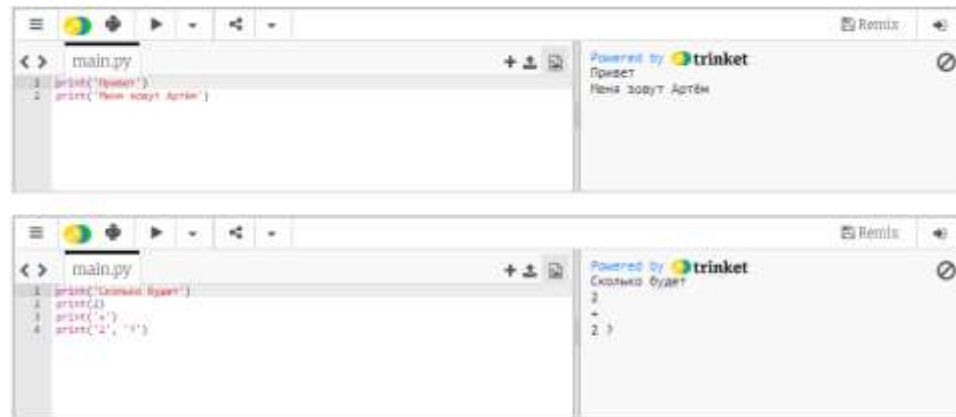


Числа и операции над ними

ТЕОРИЯ

На прошлом занятии мы с вами написали нашу первую программу, но я не рассказал, что же из себя представляет программа.

Программа - это набор инструкций, который описывает определённую последовательность действий. Например:



В первом примере программа состоит из 2 инструкций. Во втором из 4. Они будут выполняться в той последовательности, в которой указаны, то есть сверху вниз.

В свою очередь каждая инструкция создает или обрабатывает объекты и понятие "объект" является ключевым понятием в Python.

`print(5 + 6)` В этой строчке складывают 2 объекта- числа(Будет выведено число 11).

`print('Как дела?')` В этом случае объект - строка.

`print(54 > 55)` В этом случае сравниваются 2 числа и результат сравнения — объект логического типа. Конкретно в этом случае результатом исполнения будет False.

В Python существует очень много различных типов. В таблице представлено большинство из них.

Типы объектов*	Название типа объекта в Python	Пример объекта
Целое число	int (от слова «integer»)	43, -56, 0, 2548
Число с плавающей точкой (вещественные числа)	float	1.5, -43.0, 32.5
Строка	str (от слова «string»)	'hello', "Hello"
Логический тип	bool (от слова «boolean»)	True, False
Список	list	[43, -54.5, True, 'hello']
Словарь	dict (от слова «dictionary»)	{ 'a':123, 'b':456 }
Кортеж	tuple	(1, 2, 3, 4, 5)
Множество	set	{1,2,3,4}
Файлы	file	

В данном уроке мы разберем 2 типа объектов: целые числа(int, от слова "integer") и числа с плавающей точкой(float, число с дробной частью).

Целое число - число, которое не содержит дробную часть. К ним относятся все положительные, все отрицательные, а также 0. Например: 7, 6, 5, 2, 100, -5, -99.

Вещественные числа - числа, которые содержат дробную часть. Целая часть в Python отделяется от дробной знаком "точки". Если вы вместо точки поставите запятую, то Python воспримет это как перечисление двух целых чисел.

Пример: 6.3, 7.5, -9.6.

Операции над числами

В программировании, как и в математике, над числами можно выполнять различные **операции**. Многие из них вы знаете из математики. Все основные математические операции представлены в таблице:

Название операции	Используемый операнд
Сложение	+
Вычитание	-
Умножение	*
Деление	/
Возведение в степень	**
Целочисленное деление	//
Остаток от деления	%

Давайте рассмотрим некоторые из них на практике. Для этого запустите консоль Python - IDLE.

При **сложении** двух целых чисел результатом будет также целое число:

```
>>> 3 + 2
5
>>> 5 + 8
13
```

Если сложить вещественное число с целым, то результатом будет уже вещественное число - оба числа будут преобразованы к типу float:

```
>>> 3.2 + 5
8.2
>>> 4 + 0.3
4.3
```

И, естественно, если я буду складывать два вещественных числа, то получу также вещественное:

```
>>> 2.5 + 3.1
5.6
>>> 0.3 + 0.2
0.5
```

Операция **вычитания** обозначается через дефис. Логика исполнения такая же, как и при сложении:

```

>>> 5 - 3
2
>>> 6 - 9
-3
>>>
>>> 2.9 - 1
1.9
>>> 8 - 1.2
6.8
>>> 7 - 1.0
6.0
>>>
>>> 4.3 - 2.1
2.1999999999999997
>>> 3.5 - 2.3
1.2000000000000002

```

Умножение обозначается звездочкой.

```

>>> 2 * 3
6
>>> 2.0 * 3
6.0

```

Следующая операция - **деление**. Здесь есть 1 нюанс - при делении целого числа вы ВСЕГДА будете получать вещественное:

```

>>> 2 / 3
0.6666666666666666
>>> 9 / 3
3.0
>>> 2.1 / 1
2.1
>>> 2.0 / 1
2.0
>>> 5 / 1.0
5.0

```

Очень полезная операция - **возведение в степень**. Сначала вы указываете кого возводите в степень, затем уже саму степень:

```

>>> 2 ** 3
8
>>> 3 ** 2
9
>>>
>>> 2.0 ** 2
4.0
>>> 2 ** 3.0
8.0

```

Следующая операция - **деление нацело**. Она отличается от обычного деления тем, что ответ всегда будет целого типа. И результат получается исходя из ответа на вопрос "Сколько раз второе число помещается в первое?".

```
>>> 12 // 2
6
>>> 36 // 11
3
>>> 11 // 5
2
>>> 8 // 10
0
```

И последняя операция - это **остаток от деления**. Она обозначается знаком процента %. Результат получается исходя из ответа на вопрос "Какое число нужно прибавить к результату целочисленного деления этих двух чисел, чтобы получить исходное число?".

```
>>> 19 % 7
5
>>> 27 % 5
2
>>> 15 % 3
0
>>> 2 % 7 # Здесь 7-ка умещается в двойку 0 раз, соответственно ответом будет 2
2
```

Сложна, да?)