章，读者就能掌握Objective-C的基础知识。

必须掌握Cocoa的常用技术。

其中包括内存管理、委托机制（delegation）、固化机制（archiving），以及如何正确使用视图控制器（viewController）。

理解这些技术需要花些时间。

本书前半部分会介绍这些内容。

必须掌握框架（framework）。

读者的最终目标是了解iOS的所有框架，学会如何使用框架中的每一个类和方法。

但这几乎是不可能完成的任务：iOS有3000多个方法，200多个类。

而且随着iOS的版本升级，Apple还会不断地加入新的类和方法。

本书将会介绍iOS SDK中的各个组成部分，但是不会太过深入。

作者的目标是带领读者入门，使读者能够自行阅读并理解Apple的参考文档。

BigNerdRanch公司采用本书作为“iOS新手培训课程”的教材。

这些内容经过了长期的检验，并且帮助很多人成为iOS程序员。

真心希望本书也能给你带来帮助。

<<iOS编程>>

内容概要

《iOS编程（第3版）》获21届Jolt生产力大奖。

第3版更新了iOS5和Xcode4.3的内容。

全书涵盖了开发iOS应用的方方面面。

从Objective-C基础知识到新增加的语言特性；从AppKit库到常见的Cocoa设计模式；从Xcode技巧到Instrument等。

如果读者刚接触iOS编程，阅读本书前两章可以快速入门。

如果读者已经有iOS编程经验，阅读本书可以拾遗补缺。

本书的另一个特色是教读者以正确的方法解决问题。

Objective-C的习惯约定有哪些（例如命名约定，内存管理约定），创建子类时如何处理初始化方法，Cocoa的常见设计模式有哪些，如何选择数据保存方法等。

这些问题，如果读者自己摸索，难免费时费力。

本书已经将这些知识整理、归纳并清楚地呈现在读者面前。

Big Nerd Ranch培训系列中文图书网站：<http://www.iosprogrammingbook.com/>

作者简介

作者：（美国）Joe Conway Aaron Hillegass Joe Conway是Big Nerd Ranch的资深教师，自iOS平台发布以来一直从事相关的咨询服务。

Joe毕业于美国威斯康辛大学，在Mac OS X早期即开始编写Objective-C和Cocoa代码。

他编写了Big Nerd Ranch的“iOS新手培训课程”教材。

该教材广受好评，是本书的原型。

Aaron Hillegass曾就职于NeXT公司和苹果公司，目前在Big Nerd Ranch教授iOS编程课程。

Aaron在NeXT时编写了第一本OpenStep教材（OpenStep是Cocoa的前身）。

本书以Big Nerd Ranch的教学课程为基础，涵盖了近20年OpenStep和Cocoa的开发成果。

书籍目录

前言 本书适合哪些读者 第3版有哪些更新 教学理念 如何使用本书 本书是如何组织的 代码风格 版式说明 开发所需的硬件与软件

第1章 第一个简单的iOS应用 1.1 创建Xcode项目 1.2 创建界面 1.3 模型—视图—控制器 1.4 声明 1.5 创建关联 1.6 实现方法 1.7 针对模拟器构建并运行应用 1.8 安装应用 1.9 应用图标 1.10 启动图片

第2章 Objective—C 2.1 对象 2.2 使用实例 2.3 编写命令行工具RandomPossessions 2.4 创建Objective—C类的子类 2.5 异常与未知选择器 2.6 快速枚举

第3章 通过ARC管理内存 3.1 堆 3.2 栈 3.2 指针变量与对象所有权 3.4 内存管理 3.5 Strong引用与Weak引用 3.6 属性 3.7 copy特性 3.8 用点号调用存取方法

第4章 委托与CoreLocation 4.1 项目、目标与框架 4.2 CoreLocation 4.3 委托 4.4 使用调试器

第5章 MapKit与文本输入 5.1 对象图 5.2 MapKit框架 5.3 界面属性 5.4 设置MapView对象的委托对象

第6章 UIView子类与UIScrollView 6.1 视图与视图层次结构 6.2 创建自定义视图 6.3 UIView的drawRect：方法 6.4 CoreGraphics 6.5 UIKit中的绘图扩展 6.6 重画视图 6.7 运动事件 6.8 使用UIScrollView 6.9 隐藏状态条

第7章 视图控制对象 7.1 UIViewController 7.2 UITabBarController 7.3 视图控制对象的生命周期 7.4 UIViewController子类与模板

第8章 通告机制与自动转屏 8.1 通告中心 8.2 UIDevice通告 8.3 自动转屏 8.4 强制使用横排模式

第9章 UITableView与UITableViewController 9.1 编写Homepwner应用 9.2 UITableViewController 9.3 UITableView数据源 9.4 UITableViewCell对象 9.5 代码片段库

第10章 编辑UITableView 10.1 编辑模式 10.2 增加行 10.3 删除行 10.4 移动行

第11章 UINavigationController 11.1 UINavigationController 11.2 额外的视图控制对象 11.3 UINavigationController的导航功能 11.4 UINavigationController

第12章 相机 12.1 通过UIImageView对象显示图片

第13章 UIPopoverController与模态视图控制对象 13.1 将Homepwner改为通用应用 13.2 UIPopoverController 13.3 更多的模态视图控制对象

第14章 保存、读取与应用状态 14.1 固化 14.2 应用沙盒 14.3 NSKeyedArchiver与NSKeyedUnarchiver 14.4 应用状态与状态切换 14.5 通过NSData将数据写入文件 14.7 再谈内存过低警告 14.8 模型—视图—控制器—存储设计模式

第15章 创建UITableViewCell子类 15.1 创建HomepwnerItemCell 15.2 处理图片 15.3 由UITableViewCell对象转发动作消息

第16章 CoreData 16.1 对象—关系映射 16.2 用CoreData重写BNRItemStore的数据保存功能 16.3 再谈SQL 16.4 Faults 16.5 各种存取机制的优缺点

第17章 本地化 17.1 通过NSLocale实施国际化 17.2 资源的本地化 17.3 NSLocalizedString与字符串对照表

第18章 UserDefaults 18.1 更新Whereami 18.2 使用UserDefaults

第19章 触摸事件与UIResponder 19.1 触摸事件 19.2 创建TouchTracker应用 19.3 实现TouchDrawView，完成绘图功能 19.4 处理触摸事件并创建线条对象 19.5 响应对象链

第20章 UIGestureRecognizer与UIMenuController 20.1 UIGestureRecognizer子类 20.2 用UITapGestureRecognizer对象识别“按下”手势 20.3 UIMenuController 20.4 UILongPressGestureRecognizer 20.5 UIPanGestureRecognizer以及同时识别多个手势

374 第21章 Instruments 21.1 静态分析器 21.2 Instruments 21.3 Xcode方案 21.9 创建新方案 21.4 构建设置

第22章 CoreAnimationLayer 22.1 层与视图的关系 22.2 创建CALayer对象 22.3 层的显示内容 22.4 隐式可动画属性

第23章 通过CAAnimation控制动画 23.1 动画对象 23.2 通过CABasicAnimation实现旋转动画 23.3 通过CAKeyframeAnimation实现弹跳动画

第24章 UIStoryboard 24.1 创建Storyboard文件 24.2 Storyboard文件中的UITableViewController 24.3 Segue 24.4 Storyboards的优缺点

第25章 Web服务与UIWebView 25.1 Web服务 25.2 UIWebView

第26章 UISplitViewController与NSRegularExpression 26.1 在Nerdfeed中使用UISplitViewController 26.2 主、从视图控制对象间的通信 26.3 在竖排模式下显示主视图控制对象 26.4 将Nerdfeed改为通用应用 26.5 NSRegularExpression

第27章 Block对象 27.1 Block对象及其语法 27.2 浅谈如何使用Block对象 27.3 捕获变量 27.4 Block对象的常见用途

第28章 模型—视图—控制器—存储 28.1 为什么需要存储对象 28.2 创建BNRFeedStore 28.3 使用BNRFeedStore对象 28.5 通过JSON格式传输数据 28.6 再谈存储对象

第29章 再谈MVCS 29.1 缓存RSS数据 29.2 高级缓存 29.3 已读RSSItem对象与未读RSSItem对象 29.4 使用存储对象的其他益处

第30章 iCloud 30.1 使用iCloud前的准备工作 30.2 UbiquityContainer 30.3 支持iCloud的provisioningprofile 30.4 CoreData与iCloud

第31章 后记 31.1 接下来做什么 31.2 结束语 索引

章节摘录

版权页： 插图： 本书已经指导读者编写并使用了多个存储对象。

本节将向读者分析这些对象的差异，并指导读者如何根据不同的情况来创建存储对象。

读者在设计某个存储对象前，需要问自己以下三个问题。

需要处理怎样的外部数据源？

是否应该使用单实例？

如何返回请求结果？

明确外部数据源 为某个应用设计存储对象时，第一个需要考虑的问题是应用会使用哪些外部数据源？

这能帮助读者决定应该为相应的应用创建几种存储对象。

应用中的存储对象是否必须和外部数据源保持一对一的关系？

答案是否定的。

以Nerdfeed为例，BNRFeedStore对象最初只能访问一个数据源：BNR的Web服务。

更新BNRFeedStore后，BNRFeedStore对象也可以向iTunes的Web服务发起请求。

虽然针对上述两个独立的数据源，BNRFeedStore对象需要发起两个不同的请求，但是和这两个数据源交互的代码都差不多。

针对这种情况，完全可以使用单个存储对象来处理上述两种数据源。

这样可以避免因为使用两种独立的对象而产生冗余代码。

此外，存储对象也可以和外部数据源保持一对一的关系。

以Homepwner的BNRItemStore和BNRImageStore为例，BNRItemStore对象负责获取并保存RSSItem对象，BNRImageStore对象负责获取并保存UIImage对象。

这两种存储对象需要负责处理不同类型的请求和数据，代码差异很大。

针对这种情况，最好通过不同的对象分开实现不同类型的请求逻辑。

设计BNRImageStore的初衷就是用于存取图片，不牵涉BNRItem对象乃至Homepwner。

如果在BNRItemStore中实现上述功能，就会增加其复杂度，降低重用性。

应该怎样规划不同的请求逻辑和存储对象之间的关系？

如果某个应用的请求逻辑可以独立工作，不会涉及其他代码，就可以用不同的存储对象来单独实现这些请求逻辑。

另一方面，如果某个应用的请求逻辑包含类似的代码，那么即使针对的是不同的外部数据源，也可以用同一个存储对象来实现这些逻辑。

这样可以使相关的代码更容易阅读、维护、测试和重用。

确定完需要创建的存储对象，以及需要处理的外部数据源后，接下来要根据每个存储对象需要完成的任务，设计具体的实现逻辑。

以Nerdfeed的BNRFeedStore为例，BNRFeedStore对象需要从两个不同的外部数据源获取数据。

所以除了要为BNRFeedStore实现共通的Web请求代码，还要针对BNR的Web服务及iTunes的Web服务实现不同的细节代码。

编辑推荐

《Big Nerd Ranch培训系列:iOS编程(第3版)》有两个特点：第一、涵盖iOS应用开发必备知识：从Objective-C基础知识到新语言特性，从AppKit库到常见的Cocoa设计模式，从Xcode技巧到Instruments，不一而足。

第二、指导读者以正确的方法解决问题：Objective-C的习惯约定有哪些（例如命名约定，内存管理约定），创建子类时如何处理初始化方法，Cocoa的常见设计模式有哪些，如何选择数据保存方法...
...iOS开发包括iPhone开发、iPod touch开发和iPad开发，《Big Nerd Ranch培训系列:iOS编程(第3版)》中绝大部分内容可通用，有差异的部分（例如iPad界面）单独予以说明。

《Big Nerd Ranch培训系列:iOS编程(第3版)》使用最新的iOS SDK，包括4.3版本的Xcode。虽然iOS SDK一直在升级，但《Big Nerd Ranch培训系列:iOS编程(第3版)》翻译完成之际，与SDK有关的内容仍然有效。

<<iOS编程>>

名人推荐

阅读此书，转身成为iPhone/iPad程序员。
——Dietrich Kappe，Pathfinder Development公司CTO

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>