

Операции с целыми числами

```
In [1]: 2 + 5
Out[1]: 7

In [2]: 10 - 5
Out[2]: 5

In [3]: 6 * 7
Out[3]: 42

In [4]: 12345678 * 987654328765432318765
Out[4]: 12193262318244164938266047670

In [5]: 3 + 5 * 4
Out[5]: 23

In [6]: (3 + 5) * 4
Out[6]: 32

In [7]: 40 // 8
Out[7]: 5

In [8]: 42 // 8
Out[8]: 5

In [9]: 42 % 8
Out[9]: 2

In [10]: 239 % 10
Out[10]: 9

In [11]: 239 // 10
Out[11]: 23

In [12]: 2 ** 5
Out[12]: 32

In [13]: -(42)
Out[13]: -42

In [ ]: +(42)

In [15]: +--42
Out[15]: -42
```

Сообщения об ошибках

```
In [16]: --42
File "<ipython-input-16-85d397208f73>", line 1
--42
SyntaxError: invalid syntax

In [17]: print('test')
* 23
test
File "<ipython-input-17-21eaeacd21f5>", line 2
* 23
^
SyntaxError: can use starred expression only as assignment target

In [18]: 5 // 0
-----
ZeroDivisionError                                Traceback (most recent call last)
<ipython-input-18-8798db347eeb> in <module>()
----> 1 5 // 0

ZeroDivisionError: integer division or modulo by zero
```

Числа с плавающей точкой

```
In [19]: 0.5 + 0.3
Out[19]: 0.8

In [20]: 5 / 2
Out[20]: 2.5

In [21]: 5 // 2
Out[21]: 2

In [22]: 1 / 3
Out[22]: 0.3333333333333333

In [23]: 0.3 + 0.3 + 0.3
Out[23]: 0.8999999999999999

In [24]: 2 ** 5
Out[24]: 32

In [25]: 9 ** 0.5
Out[25]: 3.0

In [26]: 5e-1
Out[26]: 0.5

In [27]: 1234e2
Out[27]: 123400.0
```

Переменные

```
In [28]: a = 3
In [29]: a
Out[29]: 3

In [30]: a += 4
In [31]: a
Out[31]: 7

In [32]: a
2 * a # выводится только последнее значение и только при работе в интерактивном режиме!
Out[32]: 14
```

Для вывода значений в своих программах используйте функцию print(). Обратите внимание на наличие скобок при вызове функции print!

```
In [33]: print(a)
print(2 + a)
7
14
```

Можно выводить диалоговые сообщения при 'общении' с пользователем. Но не отправляйте в проверочную систему программы, содержащие лишний вывод

```
In [34]: name = input('Enter your name: ')
print('Hello ', name)
Enter your name: Pavel
Hello  Pavel
```

```
In [35]: a = int(input())
print(a * 2)
12
24
```

```
In [36]: a = int(input())
b = int(input())
print(a + b)
5
7
35
```

Логические операции

```
In [37]: a = int(input())
print(a > 0)
-234
False
```

```
In [38]: a = int(input())
print(a >= 10 and a < 100)
10
True
```

```
In [39]: a = int(input())
print(10 <= a < 100)
23
True
```

```
In [40]: x1, x2, x3 = False, True, False
not x1 or x2 and x3
Out[40]: True
```

Добавляя скобки в выражения, можно изменить порядок вычисления и значение результирующего выражения. Если не уверены в приоритете операций, смело добавляйте скобки, чтобы быть уверенными в том, что выражение вычисляется именно так, как вы хотите

```
In [41]: ((not x1) or x2) and x3
Out[41]: False
```

Условия

```
In [42]: a = int(input())
b = int(input())
print(a / b)
5
10
0.5
```

```
In [43]: a = int(input())
b = int(input())
if b != 0:
    print(a / b)
else:
    print('Деление невозможно')
5
0
Деление невозможно
```

```
In [44]: a = int(input())
b = int(input())
if b != 0:
    print(a / b)
else:
    print('Деление невозможно')
    b = int(input('Введите ненулевое значение '))
    print(a / b)
4
0
Деление невозможно
Введите ненулевое значение 2
2.0
```

```
In [45]: a = int(input())
b = int(input())
if b != 0:
    print(a / b)
else:
    print('Деление невозможно')
    b = int(input('Введите ненулевое значение '))
    if b == 0:
        print('Вы не справились!')
    else:
        print(a / b)
5
0
Деление невозможно
Введите ненулевое значение 0
Вы не справились!
```

```
In [47]: x = int(input())
if x % 2 == 0:
    print('Четное')
else:
    print('Нечетное')
25
Нечетное
```

Наибольшее из двух чисел

```
In [48]: a = 4
b = 7
if a >= b:
    print(a)
else:
    print(b)
7
```

```
In [49]: a = 4
b = 7
m = a
if b > m:
    m = b
print(m)
7
```

Строки

```
In [50]: a = 'string'
b = 'another string'
print(a, b)
string another string
```

```
In [51]: print(a + b) # конкатенация строк
stringanother string
```

```
In [52]: print(a)
...
multiline
comment
...
print(b)
string
another string
```

```
In [53]: print(a + '\n' + b) # вывод в двух различных строчках
string
another string
```

```
In [54]: 'string1'
Out[54]: 'string1'
```

```
In [55]: "string2"
Out[55]: 'string2'
```

```
In [56]: '''multiple lines
string'''
Out[56]: 'multiple lines\nstring'
```

```
In [57]: '''multiple lines
string with double quotes'''
Out[57]: 'multiple lines\nstring with double quotes'
```

```
In [58]: 'abc' + 'def'
Out[58]: 'abcdef'
```

```
In [59]: 'abc' * 3
Out[59]: 'abcabcabc'
```

```
In [60]: len('abcdef')
Out[60]: 6
```

```
In [61]: 'abc' == 'abc'
Out[61]: True
```

```
In [62]: 'abc' < 'ac'
Out[62]: True
```

```
In [63]: 'abc' > 'ab'
Out[63]: True
```

```
In [64]: print('First line', '\n\n\n', 'Last line')
First line

Last line
```

```
In [65]: # Это комментарий
x = 5 # комментарий к действию
...
Многострочный комментарий — это просто строка
...
Out[65]: '\nМногострочный комментарий — это просто\nстрока\n'
```