

CSDN超人气博主力作！专注于Swift开发



传承与突破

——从Swift 1.2到Swift 2.0，Swift语法世界的指南针



融合与创新

——从iOS 8到iOS 9，案例丰富，让Swift真正成为iOS开发的利器

Swift开发手册

技巧与实战

陈刚 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

Swift开发手册

技巧与实战

陈刚 编著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

全书分为6章,第1章介绍了与iOS开发环境搭建有关的知识点。第2章是Swift基础语法,基于Swift 1.2正式版本。第3章是Swift进阶语法,除了有难度的语法点之外,还加入了Swift 2.0正式版本的语法改动与新特性。第4章是作者根据斯坦福大学iOS 8公开课的计算器项目进行改进的计算器项目,通过这个简单的项目介绍了Xcode的基本用法和在iOS开发中的重要概念——MVC模式。第5章介绍了iOS中的重要内容UIKit框架,涵盖了UIKit中常用的控件,并通过丰富的实例展示了每一个控件的用法;第5章的最后几节介绍了iOS 9中的新成员,并详细讲解了他如何使用AutoLayout对页面元素进行布局。第6章介绍了iOS系统API的用法,全部基于Xcode 7.0正式版,使用Swift 2.0语法。附录A是作者精心挑选的一些帮助贴士。

本书知识点较为全面,版本也较为贴近目前日常开发所使用的版本,既可以作为初学者的入门教材,也可以作为经验老手的一个备忘手册。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Swift开发手册:技巧与实战 / 陈刚编著. —北京:电子工业出版社, 2016.1
ISBN 978-7-121-27517-3

I. ①S… II. ①陈… III. ①程序语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第263517号

责任编辑:安 娜

印 刷:三河市双峰印刷装订有限公司

装 订:三河市双峰印刷装订有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱

邮编:100036

开 本:787×980 1/16

印张:21.75

字数:495千字

版 次:2016年1月第1版

印 次:2016年1月第1次印刷

印 数:3000册 定价:69.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

推荐序

其实不是很高兴收到作者的邀请来写这本书的推荐序，因为本书对于一个 iOS 开发者了解 Swift 来说是如此的重要，如果因为我的序写得不好而影响了大家对于本书的理解，那将是莫大的遗憾。曾经在创业的过程中虽然也兼职做过一段时间的 iOS 开发，但本人并不是一个资深的 iOS 从业者，而 Swift 和本书让我重新有机会跟大家站在同一个起跑线上。

亲测阅读之后，有两点我觉得是可以跟各位读者去分享的。

第一点是关于如何去学习一门新的技能，这对于任何一个刚刚接触一门新的技术或者是正憧憬着美好未来的学生来说都是至关重要的。学习“学习的方法”永远是最重要的，所以我不打算引导大家如何去阅读本书的各个章节，因为你看完目录之后就能够发现作者在写作的时候是如此用心。在我们对一种技术体系有一定的基础认识之后，最好的学习方法是深入地了解此体系最基本的原理，然后根据最基本的原理推理出一些实践的指导思想。这是我在从过往的学习过程中觉得受益最多的一种思维模式。为了让读者能更好地理解这样种看上去并没有什么实际参考意义但又确实是可遵循的原理，下面举一些非常简单的例子。

相信大家在上高中的时候一定都学习过化学、物理这样的基础学科知识，在无机化学中有一种叫做“电解”的化学反应，当时很多同学都对这个反应感到迷惑，因为它涉及氧化、还原、电荷移动方向等诸多问题，每一部分分别来看的话都非常复杂，后来深入了解之后我发现，其实整

这个过程都是电荷“同性相斥、异性相吸”的完美体现，每一个环节都遵循这个原理，可以通过这个原理把前面提到的所有现象定性的推导出来。

再到后来，我在研究消息系统和队列的时候发现，有的思维方式让我们在教科书和参考资料的解释下感觉无比的痛苦和忧伤。索性，后来再也没有去阅读过关于这类系统的设计思想和算法阐释，而是换成了一种最原始的方式，那就是带着计算机思维去乘坐公交车体会它的运行方式（Message Buss）、去医院/火车站购票橱窗这种地方体会排队（Queue）。而用这样的方式，让我深刻地理解了计算机中的消息总线、I/O、缓存、队列、资源调度、服务降级等一些非常抽象的概念。其实，生活中有很多类似的例子，只要我们用心去体会，就能对我们的学习方法有极大的帮助，这是一种思维模式上的帮助。

第二点是关于完成和完美。很多读者在选择一门技术的时候往往都会去比较优劣，甚至去询问前辈们一个问题，就是“哪一个更好？”。然而，这个和“PHP 是最好的语言”一样能引起械斗的问题，并没有太大的实际意义。如果你不去做，其实什么都不好！如果你去完成了，那么你得到的往往比你所期望的更多，因为，完成是走向完美的必经之路。Swift 是苹果公司的一次非常伟大的尝试，敢于破旧立新，这也许正是他们的基因使然。相信很多读者都在观望：Swift 是否会成为主流？是否有公司正在使用这门新的技术去开发产品？学习完之后是否能找到更好的工作？而我的建议是，如果你致力于 iOS，那么就应该先踏上这个征程。怎么看，这一段内容都像像是广告，然而并不是这样。理解 Swift 能够让你了解到苹果公司团队的思维方式，他们为什么要在 Object-C 已经如此普及的时候要去创立全新的 Swift。笔者愚见，这正是 iOS 从完成走向完美的一种方式，曾经 OC 在早期的开发者生态中帮苹果公司的 developer 快速地走到了消费者的面前，取得了卓越的成就，而如今一些问题呈现出来，那么完美的进程上 Swift 应该是应运而生的。作为像笔者这样还不是十分资深的从业者来说，我们可以用谦卑的学习心态来了解这一切，让自己也有机会迈向完美，而本书正是一种正确的方式。

以上，是笔者曾经从一个技术 Geek 到产品经理，从产品运营到实现商业化价值的创业经历中的一些走心的体会，希望能够帮助到大家。

随身移动 CEO 孙建

旗下产品：中华万年历、微历 WeCal、生活日历、天气万年历
中国领先的手机日历服务提供商

前言

首先感谢您购买本书，这可能是您读到的最不像“前言”的前言。作为本书的作者，我并不想向您灌输 Swift 这门语言是多么优秀多么有潜力这样的鸡汤，我想要分享的是作为一名 Swifter 收获的成长与乐趣。

天意渐凉，年关将至，Swift 已经悄然走过了它的第一个年头，从 1.0 版本的毛头小孩成长为 2.0 版本的坚毅少年。和许多有资历的作者不同，Swift 是我的 iOS 入门语言，原来的我是个真正的菜鸟。

2014 年的深冬，我躲在温暖的咖啡馆中，打开 Playground 小心翼翼地敲下一行“Hello, Swift!”，从此这个世界上又多了一个 Swifter。和所有尝试新技术的程序员一样，最初的经历并不是一帆风顺的，每个版本都有不小的语法改动，Xcode 6 经常莫名出错，国内的资料十分匮乏，我买光了市面上能买到的所有 Swift 图书，不幸的是有些作者喜欢拿贴着 Swift 标签的新瓶子装一些旧酒。不久我迎来了那个学期的寒假，这是提升编程能力的好时机。那个寒假微信红包在疯狂地刷屏，而我在疯狂地使用 Swift 编写着各种小程序，连我的父母都惊讶于我的变化，我想这大概就是 Swift 的魔力。假期结束我第一次尝试使用纯粹的 Swift 开发作品去参加竞赛，因为团队人手不足还自学了 Sketch，自己动手制作 UI 素材，最终获得了不错的成绩，这让我很受鼓舞。之后接触了越来越多的英文资料，连我弱项之一的英语水平也有了很大提高。再之后只身前往北京的创业团队实习，应聘季拿到了满意的 offer，业余时间还能接一些 iOS 开发的私活赚点零花钱，我想这些机会与我

努力学习 Swift 是分不开的。2015 年 4 月份的时候电子工业出版社的安娜编辑通过我的 CSDN 博客联系到我，向我发出了约稿的邀请，让我感到受宠若惊，这是本书的由来。

我用了大概半年的时间完成了这本书的初稿，在最初的一个月里我主要的工作是编排目录，针对我自己在学习过程中遇到的那些坑，我希望本书的目录能尽可能地适合初学者，避免在基础章节有超前的知识点，建议初学者按章节阅读，通过渐进学习的方式去掌握本书的知识点。对于那些有经验的开发者，本书的知识点较为全面，版本也比较贴近目前日常开发所使用的版本，可以作为一个备忘手册，在遇到某些易错或者不易记忆的 API 时，我本人也时常翻阅此书。

全书分为 7 章，第 1 章介绍了与 iOS 开发环境搭建有关的知识点。第 2 章是 Swift 基础语法，基于 Swift 1.2 正式版本。第 3 章是 Swift 进阶语法，除了有难度的语法点之外，还加入了 Swift 2.0 正式版本的语法改动与新特性，如果你对 Swift 1.2 版本的语法已足够熟悉，可以直接翻看 3.9 小节。第 4 章是作者根据斯坦福大学 iOS 8 公开课的计算器项目进行改进的计算器项目，通过这个简单的项目介绍了 Xcode 的基本用法和在 iOS 开发中的重要概念——MVC 模式。第 5 章介绍了 iOS 中的重要内容 UIKit 框架，涵盖了 UIKit 中常用的控件，并通过丰富的实例展示了每一个控件的用法，第 5 章的最后几节介绍了 iOS 9 中的新成员，并详细讲解了何如使用 AutoLayout 对页面元素进行布局。第 6 章介绍了 iOS 系统 API 的用法，全部基于 Xcode 7.0 正式版，使用 Swift 2.0 语法。如果本书的知识点不能为您提供帮助，希望本书最后一章精心挑选的一些帮助贴士可以帮助到您。

最后，感谢父母的鼓励与支持，感谢我的舍友加竞赛队友王探云、汤闻达两位同学，感谢在“厅客”实习时刑淇翔、蔡清茂两位学长对我的照顾和夏凡对我在技术上的指导，感谢 1+1+果汁店的老板朱哥和墨点咖啡的老板老郭、大可为我提供了写作的场地，感谢好未来教育集团对我的青睐与厚望，感谢北京随身移动公司对本书的大力支持。

目录

第 1 章 搭建 Swift 开发环境.....	1
1.1 Swift 介绍.....	1
1.1.1 Swift 的前世今生	1
1.1.2 Swift 与 Objective - C.....	2
1.2 Mac OS X 操作系统.....	3
1.3 Xcode 简介和获取方法	4
1.3.1 Xcode 简介.....	4
1.3.2 Playground 简介.....	4
1.3.3 Xcode 的获取方法.....	5
1.4 iPhone SDK 简介	6
第 2 章 Swift 基础语法	7
2.1 基础知识.....	7
2.1.1 命名规则	7
2.1.2 常量与变量	8
2.1.3 类型推测	9

2.1.4	注释	9
2.1.5	输出常量和变量	10
2.2	基本数据类型	11
2.2.1	整数	11
2.2.2	浮点数	11
2.2.3	布尔类型	11
2.2.4	元组类型	12
2.2.5	可选型	13
2.3	基本运算符	15
2.3.1	赋值运算符	15
2.3.2	数值运算	15
2.3.3	自增和自减运算	16
2.3.4	复合赋值	16
2.3.5	比较运算	16
2.3.6	三元运算符	16
2.3.7	逻辑运算符	17
2.3.8	范围	17
2.3.9	括号优先级	18
2.4	字符串与字符	18
2.5	集合类型	22
2.5.1	数组	22
2.5.2	集合	25
2.5.3	字典	26
2.6	控制流	27
2.6.1	for 循环	27
2.6.2	while 循环	28
2.6.3	if 判断语句	29
2.6.4	switch 开关语句	29
2.7	函数	31
2.8	闭包	34
2.9	Swift 三杰——类、结构体、枚举	37
2.9.1	Swift 三杰简介	37
2.9.2	值引用与类型引用	38
2.9.3	类	38

2.9.4 结构体	39
2.9.5 枚举	40
2.10 属性	41
2.10.1 存储属性	41
2.10.2 计算属性	42
2.10.3 属性观察器	44
2.10.4 类型属性	45
2.11 方法	46
2.12 下标	49
2.13 继承	51
2.14 构造与析构	53
2.14.1 构造器	54
2.14.2 析构器	57
2.15 类型检查与类型转换	58
2.15.1 类型检查	59
2.15.2 类型转换	60
2.16 类型嵌套	61
2.17 扩展	62
2.17.1 扩展计算属性	62
2.17.2 扩展构造器	63
2.17.3 扩展方法	63
2.17.4 扩展下标	64
2.18 协议	64
2.18.1 声明协议	64
2.18.2 遵守协议	66
2.18.3 实现协议	66
2.19 泛型	66
2.20 断言	67
 第 3 章 Swift 进阶语法	 69
3.1 再谈可选型	69
3.1.1 可选型	69
3.1.2 为什么要用可选型	70

3.1.3	解包可选型	71
3.1.4	可选绑定	72
3.1.5	可选链	73
3.2	AnyObject	74
3.2.1	AnyObject 简介	74
3.2.2	AnyObject 的使用	74
3.3	几个数组相关的实用方法	76
3.3.1	filter 方法	76
3.3.2	map 方法	76
3.3.3	reduce 方法	78
3.3.4	sort 方法	78
3.3.5	contains 方法	78
3.3.6	find 方法	79
3.4	Objective - C 兼容性	79
3.5	ARC 自动引用计数	80
3.5.1	ARC 原理	80
3.5.2	循环强引用	82
3.5.3	弱引用与无主引用	83
3.6	带下标的遍历	84
3.6.1	C 语言风格的 for 循环	85
3.6.2	快速遍历	85
3.6.3	enumerateObjectsUsingBlock	85
3.6.4	enumerate	85
3.7	方法参数的语法甜头	86
3.7.1	可变参数	86
3.7.2	柯里化	86
3.8	闭包的“延迟调用”特性	88
3.9	Swift 2.0 新特性详解	90
3.9.1	println 简化为 print	90
3.9.2	do-while 更名为 repeat-while	91
3.9.3	where 关键字	91
3.9.4	if-case 结构	92
3.9.5	guard 关键字	92
3.9.6	错误处理	93

3.9.7	LLVM 与泛型特化.....	96
3.9.8	Protocol Extensions	97
3.9.9	API 的可用性检查.....	98
3.9.10	String 与 NSString 言归于好.....	99
3.9.11	总结.....	100
第 4 章	iOS 开发中的 MVC 模式.....	101
4.1	iOS 系统初探	101
4.1.1	核心 OS (Core OS) 层	102
4.1.2	核心服务 (Core Services) 层.....	102
4.1.3	媒体 (Media) 层.....	103
4.1.4	Cocoa Touch 层.....	103
4.2	MVC 模式	103
4.2.1	MVC 简介	103
4.2.2	iOS 中的 MVC.....	103
4.3	新建一个 Swift 工程.....	106
4.4	认识 Interface Builder	109
4.5	构建计算器界面.....	112
4.5.1	使用对象库中的对象	113
4.5.2	使用检查器设置对象	114
4.5.3	尝试运行程序	115
4.5.4	添加约束	117
4.5.5	关联代码	120
4.5.6	完善按键	125
4.6	实现计算器逻辑.....	126
4.6.1	补全键盘	127
4.6.2	给键盘添加约束	128
4.6.3	实现数字显示功能	131
4.6.4	实现运算逻辑	133
4.7	修改计算器为 MVC 模式.....	135
4.8	NSNotification	137
4.8.1	NSNotification 简介	137
4.8.2	addObserver 方法.....	138

4.8.3	addObserverForName 方法	141
4.8.4	postNotification 方法	142
第 5 章 掌控 UIKit		143
5.1	本地化	143
5.2	视图（View）	145
5.3	生命周期	149
5.3.1	APP 的生命周期	149
5.3.2	控制器的生命周期	150
5.4	Button（按钮）	151
5.4.1	按钮属性检查器	152
5.4.2	按钮的代码实现	153
5.4.3	为按钮添加事件	155
5.5	Label（标签）	156
5.5.1	Label 的属性检查器	157
5.5.2	Label 的代码实现	158
5.6	TextField（文本框）	159
5.6.1	Text Field 的属性检查器	160
5.6.2	Text Field 的代码实现	161
5.6.3	键盘	162
5.7	Switch（开关）	163
5.8	Segmented Control（分段控件）	165
5.8.1	Segmented Control 的属性检查器	166
5.8.2	Segmented Control 的代码实现	166
5.8.3	“按钮”还是“分段”	168
5.9	Slider（滑块）	169
5.9.1	Slider 的属性检查器	169
5.9.2	Slider 的代码实现	170
5.10	ProgressView&ActivityIndicatorView	171
5.10.1	多线程入门	171
5.10.2	ProgressView（进度条）	173
5.10.3	ActivityIndicatorView（环形进度条）	175
5.11	Stepper（步进器）	177

5.11.1	Stepper 的属性检查器	178
5.11.2	Stepper 的代码实现	178
5.12	ImageView (图像控件)	179
5.12.1	UIImageView&UIImage	180
5.12.2	UIImageView 的属性检查器	182
5.12.3	UIImageView 的代码实现	182
5.13	PickerView&DatePicker	186
5.13.1	PickerView (选择器) 简介	186
5.13.2	使用 delegate 和 dataSource	187
5.13.3	PickerView 的代码实现	189
5.13.4	DatePicker (日期选择器) 的属性检查器	192
5.13.5	DatePicker 的代码实现	194
5.13.6	浅谈“Delegate pattern”委托模式	194
5.14	AlertView&ActionSheet	195
5.14.1	AlertView (提醒框)	195
5.14.2	ActionSheet (操作表)	198
5.14.3	更好的选择: UIAlertController	200
5.15	多重 MVC 及 Segue (过渡)	204
5.15.1	多重 MVC 与 Segue 简介	204
5.15.2	NavigationController (导航控制器)	207
5.15.3	Segue (过渡)	214
5.15.4	SplitViewController (分屏控制器)	223
5.15.5	TabBarController (选项卡控制器)	225
5.15.6	PageViewController (页面控制器)	232
5.16	Toolbar (工具栏)	239
5.17	ScrollView (滚动视图)	242
5.17.1	ScrollView 简介	242
5.17.2	CGFloat、CGPoint、CGSize	242
5.17.3	视图的坐标系统	242
5.17.4	ScrollView 实战	243
5.18	TableViewController (表视图控制器)	246
5.18.1	UITableView 简介	246
5.18.2	UITableViewCell 简介	249
5.18.3	TableView 与 cell 的交互	250

5.19	SearchBar (搜索框)	255
5.19.1	SearchBar 的属性检查器	255
5.19.2	实时显示搜索结果的 SearchBar	257
5.20	iOS 与 Web	260
5.20.1	UIWebView	260
5.20.2	WKWebView	263
5.20.3	SFSafariViewController	264
5.21	AutoLayout&StackView	266
5.21.1	AutoLayout (自动布局)	266
5.21.2	UIStackView (堆栈视图)	276
第 6 章	iOS 开发揭秘	280
6.1	Gesture (手势)	280
6.2	KVC 与 API 设计	285
6.3	访问短信 API 与电话 API	289
6.3.1	使用 URL 访问短信 API 与电话 API	289
6.3.2	MessageUI	290
6.4	访问相册 API 与相机 API	292
6.5	快速分享 API	295
6.6	地图与定位 API	300
6.7	网络通信	310
6.7.1	网络通信初探	311
6.7.2	NSURLConnection 还是 NSURLSession	312
6.8	数据持久化	316
6.8.1	Archiving (归档)	317
6.8.2	File System (文件系统)	317
6.8.3	SQLite&Core Data	320
6.9	多线程	325
6.9.1	GCD	325
6.9.2	NSOperationQueue	328
附录 A	Swifter 帮助贴士	331

第 1 章

搭建 Swift 开发环境

正所谓“工欲善其事，必先利其器”，在开始学习 Swift 之前，我们需要做好相关的准备。由于 Swift 是苹果公司的“亲儿子”，所以 Swift 的开发必须在苹果的 Mac 操作系统下进行，编程工具这里选择的是苹果的 Xcode。

1.1 Swift 介绍

1.1.1 Swift 的前世今生

在阅读本书之前，可能你已经从其他渠道了解到 Swift 的相关信息，首先来快速了解一下 Swift 语言的前世今生。作为编程语言界的“小鲜肉”，Swift 是苹果公司在 2014 年 WWDC（苹果全球开发者大会）上发布的全新的编程语言。Swift 是供 iOS 和 OS X 应用编程的新编程语言，基于 C 和 Objective-C，没有 C 的一些兼容约束。Swift 采用了安全的编程模式和添加现代的功能以使得编程更加简单、灵活、有趣。界面则基于广受码民喜爱的 Cocoa 和 Cocoa Touch 框架，展示了软件开发的新方向。

有人笑言 Swift 语言是语言进化链顶端的语言，因为它融合了很多现代编程语言的优点，加入了诸如闭包这样的高级语言特性。在语法结构上，Swift 有点类似于 JavaScript 这样的脚本语言，更加简洁优雅。

2010 年 7 月，LLVM 编译器的原作者，暨苹果开发者工具部门总监克里斯·拉特纳（Chris Lattner）开始着手 Swift 编程语言的工作，还有一个 Dogfooding 团队大力参与其中。至 2014 年 6 月发表时，Swift 大约历经了 4 年的开发期。克里斯在开发 Swift 之前的一项伟大成就即为苹果公司开发了 LLVM 编译框架，由于他在编译框架方面的丰富经验，使得 Swift 不但语法简洁，而且在编译期的速度也有所优化，读者在使用 Swift 进行开发时一定深有感触。

程序员开发问答服务网站“StackOverflow”近期进行了一次民调。主要针对全世界 157 个国家的 2.6 万名开发者。这些样本中，6800 人为全职程序员，1900 人为移动开发员，1200 人为前端开发员，1.2 万人为其他类型的开发者。调查方对受访者问及，在过去一年中曾经使用什么编程语言来开发软件，哪种语言用得最多？调查结果显示，苹果公司推出的 Swift 以 77.6% 的覆盖率，在受欢迎度上位列第一。超过了其他许多知名度较高的开发语言。排名如图 1.1 所示。

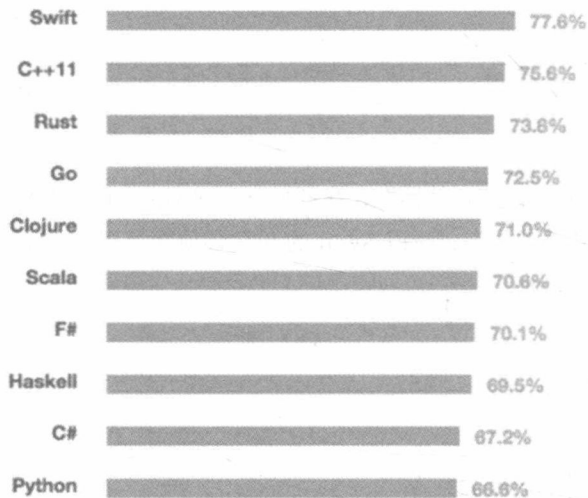


图 1.1 StackOverflow 编程语言受欢迎度民意调查排名

1.1.2 Swift 与 Objective - C

我们都知道之前进行 IOS 开发使用的是 Objective-C（在本书接下来的内容中笔者简称为 OC）这门古老的语言，Swift 语言的发布对于熟悉 OC 开发的程序员来说是一件令人兴奋的事情。