

Язык программирования Python

Основы языка Python Линейные программы



Общие сведения



о языке программирования Python

Алфавит языка Python (набор допустимых символов) состоит из букв латинского алфавита (причём *заглавные и строчные буквы различаются*), цифр и специальных знаков (знаков препинания, арифметических и других). Русские буквы могут использоваться только при выводе текста на экран и в комментариях к программе.

Служебные слова – цепочки символов, имеющие фиксированное смысловое значение.

Величины в программе представлены в виде констант и переменных.

Константы – величины, не изменяющие своего значения при выполнении программы.

Переменные – величины, которые могут изменять свое значение при выполнении программы. Каждая переменная имеет имя, тип и значение.

Имя переменной (идентификатор) – любая отличная от служебных слов последовательность латинских букв, цифр и символа подчеркивания "_", не может начинаться с цифры.

N, N1, massa, massa_tela – **правильно**;

1N, масса, massa tela – **неправильно**.

Общие сведения о языке программирования Python



Значения переменных хранятся в ячейках оперативной памяти.

Тип переменной определяет способ хранения данных в памяти компьютера и допустимые операции над ними.

Основные типы данных в языке Python

Название	Обозначение	Допустимые значения
Целочисленный	int («integer»)	Сколько угодно большие целые числа, размер ограничен оперативной памятью
Вещественный	float («floating point»)	Любые числа с дробной частью (с плавающей точкой)
Строковый	str («string»)	Произвольная последовательность символов из таблицы Unicode
Логический	bool («boolean»)	False («Ложь») или True («Истина»)

Целая часть числа от дробной отделяется точкой.

Строковое значение заключается в двойные или одинарные кавычки.

Тип переменной определяется автоматически в момент присваивания ей значения и может изменяться по ходу выполнения программы.



Выражения и операции

Выражение – это конструкция, возвращающая значение некоторого типа.

Простыми выражениями являются переменные и константы.

Сложные выражения строятся из простых с помощью операций, функций и скобок. Данные, к которым применяются операции, называются **операндами**.

Используется линейная форма записи выражений (в одну строку).

Арифметические операции

Операция	Обозначение	Пример
Сложение	+	$3 + 4 = 7$
Вычитание	-	$7 - 2 = 5$
Умножение	*	$2 * 2 = 4$
Деление	/	$8 / 2 = 4$
Целочисленное деление	//	$9 // 2 = 4$
Остаток от деления	%	$9 \% 2 = 1$
Возведение в степень	**	$2 ** 3 = 8$



Выражения и операции

Логические выражения могут содержать величины или выражения, которые сравниваются между собой с помощью операций сравнения.

Логическое выражение может принимать лишь два значения: «истина» или «ложь».

Операции сравнения

Операция	Символы	Пример
равно	<code>==</code>	<code>x == 0</code>
не равно	<code>!=</code>	<code>x != 0</code>
больше	<code>></code>	<code>x > 0</code>
меньше	<code><</code>	<code>x < 0</code>
больше или равно	<code>>=</code>	<code>x >= 0</code>
меньше или равно	<code><=</code>	<code>x <= 0</code>



Выражения и операции

Приоритет выполнения операций:

- 1) операции в скобках;
- 2) возведение в степень;
- 3) умножение и деление (в том числе // и %);
- 4) сложение и вычитание.

Операции одинакового приоритета выполняются в порядке записи слева направо.

Если выражение слишком длинное и не помещается в одной строке, необходимо заключить всё выражение в скобки (перенос внутри скобок разрешён).

Например:

$$\frac{(a+b)h}{2} \quad \longrightarrow \quad (a+b) * h / 2$$

The diagram shows the conversion of a mathematical fraction to a Python expression. The fraction $\frac{(a+b)h}{2}$ is transformed into the code `(a+b) * h / 2`. Red numbers 1, 2, and 3 are placed above the code to indicate the order of operations: 1 for the addition in the parentheses, 2 for the multiplication, and 3 for the division.

$$v + \frac{at^2}{2} \quad \longrightarrow \quad v + a * t ** 2 / 2$$

The diagram shows the conversion of a mathematical expression to a Python expression. The expression $v + \frac{at^2}{2}$ is transformed into the code `v + a * t ** 2 / 2`. Red numbers 4, 2, 1, and 3 are placed above the code to indicate the order of operations: 4 for the addition of v, 2 for the exponentiation, 1 for the multiplication, and 3 for the division.



Оператор (команда) присваивания

Оператор присваивания записывает в переменную, имя которой находится слева от знака «=» значение выражения, находящегося справа. Старое значение переменной при этом стирается.

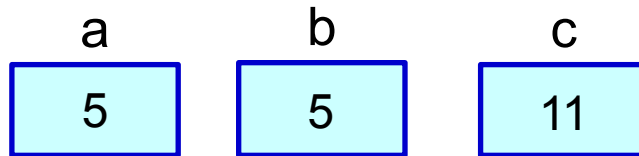
Общий вид оператора:

`<имя переменной> = <выражение>;`

Например:

```
a = 5
b = a
c = a+b
c = c+1
```

В памяти:



В языке Python допускается множественное присваивание:

<i>Запись оператора:</i>	<i>Равносильная запись:</i>
<code>a, b = 0, 1</code>	<code>a = 0</code> <code>b = 1</code>
<code>a = b = 0</code>	<code>a = 0</code> <code>b = 0</code>

Допускается запись нескольких операторов в одной строке через символ « ; ».



Оператор вывода

Вывод данных из оперативной памяти на экран осуществляется с помощью оператора (функции) вывода **print** («печатать»):

```
print (<выражение1>, <выражение2>, ..., <выражениеN>)
```

- На экран выводятся значения переменных и выражений, строковые значения выводятся на экран без кавычек.
- Выводимые значения разделяются пробелом (по умолчанию).
- После выполнения оператора происходит автоматический переход на новую строку.

Например:

```
print ("Масса равна", m, "кг");
```

Для $m=15$ на экране появится:

```
Масса равна 15 кг
```

Здесь и далее символом \square обозначен пробел.



Оператор ввода

Для ввода значений переменных с клавиатуры в процессе выполнения программы используется оператор (функция) ввода **input** («ввод»):

```
<имя_переменной> = input()
```

При выполнении оператора:

- компьютер переходит в режим ожидания данных;
- пользователь вводит с клавиатуры данные в виде строки символов;
- для завершения ввода пользователь нажимает клавишу Enter;
- введенная строка записывается в указанную переменную.

Если вводится не строка, а число, необходимо выполнить преобразование типов с помощью функций **int** (для целых) и **float** (для вещественных).

Например:

На экране:

```
print("Введите слово и два числа:")  
x = input()  
y = int(input())  
z = float(input())  
print(x, y, z)
```

```
Введите слово и два числа:  
ноль  
1  
2  
ноль 1 2.0
```



Оператор ввода

Можно в скобках указать текст подсказки для пользователя.

Например:

```
x = input("Введите слово: ")
y = int(input("Введите целое число: "))
z = float(input("Введите вещественное число: "))
print(x, y, z)
```

На экране:

```
Введите слово: ноль
Введите целое число: 1
Введите вещественное число: 2
ноль 1 2.0
```