



ОРГАНИЗАЦИЯ ВВОДА И ВЫВОДА ДАННЫХ

НАЧАЛА ПРОГРАММИРОВАНИЯ

8 класс

Ключевые слова

- оператор вывода `print`
- формат вывода
- оператор ввода `input`



Вывод данных

Вывод данных из оперативной памяти на экран монитора:

```
print(<выражение 1> ,< выражение 2> , ..., < выражение N>)
```

СПИСОК ВЫВОДА

Выражения - символьные, числовые, логические, в том числе переменные и константы

Пример:

```
s = 10
```

```
print ('s=',s)
```

На экране:

```
s= 10
```



**Информация в кавычках
выводится на экран без
изменений**

Варианты организации вывода

Вариант организации вывода	Оператор вывода	Результат
По умолчанию	<code>print (1, 20, 300)</code>	1 20 300
Убрать разделители — пробелы	<code>print (1, 20, 300, sep="")</code>	120300
Добавить другой разделитель	<code>print (1, 20, 300, sep=", ")</code>	1, 20, 300

Формат вывода

Формат вывода позволяет установить количество позиций на экране, занимаемых выводимой величиной.

d - вывод целого значения

f – вывод вещественного значения

Оператор вывода	Результат выполнения оператора
<pre>a = 4 print ("a=", "{:5d} {:5d}".format(a, a*a))</pre>	a=○○○○4○○16
<pre>a = 1/3 b = 1/9 print ("{:7.3f} {:7.3f}".format(a, b))</pre>	○○0.333○○0.111

Первая программа

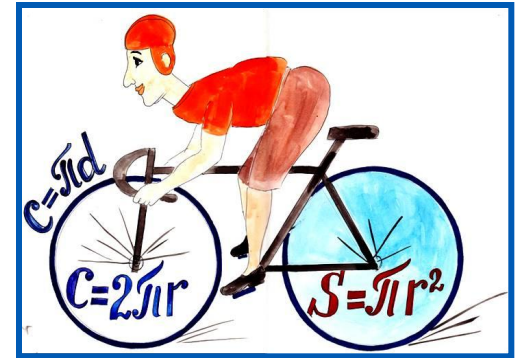
```
r = 5.4
```

```
c = 2 * 3.14 * r
```

```
s = 3.14 * r * r
```

```
print ('c=', c)
```

```
print ('s=', s)
```



```
Python 3.4.3 Shell
File Edit Shell Debug Options Window Help
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [M
SC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more informat
ion.
>>> ===== RESTART =====
>>>
c= 33.912000000000006
s= 91.562400000000003
>>> |
```

Ввод данных с клавиатуры

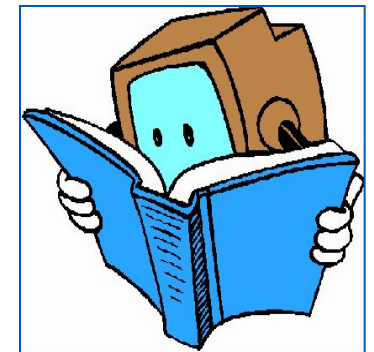
Ввод в оперативную память значений переменных :

```
<имя переменной> = input()
```

Переменная будет строкового типа (str)

Выполнение оператора **input**:

- 1) компьютер переходит в режим ожидания данных:
- 2) пользователь вводит данные с клавиатуры:
- значение переменной
- 3) пользователь нажимает клавишу **Enter**.



Ввод данных с клавиатуры

Ввод переменных целого типа:

```
<имя переменной> = int (input())
```

Переменная из строковой сразу преобразуется в целую

Ввод переменных вещественного типа (float):

```
<имя переменной> = float (input())
```

Переменная из типа str сразу преобразуется в тип float

Ввод переменных с приглашением:

```
n = int (input ("введи n:"))
```


Ввод данных с клавиатуры

! *Количество переменных* в операторе присваивания в левой части должно *соответствовать количеству переменных*, введенных с клавиатуры пользователем

Ввод двух значений в одной строке

a и *b* - целые числа, данные разделены пробелом

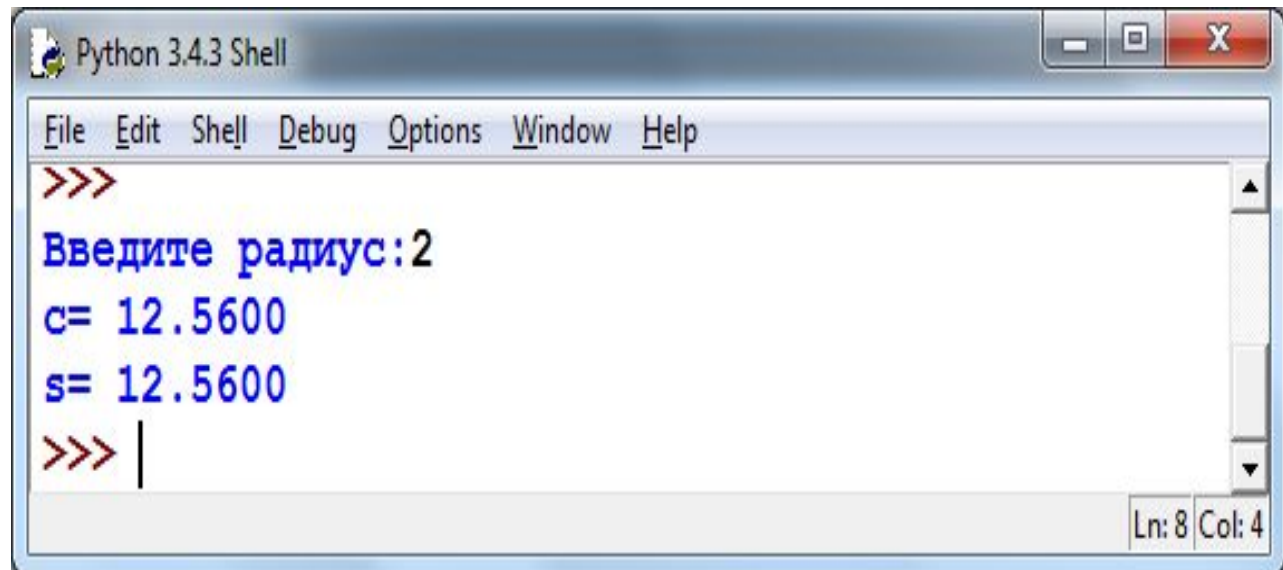
```
a, b = map ( int, input().split() )
```

При вводе числа ввести через пробел:

```
4 5 <Enter>
```

Улучшенная программа

```
r = float(input('Введите радиус:'))  
c = 2 * 3.14 * r  
s = 3.14 * r * r  
print ("c=", "{:6.4f}".format(c))  
print ("s=", " {:6.4f}".format(s))
```



```
Python 3.4.3 Shell  
File Edit Shell Debug Options Window Help  
>>>  
Введите радиус:2  
c= 12.5600  
s= 12.5600  
>>> |  
Ln: 8 Col: 4
```

Самое главное

Для **ввода** в оперативную память значений переменных используется оператор ввода ***input***.

Для **вывода данных** из оперативной памяти на экран монитора используется оператор вывода ***print***.

Ввод исходных данных и вывод результатов должны быть организованы ***понятно и удобно***; это обеспечивает дружелюбность пользовательского интерфейса.



Опорный конспект

Ввод в оперативную память значений переменных выполняет оператор ввода *input*.

```
<имя переменной> = input ("текст - подсказка")
```

приглашение

Вывод данных из оперативной памяти на экран монитора выполняет оператор вывода *print*.

```
print( <выражение 1> , < выражение 2> , ..., < выражение N> )
```

СПИСОК ВЫВОДА

Вопросы и задания

Дан фрагмент программы. Выберите правильный оператор, если необходимо, используйте операторы вхождения и вывода их в стандартный поток вывода. Одно из входов вводится в обратном порядке. Одних данных для вычисления запишите оператор ввода, соответствующий входному потоку.

```
a) print('a', a)
b) print('a', a)
c) print('a', a)
d) print('a', a)
```

Какие числа будут выведены на экран компьютера?
Упростите его, сократив число операторов.