

Целые числа (int)

Числа в Python 3 ничем не отличаются от обычных чисел. Они поддерживают набор самых обычных математических операций:

$x + y$	Сложение
$x - y$	Вычитание
$x * y$	Умножение
x / y	Деление
$x // y$	Получение целой части от деления
$x \% y$	Остаток от деления
$-x$	Смена знака числа
$\text{abs}(x)$	Модуль числа
$\text{divmod}(x, y)$	Пара ($x // y$, $x \% y$)
$x ** y$	Возведение в степень
$\text{pow}(x, y, z)$	x^y по модулю (если модуль задан)

In [1]:

```
print('255 + 34 =', 255 + 34)

print('5 * 2 =', 5 * 2)

print('20 / 3 =', 20 / 3)

print('20 // 3 =', 20 // 3)

print('20 % 3 =', 20 % 3)

print('3 ** 4 =', 3 ** 4)

print('pow(3, 4) =', pow(3, 4))

print('pow(3, 4, 27) =', pow(3, 4, 27))

print('3 ** 150 =', 3 ** 150)
```

```
255 + 34 = 289
5 * 2 = 10
20 / 3 = 6.666666666666667
20 // 3 = 6
20 % 3 = 2
3 ** 4 = 81
```

```
pow(3, 4) = 81
pow(3, 4, 27) = 0
3 ** 150 = 369988485035126972924700782451696644186473100389722973815184405301748249
```

Вещественные числа (float)

Вещественные числа поддерживают те же операции, что и целые

```
In [2]: 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1
```

Out[2]: 0.9999999999999999

Для высокой точности используют другие объекты (например Decimal и Fraction)).

Также вещественные числа не поддерживают длинную арифметику:

```
In [3]: a = 3 ** 1000
a + 0.1
```

```
-----
OverflowError                                Traceback (most recent call last)
<ipython-input-3-88fe2c7be406> in <module>
      1 a = 3 ** 1000
----> 2 a + 0.1
```

OverflowError: int too large to convert to float

Простенькие примеры работы с числами:

```
In [4]: c = 150.8
d = 12.7
print('c + d=', c + d)
print()
p = abs(d - c) # Модуль числа
print(p)
print()
round(p, 1) # Округление до одного знака после запятой, вместо 1 может стоять любое
```

c + d= 163.5

138.10000000000002

Out[4]: 138.1

Помимо стандартных выражений для работы с числами (а в Python их не так уж и много), в составе Python есть несколько полезных модулей.

Модуль math предоставляет более сложные математические функции.

```
In [5]: import math

math.sqrt(64) # корень из числа
```

Out[5]: 8.0

Модуль random реализует генератор случайных чисел и функции случайного выбора.

```
In [6]: import random
```

```
random.random()
```

Out[6]: 0.33201266950476604