

9 занятие

Школа::Кода

«Основы программирования на языке Python»

```
print (48)
```

```
48
```

```
print (22 + 35)
```

```
57
```

```
print ("Hello World")
```

```
Hello World
```

Что будет, если забыть скобки? Или кавычки?

```
print Hello World  
print (Hello World)
```

```
SyntaxError: Missing parentheses in call to 'print'.  
Did you mean print(Hello world)?
```

```
SyntaxError: invalid syntax
```

```
print ("Привет, мир")
```

```
Привет, мир
```

```
print ("Текст в двойных кавычках")
```

```
print ('Текст в одинарных кавычках')
```

```
Текст в двойных кавычках
```

```
Текст в одинарных кавычках
```

А если использовать разные кавычки?

```
print ("Я учусь в школе ')
```

```
SyntaxError: EOL while scanning string literal
```

```
print ("Факультатив `Школа::Кода' .")  
print ('Факультатив "Школа::  
Кода" . `')  
print ("""Факультатив  
"Школа::Кода".""")
```

Факультатив `Школа::Кода' .

Факультатив "Школа::Кода" .

Факультатив

"Школа::Кода"

КОММЕНТАРИИ

```
print (598)
print (22 + 35) # это комментарий
#print 98
print (17 * 50)
""" Это многострочный комментарий,
содержащий код
print 9 * 11
"""
```

598

57

850

ТИПЫ ДАННЫХ

```
# int (integer) – целое число
print (4)      # целое положительное
print (-5)     # целое отрицательное

# float – дробное (вещественное) число
print (9.35)   # дробное положительное
print (-23.5)  # дробное отрицательное
```

4

-5

9.35

-23.5

```
# string - строка (текст)
print ('text content')      # строка
print ("other text")       # строка

# bool (boolean) - логическая (двоичная)
print (True)             # истина
print (False)           # ложь
```

```
text content
other text
True
False
```

МАТЕМАТИЧЕСКИЕ ОПЕРАТОРЫ

```
print (5 + 3)
```

```
print (10 - 4)
```

```
print (4 * 8)
```

```
print (20 / 4)
```

```
#остаток от деления:
```

```
print (5 % 2)
```

```
#возведение в степень:
```

```
print (2 ** 4)
```

8

6

32

5

1

16

$$\begin{array}{r|l} 7 & 2 \\ \hline 6 & 3 \\ \hline 1 & \end{array}$$

```
print (7 % 2)
```

1

$$\begin{array}{r|l} 11 & 4 \\ \hline 8 & 2 \\ \hline 3 & \end{array}$$

```
print (11 % 4)
```

3

$$\begin{array}{r|l} 5 & 2 \\ 4 & 2 \\ \hline 1 & \end{array}$$

`print (5 % 2)`

1

$$\begin{array}{r|l} 12 & 2 \\ 12 & 6 \\ \hline 0 & \end{array}$$

`print (12 % 2)`

0

$$\begin{array}{r|l} 58 & 10 \\ \hline 50 & 5 \\ \hline 8 & \end{array}$$

$$\begin{array}{r|l} 32 & 10 \\ \hline 30 & 3 \\ \hline 2 & \end{array}$$

```
print(58 % 10)
```

8

```
print(32 % 10)
```

2

Возведение в степень

$$2 * 2 * 2 * 2 = 2^4$$

$$5 * 5 * 5 = 5^3$$

Что выведет программа?

```
print (38 + 12)
```

```
print (15 - 5)
```

```
print (5 * 3)
```

```
print (30 / 5)
```

```
print (38 % 10)
```

```
print (2 ** 3)
```

ПЕРЕМЕННЫЕ

```
my_int = 5  
print (my_int)
```



5

Как НЕЛЬЗЯ называть переменные

русский	ИМЯ
«*»	*level
«+»	weight+
«-»	new-num
«?» или «!»	warning!

с числа	1name
число	5
пробел	my name

Обычно различают следующие стили (1)

- b (одиночная маленькая буква)
- B (одиночная заглавная буква)
- lowercase (слово в нижнем регистре)
- lower_case_with_underscores (слова из маленьких букв с подчеркиваниями)
- UPPERCASE (заглавные буквы)
- UPPERCASE_WITH_UNDERSCORES (слова из заглавных букв с

Обычно различают следующие стили (2)

- CapitalizedWords (слова с заглавными буквами, или CapWords, или CamelCase. Иногда называется StudlyCaps). Когда вы используете аббревиатуры в таком стиле, пишете все буквы аббревиатуры заглавными – `HTTPServerError`.
- mixedCase (отличается от CapitalizedWords тем, что первое слово начинается с маленькой буквы)

Как МОЖНО называть переменные

английский	name
цифры	name1
«_»	last_name
(классы)	HumanClass
(константы)	BASE_PATH


```
summ = 0
print (summ)
summ = summ + 50 * 2
print (summ)
summ = summ + 400 * 1
print (summ)
```

```
0
100
500
```

Как вывести возраст?

```
current_year = 2015
```

```
birth_year = 1997
```

???

```
name = "Валерий"  
print ("Здравствуйте, " + name)  
print ("...")      # другой текст  
print (name + ", предлагаем Вам...")
```

Здравствуйте, Валерий

...

Валерий, предлагаем Вам...

```
name = "Анна"  
action = "говорит"  
print (name + action)  
print (name + ' ' + action)  
print (name, action)
```

Аннаговорит

Анна говорит

Анна говорит

```
name = "Светлана"  
age = 23  
print (name + ' - ' + age)
```

```
TypeError: must be str, not int
```

```
name = "Светлана"
```

```
age = 23
```

```
print (name + ' - ' + str(age))
```

```
print (name, '- ', age)
```

```
Светлана - 23
```

```
Светлана - 23
```

Ввод данных

```
print ('Как вас зовут?')  
#записываем ввод в переменную name:  
name = input()  
print ('Здравствуйте, ' + name + '!')
```

Как вас зовут?

Петр

Здравствуйте, Петр!

input читает значение из консоли как строку

```
a = input()  
b = input()  
print a + b
```

100

500

100500

Перевод в целое число

```
a = input ()  
b = input ()  
print (int (a) + int (b) )
```


100


500


600

Функции str(), int() и float()

`print (str (18))`  `print ("18")`


`print (int ("25"))`  `print (25)`

`print (int (38.5))`  `print (38)`

`print (float ("32.5"))`  `print (32.5)`

`print (float (38))`  `print (38.0)`

УСЛОВНЫЕ ВЫРАЖЕНИЯ

```
print (5 > 3) True
```

```
print (4 > 8) False
```

```
print (18 < 18) False
```

```
print (18 >= 18) True
```

```
print (18 == 18) True
```

Присваивание vs сравнение

```
a = 18  
print (a)
```

18

```
print (18 == 18.0)
```

True

```
print (18 == 18.0)
```

True

```
print (18 == "18")
```

False

```
print (6 != 5)
```

True

```
num = 30
```

```
print (num > 10)
```

True

```
print (num == 30)
```

True

Что выведет программа?

```
age = 25  
print (age == 25.0)  
  
print (age == "25")  
  
print (age != 5)
```

УСЛОВИЯ


```
if 5 > 3:  
    print ("Ok")
```

Ok

if УСЛОВНОЕ_ВЫРАЖЕНИЕ :

•••• КОД_ЕСЛИ_УСЛ_ВЫР_ИСТИННО

```
weather = "дождь"  
if weather == "дождь":  
    print ("Взять зонт")  
    print ("Одеть зимнюю куртку")  
else:  
    print ("Одеть весеннюю куртку")  
print ("Выйти на улицу")
```

Взять зонт

Одеть зимнюю куртку

Выйти на улицу

```
age = 19
if age >= 18:
    print ("Вам есть 18 лет")
else:
    print ("Вам еще нет 18 лет")
```

Вам есть 18 лет

```
age = 15
if False :
    print ("Вам больше 18 лет")
elif False :
    print ("Вам 18 лет")
else:
    print ("Вам меньше 18 лет")
```

Вам меньше 18 лет

```
age = 32
if age > 18:
    print ("Вам больше 18 лет")
if age == 18:
    print ("Вам 18 лет")
else:
    print ("Вам меньше 18 лет")
```

Вам больше 18 лет

Вам меньше 18 лет

```
points = 75      # 75 баллов
if points > 90:
    print ("Ваша оценка - пять")
elif points > 75:
    print ("Ваша оценка - четыре")
elif points > 60:
    print ("Ваша оценка - три")
else:
    print ("Ваша оценка - два")
```

ИТОГИ

- для вывода используется `print()`
- для ввода используется `input()`
- числа записываются без кавычек
- текст (строка) пишется в одинарных или двойных кавычках
- основные типы данных в python:
 - целые (положительные и отрицательные):
4, -5
 - дробные числа (положительные и отрицательные): 49.5, -89.396
 - строка: 'разные' или "кавычки"
 - логический тип (двоичный): True или False
- преобразование типов: `int()`, `float()` и `str()`

- для написания комментария до конца строки используется знак **#**
- математические операторы: **+**, **-**, *****, **/**, **%**, ******
- переменные определяется так: **var_name = "value"**
- условные операторы: **>**, **<**, **>=**, **<=**, **==**, **!=**
- условия:
 - синтаксис: **if** условное_выражение:
 - после двоеточия идет табулированный блок, который нужно выполнить, если условное выражение истинно
 - можно писать одно или несколько дополнительных условий: **elif**
 - можно использовать блок **else** (иначе)

PYTHON 2 VS PYTHON 3

```
print "The answer is", 2*2
```

```
print ("The answer is", 2*2)
```

Запятая в конце подавляет перевод строки:

```
print x,
```

```
print (x, end=" ")
```

Печатает перевод строки:

```
print
```

```
print ()
```

Русские символы:

```
# -*- coding: utf-8 -*-
```

```
print len(u"Маша")
```

```
print len("Маша")
```

Ввод из консоли:

```
raw_input()
```

```
input()
```

```
print 5 / 3
```

```
1
```

```
print float(5) / 3
```

```
1.666666666666667
```

```
print (5 / 3)
```

```
1.666666666666667
```

```
print (5 // 3)
```

```
1.0
```