

# 《从零开始学Swift》

## 图书基本信息

书名：《从零开始学Swift》

13位ISBN编号：9787115415897

出版时间：2016-3

作者：关东升

页数：600

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

# 《从零开始学Swift》

## 内容概要

本书基于Swift 2.x，通过大量案例全面介绍苹果平台的应用开发。全书共分5部分，包括Swift 语法篇、Cocoa Touch 框架编程篇、混合编程篇、进阶篇、项目实战篇，囊括了开发工具介绍、语言基础知识、编写HelloWorld 程序，直至应用在App Store 上线的所有内容。此外，本书前20章专门配有同步练习，以帮助读者掌握所学知识。

## 作者简介

关东升

国内著名iOS和Cocos技术作家、iOS技术顾问、Cocos最有价值专家（CVP）、智捷课堂首席培训专家、51CTO社区iOS技术顾问。

他精通iOS、Android和Windows及HTML5等移动开发技术，擅长应用和游戏类项目开发，曾先后主持开发：大型网络游戏《神农诀》的iOS和Android客户端；国家农产品追溯系统的iPad客户端；酒店预订系统的iOS客户端；金融系统的iOS、Windows及Android客户端。与此同时，他长期为中国移动研究院、方正科技、大唐电信、中国石油、工商银行、南方航空、惠普、东软集团、NTT等企事业单位提供技术咨询及员工培训。

另著有：

《iOS开发指南》

《Swift开发指南》

《iOS传感器应用开发最佳实践》

《iOS网络编程与云端应用最佳实践》

《iOS图形图像、动画和多媒体编程技术最佳实践》

《品味移动设计》

《交互设计的艺术》

《iPhone与iPad开发实战》

《Android开发案例驱动教程》

《Android网络游戏开发实战》

《Cocos2d-x实战：C++卷》

《Cocos2d-x实战：JS卷》

《Cocos2d-x实战：Lua卷》

《Cocos2d-x实战：工具卷》

## 书籍目录

第一部分 Swift语法篇	
第1章 准备起航	2
1.1 本书约定	2
1.1.1 示例代码约定	2
1.1.2 图示约定	3
1.1.3 函数和方法签名约定	4
1.1.4 承接上一行代码约定	5
1.1.5 代码行号约定	5
1.2 Swift开发工具	6
1.2.1 Xcode开发工具	6
1.2.2 AppCode开发工具	12
1.3 本章小结	13
1.4 同步练习	13
第2章 第一个Swift程序	14
2.1 使用REPL	14
2.1.1 启动Swift REPL	14
2.1.2 使用Swift REPL	15
2.2 使用Playground	16
2.2.1 编程利器Playground	16
2.2.2 编写HelloWorld程序	18
2.3 使用Web 网站	21
2.4 通过Xcode创建OS X工程	22
2.4.1 创建OS X工程	22
2.4.2 编译和运行	23
2.5 通过AppCode创建OS X工程	24
2.5.1 创建OS X工程	24
2.5.2 编译和运行	25
2.6 使用swiftc命令	27
2.6.1 编译	27
2.6.2 运行	28
2.7 代码解释	28
2.8 在Linux下编译和运行Swift程序	29
2.9 本章小结	32
2.10 同步练习	32
第3章 Swift语法基础	33
3.1 标识符和关键字	33
3.1.1 标识符	33
3.1.2 关键字	34
3.2 常量和变量	35
3.2.1 常量	35
3.2.2 变量	36
3.2.3 使用var还是let	36
3.3 注释	37
3.4 表达式	38
3.5 本章小结	39
3.6 同步练习	39
第4章 运算符	41

4.1	算术运算符	41
4.1.1	一元运算符	41
4.1.2	二元运算符	41
4.1.3	算术赋值运算符	43
4.2	关系运算符	43
4.3	逻辑运算符	45
4.4	位运算符	46
4.5	其他运算符	47
4.6	本章小结	48
4.7	同步练习	48
第5章	Swift原生数据类型	50
5.1	Swift数据类型	50
5.2	整型	50
5.3	浮点型	51
5.4	数字表示方式	52
5.4.1	进制数字表示	52
5.4.2	指数表示	52
5.4.3	其他表示	53
5.5	数字类型之间的转换	53
5.5.1	整型之间的转换	53
5.5.2	整型与浮点型之间的转换	54
5.6	布尔型	54
5.7	元组类型	54
5.8	可选类型	56
5.8.1	可选类型概念	56
5.8.2	可选类型值拆包	56
5.8.3	可选绑定	57
5.9	本章小结	57
5.10	同步练习	57
第6章	Swift原生字符串	59
6.1	字符	59
6.1.1	Unicode编码	59
6.1.2	转义符	60
6.2	创建字符串	61
6.3	可变字符串	61
6.3.1	字符串拼接	61
6.3.2	字符串插入、删除和替换	62
6.4	字符串比较	63
6.4.1	大小和相等比较	63
6.4.2	前缀和后缀比较	64
6.5	本章小结	65
6.6	同步练习	65
第7章	控制语句	66
7.1	分支语句	66
7.1.1	if语句	66
7.1.2	switch语句	68
7.1.3	guard语句	69
7.2	循环语句	71
7.2.1	while语句	71

7.2.2	repeat-while语句	73
7.2.3	for语句	74
7.2.4	for-in语句	75
7.3	跳转语句	76
7.3.1	break语句	76
7.3.2	continue语句	78
7.3.3	fallthrough语句	79
7.4	范围与区间运算符	81
7.4.1	switch中使用区间运算符	81
7.4.2	for-in中使用区间运算符	82
7.5	值绑定	83
7.5.1	if中的值绑定	83
7.5.2	guard中的值绑定	84
7.5.3	switch中的值绑定	85
7.6	where语句	85
7.6.1	switch中使用where语句	86
7.6.2	for-in中使用where语句	86
7.6.3	guard中使用where语句	86
7.7	本章小结	87
7.8	同步练习	87
第8章	Swift原生集合类型	91
8.1	Swift中的数组集合	91
8.1.1	数组声明和初始化	92
8.1.2	可变数组	93
8.1.3	数组遍历	94
8.2	Swift中的字典集合	94
8.2.1	字典声明与初始化	95
8.2.2	可变字典	96
8.2.3	字典遍历	97
8.3	Swift中的Set集合	98
8.3.1	Set声明和初始化	98
8.3.2	可变Set集合	99
8.3.3	Set集合遍历	100
8.3.4	Set集合运算	101
8.4	本章小结	102
8.5	同步练习	102
第9章	函数	104
9.1	使用函数	104
9.2	传递参数	105
9.2.1	使用外部参数名	105
9.2.2	省略外部参数名	105
9.2.3	参数默认值	106
9.2.4	可变参数	106
9.2.5	参数的传递引用	107
9.3	函数返回值	107
9.3.1	无返回值函数	107
9.3.2	多返回值函数	108
9.4	函数类型	108
9.4.1	作为函数返回类型使用	109

9.4.2	作为参数类型使用	110
9.5	嵌套函数	111
9.6	本章小结	112
9.7	同步练习	112
第10章	闭包	115
10.1	回顾嵌套函数	115
10.2	闭包的概念	115
10.3	使用闭包表达式	116
10.3.1	类型推断简化	117
10.3.2	隐藏return关键字	117
10.3.3	省略参数名	118
10.3.4	使用闭包返回值	118
10.4	使用尾随闭包	119
10.5	捕获上下文中的变量和常量	120
10.6	本章小结	121
10.7	同步练习	121
第11章	Swift语言中的面向对象特性	123
11.1	面向对象概念和基本特征	123
11.2	Swift中的面向对象类型	123
11.3	枚举	124
11.3.1	成员值	124
11.3.2	原始值	126
11.3.3	相关值	127
11.4	结构体与类	128
11.4.1	类和结构体定义	128
11.4.2	再谈值类型和引用类型	129
11.4.3	引用类型的比较	131
11.4.4	运算符重载	132
11.5	类型嵌套	133
11.6	可选链	134
11.6.1	可选链的概念	134
11.6.2	使用问号(?)和感叹号(!)	136
11.7	访问限定	136
11.7.1	访问范围	137
11.7.2	访问级别	138
11.7.3	使用访问级别最佳实践	138
11.8	选择类还是结构体最佳实践	140
11.8.1	类和结构体的异同	141
11.8.2	选择的原则	141
11.9	本章小结	142
11.10	同步练习	142
第12章	属性与下标	146
12.1	存储属性	146
12.1.1	存储属性概念	146
12.1.2	延迟存储属性	147
12.2	计算属性	148
12.2.1	计算属性的概念	148
12.2.2	只读计算属性	150
12.2.3	结构体和枚举中的计算属性	150

12.3	属性观察者	151	
12.4	静态属性	153	
12.4.1	结构体静态属性	154	
12.4.2	枚举静态属性	155	
12.4.3	类静态属性	156	
12.5	使用下标	156	
12.5.1	下标概念	157	
12.5.2	示例：二维数组	157	
12.6	本章小结	159	
12.7	同步练习	159	
第13章	方法	160	
13.1	实例方法	160	
13.1.1	使用规范的命名	160	
13.1.2	结构体和枚举方法变异	163	
13.2	静态方法	165	
13.2.1	结构体静态方法	165	
13.2.2	枚举静态方法	166	
13.2.3	类静态方法	166	
13.3	本章小结	167	
13.4	同步练习	167	
第14章	构造与析构	168	
14.1	构造函数	168	
14.1.1	默认构造函数	168	
14.1.2	构造函数与存储属性初始化	169	
14.1.3	使用外部参数名	170	
14.2	构造函数重载	172	
14.2.1	构造函数重载概念	172	
14.2.2	构造函数代理	172	
14.2.3	类构造函数横向代理	173	
14.3	析构函数	174	
14.4	本章小结	176	
14.5	同步练习	176	
第15章	类继承	178	
15.1	从一个示例开始	178	
15.2	构造函数继承	179	
15.2.1	构造函数调用规则	179	
15.2.2	构造过程安全检查	180	
15.2.3	构造函数继承	184	
15.3	重写	186	
15.3.1	重写实例属性	186	
15.3.2	重写静态属性	188	
15.3.3	重写实例方法	189	
15.3.4	重写静态方法	190	
15.3.5	下标重写	190	
15.3.6	使用final关键字	191	
15.4	类型检查与转换	193	
15.4.1	使用is进行类型检查	194	
15.4.2	使用as、as!和as?进行类型转换	195	
15.4.3	使用Any和AnyObject类型	197	



15.5	本章小结	198
15.6	同步练习	198
第16章	扩展	200
16.1	“轻量级”继承机制	200
16.2	声明扩展	200
16.3	扩展计算属性	201
16.4	扩展方法	202
16.5	扩展构造函数	203
16.5.1	值类型扩展构造函数	203
16.5.2	引用类型扩展构造函数	204
16.6	扩展下标	204
16.7	本章小结	205
16.8	同步练习	205
第17章	协议	206
17.1	协议概念	206
17.2	协议	206
17.3	协议方法	207
17.3.1	协议实例方法	207
17.3.2	协议静态方法	208
17.3.3	协议变异方法	208
17.4	协议属性	210
17.4.1	协议实例属性	210
17.4.2	协议静态属性	211
17.5	面向协议编程	211
17.5.1	协议类型	212
17.5.2	协议的继承	214
17.5.3	协议扩展	215
17.5.4	协议的合成	216
17.5.5	扩展中遵从协议	217
17.5.6	面向协议编程示例：表视图中使用扩展协议	217
17.6	本章小结	220
17.7	同步练习	220
第18章	泛型	222
18.1	一个问题的思考	222
18.2	泛型函数	222
18.2.1	使用泛型函数	223
18.2.2	多类型参数	223
18.3	泛型类型	223
18.4	泛型扩展	225
18.5	本章小结	226
18.6	同步练习	226
第19章	Swift内存管理	227
19.1	Swift内存管理概述	227
19.1.1	引用计数	227
19.1.2	示例：Swift自动引用计数	228
19.2	强引用循环	229
19.3	打破强引用循环	233
19.3.1	弱引用	233
19.3.2	无主引用	236

19.4	闭包中的强引用循环	238
19.4.1	一个闭包中的强引用循环示例	238
19.4.2	解决闭包强引用循环	239
19.5	本章小结	241
19.6	同步练习	241
第20章	错误处理	243
20.1	Cocoa错误处理模式	243
20.2	do-try-catch错误处理模式	244
20.2.1	捕获错误	244
20.2.2	错误类型	244
20.2.3	声明抛出错误	245
20.2.4	在函数或方法中抛出错误	245
20.2.5	try?和try!的使用区别	246
20.3	示例：MyNotes应用数据持久层实现	247
20.3.1	MyNotes应用介绍	247
20.3.2	MyNotes应用数据持久层设计	248
20.3.3	实现Note实体类	248
20.3.4	NoteDAO代码实现	249
20.3.5	使用defer语句释放资源	250
20.3.6	测试示例	251
20.4	本章小结	252
20.5	同步练习	252
第21章	Swift编码规范	253
21.1	命名规范	253
21.2	注释规范	255
21.2.1	文件注释	255
21.2.2	文档注释	255
21.2.3	代码注释	256
21.2.4	使用地标注释	257
21.3	声明	259
21.3.1	变量或常量声明	259
21.3.2	属性声明	260
21.4	代码排版	260
21.4.1	空行	260
21.4.2	空格	261
21.4.3	断行	262
21.4.4	缩进	263
21.5	本章小结	265
第二部分	Cocoa Touch框架编程篇	
第22章	Foundation框架	268
22.1	数字类NSNumber	268
22.1.1	获得NSNumber对象	268
22.1.2	比较NSNumber对象	270
22.1.3	数字格式化	271
22.1.4	NSNumber与Swift原生数字类型之间的桥接	272
22.2	字符串类	273
22.2.1	NSString类	273
22.2.2	NSMutableString类	275
22.2.3	NSString与String之间的桥接	277

22.3	数组类	278
22.3.1	NSArray类	278
22.3.2	NSMutableArray类	279
22.3.3	NSArray与Swift原生数组之间的桥接	280
22.4	字典类	281
22.4.1	NSDictionary类	281
22.4.2	NSMutableDictionary类	282
22.4.3	NSDictionary与Swift原生字典之间的桥接	283
22.5	NSSet集合类	283
22.5.1	NSSet类	284
22.5.2	NSMutableSet类	284
22.5.3	NSSet与Swift原生Set之间的桥接	285
22.6	文件管理	286
22.6.1	访问目录	286
22.6.2	目录操作	287
22.6.3	文件操作	288
22.7	字节缓存	289
22.7.1	NSData和NSMutableData类	289
22.7.2	示例：Base64解码与编码	291
22.8	日期与时间	292
22.8.1	使用NSDate类	292
22.8.2	日期时间格式化	294
22.8.3	使用NSCalendar和NSDateComponents类	295
22.8.4	示例：时区转换	296
22.9	使用谓词NSPredicate过滤数据	297
22.9.1	一个过滤员工花名册的示例	298
22.9.2	使用谓词NSPredicate	300
22.9.3	NSPrdicate与集合	300
22.9.4	格式说明符	301
22.9.5	运算符	302
22.10	使用正则表达式	304
22.10.1	在NSPredicate中使用正则表达式	305
22.10.2	使用NSRegularExpression	305
22.10.3	示例：日期格式转换	307
22.11	本章小结	309
第23章	iOS开发基础	310
23.1	iOS介绍	310
23.2	第一个iOS应用：HelloWorld	310
23.2.1	创建工程	311
23.2.2	Xcode中的iOS工程模板	316
23.2.3	程序剖析	317
23.3	iOS API简介	319
23.4	本章小结	319
第24章	Core Foundation框架	320
24.1	数据类型映射	320
24.2	内存管理	321
24.2.1	Objective-C的MRC内存管理	321
24.2.2	对象所有权	322
24.2.3	内存托管对象	323

24.2.4	内存非托管对象	324
24.3	本章小结	324
第25章	Cocoa Touch设计模式及应用	325
25.1	单例模式	325
25.1.1	问题提出	325
25.1.2	解决方案	325
25.2	目标与动作	326
25.2.1	问题提出	326
25.2.2	解决方案	327
25.3	输出口	331
25.3.1	问题提出	331
25.3.2	解决方案	331
25.4	选择器	334
25.4.1	问题提出	334
25.4.2	解决方案	334
25.4.3	示例：使用定时器变换视图背景	335
25.5	委托与数据源	337
25.5.1	问题提出	337
25.5.2	委托解决方案	339
25.5.3	数据源解决方案	343
25.6	通知机制	346
25.6.1	问题提出	346
25.6.2	解决方案	346
25.6.3	示例：关闭和打开键盘	347
25.7	MVC模式	348
25.7.1	MVC模式概述	348
25.7.2	Cocoa Touch中的MVC模式	349
25.8	响应者链与触摸事件	351
25.8.1	响应者对象与响应链	351
25.8.2	触摸事件	352
25.9	本章小结	357
第26章	UIKit框架	358
26.1	视图控制器	358
26.1.1	与导航相关的视图控制器	358
26.1.2	视图控制器的生命周期	358
26.2	视图与UIView	360
26.2.1	UIView继承层次结构	360
26.2.2	视图分类	362
26.2.3	应用界面的构建层次	362
26.3	界面构建技术	363
26.3.1	使用故事板	363
26.3.2	使用XIB文件	365
26.3.3	使用代码	368
26.4	开关控件	370
26.5	滑块控件	371
26.6	分段控件	373
26.7	工具栏	374
26.8	导航栏	376
26.9	表视图	379

26.9.1	表视图组成	379
26.9.2	单元格组成和样式	380
26.9.3	示例：纯代码创建简单表视图	381
26.10	本章小结	386
第三部分 混合编程篇		
第27章	Swift与Objective-C混合编程	388
27.1	选择语言	388
27.2	文件扩展名	388
27.3	Swift与Objective-C API映射	388
27.3.1	构造函数映射	389
27.3.2	方法名映射	391
27.4	同一应用目标中的混合编程	393
27.4.1	什么是目标	394
27.4.2	Swift调用Objective-C	396
27.4.3	Objective-C调用Swift	399
27.5	同一框架目标中的混合编程	402
27.5.1	链接库和框架	402
27.5.2	Swift调用Objective-C	404
27.5.3	测试框架目标	407
27.5.4	Objective-C调用Swift	410
27.6	本章小结	413
第28章	Swift与C/C++混合编程	414
28.1	数据类型映射	414
28.1.1	C语言基本数据类型	414
28.1.2	C语言指针类型	415
28.2	应用目标中的混合编程	419
28.2.1	Swift调用C API	419
28.2.2	Swift调用C++ API	421
28.3	框架目标中的混合编程	425
28.3.1	同一框架目标中Swift调用C或C++ API	425
28.3.2	Swift调用第三方库中的C或C++ API	428
28.4	示例：用SQLite嵌入式数据库实现MyNotes数据持久层	430
28.4.1	Note实体类代码	430
28.4.2	创建表	431
28.4.3	插入数据	432
28.4.4	查询数据	434
28.4.5	应用沙箱目录	435
28.4.6	表示层开发	436
28.5	本章小结	440
第四部分 进阶篇		
第29章	程序调试	442
29.1	Xcode调试工具	442
29.1.1	调试功能	442
29.1.2	输出窗口	443
29.1.3	变量查看窗口	444
29.2	设置和查看断点	445
29.2.1	断点设置	445
29.2.2	管理断点	446
29.3	使用Xcode中的Instruments工具	447

29.3.1	跟踪内存使用	447
29.3.2	内存泄漏查找	449
29.4	iOS 真机调试	453
29.4.1	Xcode设置	453
29.4.2	设备设置	455
29.5	本章小结	456
第30章	并发编程	457
30.1	并发相关概念	457
30.1.1	CPU与并发执行	457
30.1.2	进程与线程	457
30.2	GCD技术	457
30.2.1	GCD概念	458
30.2.2	调度队列	458
30.3	GCD示例	458
30.3.1	主线程阻塞问题	458
30.3.2	使用GCD 解决主线程阻塞	460
30.4	GCD 与单例模式	460
30.5	本章小结	461
第31章	SpriteKit游戏引擎	462
31.1	移动平台游戏引擎介绍	462
31.2	第一个SpriteKit游戏	462
31.2.1	创建工程	462
31.2.2	工程剖析	465
31.3	一切都是节点	468
31.3.1	节点“家族”	469
31.3.2	节点树	469
31.3.3	节点中重要的方法	470
31.3.4	节点中重要的属性	470
31.4	精灵	471
31.4.1	精灵类SKSpriteNode	471
31.4.2	示例：沙漠英雄场景	473
31.4.3	使用纹理图集性能优化	478
31.5	场景切换	481
31.5.1	场景切换方法	481
31.5.2	场景过渡动画	481
31.5.3	示例：沙漠英雄场景切换	481
31.6	动作	484
31.6.1	常用动作	484
31.6.2	组合动作	487
31.6.3	示例：帧动画实现	491
31.7	粒子系统	493
31.7.1	粒子系统属性	494
31.7.2	内置粒子系统模板	495
31.8	游戏音乐与音效	499
31.8.1	音频文件介绍	499
31.8.2	iOS和OS X平台音频优化	499
31.8.3	背景音乐	500
31.8.4	3D音效	501
31.9	物理引擎	502

31.9.1	物理引擎核心概念	502
31.9.2	物理引擎中的物体	503
31.9.3	接触与碰撞	504
31.10	本章小结	507
第五部分 项目实战篇		
第32章	项目实战：基于分层架构的多版本iPhone计算器应用	510
32.1	应用分析与设计	510
32.1.1	应用概述	510
32.1.2	需求分析	510
32.1.3	原型设计	511
32.2	分层架构设计	511
32.2.1	低耦合应用架构	511
32.2.2	详细设计	512
32.2.3	分层架构设计有多种模式	514
32.2.4	选择语言	514
32.3	基于同一工程同一目标纯Swift的实现	515
32.3.1	创建工程	515
32.3.2	业务逻辑层开发	516
32.3.3	表示层开发	520
32.4	基于同一工程同一目标Swift调用Objective-C的实现	531
32.4.1	在Swift 工程中添加Objective-C类	531
32.4.2	Objective-C实现业务逻辑层	532
32.5	基于同一工程不同目标Swift调用Objective-C的实现	534
32.5.1	添加业务逻辑层目标	535
32.5.2	暴露的头文件CalcLogic.h	535
32.5.3	修改表示层	536
32.6	基于同一工作空间不同工程下Swift调用Objective-C的实现	536
32.6.1	创建工作空间	537
32.6.2	暴露头文件CalcLogic.h	537
32.6.3	修改表示层	537
32.7	本章小结	537
第33章	游戏App实战：《迷失航线》	538
33.1	《迷失航线》游戏分析与设计	538
33.1.1	《迷失航线》故事背景	538
33.1.2	需求分析	538
33.1.3	原型设计	539
33.1.4	游戏脚本	540
33.2	任务1：游戏工程的创建与初始化	540
33.2.1	迭代1.1：创建工程	540
33.2.2	迭代1.2：自定义类型维护	540
33.2.3	迭代1.3：添加资源文件	543
33.3	任务2：创建Loading场景	543
33.3.1	迭代2.1：设计场景	544
33.3.2	迭代2.2：Loading动画	545
33.3.3	迭代2.3：预处理加载纹理	545
33.4	任务3：创建Home场景	546
33.4.1	迭代3.1：设计场景	546
33.4.2	迭代3.2：实现代码	547
33.5	任务4：创建设置场景	549



33.5.1	迭代4.1：设计场景	550
33.5.2	迭代4.2：实现代码	550
33.6	任务5：创建帮助场景	552
33.6.1	迭代5.1：设计场景	553
33.6.2	迭代5.2：实现代码	553
33.7	任务6：实现游戏场景	554
33.7.1	迭代6.1：设计场景	555
33.7.2	迭代6.2：创建敌人精灵	556
33.7.3	迭代6.3：创建玩家飞机精灵	560
33.7.4	迭代6.4：创建子弹精灵	560
33.7.5	迭代6.5：初始化游戏场景	561
33.7.6	迭代6.6：玩家移动飞机	565
33.7.7	迭代6.7：游戏循环与任务调度	566
33.7.8	迭代6.8：游戏场景菜单实现	566
33.7.9	迭代6.9：玩家飞机发射子弹	568
33.7.10	迭代6.10：子弹与敌人的碰撞检测	569
33.7.11	迭代6.11：玩家飞机与敌人的碰撞检测	571
33.8	任务7：游戏结束场景	573
33.8.1	迭代7.1：设计场景	573
33.8.2	迭代7.2：实现代码	574
33.9	本章小结	575
第34章	《迷失航线》游戏上架苹果App Store	576
34.1	苹果App Store	576
34.2	还有“最后一公里”	577
34.2.1	添加图标	577
34.2.2	添加启动界面	579
34.2.3	修改发布产品属性	581
34.3	为发布进行编译	582
34.3.1	创建开发者证书	583
34.3.2	创建App ID	587
34.3.3	创建描述文件	589
34.3.4	发布编译	593
34.4	发布上架	594
34.4.1	创建应用	595
34.4.2	应用定价	597
34.4.3	基本信息输入	597
34.4.4	上传应用	601
34.4.5	提交审核	603
34.5	审核不通过的常见原因	604
34.5.1	功能问题	604
34.5.2	用户界面问题	604
34.5.3	商业问题	605
34.5.4	不当内容	605
34.5.5	其他问题	605
34.6	本章小结	605



# 《从零开始学Swift》

## 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:[www.tushu000.com](http://www.tushu000.com)