



ЛЕКЦИЯ 2

Примитивы. Синтаксис

Примитивы. Числа

Целые числа (со знаком):

int (платформозависимый тип)

int8

int16

int32 (псевдоним "rune" для работы с символами)

int64

Целые числа (без знака):

uint (платформозависимый тип)

uint8 (псевдоним "byte" для работы с бинарными данными)

uint16

uint32

uint64



Примитивы. Числа

Числа с плавающей запятой:

float32

float64

Комплексные числа (пары из вещественной и мнимой части):

complex64 (пара из float32)

complex128 (пара из float64)



Примитивы. Числа

Числовые типы с неограниченной точностью (пакет big):

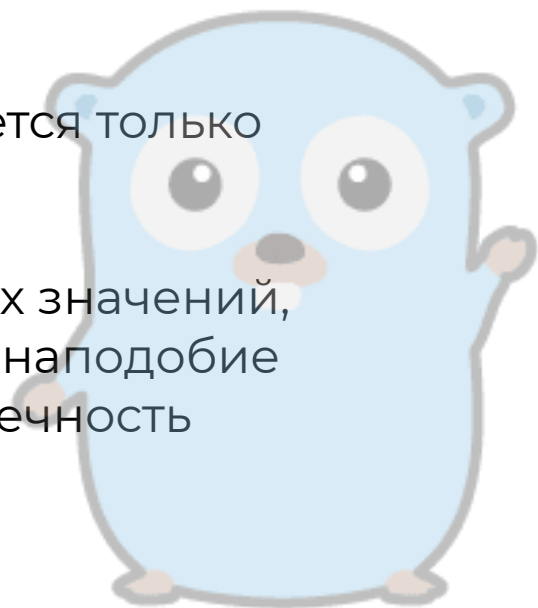
`big.Int`

`big.Rat`

`big.Float`

Размер этих чисел может быть любым и ограничивается только объёмом доступной памяти.

В дополнение к числам существуют несколько других значений, таких как: «not a number» (не число) (NaN, для вещей наподобие $0/0$), а также положительная и отрицательная бесконечность ($+\infty$ и $-\infty$)



Числа. Операторы

Арифметические операторы:

$x + y$	сумма x и y
$x - y$	разница между x и y
$x * y$	произведение x и y
x / y	результат деления x на y
$x \% y$	остаток деления x на y
$-x$	изменение знака x
$+x$	тождественность x
$x++$	инкремент x
$x--$	декремент x



Числа. Операторы

Операторы сдвига (поразрядные):

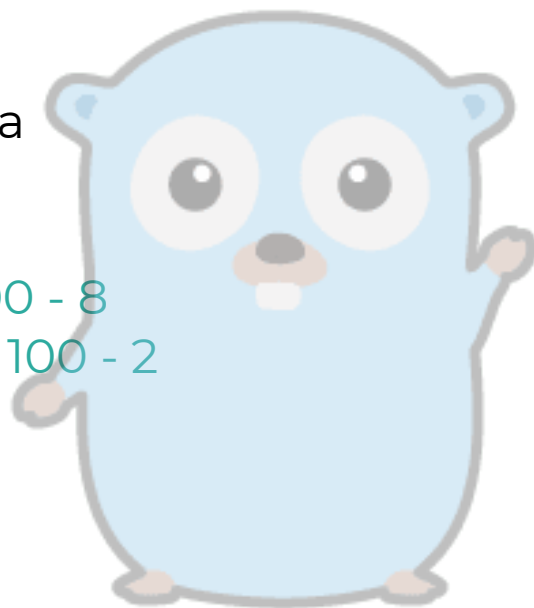
<< - сдвигает битовое представление числа, представленного первым операндом, влево на определенное количество разрядов, которое задается вторым операндом.

>> - сдвигает битовое представление числа вправо на определенное количество разрядов.

```
var b int = 2 << 2  
var c int = 16 >> 3
```

// 10 на два разряда влево = 1000 - 8

// 10000 на три разряда вправо = 100 - 2



Числа. Операторы

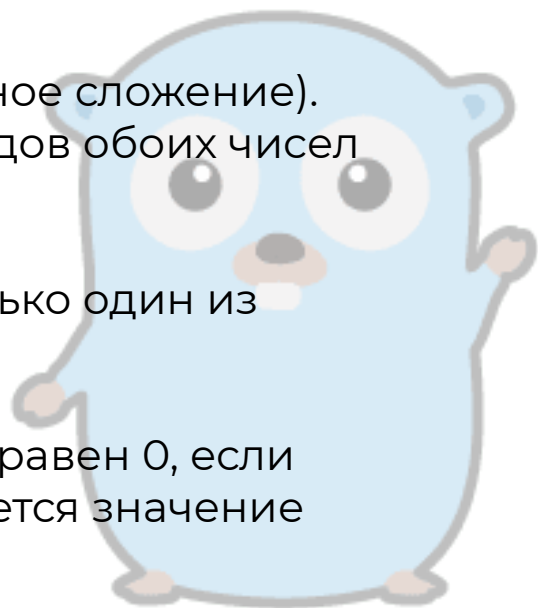
Поразрядные операции:

& - поразрядная конъюнкция (операция И или поразрядное умножение). Возвращает 1, если оба из соответствующих разрядов обоих чисел равны 1. Возвращает 0, если разряд хотя бы одного числа равен 0

| - поразрядная дизъюнкция (операция ИЛИ или поразрядное сложение). Возвращает 1, если хотя бы один из соответствующих разрядов обоих чисел равен 1

^ - поразрядное исключающее ИЛИ. Возвращает 1, если только один из соответствующих разрядов обоих чисел равен 1

&^ - сброс бита (И НЕ). В выражении $z = x \&^ y$ каждый бит z равен 0, если соответствующий бит y равен 1. Если бит y равен 0, то берется значение соответствующего бита из x .



Числа. Операторы

Поразрядные операции:

var a int = 5 | 2 // 101 | 010 = 111 7

var b int = 6 & 2 // 110 & 010 = 10 2

var c int = 5 ^ 2 // 101 ^ 010 = 111 7

var d int = 5 &^ 6 // 101 &^ 110 = 001 1



Примитивы. Булев тип

Логические типы:

bool

Логические операторы:

&&	И
	ИЛИ
!	НЕ



Примитивы. Строки

Строковые типы:

string

Операции над строками:

"Hello " + "world"

Конкатенация

"Hello world"[0]

Выбор символа (rune)



Инициализация

Объявление переменных или констант:

```
var test1 int
var test2 int = 1
var test3, test4 int
var test5, test6 int = 1, 2
var (
    test7 int
    test8 int
)
test9 := 1    // для локальных
```

```
const test1 int = 1
const test2, test3 int = 1, 2
const (
    test4 int = 1
    test5 int = 2
)
```



Константы. Iota

```
const (  
    C1 = iota + 1  
    C2  
    _ // псевдопеременная  
    C3  
)  
  
fmt.Println(C1, C2, C3)    // "1 2 4"
```



Присваивание

$a = b$ // присвоить переменной a значение b

$v1 := v2$ // аналог $\text{var } v1 = v2$

$i, j = j, i$ // поменять местами значения i и j

