

Оглавление

Читателю	20
Присоединяйтесь к нам	21
Введение	22
Для кого написана книга	22
Что нужно знать, прежде чем начать читать.....	23
Особенности Swift	23
Современность	23
Объектно-ориентированность	23
Читабельность, экономичность и лаконичность кода	24
Безопасность	24
Производительность	24
Актуальность	24
О том, как работать с книгой.....	24
О домашних заданиях	26
Исправления в шестом издании	27
Структура книги.....	27
Условные обозначения	28
О важности изучения английского языка	29
От издательства	30
Подготовка к разработке Swift-приложений	31
Какие варианты обучения доступны	31
Подготовка к разработке на Mac.....	32
Компьютер Mac.....	32
Зарегистрируйте учетную запись Apple ID	32
Скачайте и установите среду разработки Xcode.....	33
Обзор Xcode	34
Интерфейс playground-проекта	37
Возможности playground-проекта.....	40

Часть I Базовые возможности Swift

Глава 1. Отправная точка.....	45
1.1. Вычислительное мышление	45
1.2. Как компьютер работает с данными.....	48
Аппаратный уровень.....	49
Уровень операционной системы.....	51
Программный уровень	51
1.3. Базовые понятия.....	52
1.4. Введение в операторы	54
Ваш первый программный код	54
Классификация операторов	56
1.5. Оператор инициализации	57
1.6. Переменные и константы.....	57
Переменные	58
Константы	61
Объявление нескольких параметров в одном выражении	62
Где использовать переменные и константы	63
1.7. Инициализация копированием	63
1.8. Правила именования переменных и констант	65
1.9. Возможности автодополнения и кодовые сниппеты	65
1.10. Область видимости (scope)	67
1.11. Комментарии	68
Классические комментарии.....	68
Markdown-комментарии	69
1.12. Точка с запятой	71
1.13. Отладочная консоль и функция print(_:)	71
Консоль.....	71
Вывод текстовой информации	72
Глава 2. Фундаментальные типы данных	78
2.1. Предназначение типов данных	80
2.2. Числовые типы данных	82
Целочисленные типы данных.....	82
Объектные возможности Swift.....	84
Числа с плавающей точкой.....	85

Арифметические операторы	86
Приведение числовых типов данных.....	90
Составной оператор присваивания	91
Способы записи числовых значений	92
Тип данных Decimal и точность операций	94
2.3. Строковые типы данных.....	96
Как компьютер видит строковые данные	97
Инициализация строковых значений.....	99
Тип данных Character.....	99
Тип данных String	99
Пустые строковые литералы	100
Многострочные строковые литералы	101
Приведение к строковому типу данных.....	102
Объединение строк.....	102
Сравнение строк.....	103
Юникод в строковых типах данных.....	104
2.4. Логический тип данных	104
Логические операторы.....	105
Операторы сравнения.....	107
2.5. Псевдонимы типов.....	108
2.6. Дополнительные сведения о типах данных	109
Как узнать тип данных параметра.....	109
Хешируемые и сопоставимые типы данных.....	110
2.7. Где использовать фундаментальные типы.....	112

Часть II

Контейнерные типы данных

Глава 3. Кортежи (Tuple)	116
3.1. Основные сведения о кортежах	116
Литерал кортежа	116
Тип данных кортежа	118
3.2. Взаимодействие с элементами кортежа.....	119
Инициализация значений в параметры	119
Доступ к элементам кортежа через индексы	121
Доступ к элементам кортежа через имена	121
Редактирование кортежа	122

3.3. Сравнение кортежей.....	123
3.4. Где используются кортежи.....	124
Глава 4. Последовательности и коллекции.....	125
4.1. Классификация понятий	125
4.2. Последовательности (Sequence)	127
4.3. Коллекции (Collection).....	129
4.4. Работа с документацией	130
Глава 5. Диапазоны (Range).....	134
5.1. Оператор полукоткрытого диапазона	134
Бинарная форма оператора	134
Префиксная форма оператора	136
5.2. Оператор закрытого диапазона.....	137
Бинарная форма оператора	137
Постфиксная форма оператора.....	138
Префиксная форма оператора	139
5.3. Базовые свойства и методы	139
5.4. Классификация диапазонов	140
5.5. Где использовать диапазоны	144
Глава 6. Массивы (Array)	145
6.1. Введение в массивы.....	145
Хранение массива в памяти компьютера.....	145
Создание массива с помощью литерала	147
Создание массива с помощью Array(arrayLiteral:)	148
Создание массива с помощью Array(_:)	148
Создание массива с помощью Array(repeating:count:).....	149
Доступ к элементам массива.....	150
6.2. Тип данных массива.....	151
6.3. Массив — это value type.....	152
6.4. Пустой массив.....	153
6.5. Операции с массивами.....	153
Сравнение массивов	153
Слияние массивов.....	154
6.6. Многомерные массивы	154
6.7. Базовые свойства и методы массивов.....	155

6.8. Срезы массивов (ArraySlice)	159
Операции с ArraySlice.....	159
6.9. Где использовать массивы	160
Глава 7. Множества (Set)	161
7.1. Введение во множества	161
Варианты создания множества	161
7.2. Пустое множество	163
7.3. Базовые свойства и методы множеств.....	163
Операции со множествами.....	165
Отношения множеств.....	166
7.4. Где использовать множества.....	168
Глава 8. Словари (Dictionary)	169
8.1. Введение в словари	169
Создание словаря с помощью литерала словаря.....	169
Создание словаря с помощью Dictionary(dictionaryLiteral:).....	170
Создание словаря с помощью Dictionary(uniqueKeysWithValues:)	171
8.2. Тип данных словаря.....	172
8.3. Взаимодействие с элементами словаря	173
8.4. Пустой словарь	174
8.5. Базовые свойства и методы словарей	175
8.6. Вложенные типы.....	176
8.7. Где использовать словари	177
Глава 9. Строка — коллекция символов (String).....	178
9.1. Character в составе String.....	178
9.2. Графем-кластеры.....	179
9.3. Строковые индексы.....	181
9.4. Подстроки (Substring)	184

Часть III Основные возможности Swift

Глава 10. Операторы управления	187
10.1. Утверждения.....	188
10.2. Оператор условия if	189
Сокращенный синтаксис оператора if	190
Стандартный синтаксис оператора if.....	192

Расширенный синтаксис оператора if.....	195
Тернарный оператор условия	198
10.3. Оператор ветвления switch	200
Диапазоны в операторе switch.....	202
Кортежи в операторе switch	203
Ключевое слово where в операторе switch.....	205
Связывание значений	205
Оператор break в конструкции switch-case	208
Ключевое слово fallthrough	208
10.4. Операторы повторения while и repeat while.....	209
Оператор while	209
Оператор repeat while.....	210
Использование оператора continue	211
Использование оператора break	211
10.5. Оператор повторения for	212
Использование where в конструкции for-in.....	217
Многомерные коллекции в конструкции for-in.....	218
Использование continue в конструкции for-in	219
Использование break в конструкции for-in.....	219
10.6. Оператор досрочного выхода guard	221
10.7. Где использовать операторы управления	222
Глава 11. Опциональные типы данных	226
11.1. Введение в опционалы.....	226
Опционалы в кортежах.....	230
11.2. Извлечение опционального значения.....	230
Принудительное извлечение значения	231
Косвенное извлечение значения.....	232
11.3. Проверка наличия значения в опционале	233
11.4. Опциональное связывание	234
11.5. Опциональное связывание как часть оптимизации кода.....	236
11.6. Оператор объединения с nil	238
11.7. Где использовать опциональные значения	239
Глава 12. Функции	241
12.1. Введение в функции	241
12.2. Входные параметры и возвращаемое значение	245
Входные параметры.....	245

Внешние имена входных параметров	246
Возвращаемое значение	246
Изменяемые копии входных параметров.....	247
Сквозные параметры	248
Функция в качестве входного параметра	248
Входной параметр с переменным числом значений	249
Кортеж в качестве возвращаемого значения	249
Значение по умолчанию для входного параметра	250
12.3. Функциональный тип	251
Простой функциональный тип	252
Сложный функциональный тип.....	252
12.4. Функция в качестве входного и возвращаемого значений	253
Возвращаемое значение функционального типа	253
Входное значение функционального типа.....	254
Параметры функционального типа для ленивых вычислений	256
12.5. Возможности функций	257
Вложенные функции.....	257
Перегрузка функций.....	258
Рекурсивный вызов функций	259
12.6. Где использовать функции	259
Глава 13. Замыкания (closure)	261
13.1. Виды замыканий	261
13.2. Введение в безымянные функции	261
13.3. Возможности замыканий	265
Пропуск указания типов	266
Неявное возвращение значения	266
Сокращенные имена параметров	266
Вынос замыкания за скобки	267
Вынос нескольких замыканий за скобки.....	267
13.4. Безымянные функции в параметрах	268
13.5. Пример использования замыканий при сортировке массива.....	269
13.6. Захват переменных	270
Синтаксис захвата переменных.....	270
Захват вложенной функцией	271
13.7. Замыкания передаются по ссылке	272
13.8. Автозамыкания	273

13.9. Выходящие (сбегающие) замыкания	275
13.10. Где использовать замыкания.....	277
Глава 14. Дополнительные возможности.....	279
14.1. Метод map(_:):.....	279
14.2. Метод mapValues(_:):	281
14.3. Метод flatMap(_:):	281
14.4. Метод compactMap(_:):.....	282
14.5. Метод filter(_:):	282
14.6. Метод reduce(_:_:):	283
14.7. Метод zip(_:_:):	284
14.8. Оператор guard для опционалов	285
14.9. Оператор отложенных действий defer	286
Глава 15. Ленивые вычисления	288
15.1. Понятие ленивых вычислений.....	288
15.2. Замыкания в ленивых вычислениях.....	288
15.3. Свойство lazy	289

Часть IV

Введение в разработку приложений

Глава 16. Консольное приложение «Сумма двух чисел».....	291
16.1. Обзор интерфейса Xcode.....	291
Создание Xcode-проекта	291
Интерфейс и элементы управления Xcode-проектом	294
16.2. Подготовка к разработке приложения.....	296
16.3. Запуск приложения	299
16.4. Код приложения «Сумма двух чисел»	301
Глава 17. Консольная игра «Угадай число».....	305
17.1. Код приложения «Угадай число».....	306
17.2. Устраняем ошибки приложения.....	307

Часть V

Нетривиальные возможности Swift

Глава 18. Введение в объектно-ориентированное и протокол-ориентированное программирование	310
18.1. Экземпляры	310

18.2. Модули	312
18.3. Пространства имен	313
18.4. API Design Guidelines	313
Глава 19. Перечисления	315
19.1. Синтаксис перечислений	315
19.2. Ассоциированные параметры	317
19.3. Вложенные перечисления	319
19.4. Оператор switch для перечислений	320
19.5. Связанные значения членов перечисления	320
Указание связанных значений	321
Доступ к связанным значениям	321
19.6. Инициализатор	322
19.7. Свойства в перечислениях	323
19.8. Методы в перечислениях	323
19.9. Оператор self	324
19.10. Рекурсивные перечисления	325
19.11. Где использовать перечисления	328
Глава 20. Структуры	329
20.1. Синтаксис объявления структур	329
20.2. Свойства в структурах	330
Объявление свойств	330
Встроенный инициализатор	331
Значения свойств по умолчанию	331
20.3. Структура как пространство имен	332
20.4. Собственные инициализаторы	333
20.5. Методы в структурах	334
Объявление методов	334
Изменяющие методы	335
Глава 21. Классы	336
21.1. Синтаксис классов	336
21.2. Свойства классов	337
21.3. Методы классов	339
21.4. Инициализаторы классов	340
21.5. Вложенные в класс типы	341
Ссылки на вложенные типы	342

Глава 22. Свойства.....	343
22.1. Типы свойств	343
Хранимые свойства.....	343
Ленивые хранимые свойства	343
Вычисляемые свойства	345
22.2. Контроль значений свойств.....	346
Геттер и сеттер вычисляемого свойства	346
Наблюдатели хранимых свойств	348
22.3. Свойства типа	350
Глава 23. Сабскрипты.....	352
23.1. Назначение сабскриптов	352
23.2. Синтаксис сабскриптов	352
Глава 24. Наследование.....	357
24.1. Синтаксис наследования	357
Доступ к наследуемым характеристикам.....	358
24.2. Переопределение наследуемых элементов.....	359
Переопределение методов.....	359
Доступ к переопределенным элементам суперкласса.....	360
Переопределение инициализаторов	361
Переопределение наследуемых свойств	361
24.3. Модификатор final.....	362
24.4. Подмена экземпляров классов	362
24.5. Приведение типов.....	363
Проверка типа	363
Преобразование типа	363
Глава 25. Контроль доступа.....	365
Глава 26. Псевдонимы Any и AnyObject.....	369
26.1. Псевдоним Any.....	369
Приведение типа Any.....	370
26.2. Псевдоним AnyObject	370
Приведение типа AnyObject	371
Глава 27. Инициализаторы и деинициализаторы.....	372
27.1. Инициализаторы	372
Назначенные инициализаторы.....	372

Вспомогательные инициализаторы	373
Наследование инициализаторов	374
Отношения между инициализаторами	374
Проваливающиеся инициализаторы.....	375
Обязательные инициализаторы	377
27.2. Деинициализаторы.....	378
Глава 28. Управление памятью в Swift.....	380
28.1. Что такое управление памятью	380
Статическая память	382
Автоматическая память	382
Динамическая память	385
28.2. Уничтожение экземпляров	387
Количество ссылок на экземпляр.....	388
28.3. Утечки памяти и ARC.....	390
Пример утечки памяти.....	390
Сильные (strong), слабые (weak) и бесхозные (unowned) ссылки	393
Automatic Reference Counting (ARC).....	396
28.4. Ссылки в замыканиях.....	396
Глава 29. Опциональные цепочки.....	400
29.1. Доступ к свойствам через опциональные цепочки.....	400
29.2. Установка значений через опциональные цепочки.....	402
29.3. Доступ к методам через опциональные цепочки	403
29.4. Доступ к сабскриптам через опциональные цепочки.....	403
Глава 30. Протоколы	405
30.1. Понятие протокола	405
30.2. Требуемые свойства.....	406
30.3. Требуемые методы.....	408
30.4. Требуемые инициализаторы.....	409
30.5. Протокол в качестве типа данных.....	410
Протокол, указывающий на множество типов	410
Протокол и операторы as? и as!.....	410
Протокол и оператор is.....	411
30.6. Наследование протоколов.....	412
30.7. Классовые протоколы.....	413
30.8. Композиция протоколов.....	413

Глава 31. Расширения	415
31.1. Вычисляемые свойства в расширениях.....	415
31.2. Методы в расширениях	416
31.3. Инициализаторы в расширениях	417
31.4. Сабскрипты в расширениях.....	418
31.5. Расширения протоколов.....	419
Подпись объектного типа на протокол.....	419
Расширение протоколов и реализации по умолчанию.....	420
Глава 32. Протокол-ориентированное программирование	422
32.1. Важность использования протоколов.....	422
Целостность типов данных	423
Инкапсуляция	423
Полиморфизм	424
32.2. Протокол-ориентированное программирование	425
32.3. Где использовать class и struct.....	427
Правила выбора между классом и структурой.....	428
Глава 33. Разработка приложения в Xcode Playground	433
33.1. UIKit и SwiftUI	433
33.2. Разработка интерактивного приложения.....	434
Библиотека PlaygroundSupport	434
Структура проекта	435
Класс Ball.....	436
Класс SquareArea	438
Глава 34. Универсальные шаблоны (Generic).....	446
34.1. Зачем нужны дженерики	446
34.2. Универсальные функции	447
34.3. Ограничения типа	449
34.4. Универсальные объектные типы	452
Расширения универсального типа.....	453
34.5. Универсальные протоколы	454
Использование ассоциированных параметров.....	456
Дальнейшая доработка сущности	457
34.6. Непрозрачные типы (Opaque types) и ключевое слово some	458
Решение проблемы	460

Глава 35. Обработка ошибок	465
35.1. Выбрасывание ошибок	466
35.2. Обработка ошибок	467
Передача ошибки	467
Оператор do-catch	469
Преобразование ошибки в опционал	470
Подавление выброса ошибки.....	471
35.3. Структуры и классы для обработки ошибок.....	471
Глава 36. Нетривиальное использование операторов	473
36.1. Операторные функции	473
Префиксные и постфиксные операторы.....	474
Составной оператор присваивания	474
Оператор эквивалентности	475
36.2. Пользовательские операторы.....	475

Часть VII

Введение в мобильную разработку

Глава 37. Разработка приложения с использованием UIKit	478
37.1. Создание проекта MyName	478
37.2. Структура проекта.....	481
37.3. Разработка простейшего UI	487
Шаг 1. Размещение кнопки	488
Шаг 2. Изменение текста кнопки	489
37.4. Запуск приложения в симуляторе.....	490
37.5. View Controller сцены и класс UIViewController	494
37.6. Доступ UI к коду. Определитель типа @IBAction	497
37.7. Создание дополнительной сцены.....	499
37.8. Отображение всплывающего окна. Класс UIAlertController.....	505
37.9. Изменение атрибутов кнопки	510
37.10. Доступ кода к UI. Определитель типа @IBOutlet.....	512
Глава 38. Разработка приложения с использованием SwiftUI	519
38.1. Создание нового проекта	520
38.2. Структура проекта на SwiftUI	522

38.3. Редактирование элементов интерфейса	529
38.4. Создание приложения.....	531
Глава 39. Паттерны проектирования	538
39.1. Паттерн MVC. Фреймворк Cocoa Touch	538
39.2. Паттерн Singleton. Класс UIApplication	539
39.3. Паттерн Delegation. Класс UIApplicationDelegate	541
Заключение.....	542