

Ссылка на урок: <https://coreapp.ai/app/player/lesson/673c8ffb7a79a7d792108ab0>
(для учеников).

Занятие 7

Тема. Выражения. Математические функции. Часть 1

1. Теоретическая часть

На этом занятии мы поговорим о выражениях и математических функциях, которые используются в ЯП «Python». Рассмотрим следующий пример:

```
my_count = 50 * 1.25
print(my_count) # 62.5
```

Здесь в переменной `my_count` справа от знака «равно» записано выражение `50 * 1.25`.

Интерпретатор вычислит результат (62.5) и запишет его в переменную `my_count`. Интерпретатору не важно, в каком виде записаны данные: 62.5 или `50 * 1.25`. Для него оба варианта — выражения, которые надо вычислить. Он проводит вычисления и приходит к одному и тому же значению — 62.5.

Выражение — это языковая конструкция для вычисления значения с помощью одного или нескольких операндов

или

Выражение — это любой корректный блок кода, который возвращает результат (значение выражения), который можно использовать.

Выражения записываются в виде линейной последовательности символов; знаки операций пропускать нельзя.

Примеры выражений (в комментариях справа от каждого выражения записаны итоговые значения):

<code>62.5</code>	<code># 62.5</code>
<code>50 * 1.25</code>	<code># 62.5</code>
<code>120 / 10 * 2</code>	<code># 24.0</code>
<code>int('100')</code>	<code># 100</code>
<code>'hello'</code>	<code># hello</code>
<code>'Good' + 'will'</code>	<code># Goodwill</code>

Напоминаю, что в языке Python реализованы *операции целочисленного деления*: для деления нацело используется оператор `//`, для взятия остатка от деления — оператор `%`, которые тоже можно использовать при записи выражений.

В состав выражения могут входить не только математические операции, но и функции, например, функция `abs()`, определяющая модуль числа, или функция `pow()`, которая возводит число в степень. К ним мы чуть позже вернёмся.

Функция (подпрограмма) – это самостоятельный блок кода, написанный для выполнения определенной задачи, возвращающий единственный результат. Функцию можно вызывать по имени необходимое число раз.

Вопрос: Является ли вызов функции выражением? Например, `print(my_count)`?

Да, является. Мы знаем, что функции, как и выражения, возвращают результат, значит, они тоже выражения. Функции также могут быть частью других выражений, и мы можем использовать вызов функции прямо в математических операциях.

Все языки программирования позволяют создавать и использовать функции для решения различных задач.

Приоритет операций

При вычислении выражения необходимо соблюдать правильный порядок (последовательность) выполнения операций (*приоритет*). Из школьного курса математики мы знаем, что умножение и деление имеют больший приоритет, чем сложение и вычитание, а *приоритет возведения в степень выше всех остальных арифметических операций*.

Например:

```
print (2 ** 3 * 2) # 16
print (2 * 3 ** 2) # 36
```

Операции одного приоритета выполняются слева направо, например:

```
print (7 + 3 + 2) # 12
print (7 + 3 - 2) # 8
print(2 * 4 * 5 * 10) # 400
print (7 / 2 * 2 / 7) # 1
```

Но нередко вычисления должны происходить в порядке, отличном от стандартного приоритета. Тогда приоритет нужно задавать круглыми скобками, например: $(2 + 2) * 2 = 8$. В скобки можно заключать любую операцию. Скобки могут вкладываться друг в друга сколько угодно раз. Вот примеры:

Например:

```
print(3 ** (4 - 2)) # 9
print(7 * 3 + (4 / 2) - (8 + (2 - 1))) # 14.0
```

Иногда выражение сложно воспринимать визуально. Тогда можно расставить скобки, не повлияв на приоритет:

Было

```
print(8 / 2 + 5 - -3 / 2) # => 10.5
```

Стало

```
print(((8 / 2) + 5) - (-3 / 2)) # => 10.5
```

Главное при этом соблюдать *парность* — *закрывать скобки в правильном порядке* (“раньше открылась – позже закрылась”). Для удобства ставьте сразу открывающую и закрывающую скобку, а потом пишите внутреннюю часть.

Итак, подытожим вышесказанное. Порядок вычисления выражения определяется скобками и приоритетом (старшинством) операций:

1. Круглые скобки.
2. Вычисление функции.
3. Умножение и деление.
4. Сложение и вычитание.

2. Просмотр учебного видеоролика «Правила записи арифметических выражений. Python» (с целью расширения знаний, наглядного представления и закрепления материала):

<https://yandex.ru/video/preview/4432105243778961377> (длина ролика: 05:17).

3. Закрепление знаний

ЗАДАНИЕ 1. Запишите выражения на ЯП Python:

а)
$$\frac{9,2^3 - 2 \cdot 2,4}{3 \cdot 2 + 3} + \frac{7,1^2 \cdot 4}{3 \cdot 7,1^4 - 6 \cdot 1.8}$$

Ответ: $(9.2 ** 3 - 2 * 2.4) / (3 * 2 + 3) + (7.1 * 7.1 * 4) / (3 * 7.1 ** 4 - 6 * 1.8)$

б)
$$\frac{34 + 2 \cdot 4 - 42}{26 \cdot 3^3} - \frac{9 \cdot 2^4}{2 + 1}$$

Ответ: $(34 + 2 * 4 - 42) / (26 * 3 ** 3) - 9 * 2 ** 4 / (2 + 1)$

ЗАДАНИЕ 2. Определите значение переменной *c* после выполнения программы:

а)

```
a = 13
b = a % 5      # 3
b = b * 8      # 24
a = b // 7 + 2  # 5
b = b - b // 7 * 8 # 0
c = a + b      # 5
```

Ответ: 5

б)

```
a = 220
b = a // 5      # 44
b = 10 + b / 4 - 1 # 20
a = b % 7 + 2    # 8
a = b + b // 3    # 26
b = a % b        # 6
c = a + b        # 32
```

Ответ: 32