程序设计丛书

WILEY

- 热门App的开发者现身说法
- Objective-C编程知识一应俱全
- 掌握移动互联时代最热门的开发语言

好学的 Objective-C

Objective-C Developer Reference

[美] Jiva DeVoe 著
林本杰 译

人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

好学的 Objective-C

Objective-C Developer Reference

[美] Jiva DeVoe 著
林本杰 译

人民邮电出版社
北京

图书在版编目（CIP）数据

好学的Objective-C / (美) 德沃 (DeVoe, J.) 著 ; 林本杰译. -- 北京 : 人民邮电出版社, 2012.3
(图灵程序设计丛书)
书名原文: Objective-C Developer Reference
ISBN 978-7-115-27358-1

Ⅰ. ①好… Ⅱ. ①德… ②林… Ⅲ. ①C语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2012)第004948号

内 容 提 要

本书共分为4部分。第一部分介绍了Objective-C的基础知识，包括Objective-C的基本语法、对象、内存管理等；第二部分深入挖掘Objective-C提供的一些功能，包括如何使用代码块，使用键值编码和键值观察，使用协议，扩展现有类的功能，编写宏以及处理错误和异常；第三部分介绍了Foundation框架及其相关知识；第四部分介绍了一些高级主题，包括多线程处理、Objective-C设计模式、利用NSCoder读写数据以及在其他平台上使用Objective-C等内容。

本书适合对Objective-C程序设计感兴趣的人阅读。

图灵程序设计丛书

好学的Objective-C

◆ 著 [美] Jiva DeVoe
译 林本杰
责任编辑 王军花
执行编辑 李 静

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京鑫正大印刷有限公司印刷

◆ 开本：800×1000 1/16
印张：16.25
字数：384千字 2012年3月第1版
印数：1－3 500册 2012年3月北京第1次印刷

著作权合同登记号 图字：01-2011-2966号

ISBN 978-7-115-27358-1

定价：55.00元

读者服务热线：(010)51095186转604 印装质量热线：(010)67129223
反盗版热线：(010)67171154

版权声明

致　　谢

写这本书很有收获并富有挑战，在这里我想感谢一些人，没有他们的帮助和支持我不可能完成本书。

首先，我想感谢Cynical Peak Software的Brad Miller，他是该公司最好的技术编辑之一。他对细节的关注以及在帮我改正错误方面的孜孜不倦的坚持真令人难以置信。谢谢你的努力。你太棒了。

我想感谢Wiley的Aaron Black的帮助和支持，尽管过程中有一些阻碍。非常感谢你的赞助和帮助。

感谢我的父亲Robert A. DeVoe，是你让我感受到技术的神奇并鼓励我在计算机方面追求梦想。

我儿子Alex花了很多时间帮我处理本书的格式，谢谢你。你的努力和帮助是本书完成的关键。

最后，也是最重要的，我要感谢我的妻子，感谢她不仅仅因为这个项目，而是因为她对我所有工作的一贯支持。没有她，本书就不会完成。你在我情绪低落和疲惫的时候鼓励我，并激励我追求新的成就和目标。真不知道该怎么感谢你。

序　言

Objetive-C在IT行业可谓受到了不公平的对待。尽管它很强大，并且是动态的面向对象语言，但却没有像C++、Java等语言一样得到足够认可。

在为iPhone OS 3写Cocoa Touch时，我意识到了需要写一本配套的书，以帮助新手们在接触Cocoa和Cocoa Touch等高层框架之前克服学习Objective-C的障碍。

所以当有人请我写一本专门介绍Objective-C语言的书时，我欣然接受了。

最后，我感觉到可以通过这本书向Mac、iPhone和iPad开发新手们介绍基础知识，因此万分激动。我期待这本书可以催化Objective-C在更多不同平台上发展。Objective-C完全有理由在Unix、Windows等平台上使用。

读者只需具备有限的计算机知识。我会从最基础的知识开始阐述，但是你至少需要懂得一些操作计算机的基础知识。

如果你已经熟悉了其他一些编程语言，这也不会有任何负面影响。我介绍的一些东西对你而言可能是一种回顾，不要担心，你会学到很多关于Objective-C的细节。

如果你接触过Objective-C，希望你可以在本书中发现一些有价值的新信息。我会努力将这些知识设计得便于你查找。这样一来，你无需逐页浏览，就能跳到某一部分并了解如何完成你想完成的任务。

对于本书中使用的一些约定，我尽量确保一致，同时尽量遵照苹果的约定。唯一一个比较明显的例外就是使用“方法”来表示实例和类的函数。苹果通常会倾向于使用“消息”。某种程度上这是缘于Objective-C受到Smalltalk的影响。

关于键盘快捷方式，我选用“Command键”这一术语来表示多数苹果键盘上空格键左侧的键。大家可能知道它也叫苹果键，因为就在几年前它上面会印有一个苹果标志。此外Command键旁边的键称为Option键，Option键旁边的就是Control键。这些是和苹果文档的约定保持一致的。

关于存储对象的变量，我通常会把它们称作“实例变量”。有些书会习惯用该术语或者其缩写“ivar”来指代作为类的一部分的变量。对此，我喜欢使用“成员变量”。在我看来，成员变量可以是实例变量，但不是所有的实例变量都是成员变量。

在文中提及方法时，我会遵照苹果引用它们的约定：使用方法名，但不包括参数。比如以下方法：

```
-(void)someMethodUsingParam1:(NSString *)param1 andParam2:(NSString *)param2;
```

就会被写做：`-someMethodUsingParam1:andParam2`。如果它是一个类方法，打头的连字符就会被替换成一个+号，就像你在写类定义中的方法一样。

关于示例代码，在需要构建完整项目的章节，通常我会尽可能提供代码的完整列表。在没有提供的情况下，你可以从本书网站上下载包含图片资源和其他相关支持文件的项目。有部分章节可能无法创建一个完整的项目来展示相关技术。在这种情况下，代码列表可能只是一些片段，你可用作自定义代码的基础。由于这些代码片段无法构成功能完整的项目，在网站上也就没有提供示例项目。

我希望你在阅读本书时会有一种和我写作时一样的愉悦体验。在我看来，一本好的技术书的标志就是它不会被束之高阁。它会被好好地放在书桌上或者书桌旁，因为经常需要翻阅它。我希望这本书在你的手中也会有这样的地位，并且希望它书角翘起、封面破损，每页都留有潦草的笔迹，但仍然能在未来几年对你有所帮助。

Jiva DeVoe
book@random-ideas.net

目　录

第一部分　Objective-C 简介

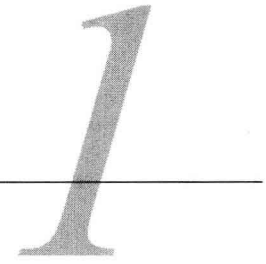
第四部分 高级主题

Part 1 第一部分

Objective-C 简介

本部分内容

第 1 章 Objective-C 简介



本章概要

- 学习 Objective-C 历史
- 了解编写 Objective-C 代码的 Xcode
- 配置开发环境

这一年是 1986 年，是哈雷彗星 75 年来最接近太阳的一年。英国和法国宣布建造英法海底隧道的计划。宝丽来盛行并刚刚迫使柯达退出了快速相机业务。人们使用 C 语言差不多 15 年，而 C++还是该领域的新军并鲜为人知。Smalltalk 语言在技术界打了个翻身仗，人们开始对一种称作面向对象编程（缩写是 OOP）的新概念感到兴奋。

Tom Love 和 Brad Cox 这两名开发人员在 ITT 公司的编程技术中心接触了 Smalltalk。Cox 想，要是在 C 语言中加入面向对象功能，只用 C 就可以进行面向对象编程，那定会很有意思。实际上，他将这种扩展命名为 COOPC，表示是用 C 实现的面向对象编程。最终，两个人成立了一家公司来商业化这些扩展并将其作为一种语言向开发人员推销。这一新语言也更名为 Objective-C。若干年后，Steve Jobs 领导的一家名为 NeXT 的小型创业公司，获准使用并标准化了 Objective-C，以作为将要开发的 NeXTstep 操作系统的主要语言。NeXT 计算机公司最终被苹果收购，NeXTstep 操作系统最终发展成为 Mac OS X。

很少有人会想到 Objective-C 历史悠久，并且它实际上影响了很多其他的编程技术。比如，Java 编程语言和 Objective-C 就有很多共同点。原因就是在 Objective-C 的早期，NeXT 和 Sun Microsystems 合作开发 OpenStep 平台，他们用来开发这种技术的语言就是 Objective-C。当 NeXT 计算机的表现没有达到他们预期的要求时，该公司走向了失败，Sun 决定开发自己的语言和跨平台开发包——Java。Java 工程师们都是谙熟 Objective-C 的，因为 Objective-C 是他们在使用 Java 之前首选的语言。后来他们就将 Objective-C 的一些较好的功能引入到了他们所开发的语言中。

Objective-C 现已成为了 Mac OS X 和 iPhone OS 上首选的开发语言。它已经发展成为了一种优雅的解决方案，在纯静态语言和纯动态语言之间实现了平衡。它是少有的几种通常进行编译的语言，不仅能从类似 C 和 C++的编译时语法检查受益，还能从负责处理动态对象类型的动态运行时受益。

除了 Mac OS X 和 iPhone OS，Objective-C 在其他平台上也发展了一批追随者，可以在 Linux、Windows 和其他支持 GNU 编译器的平台上开发应用。在 iPhone OS 上的使用增加了该语言的知

名度并吸引了很多新的程序员。可以说 Objective-C 如今正在经历一次复兴——成千上万的开发者正涌向该语言，使其成为了最热门的技术之一。

本书将介绍 Objective-C，并展示为什么我觉得它也是世界上一流的编程语言。我觉得一个好的程序员需要三种语言。第一种是工作流程自动化语言。通常这是一种脚本语言，可用于自动化工作空间并构建一个用于优化工作流的临时工具。第二种是编辑器宏语言。作为程序员，我们会花 99%的时间用于将文本打造成软件。有一个可以帮助你控制编辑器的重要工具。最后一种是用于构建系统和应用的语言，可以用于部署要求高性能和高稳定性的应用。通常这些语言都是编译型的，这样你就可以从所选平台中获得最佳性能。但是，这些语言最重要的特点就是可以最大限度地利用系统库。

希望你看完这本书后，Objective-C 会成为你的应用开发语言。对于实现各种任务，和其他语言相比，该语言绝对略胜一筹。

1.1 使用 Xcode 进行开发

本书假定你使用 Xcode 开发环境进行编码。Xcode 是一个加入苹果开发者计划就可以免费获得的优秀的 IDE。它默认支持 C、Objective-C、C++、Java 以及其他几种语言，不过本书中我们只用它来编写 Objective-C 程序。

1.1.1 新建项目

首先启动 Xcode，你可以选择打开最近打开的项目或者创建一个新项目。为了便于讨论，选择新建一个项目，这样大家就可以跟着练习。之后会弹出一个如图 1-1 所示的“新建项目”对话框。

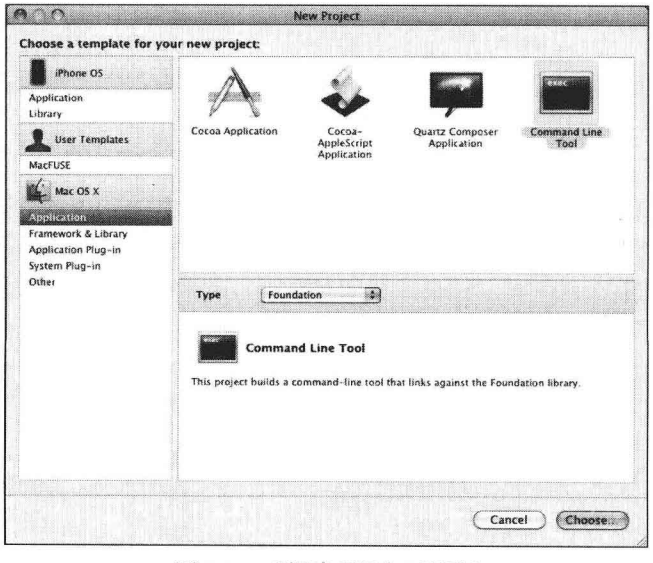


图 1-1 “新建项目”对话框

在该对话框中，你可以选择创建各种不同类型的项目。从命令行应用到桌面图形应用，你可以找到几乎所有的模板。此外，你如果安装了 iPhone SDK，就有多种不同的 iPhone 和 iPad 应用的模板。由于目前我们主要关注的是 Objective-C，选择其中最简单的一种项目就好了。

(1) 从 Mac OS X 组中选择 Application 后选择 Command Line Tool。

(2) 在 Type 的下拉列表中选择 Foundation。

(3) 单击 Choose 按钮，选择一个保存新项目的位置后单击 Finish。

在下面几节中，我们会简要介绍 Xcode 开发环境，这样你就可以慢慢熟悉它。我们就从图 1-2 所示的 Xcode 窗口开始。

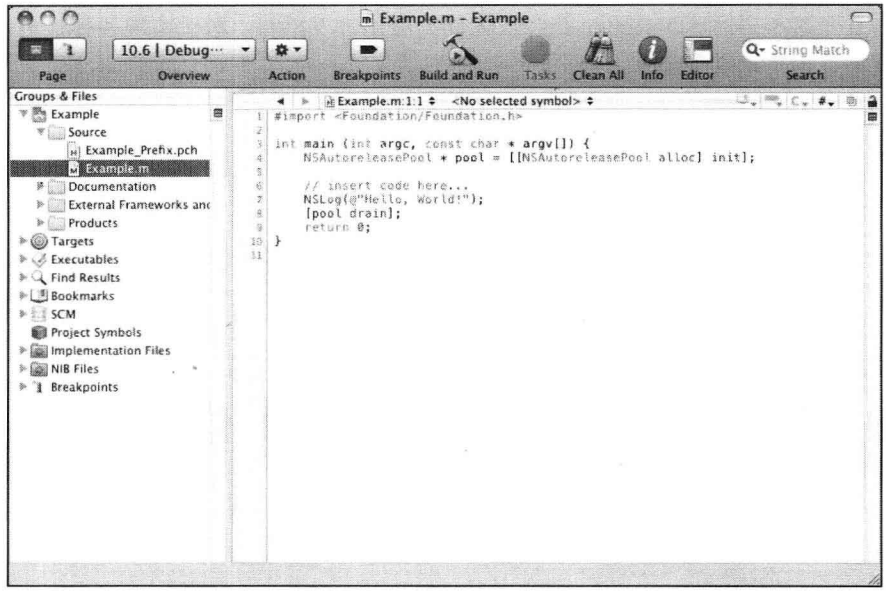


图 1-2　主 Xcode 窗口

主 Xcode 窗口包括两个面板：左边的面板包含了项目中的所有文件。选择其中的一个文件就会在位于窗口右侧的编辑面板中显示。在 Xcode 中，可以将项目文件移到项目中的目录中进行分组。大多数情况下，这些组仅仅在开发期间有用，对最后完成的项目影响很小，甚至没有任何影响。

除了源码文件外，还显示了链接项目所需的框架。

在项目文件下方是一系列的智能组。这包括了项目将要生成的目标、搜索结果和断点等。

目标组包括了项目编译生成的各种目标。改变组内对象的设置，就可以重写项目范围的编译设置。在这里还可以为项目添加和编辑一些自定义编译步骤。该组的编译设置如图 1-3 所示。

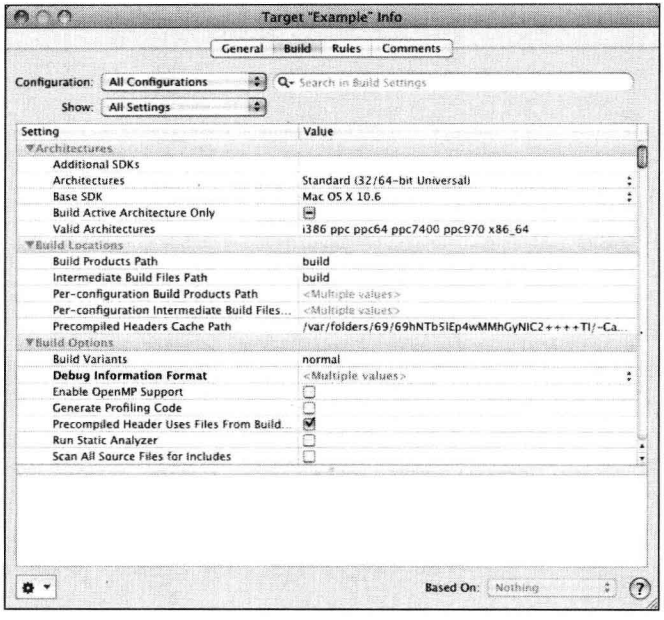


图 1-3 编译设置

1.1.2 项目文件

在这个简单的项目中，源码文件包含在源码文件组内。目前你可以看到只有一个源码文件，文件名和项目名是一致的。文件的扩展名是.m。单击该文件应该会在 Xcode 的编辑器面板中显示它，如图 1-4 所示。

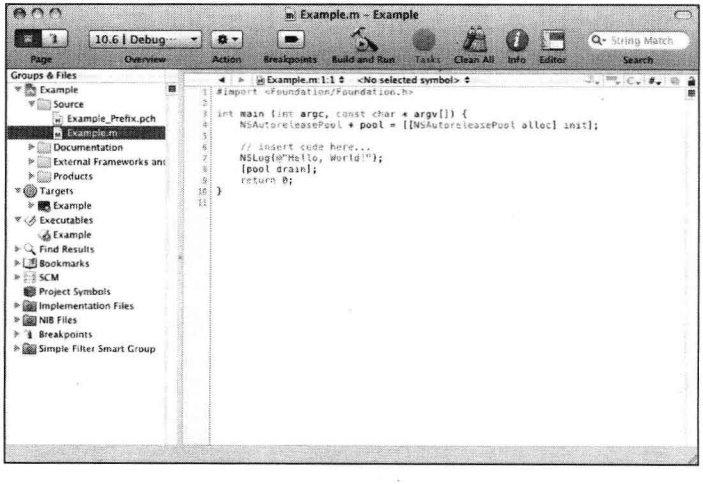


图 1-4 Xcode 中的编辑器面板

说明

虽然我们只提到一个源码文件，但这里其实还有一个扩展名为.pch 的文件。这是预编译头文件，不需要我们编辑或处理，它是编译器自动生成的。

不必担心看不懂文件中的源代码，下一章会介绍 Objective-C 的基本语法。现在，关键是要理解 XCode 以及它的工作原理。

说明

如果没有显示源文件，可能就需要拖动 Xcode 窗口的底部的分割条来显示源码编辑器。

默认项目中包括的其他文件有：文档组（Documentation）中的一个程序文档文件、外部框架和库组（External Frameworks and Libraries）中的项目相关的框架，以及位于产品组（Products）中的可执行文件。现在的可执行文件是红色的，这是因为项目还没有编译出可执行文件。如果单击编译和运行（Build and Run）按钮，就会编译可执行文件、运行它并在控制台窗口中显示输出结果。好好熟悉一下控制台窗口，因为接下来的几章会经常使用它来检查所编写程序的输出结果。

1.1.3　添加源码文件

在项目中新建一个源码文件，你可以选择文件组织面板中的源码文件组，然后单击 File ➤ New File 菜单，之后就会弹出一个如图 1-5 所示的“新建文件”对话框。

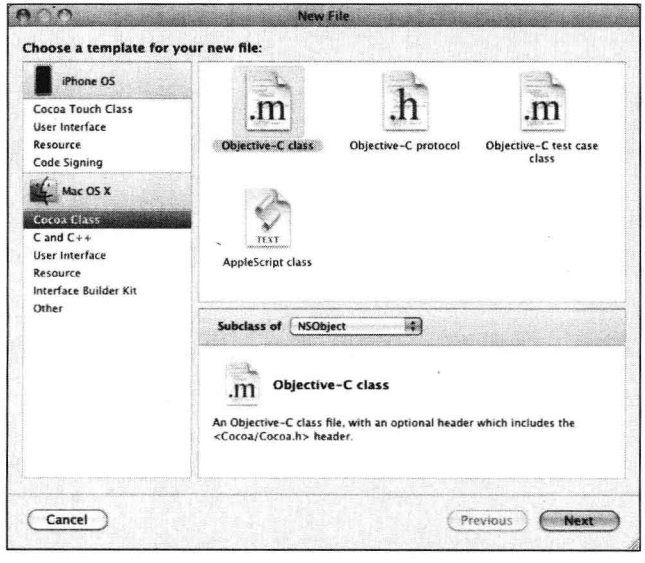


图 1-5　“新建文件”对话框