Основы Swift

Операторы

Введение. Типы операторов

Операторы — это специальные символы или фразы, которые вы используете для комбинирования, проверки или изменения значений.

Большая часть операторов — стандартные операторы языка C.

Унарный
    -variable; !variable
Бинарный
    variable + variable
Тернарный
    boolValue ? value1 : value2

https://developer.apple.com/documentation/swift/operator_declarations

Оператор присвоения (=)

Для объявления или обновления значения

```
let b = 10
var a = 5
a = b
let (x, y) = (1, 2)
```

В отличие от C, подобный код приведет к ошибке

```
let x = 1
if x = 5 {

}
```

Арифметические операторы

Бинарные операторы

```
let a = 5
let b = 3
var c = 0
c = a + b
c = a / b
```

Унарные операторы

```
c = -a
```

Конкатенация строк

```
let helloWorld = "Hello" + "world!"
```

Составные операторы присваивания

Позволяют немного сократить запись.

Сочетание оператора присваивания
и арифметического оператора.

```
var a = 1
a += 2 // значение a равно 3,
       a += 2 эквивалентно a = a + 2
```

Также используются

```
a -= 2
a *= 2
a /= 2
```

Операторы сравнения

Все основные C операторы сравнения (==, <, >).
Операторы эквивалентности (=== и !==)
возвращают Bool.

Для кортежей до 7 элементов

```
(1, "zebra") < (2, "apple")
(4, "dog") == (4, "dog")
```

Типы и количество элементов должны совпадать.

Тернарный условный оператор

Не имеет особенностей реализации, используется для сокращения записи if-else.

```
question ? answer1 : answer2
```

```
if question {
    answer1
} else {
    answer2
}
```

Nil-coalescing оператор

Для работы с optional значениями.
Бинарный оператор (??).

```
var a: Int?
let b: Int = 5
var c: Int = 0

c = (a ?? b) // c = 5
```

```
if a != nil {
    c = a!
} else {
    c = b
}
```

Операторы диапазона

- Сокращенная форма для представления ряда значений.
- Поддерживается только тип Int.
- Обязательно a < b.
- Восходящий диапазон от a до b (a...b).
- Для нисходящего диапазона используем функцию reversed() (a...b).reversed().

Операторы диапазона

Три типа: закрытые, полуоткрытые и односторонние.

- Закрытый диапазон (a...b)
  включает оба значения 1...5
- Полуоткрытый (a..<b)
  в отличие от закрытого, b не входит. 1..<5

Операторы диапазона (односторонний диапазон)

- От минимально возможного значения, до значения указанного вами (…a)
- От значения указанного вами до максимально возможного значения (a...)

Обычно применяется для работы с массивами

```
let letters = ["a", "b", "c", "d", "e"]
for letter in letters[2...] {
    print(letter) // c, d, e
}

for letter in letters[...2] {
    print(letter) // a, b, c
}
```

Операторы диапазона (без ограничений)

Исключительные ситуации

```
let rangeLeft = ...5
for index in rangeLeft {
    print(index)
} // Приведет к ошибке

let rangeRight = 5...
for index in rangeRight {
    print(index)
} // Правая граница равна
            максимальному значению Int
```

Логические операторы

Стандартная реализация операторов

Модифицируют или комбинируют булевы значения

- Логическое НЕ
  (!a)
- Логическое И
  (a && b)
- Логическое ИЛИ
  (a || b)