

ПРОГРАММИРОВАНИЕ  
ОТ ЭКСПЕРТОВ

**САМОЕ ПОЛНОЕ  
РУКОВОДСТВО  
ПО РАЗРАБОТКЕ**

**SWIFT**

**в примерах от сообщества  
Stack Overflow**

МОСКВА  
ИЗДАТЕЛЬСТВО АСТ

УДК 004.43  
ББК 32.973.26-018.1  
С24

Последнюю версию этой книги на английском языке можно скачать с сайта: <https://goalkicker.com/SwiftBook>. Пожалуйста, не стесняйтесь поделиться этим PDF-файлом с кем угодно бесплатно.

Книга Swift™ Notes for Professionals составлена на основе документации Stack Overflow (<https://archive.org/details/documentation-dump.7z>), содержание которой написано прекрасными людьми из Stack Overflow. **Текстовые материалы публикуются на условиях Creative Commons BY-SA.** В конце книги указаны авторы, внесшие вклад в создание различных разделов. Изображения могут являться объектами авторского права соответствующих владельцев, если не указано иное.

**Swift.** Самое полное руководство по разработке в примерах от сообщества Stack Overflow. — Москва : Издательство АСТ, 2025. — 480 с. : ил. — (Программирование от экспертов).  
ISBN 978-5-17-160275-8.

Эта книга знакомит читателя с основами программирования на широко распространенном языке Swift, который специально разработан компанией Apple для создания приложений под iOS и macOS. Основное внимание в ней уделено практическим аспектам, что позволит и новичкам, и опытным разработчикам сразу применять теорию на практике. Специалисты разного уровня подготовки найдут здесь всю необходимую информацию — от основ синтаксиса языка и работы со Swift Playgrounds до знакомства с инструментарием JSON, рассмотрения примеров взаимодействия Swift с Objective-C, использования в Swift шаблонов проектирования, парадигм протоколно-ориентированного и функционального программирования. Кроме того, в этом практическом руководстве представлена информация о применении Swift в сферах безопасности и криптографии, а также о повышении производительности Swift-кода.

**УДК 004.43**  
**ББК 32.973.26-018.1**

**ISBN 978-5-17-160275-8**

Перевод на русский язык: ООО «Интеджер»  
Издание на русском языке: ООО «Издательство АСТ»

# Содержание

<b>Раздел 1. Начало работы с языком Swift</b> .....	14
Глава 1.1. Ваша первая программа на Swift.....	15
Глава 1.2. Ваша первая программа на Swift на Mac (с использованием Playground).....	17
Глава 1.3. Ваша первая программа в приложении Swift Playgrounds на iPad.....	20
Глава 1.4. Установка Swift.....	22
Глава 1.5. Опциональные значения и Optional enum (перечисления).....	22
<b>Раздел 2. Переменные и свойства</b> .....	24
Глава 2.1. Создание переменной.....	24
Глава 2.2. Наблюдатели свойств.....	25
Глава 2.3. Ленивые хранимые свойства.....	26
Глава 2.4. Основы свойств.....	27
Глава 2.5. Вычисляемые свойства.....	27
Глава 2.6. Локальные и глобальные переменные.....	28
Глава 2.7. Свойства типов.....	29
<b>Раздел 3. Числа</b> .....	31
Глава 3.1. Числовые типы и литералы.....	31
Глава 3.2. Преобразование чисел в строки и обратно.....	33
Глава 3.3. Округление.....	33
Глава 3.4. Генерация случайных чисел.....	35
Глава 3.5. Преобразование одного числового типа в другой.....	36
Глава 3.6. Возведение в степень.....	37
<b>Раздел 4. Строки и символы</b> .....	39
Глава 4.1. Литералы строк и символов.....	39
Глава 4.2. Конкатенация строк.....	41
Глава 4.3. Кодирование и декомпозиция строк.....	42
Глава 4.4. Исследование и сравнение строк.....	44
Глава 4.5. Реверсирование строк.....	44

Глава 4.6. Проверка наличия в строке символов из заданного набора.....	45
Глава 4.7. Итерация строк.....	47
Глава 4.8. Разбиение строки на массив.....	50
Глава 4.9. Юникод.....	51
Глава 4.10. Преобразование строки Swift в числовой тип.....	52
Глава 4.11. Преобразование строки в Data / NSData и обратно.....	53
Глава 4.12. Форматирование строк.....	53
Глава 4.13. Строки в верхнем и нижнем регистре.....	54
Глава 4.14. Удаление символов из строки, не определенной в Set.....	55
Глава 4.15. Подсчет вхождений символа в строку.....	56
Глава 4.16. Удаление ведущих и завершающих пробелов и символов новой строки.....	56
<b>Раздел 5. Логические значения (Booleans).....</b>	<b>58</b>
Глава 5.1. Что такое Bool?.....	58
Глава 5.2. Логические значения и условный оператор.....	58
Глава 5.3. Логические операторы.....	60
Глава 5.4. Отрицание логического значения с помощью префиксного оператора !.....	61
<b>Раздел 6. Массивы.....</b>	<b>62</b>
Глава 6.1. Основы работы с массивами.....	62
Глава 6.2. Извлечение значений заданного типа из массива с помощью flatMap(:).....	64
Глава 6.3. Объединение элементов массива с помощью reduce(:combine:).....	65
Глава 6.4. Сведение результата преобразования массива с помощью flatMap(:).....	65
Глава 6.5. Ленивое сглаживание многомерного массива с помощью функции flatten().....	67
Глава 6.6. Фильтрация nil из преобразования массива с помощью flatMap(:).....	67
Глава 6.7. Индексация массива с диапазоном.....	68
Глава 6.8. Удаление элемента из массива без знания его индекса.....	69
Глава 6.9. Сортировка массива строк.....	70
Глава 6.10. Безопасный доступ к индексам.....	72
Глава 6.11. Фильтрация массива.....	73
Глава 6.12. Преобразование элементов массива с помощью map(:).....	73
Глава 6.13. Полезные методы.....	74
Глава 6.14. Сортировка массива.....	75

---

Глава 6.15. Нахождение минимального или максимального элемента массива .....	77
Глава 6.16. Изменение значений в массиве.....	79
Глава 6.17. Сравнение двух массивов с помощью zip .....	79
Глава 6.18. Группировка значений массива.....	80
Глава 6.19. Семантика значений.....	81
Глава 6.20. Доступ к значениям массива.....	82
<b>Раздел 7. Кортежи .....</b>	<b>84</b>
Глава 7.1. Что такое кортежи? .....	84
Глава 7.2. Декомпозиция на отдельные переменные.....	85
Глава 7.3. Кортежи как возвращаемое значение функций.....	86
Глава 7.4. Использование typealiases для именованного типа кортежа.....	87
Глава 7.5. Замена значений .....	88
Глава 7.6. Кортежи как case в switch.....	89
<b>Раздел 8. Перечисления .....</b>	<b>91</b>
Глава 8.1. Основные перечисления.....	91
Глава 8.2. Перечисления со связанными значениями .....	92
Глава 8.3. Косвенные значения .....	94
Глава 8.4. Значения Raw и Nash .....	95
Глава 8.5. Инициализаторы.....	97
Глава 8.6. Перечисления имеют много общих черт с классами и структурами .....	99
Глава 8.7. Вложенные перечисления .....	100
<b>Раздел 9. Структуры .....</b>	<b>102</b>
Глава 9.1. Структуры — это типы значений.....	102
Глава 9.2. Доступ к членам структуры .....	103
Глава 9.3. Основы работы со структурами.....	103
Глава 9.4. Изменение (mutating) структуры .....	104
Глава 9.5. Структуры не могут наследоваться.....	105
<b>Раздел 10. Множества.....</b>	<b>106</b>
Глава 10.1. Объявление множеств .....	106
Глава 10.2. Выполнение операций над множествами.....	106
Глава 10.3. CountedSet.....	108
Глава 10.4. Изменение значений в множестве.....	108
Глава 10.5. Проверка наличия значения в множестве .....	109
Глава 10.6. Добавление значений собственного типа в множество .....	109

<b>Раздел 11. Словари</b> .....	<b>111</b>
Глава 11.1. Объявление словарей .....	111
Глава 11.2. Доступ к значениям .....	112
Глава 11.3. Изменение значения словаря с помощью ключа .....	113
Глава 11.4. Получение всех ключей в словаре .....	113
Глава 11.5. Изменение словарей .....	114
Глава 11.6. Слияние двух словарей .....	114
<b>Раздел 12. Оператор switch</b> .....	<b>116</b>
Глава 12.1. Switch и опциональные значения .....	116
Глава 12.2. Основное использование .....	117
Глава 12.3. Сопоставление диапазона .....	117
Глава 12.4. Частичное сопоставление .....	118
Глава 12.5. Использование оператора where в switch .....	120
Глава 12.6. Сопоставление нескольких значений .....	121
Глава 12.7. Switch и перечисления (enums) .....	121
Глава 12.8. Switch и кортежи (Tuples) .....	122
Глава 12.9. Удовлетворение одного из нескольких ограничений с помощью switch .....	123
Глава 12.10. Сопоставление на основе класса — полезно для prepareForSegue .....	124
Глава 12.11. Оператор fallthrough в switch .....	125
<b>Раздел 13. Опционалы</b> .....	<b>127</b>
Глава 13.1. Типы опционалов .....	127
Глава 13.2. Извлечение опционала .....	128
Глава 13.3. Оператор объединения с nil .....	130
Глава 13.4. Опциональная цепочка .....	131
Глава 13.5. Обзор: зачем использовать опционалы? .....	133
<b>Раздел 14. Условные выражения</b> .....	<b>135</b>
Глава 14.1. Опциональная привязка и where условия .....	135
Глава 14.2. Использование Guard .....	137
Глава 14.3. Базовые условные выражения: if-выражения .....	138
Глава 14.4. Тернарный оператор .....	140
Глава 14.5. Оператор объединения с nil .....	141
<b>Раздел 15. Обработка ошибок</b> .....	<b>143</b>
Глава 15.1. Основы обработки ошибок .....	143
Глава 15.2. Перехват различных типов ошибок .....	146
Глава 15.3. Шаблон catch и switch для явной обработки ошибок .....	147

---

Глава 15.4. Отключение распространения ошибок .....	149
Глава 15.5. Создание пользовательской ошибки с локализованным описанием .....	149
<b>Раздел 16. Циклы.....</b>	<b>151</b>
Глава 16.1. Цикл For-in.....	151
Глава 16.2. Цикл Repeat-while.....	154
Глава 16.3. Цикл For-in с фильтрацией.....	155
Глава 16.4. Метод forEach для типов последовательностей .....	156
Глава 16.5. Цикл while.....	157
Глава 16.6. Разрыв цикла .....	157
<b>Раздел 17. Протоколы.....</b>	<b>159</b>
Глава 17.1. Основы протоколов .....	159
Глава 17.2. Паттерн делегирования.....	163
Глава 17.3. Требования к связанным типам.....	166
Глава 17.4. Протоколы только для классов .....	169
Глава 17.5. Расширение протокола для конкретного соответствующего класса .....	171
Глава 17.6. Использование протокола RawRepresentable (расширяемое перечисление).....	172
Глава 17.7. Реализация протокола Hashable.....	173
<b>Раздел 18. Функции .....</b>	<b>175</b>
Глава 18.1. Основное использование.....	175
Глава 18.2. Функции с параметрами.....	175
Глава 18.3. Индексы .....	177
Глава 18.4. Методы .....	178
Глава 18.5. Параметры с переменным числом аргументов.....	179
Глава 18.6. Операторы как функции .....	181
Глава 18.7. Передача и возврат функций.....	181
Глава 18.8. Типы функций .....	182
Глава 18.9. Параметры inout.....	182
Глава 18.10. Выбрасывание ошибок .....	183
Глава 18.11. Возврат значений.....	184
Глава 18.12. Синтаксис завершающего замыкания .....	184
Глава 18.13. Функции с замыканиями.....	185
<b>Раздел 19. Расширения.....</b>	<b>187</b>
Глава 19.1. Что такое расширения? .....	187
Глава 19.2. Переменные и функции.....	187

Глава 19.3. Инициализаторы в расширениях.....	188
Глава 19.4. Сабскрипты.....	189
Глава 19.5. Расширения протокола.....	190
Глава 19.6. Ограничения.....	191
Глава 19.7. Что такое расширения и когда их использовать .....	191
<b>Раздел 20. Классы.....</b>	<b>193</b>
Глава 20.1. Определение класса .....	193
Глава 20.2. Свойства и методы.....	193
Глава 20.3. Семантика ссылок.....	194
Глава 20.4. Классы и множественное наследование .....	196
Глава 20.5. deinit.....	197
<b>Раздел 21. Приведение типов.....</b>	<b>198</b>
Глава 21.1. Понижающее приведение (downcasting).....	198
Глава 21.2. Приведение типов в языке Swift.....	199
Глава 21.3. Повышающее приведение (upcasting).....	202
Глава 21.4. Пример использования downcast в функции .....	203
Глава 21.5. Приведение с использованием switch.....	203
<b>Раздел 22. Обобщенные типы (Generics) .....</b>	<b>205</b>
Глава 22.1. Основы обобщенных типов .....	205
Глава 22.2. Ограничение типов заполнителей обобщенных типов .....	207
Глава 22.3. Примеры обобщенных классов .....	208
Глава 22.4. Использование обобщенных типов для упрощения функций массива .....	210
Глава 22.5. Расширенные ограничения типов.....	211
Глава 22.6. Наследование обобщенных классов .....	213
Глава 22.7. Использование обобщений для повышения безопасности типов .....	214
<b>Раздел 23. OptionSet.....</b>	<b>216</b>
Глава 23.1. Протокол OptionSet.....	216
<b>Раздел 24. Чтение и запись JSON .....</b>	<b>218</b>
Глава 24.1. Сериализация, кодирование и декодирование JSON с использованием Apple Foundation и стандартной библиотеки Swift .....	218
Глава 24.2. SwiftyJSON.....	223
Глава 24.3. Freddy.....	226
Глава 24.4. Парсинг JSON Swift 3.....	230
Глава 24.5. Простой разбор JSON в пользовательские объекты.....	234
Глава 24.6. Arrow.....	236

---

<b>Раздел 25. Расширенные операторы.....</b>	<b>240</b>
Глава 25.1. Побитовые операторы.....	240
Глава 25.2. Пользовательские операторы.....	242
Глава 25.3. Операторы переполнения.....	244
Глава 25.4. Коммутативные операторы .....	244
Глава 25.5. Перегрузка + для словарей .....	245
Глава 25.6. Приоритет стандартных операторов Swift.....	246
<b>Раздел 26. Swizzling методов .....</b>	<b>249</b>
Глава 26.1. Расширение UIViewController и Swizzling viewDidLoad .....	249
Глава 26.2. Основы Swizzling в Swift.....	251
Глава 26.3. Основы Swizzling в Objective-C .....	252
<b>Раздел 27. Рефлексия.....</b>	<b>254</b>
Глава 27.1. Основы использования Mirror.....	254
Глава 27.2. Получение типа и имен свойств класса без необходимости создания экземпляра .....	255
<b>Раздел 28. Управление доступом.....</b>	<b>260</b>
Глава 28.1. Базовый пример с использованием структуры.....	260
Глава 28.2. Пример использования наследования.....	262
Глава 28.3. Пример использования геттеров и сеттеров .....	262
<b>Раздел 29. Замыкания .....</b>	<b>263</b>
Глава 29.1. Основы использования замыканий .....	263
Глава 29.2. Вариации синтаксиса .....	264
Глава 29.3. Передача замыканий в функции .....	266
Глава 29.4. Захваты, сильные/слабые ссылки и циклы удержания.....	269
Глава 29.5. Использование замыканий для асинхронного кодирования .....	271
Глава 29.6. Замыкания и псевдонимы типов.....	273
<b>Раздел 30. Инициализаторы.....</b>	<b>274</b>
Глава 30.1. Инициализация .....	274
Глава 30.2. Установка значений по умолчанию для свойств .....	277
Глава 30.3. Настройка инициализации с параметрами .....	278
Глава 30.4. Инициализатор, который может выбрасывать ошибки .....	280
<b>Раздел 31. Связанные объекты .....</b>	<b>282</b>
Глава 31.1. Свойство в расширении протокола, достигнутое с помощью связанного объекта.....	282

<b>Раздел 32. Параллелизм</b> .....	<b>288</b>
Глава 32.1. Получение очереди Grand Central Dispatch (GCD).....	288
Глава 32.2. Параллельные циклы.....	290
Глава 32.3. Выполнение задач в очереди Grand Central Dispatch (GCD).....	292
Глава 32.4. Выполнение задач в OperationQueue.....	294
Глава 32.5. Создание высокоуровневых операций.....	296
<b>Раздел 33. Начало работы с протоколно-ориентированным программированием</b> .....	<b>299</b>
Глава 33.1. Использование протоколов в качестве типов первого класса.....	299
Глава 33.2. Использование протоколно-ориентированного программирования для юнит-тестирования.....	304
<b>Раздел 34. Функциональное программирование в Swift</b> .....	<b>308</b>
Глава 34.1. Извлечение списка имен из списка Person.....	308
Глава 34.2. Перебор.....	309
Глава 34.3. Фильтрация.....	309
Глава 34.4. Использование фильтра со структурами.....	311
Глава 34.5. Проекция.....	313
<b>Раздел 35. Функции как объекты первого класса в Swift</b> .....	<b>315</b>
Глава 35.1. Присвоение функции переменной.....	315
Глава 35.2. Передача функции в качестве аргумента другой функции и создание тем самым функции более высокого порядка...	316
Глава 35.3. Функция как тип возвращаемого значения из другой функции.....	317
<b>Раздел 36. Замыкания</b> .....	<b>318</b>
Глава 36.1. Замыкание без выхода («несбегающее» замыкание).....	318
Глава 36.2. Замыкание с выходом.....	319
<b>Раздел 37. Оператор defer</b> .....	<b>320</b>
Глава 37.1. Когда использовать оператор defer.....	320
Глава 37.2. Когда НЕ использовать оператор defer.....	321
<b>Раздел 38. Стиливые соглашения</b> .....	<b>322</b>
Глава 38.1. Плавное использование.....	322
Глава 38.2. Ясное использование.....	324
Глава 38.3. Капитализация.....	326
<b>Раздел 39. NSRegularExpression в Swift</b> .....	<b>329</b>
Глава 39.1. Расширение String для простой проверки шаблонов.....	329
Глава 39.2. Основное использование.....	331

---

Глава 39.3. Замена подстрок.....	333
Глава 39.4. Специальные символы.....	333
Глава 39.5. Валидация.....	334
Глава 39.6. NSRegularExpression для валидации электронной почты.....	335
<b>Раздел 40. Библиотека RxSwift.....</b>	<b>336</b>
Глава 40.1. Удаление.....	336
Глава 40.2. Основы RxSwift.....	337
Глава 40.3. Создание Observable.....	338
Глава 40.4. Связывание.....	340
Глава 40.5. RxCocoa и ControlEvents.....	341
<b>Раздел 41. Swift Package Manager.....</b>	<b>344</b>
Глава 41.1. Создание и использование простого Swift-пакета.....	344
<b>Раздел 42. Работа с C и Objective-C.....</b>	<b>346</b>
Глава 42.1. Использование карты модулей для импорта заголовков C.....	346
Глава 42.2. Использование классов Objective-C в коде Swift.....	347
Глава 42.3. Указание файла связывания для swiftc.....	349
Глава 42.4. Использование стандартной библиотеки C.....	349
Глава 42.5. Детализированная совместимость между Objective-C и Swift.....	350
Глава 42.6. Использование классов Swift из кода Objective-C.....	352
<b>Раздел 43. Разметка документации.....</b>	<b>354</b>
Глава 43.1. Документация класса.....	354
Глава 43.2. Стили документации.....	355
<b>Раздел 44. Typealias (псевдоним типа).....</b>	<b>359</b>
Глава 44.1. typealias для замыканий с параметрами.....	359
Глава 44.2. typealias для пустых замыканий.....	359
Глава 44.3. typealias для других типов.....	360
<b>Раздел 45. Внедрение зависимостей.....</b>	<b>361</b>
Глава 45.1. Внедрение зависимостей с помощью контроллеров представлений.....	361
Глава 45.2. Типы внедрения зависимостей.....	368
<b>Раздел 46. Кеширование на дисковом пространстве.....</b>	<b>372</b>
Глава 46.1. Чтение.....	372
Глава 46.2. Сохранение.....	373

<b>Раздел 47. Алгоритмы в Swift</b> .....	374
Глава 47.1. Сортировка .....	374
Глава 47.2. Сортировка вставками .....	379
Глава 47.3. Сортировка выбором .....	380
Глава 47.4. Асимптотический анализ .....	381
Глава 47.5. Быстрая сортировка — сложность $O(n \log n)$ .....	382
Глава 47.6. Графы, префиксные деревья, стеки .....	384
<b>Раздел 48. Расширенные функции</b> .....	404
Глава 48.1. Сглаживание многомерного массива.....	404
Глава 48.2. Введение в расширенные функции.....	405
<b>Раздел 49. Обработчик завершения (Completion Handler) .....</b>	<b>408</b>
Глава 49.1. Обработчик завершения без входного аргумента .....	408
Глава 49.2. Обработчик завершения с входным аргументом.....	408
<b>Раздел 50. Swift HTTP-сервер от Kitura .....</b>	<b>411</b>
Глава 50.1. Приложение Hello world.....	411
<b>Раздел 51. Генерация UIImage из инициалов из строки .....</b>	<b>415</b>
Глава 51.1. InitialsImageFactory.....	415
<b>Раздел 52. Порождающие (Creational) шаблоны проектирования ...</b>	<b>417</b>
Глава 52.1. Синглтон (Singleton) .....	417
Глава 52.2. Шаблон «Строитель» (Builder Pattern) .....	418
Глава 52.3. Шаблон «Фабричный метод» .....	428
Глава 52.4. Шаблон «Наблюдатель».....	430
Глава 52.5. Шаблон «Цепочка обязанностей» .....	432
Глава 52.6. Шаблон «Итератор».....	434
<b>Раздел 53. Структурные шаблоны проектирования .....</b>	<b>436</b>
Глава 53.1. Адаптер .....	436
Глава 53.2. Шаблон «Фасад» .....	437
<b>Раздел 54. (Небезопасные) указатели на буферы.....</b>	<b>439</b>
Глава 54.1. UnsafeMutablePointer .....	439
Глава 54.2. Практический случай использования указателей на буферы .....	441
<b>Раздел 55. Криптографическое хеширование.....</b>	<b>443</b>
Глава 55.1. HMAC с MD5, SHA1, SHA224, SHA256, SHA384, SHA512 (Swift 3).....	443
Глава 55.2. Функции MD2, MD4, MD5, SHA1, SHA224, SHA256, SHA384, SHA512 в Swift 3 .....	446

---

<b>Раздел 56. Шифрование по стандарту AES (Advanced Encryption Standard)</b> .....	<b>449</b>
Глава 56.1. Шифрование по стандарту AES в режиме CBC со случайным IV (Swift 3.0).....	449
Глава 56.2. Шифрование по стандарту AES в режиме CBC со случайным IV (Swift 2.3).....	454
Глава 56.3. Шифрование по стандарту AES в режиме ECB с PKCS7 padding.....	458
<b>Раздел 57. Формирование ключа с использованием стандарта PBKDF2</b> .....	<b>460</b>
Глава 57.1. Использование стандарта формирования ключа на основе пароля PBKDF2 в Swift 3.....	460
Глава 57.2. Использование стандарта формирования ключа на основе пароля PBKDF2 в Swift 2.3.....	462
Глава 57.3. Калибровка формирования ключа на основе пароля в Swift 2.3.....	464
Глава 57.4. Калибровка вывода ключа на основе пароля в Swift 3.....	465
<b>Раздел 58. Логирование в Swift</b> .....	<b>467</b>
Глава 58.1. Использование функции dump .....	467
Глава 58.2. Использование функции debugPrint .....	469
Глава 58.3. print() vs dump() .....	470
Глава 58.4. print vs NSLog.....	471
<b>Раздел 59. Управление памятью</b> .....	<b>473</b>
Глава 59.1. Циклы ссылок и слабые ссылки .....	473
Глава 59.2. Ручное управление памятью .....	474
<b>Раздел 60. Производительность</b> .....	<b>476</b>
Глава 60.1. Производительность выделения памяти.....	476
Благодарности.....	478

# Раздел 1.

## Начало работы с языком Swift

Версия Swift	Версия Xcode	Дата выхода
Разработка началась (первый коммит)	–	2010-07-17
1.0	Xcode 6	2014-10-02
1.1	Xcode 6.1	2014-10-16
1.2	Xcode 6.3	2015-02-09
2.0	Xcode 7	2015-06-08
2.1	Xcode 7.1	2015-09-23
Дебют открытого исходного кода	–	2015-12-03
2.2	Xcode 7.3	2016-03-21
2.3	Xcode 8	2016-09-13
3.0	Xcode 8	2016-09-13
3.1	Xcode 8.3	2017-03-27
4.0	Xcode 9	2017-11-19
4.1	Xcode 9.3	2018-03-29
4.2	Xcode 10.0	2018-09-17
5.0	Xcode 10.0	2019-03-25
5.1	Xcode 10.0	2019-09-20
5.2	Xcode 11.4	2020-03-24
5.3	Xcode 12.0	2020-09-16

5.4	Xcode 12.5	2021-04-26
5.5	Xcode 13.0	2021-09-20
5.6	Xcode 13.3	2022-03-14
5.7	Xcode 14.0	2022-09-12
5.8	Xcode 14.3	2023-03-30
5.9	Xcode 15.0	2023-09-18
5.10	Xcode 15.3	2024-03-05

## Глава 1.1. Ваша первая программа на Swift

Напишите свой код в файле с именем `hello.swift`:

```
print("Hello, world!")
```

- Чтобы скомпилировать и запустить скрипт за один шаг, наберите в терминале `swift` (в каталоге, где находится этот файл):

Чтобы запустить терминал, нажмите **CTRL + ALT + T** в *Linux* или найдите его в Launchpad в *macOS*. Чтобы сменить каталог, введите **cd/имя\_каталога** (или **cd ..** для возврата назад).

```
$ swift hello.swift
Hello, world!
```

**Компилятор** — это компьютерная программа (или набор программ), которая преобразует исходный код, написанный на языке программирования (исходном языке), в другой компьютерный язык (целевой язык), причем последний часто имеет двоичную форму, называемую объектным кодом.

- Для отдельной компиляции и запуска используйте `swiftc`:

```
$ swiftc hello.swift
```